

Universidad de Las Ciencias Informáticas

Facultad 2



“Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática y diseño de una aplicación que informatice el cálculo de los costos de calidad.”

Trabajo de Diploma para optar por el Título de:

Ingeniero en Ciencias Informáticas

Autor:

Dairys Machado López

Tutores:

Ing. Aymara Marín Díaz

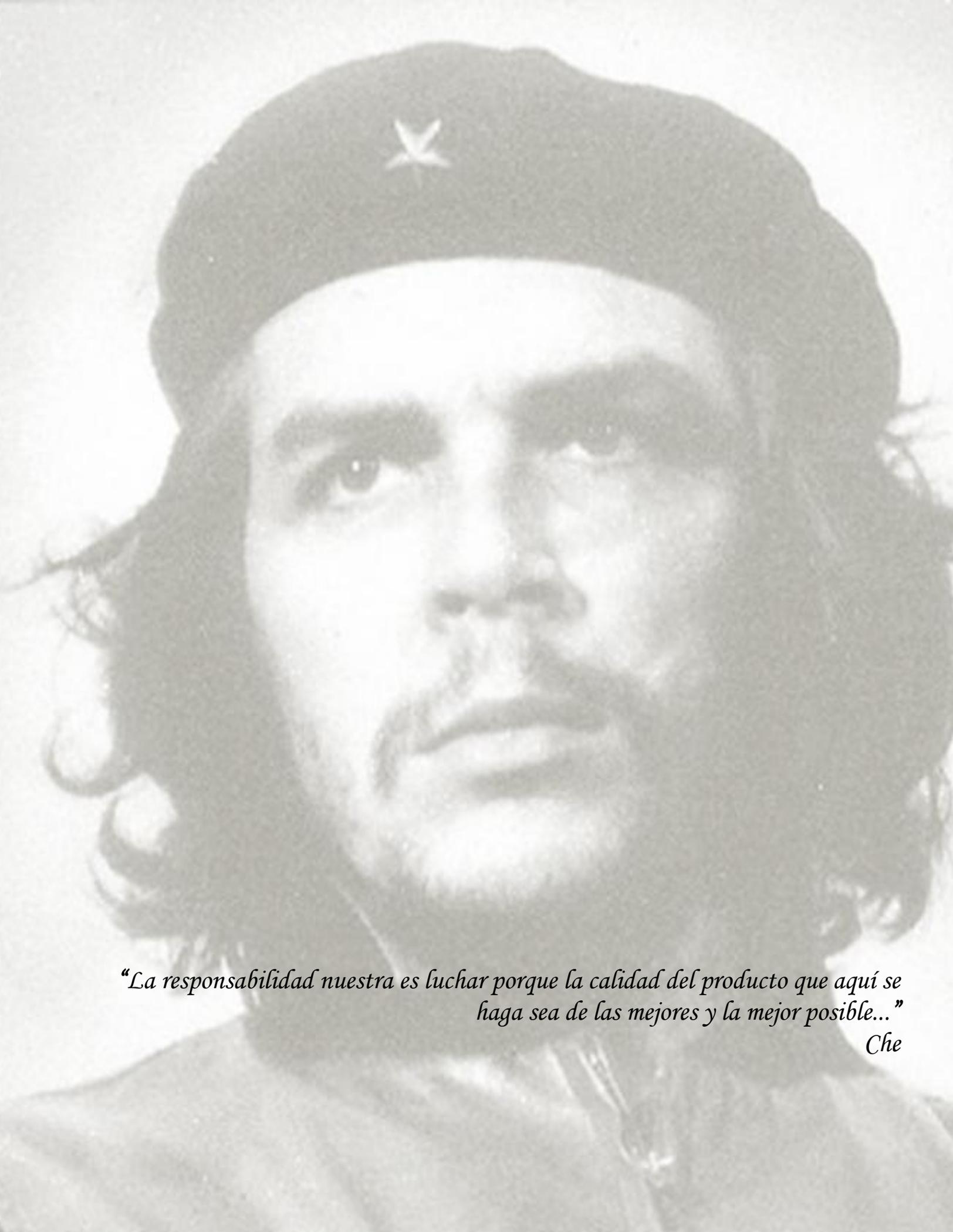
Ing. Denys Buedo Hidalgo.

Co-tutor:

Ing. Leslye Bravo García

La Habana, Junio de 2011

“Año 53 de La Revolución”



*“La responsabilidad nuestra es luchar porque la calidad del producto que aquí se
haga sea de las mejores y la mejor posible...”*

Che

Declaración de Autoría

Por este medio declaro que soy la única autora de este trabajo y autorizo a la Universidad de las Ciencias Informáticas para que haga el uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año 2011.

Dairys Machado López

Firma del Autor

Aymara Marín Díaz

Firma del Tutor

Denys Buedo Hidalgo

Firma del Tutor

Leslye Bravo García

Firma del Co-Tutor

Tutor: Ing. Aymara Marín Díaz.

Email: amarin@uci.cu.

Currículum:

- Graduado de la Universidad de las Ciencias Informáticas, con categoría docente: Instructor.
- 2 años de experiencia productiva.
- Ocupa el cargo de Asesor de Calidad del Centro de Telemática.

Tutor: Ing. Denys Buedo Hidalgo.

Email: dbuedo@uci.cu.

Currículum:

- Graduado de la Universidad de las Ciencias Informáticas, con categoría docente: Instructor.
- 4 años de experiencia productiva.
- Ocupa el cargo de Director del Centro de Telemática.

Co-Tutor: Ing. Leslye Bravo García.

Email: lbravo@uci.cu

Currículum:

- Graduado de la Universidad de las Ciencias Informáticas, con categoría docente: Instructor
- 1 año de experiencia productiva.

Agradecimientos

A mi mamá querida, por ser la mejor madre del mundo, por ser la luz que ilumina mi vida, por todo el amor que me ha dado, por confiar incondicionalmente en mí, por apoyarme en todas mis decisiones, por creer en mí y darme las fuerzas y enseñanzas necesarias para enfrentar los retos de la vida... gracias Mami...te quiero mucho.

A mi papá, por ser el mejor padre del mundo, por ese amor ciego que me tienes, por estar ahí continuamente aconsejándome, por confiar en mí y por siempre darme fuerzas para seguir adelante.

A mis hermanos Tamara, Kenia y Liván, por estar en cada etapa de mi vida y alegrarme la vida con su presencia, por creer siempre en mí, por su confianza y todo su apoyo...Los quiero mucho.

A mi familia en general, especialmente a mis sobrinos: Nachy, Manolito, Libancito, Yoenis y Greity, que a pesar de saber poco de la vida, me dan confianza, amor, apoyo y me quieren. Mi cariño es para Uds.

A mis buenas amistades que nunca olvidaré, por compartir tantos momentos inolvidables, por su preocupación constante, y porque siempre me han apoyado, me han dado cariño, consejos, fuerzas y aliento para seguir adelante: Mary, Yanara, Dailenis, Gleidis, Evelyn.

A todos los compañeros que tuve en la UCI, por compartir conmigo estos 5 años de universidad.

A todas las personas que de una forma u otra ayudaron a que mi graduación como ingeniera fuera un hecho e hicieron de mí una mejor persona...gracias.

Dairys

Dedicatoria

Este gran triunfo está dedicado principalmente a mi mamá, Neyda López Lozano, que sin ella nada de esto fuera posible, a ella que toda una vida ha sido una excelente madre, amiga, compañera, por ser lo más grande que tengo en el mundo, por dejarme ver a través de sus ojos.

A mi papá, mis hermanos y sobrinos, por hacer de mí lo que soy hoy y por haber hecho posible que este sueño sea una realidad.

Resumen

Dentro de la gestión de la calidad los Costos Totales de la Calidad se consideran una herramienta fundamental en la mejora continua de las organizaciones. Estos se basan fundamentalmente en lo que invierte una empresa para desarrollar productos con la calidad requerida. Es por ello que el aumento de las ganancias de las empresas, como resultado de un elevado nivel de calidad de los productos y una disminución de los costos, debe ser un objetivo a alcanzar. Dada la importancia que requiere este tema y contribuyendo al logro del mismo, en el presente trabajo de diploma se describen los antecedentes de la calidad, el papel que juegan los Costos Totales de la Calidad en la misma y el mejoramiento continuo como principio básico de la gestión de la calidad.

Para lograr su demostración se hace necesario realizar un análisis de los diferentes indicadores que permiten conocer el estado de los costos de la calidad en los proyectos de software del Centro de Telemática, con el objetivo de proponer acciones concretas que permitan disminuir y mejorar costos de calidad, una vez que se indique que son desfavorables. Asimismo, el presente trabajo de diploma propone el diseño de un sistema informático para calcular los Costos Totales de la Calidad en los proyectos del centro. La consecución de dicho sistema conllevará a minimizar la deficiente gestión de la calidad que existe en la ejecución del proceso de desarrollo del software llevado a cabo por los proyectos de software del centro mediante la optimización de los costos de la calidad en que incurren, facilitando de manera efectiva la toma de decisiones.

Palabras Clave: Calidad, Costos, Gestión, Mejoramiento.

Tabla de Contenidos

Agradecimientos	IV
Dedicatoria	V
Resumen	VI
Introducción	1
Capítulo 1. Fundamentación Teórica.....	5
1.1 Introducción.....	5
1.2 Calidad del Software.....	5
1.3 Gestión de la Calidad	6
1.3.1 Mejoramiento Continuo	8
1.4 Modelo de Madurez de Capacidad Integrada	9
1.5 Costos Totales de la Calidad y su clasificación	11
1.5.1 Procedimiento para determinar los Costos Totales de la Calidad en los proyectos de software del Centro de Telemática.....	14
1.5.1.1 Análisis de indicadores que permiten conocer el estado de los costos.....	16
1.6 Metodología de Desarrollo.....	17
1.6.1 Metodología de Desarrollo de Software	17
1.6.1.1 Proceso Unificado de Desarrollo de Software	18
1.7 Lenguaje de Modelado UML	18
1.8 Herramienta CASE	18
1.8.1 Visual Paradigm	19
1.9 Herramientas de desarrollo y tecnologías.....	19

1.9.1 Marcos de trabajo	19
1.9.2 Lenguajes de programación.....	21
1.9.3 Servidor Web.....	21
1.9.4 Sistema gestor de Base de Datos.....	22
Conclusiones parciales	23
Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática. Características del Sistema.....	24
2.1 Introducción.....	24
2.2 Centro de Telemática de la UCI.....	24
2.3 Objeto de estudio	26
2.3.1 Problema y situación problemática	26
2.3.2 Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.....	26
2.3.3 Objeto de Automatización	30
2.3.4 Información que se maneja	30
2.3.5 Propuesta del sistema.....	31
2.3.6 Modelo de dominio	31
2.3.7 Especificación de los requerimientos del software.....	33
2.3.8 Definición de los Casos de Uso.....	46
2.3.8.1 Definición de los actores	46
2.3.8.2 Diagrama de Paquetes del Sistema	46
2.3.8.3 Diagrama de Casos de Uso del Sistema.....	47
2.3.8.4 Descripción de los Casos de Uso del Sistema	47

Conclusiones parciales	51
Capítulo 3. Diseño del Sistema	52
3.1 Introducción.....	52
3.2 Diseño del sistema.....	52
3.2.1 Diagrama de clases del diseño.....	52
3.2.2 Diagramas de Interacción.....	53
3.2.2.1 Diagramas de Colaboración	54
3.3 Arquitectura y Patrones de Diseño.....	54
3.4 Diseño de la Base de Datos	57
Conclusiones parciales	57
Capítulo 4. Pruebas	58
4.1 Introducción.....	58
4.2 Propuesta de diseño de Casos de Pruebas.	58
Conclusiones parciales	60
Conclusiones.....	61
Recomendaciones.....	62
Bibliografía	66
Anexos.....	68

Introducción

La calidad es un término que ha adquirido gran relevancia con el paso del tiempo, pues se sitúa entre los principales activos con los que cuenta una institución para mejorar su posición competitiva. Mundialmente las empresas productoras de software tienen la necesidad de construir productos con la calidad requerida y que satisfagan las expectativas de los clientes, pues es la base para lograr que el cliente sienta confianza en el producto obtenido.

Hoy en día para poder mantener la efectividad de las organizaciones es necesario adoptar una posición favorable en el mercado, que solo se logra a través de la calidad de los productos y servicios, la satisfacción de los clientes con los mismos, más el control de los costos generados en este empeño. Una de las herramientas que las organizaciones han utilizado para mantener control sobre la efectividad de las mismas es el trabajo con los costos de calidad [1].

Los costos de calidad son una poderosa herramienta para el incremento de la competitividad de las empresas, pues permiten el aumento de la calidad de los productos y una utilización más eficiente de los recursos tecnológicos, humanos y financieros, al señalar dónde se están ocasionando las deficiencias. Para los países en vías de desarrollo, el logro de producciones capaces de entrar en mercados altamente competitivos depende en gran medida de la capacidad de estos de desarrollar productos de alta calidad con costos muy controlados [1].

La economía cubana con recursos limitados, necesita ser cada día más eficiente y competitiva, la búsqueda de mayores resultados económicos y sociales es fundamental en el sector empresarial. Para ello se necesita alcanzar una producción cada vez mayor con el fin de lograr su inserción en el mercado internacional, donde la calidad debe desempeñar un papel fundamental y estratégico, pues permite garantizar la elevación de la eficiencia y de la competitividad, tanto para las empresas del sector productivo, como las del sector de servicios.

La Universidad de las Ciencias Informáticas (UCI) se encuentra entre los principales avances de la informática con que cuenta el país, con el propósito fundamental de situar la producción de software en el mercado mundial. La mencionada universidad tiene la misión de producir software y servicios informáticos a partir de la vinculación estudio-trabajo como modelo de formación. Es considerada la mayor organización productora de software en el país [2].

Uno de los caminos a seguir por la UCI para asegurar el correcto desarrollo y terminación de proyectos, productos y servicios es el de institucionalizar el Modelo de Madurez de Capacidad Integrada (CMMI, por sus siglas en inglés), con el propósito de organizar la calidad en los proyectos de software que se ejecutan en la universidad y por lo tanto, lograr un mayor número de proyectos culminados en plazo, dentro del presupuesto fijado y con las funcionalidades definidas.

El centro de Calisoft¹ de la UCI hasta noviembre de 2009 realizó una serie de comprobaciones a los productos, los principales problemas detectados se manifestaron en la metodología de desarrollo, uso de planillas que no están vinculadas a ningún lineamiento de calidad, entregables sin fechas definidas y proyectos que no usan herramientas para la gestión de errores. Dicha problemática generaba un gran número de no conformidades en los proyectos que atentaba contra la entrega en fecha del producto al cliente, principalmente porque no se concibe como parte del proceso de desarrollo de software de manera planificada, actividades relacionadas con la prevención y la evaluación de errores y fallas del producto de software en su ciclo de vida.

Para responder a esta situación el Centro de Telemática (TLM) de la Universidad de las Ciencias Informáticas, donde se enmarcará esta investigación, tiene definido un procedimiento para calcular los costos de calidad de los proyectos de investigación y desarrollo. Este procedimiento permite tener una visión objetiva sobre el desempeño del proyecto en cuanto a la Gestión de la Calidad de manera que se contribuya a reducir el tiempo invertido en corregir fallas internas y externas detectadas en los diferentes momentos destinados a evaluar el software. Por otra parte, el mismo se realiza de forma manual, lo que conlleva a grandes pérdidas de tiempo al realizar los cálculos que este posee y da margen a que se cometan errores derivados del factor humano en la aplicación del procedimiento. Además, este no tiene precisadas las acciones concretas a considerar que permitan mejorar y disminuir los costos de calidad en caso de que los resultados al aplicar el mismo sean desfavorables, por tanto, él por sí solo no responde la necesidad de disminuir dichos costos, sino que brinda un indicador que permite conocer el estado de estos.

Por todo lo anteriormente planteado el **Problema científico** que se presenta es el siguiente: ¿Cómo optimizar el procedimiento para determinar y disminuir los costos de la calidad en los proyectos del Centro de Telemática?

¹ Calisoft: Centro de Calidad de la UCI para soluciones tecnológicas, que tiene la misión de garantizar el crecimiento continuo de una producción de software con calidad en la organización.

Para dar respuesta a esta problemática se puede afirmar como **Objeto de estudio**: Los costos totales de la calidad que intervienen en el proceso de desarrollo del software. Se ha definido como **Campo de acción**: Los costos totales de la calidad que intervienen en el proceso de desarrollo del software del Centro de Telemática, trazándose como **Objetivo general**:

- ✓ Diseñar un sistema que permita calcular los costos de calidad de los proyectos del Centro de Telemática.

En función de lograr el objetivo planteado se proponen como **Objetivos específicos**:

- ✓ Proponer medidas y acciones que permitan disminuir los costos de calidad en los proyectos del Centro de Telemática.
- ✓ Diseñar un sistema que permita informatizar el cálculo de costos de calidad en los proyectos del Centro de Telemática.

Siendo las **Tareas de la investigación** para dar cumplimiento a los objetivos específicos las siguientes:

- ✓ Caracterización de los proyectos productivos en el Centro de Telemática.
- ✓ Caracterización de los costos de calidad en los proyectos productivos.
- ✓ Análisis del procedimiento que se utiliza de forma manual para los cálculos de costos de calidad en los proyectos.
- ✓ Análisis de indicadores que permitan conocer el estado de los costos.
- ✓ Propuesta de acciones concretas que deban ser implementadas para mejorar los costos de calidad una vez que se indique que estos son desfavorables.
- ✓ Análisis de las herramientas y tecnologías a utilizar en el desarrollo de la aplicación.
- ✓ Selección de la metodología de desarrollo del software que permita guiar el proceso de confección del producto.
- ✓ Modelación de los procesos que se realizan para calcular los costos de calidad.
- ✓ Especificación de los requisitos que se requieren para la correcta confección de la propuesta.
- ✓ Diseño de una aplicación que permita calcular los costos de calidad en los proyectos del Centro de Telemática.

- ✓ Diseñar casos de pruebas para la realización de pruebas a la aplicación.

La **Idea a defender** de la investigación es: Con la informatización del procedimiento que permite calcular los costos totales de la calidad y la propuesta de medidas y acciones a tomar para disminuir estos, se optimizará dicho procedimiento, lo cual permitirá minimizar la deficiente gestión de la calidad que existe en la ejecución del proceso de desarrollo del software y reducir el tiempo invertido en corregir fallas internas y externas detectadas en los diferentes momentos destinados a evaluar el producto.

Para la realización de las tareas antes mencionadas se emplearon los siguientes **Métodos Científicos**:

- ✓ **Métodos teóricos**:
 - El método **analítico-sintético** para obtener de las consultas bibliográficas los conocimientos más actuales del problema a resolver.
 - El método **inductivo-deductivo**, para a través de un razonamiento llegar a un grupo de conocimientos particulares y generales.

El presente trabajo de diploma ha sido organizado en 4 capítulos, a continuación se muestra una breve descripción de cada uno de ellos: **Capítulo 1. “Fundamentación teórica”**: Este capítulo contendrá todo el fundamento teórico de la investigación. En el mismo se mostrarán los principales conceptos y términos a tratar a lo largo del documento relacionados con el tema en cuestión. **Capítulo 2. “Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática. Características del Sistema”**: Este capítulo se desglosará en dos partes fundamentales: en la primera parte se expondrán las medidas y acciones propuestas para mejorar y disminuir los costos de calidad en el proceso de desarrollo de software que se desempeña en el Centro de Telemática de la UCI. En la segunda parte se identifican los principales conceptos mediante un Modelo de Dominio necesario para un buen entendimiento del sistema y los requisitos que debe cumplir el sistema una vez concluido. **Capítulo 3. “Diseño del Sistema”**: Este capítulo se enfoca en los diagramas de clases del diseño para cada caso de uso del sistema propios de la metodología seleccionada para el desarrollo del sistema. **Capítulo 4. “Pruebas”**: En este capítulo se describe la propuesta de diseño de Casos de Pruebas basados en Casos de Uso para la aplicación que se propone desarrollar.

Capítulo 1. Fundamentación Teórica

1.1 Introducción

Este capítulo contiene todo el fundamento teórico de la investigación. En el mismo se muestran los principales conceptos y términos a tratar a lo largo del documento relacionados con el tema en cuestión, los cuales están avalados por referencias actualizadas. Se hace además un análisis de las herramientas y tecnologías seleccionadas para el desarrollo del sistema que se propone implementar.

1.2 Calidad del Software

La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. Calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad [3]. Esta ha sido estudiada por varios expertos y algunos de ellos la definen como: “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” [4].

La calidad puede estar presente en todas las actividades del ciclo de vida de un proyecto de software, para ello, se hace imprescindible disponer de una serie de pilares sobre los que sustentan las actividades: infraestructura apropiada de soporte, un grupo de personas especializadas en esta disciplina y procesos alineados con los objetivos de negocio que permitan una mayor industrialización en el desarrollo y mantenimiento. Para lograr un mejor desarrollo del proyecto es importante controlar todas las fases y áreas de gestión del proyecto así como la implantación de metodologías y mejores prácticas que aseguren la correcta gestión de las mismas. Dentro de un proyecto productivo se puede gestionar la calidad de dos formas, mediante la calidad del proceso o del producto como muestra la Figura 1.

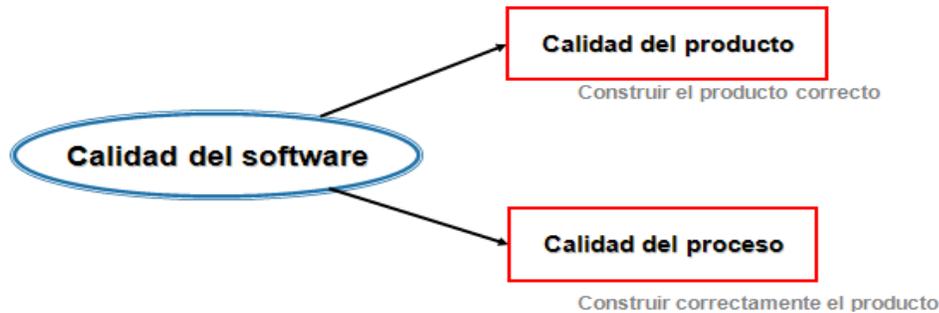


Fig.1. Estructura de la calidad de software.

Calidad del producto: Cuando se centra en el proceso de desarrollo de software y se hace una serie de pruebas en paralelo con cada etapa, para detectar y corregir los posibles defectos que puedan surgir [5].

Calidad del proceso: Cuando se centra en la gestión de todas las áreas de proceso de una organización, mediante la implantación de una metodología o estrategia. Así se consigue tener mayor información de los procesos de modo que puedan controlarse y mejorarse, y produzcan así un aumento de la calidad de los productos y servicios relacionados con ellos [5].

1.3 Gestión de la Calidad

La evolución más avanzada de los conceptos relacionados con la calidad conduce inevitablemente a la denominada Gestión de la Calidad Total o Gestión Estratégica de la Calidad, ambos conceptos representan las tendencias actuales que consideran a la calidad como parte integrante de la estrategia global de la empresa [6]. Es el modo en que la dirección planifica el futuro, implanta programas de actuación y controla los resultados con vistas a la mejora. El fundamento principal para la Gestión de la Calidad es conseguir que la calidad realizada, la calidad programada y la calidad necesaria coincidan entre sí [7]. Si se desea producir una buena calidad para el consumidor, es necesario decidir por adelantado cuál es la calidad de diseño, la calidad de fabricación y la calidad que desea el cliente. Para ello se deben tener en cuenta los cuatro aspectos de la calidad y planificarla siguiendo el Ciclo de Deming [6]. Ver Figura 2.

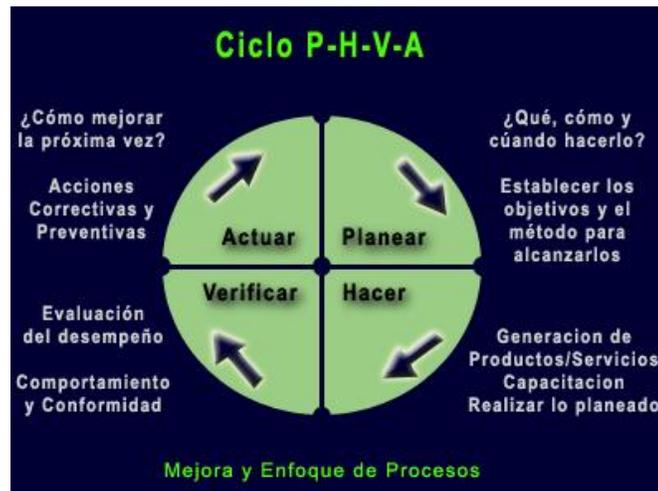


Fig. 2. Ciclo de Deming.

Asimismo, la Gestión de la Calidad se puede entender como el conjunto de actividades y medios necesarios para definir e implantar un sistema de la calidad, por una parte, y responsabilizarse de su control, aseguramiento y mejora continua por otra. En este sentido, la gestión de la calidad en cualquier organización (y, por supuesto, en las dedicadas al desarrollo y mantenimiento de software) cuenta con dos niveles de trabajo [8]:

- ✓ **El nivel de entidad u organización**, donde se trata de crear y gestionar una infraestructura que fomente la calidad de los productos software mediante la adecuación y mejora de las actividades y procesos involucrados en su producción e, incluso, en su comercialización y en la interacción con los clientes.
- ✓ **El nivel de proyecto**, donde las guías que la infraestructura organizativa prevé para las distintas actividades de desarrollo y mantenimiento de software deben ser adaptadas a las características concretas del proyecto y de su entorno para ser aplicadas en la práctica.

En la actualidad al reflexionar sobre la gestión de la calidad, se puede observar que una de las mayores debilidades que presentan muchas empresas es la poca atención a la importancia, alcance y repercusión que genera el no identificarse con la relevancia del mejoramiento continuo. Este constituye un aspecto significativo que no debe ser descuidado hoy en día, debido a que solo triunfan las empresas que logran ser cada día más competitivas, eficientes y para ello logran identificarse plenamente con la calidad total y lo que el mejoramiento continuo ofrece.

El mejoramiento continuo permite visualizar un horizonte más amplio, donde se buscará siempre la excelencia y la innovación que llevarán a los empresarios a aumentar su competitividad, disminuir los costos, orientando los esfuerzos a satisfacer las necesidades y expectativas de los clientes. [9].

1.3.1 Mejoramiento Continuo

El mejoramiento continuo es un principio básico de la gestión de la calidad, donde la mejora continua debe ser un objetivo imprescindible de la organización para incrementar la ventaja competitiva a través del perfeccionamiento de las capacidades organizativas. Es un proceso que describe la esencia de la calidad y refleja lo que las empresas necesitan hacer si quieren ser competitivas a lo largo del tiempo [10]. Ver Figura 3.



Fig.3. Ciclo de Mejoramiento Continuo.

Para llevar a cabo este proceso de mejoramiento continuo tanto en un departamento determinado como en toda la empresa, se debe tomar en consideración que dicho proceso debe ser: económico, es decir, debe requerir menos esfuerzo que el beneficio que aporta; y acumulativo, que la mejora que se haga permita abrir las posibilidades de sucesivas mejoras a la vez que se garantice el cabal aprovechamiento del nuevo nivel de desempeño logrado [9].

Hoy en día ha aumentado de forma significativa y como factor estratégico de primer orden, la exigencia por obtener una sustancial mejora de la calidad en los programas informáticos, y por consiguiente, ha aumentado la complejidad en el desarrollo de los sistemas automatizados de información, debido a esto, en la mayoría de los casos resulta engorroso generar productos que cumplan exactamente con los requerimientos y expectativas del cliente. Para responder a esta situación, muchos son los modelos, estándares, metodologías y filosofías que existen en la actualidad para la industria del software; con el propósito de garantizar y mejorar la calidad del producto de software y del proceso de desarrollo.

Dentro del listado se encuentran la norma ISO 9000:2000, la ISO/IEC TR 15504, el modelo CMMI [11]. Por otra parte también, se cuenta con la norma ISO/IEC 9003:2006 Ingeniería de Software, Directivas para la aplicación de la ISO 9001:2000 al software de computación como guía a las organizaciones de software.

La UCI optó por seleccionar el modelo CMMI, el cual permite determinar la capacidad de las organizaciones de desarrollo de software para producir de manera consistente y predecible productos de calidad superior. El modelo brinda guías para seleccionar estrategias de mejoramiento del proceso mediante la determinación de las capacidades actuales del proceso y la identificación de los puntos críticos para mejorar el proceso y la calidad del software [12].

1.4 Modelo de Madurez de Capacidad Integrada

El Modelo de Madurez de Capacidad Integrada (del inglés Capability Maturity Model, CMMI), es un modelo para la mejora o evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. Uno de sus principales principios es que la calidad de un producto o de un sistema es en su mayor parte consecuencia de la calidad de los procesos empleados en su desarrollo y mantenimiento. Este modelo para software establece 5 niveles de madurez para clasificar a las organizaciones, en función de qué áreas de procesos consiguen sus objetivos y se gestionan con principios de ingeniería. Para que una organización se encuentre en un determinado nivel es necesario cumplir con todas las actividades definidas para ese nivel y para los niveles anteriores.

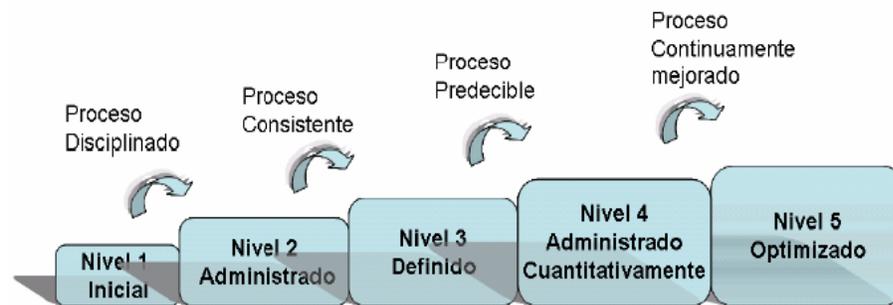


Fig.4. Niveles de CMMI.

En el **Nivel 2 Administrado**: Se normalizan las buenas prácticas en el desarrollo de proyectos (basándose en la experiencia y el método).

- ✓ En este nivel consolidado, las buenas prácticas se mantienen en los momentos de estrés.
- ✓ Están definidos los productos a realizar.
- ✓ Se definen hitos para la revisión de los productos.

Asimismo, en dicho nivel son establecidas las actividades básicas para la administración de proyectos de software para el seguimiento de costos, programación y funcionalidad.

CMMI es un modelo, una abstracción, el cual la UCI ha decidido institucionalizar el nivel 2 como vía de solución a sus necesidades, pero cabe destacar que nunca se va a encontrar en CMMI un tema específico como costos de calidad. Es decir, CMMI no tiene elementos específicos que indiquen cómo se mide o controla un elemento específico, solo le pregunta: ¿Qué le interesa medir y monitorear? Por tanto, las áreas del nivel 2 que se relacionan con los costos de calidad son **Monitoreo y Control del Proyecto (PMC)** y **Medición y Análisis (MA)** que indican que se deben medir y monitorear parámetros del proyecto (los que le sean útil al proyecto conocer, el modelo no especifica cuáles). Por otra parte, si para la organización o el proyecto le es importante conocer cuánto les está costando la falta de calidad o cuánto se están ahorrando en este tema y monitorear que se esté controlando este costo, entonces estas dos áreas ayudan a definir cómo mostrar este indicador, cómo recolectarlo, cómo mostrarlo o comunicarlo (reportes) y asegurar que se está recogiendo y mostrando (monitoreo).

El Monitoreo y Control de proyectos permite seguir el desempeño del proyecto en cada paso de su ejecución, de forma que se puedan identificar los posibles problemas oportunamente y adoptar las acciones correctivas que permitan mantener al proyecto dentro de los límites establecidos en la líneas base: Alcance, Costo, Tiempo y Calidad. Una apropiada visibilidad permitirá tomar acciones correctivas antes de que el trabajo real se desvíe mucho del plan, en fin, es controlar el cumplimiento del alcance, tiempo y costo. La Medición y Análisis permiten coleccionar datos y seleccionar indicadores que permitan medir la evolución de los procesos críticos del proyecto, comparar lo real contra lo planificado. En esta área de proceso se buscan maneras de conocer los puntos críticos donde están los mayores costos y actuar sobre ellos para reducirlos. Solo con métricas se puede determinar donde mejorar.

Se puede concluir que, CMMI no tiene un área de proceso específica que se encargue de los costos de calidad, pero al mismo tiempo todo tiene que ver desde el punto de vista que se mire. El aseguramiento de la calidad, el establecimiento de normas, estándares, procedimientos, la validación, la verificación entre otras actividades contribuyen al costo de la calidad.

Los Costos de la Calidad, son los costos asociados con la obtención, identificación, reparación y prevención de fallas o defectos [13].

1.5 Costos Totales de la Calidad y su clasificación

Actualmente, se entienden como costos de calidad aquellos incurridos en el diseño, implementación, operación y mantenimiento de los sistemas de calidad de una organización. Se define además como aquellos costos de la organización comprometidos en los procesos de mejoramiento continuo de la calidad, y los costos de sistemas, productos y servicios frustrados o que han fracasado al no tener en el mercado el éxito que se esperaba. El costo de la calidad tiene dos componentes, lo que invertimos en obtener buena calidad y lo que pagamos por no lograrla. El primer componente es decidido y controlado por la institución; el segundo se manifiesta en las fallas del producto. Invertimos en tener buena calidad mediante prevención (evitar errores) y evaluación (verificar que no tenemos errores). Por otro lado, las fallas pueden ser de dos tipos: internas (las que encuentran los desarrolladores) y externas (las que encuentran los clientes), al costo resultante de sumar todas las categorías se le conoce como Costos Totales de la Calidad [14]. Ver Figura 5.

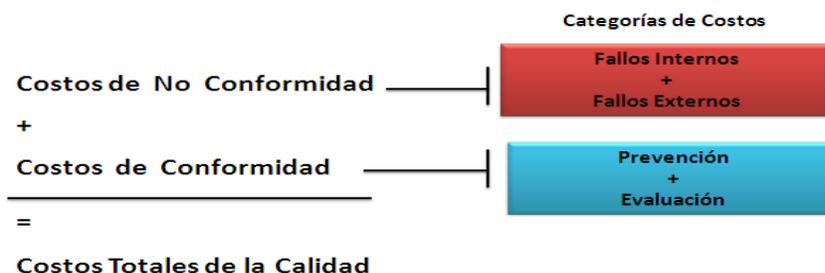


Fig. 5. Costos Totales de la Calidad.

El costo de la calidad no es exclusivamente una medida absoluta del desempeño, su importancia consiste en que brindan un indicador importantísimo de donde será más provechosa aplicar acciones correctivas apropiadas para reducirlos. En este sentido, varios estudios, autores y empresas señalan que los costos de calidad representan alrededor del 5 al 25 % sobre las ventas anuales. Estos costos varían según sea el tipo de industria, circunstancias en que se encuentre el negocio o servicio, la visión que tenga la organización acerca de los costos relativos a la calidad, su grado de avance en calidad total, así como las experiencias en mejoramiento de procesos.

Por otra parte, el costo de la calidad sirve para comprender mejor el funcionamiento de los procesos, identificar y priorizar oportunidades de mejora en los mismos, estimar el impacto monetario de dichas mejoras y hacer un seguimiento de los beneficios reales de dichas mejoras. Se trata, pues, de una potente herramienta para la mejora continua en la empresa. En contraste, reducir los costos de la calidad significa eliminar o reducir en parte el trabajo de la función calidad, pero esta reducción de costo se basa en una mejora de la calidad que sirve para añadir valor a nuestros procesos y para aumentar la satisfacción del cliente [15].

Los Costos de Calidad pueden clasificarse en cuatro categorías: costos de prevención, costos de evaluación, costos de fallas internas y costos de fallas externas. A continuación se describe cada una de ellas, así como, las actividades que incurren dentro de cada categoría de costo:

Costos de prevención: Aquellos en los que se incurre buscando que la fabricación de productos esté apegada a las especificaciones. Representan el costo de todas las actividades llevadas a cabo para evitar

defectos en el diseño, desarrollo y en todos aquellos aspectos que tienen que ver desde el inicio y diseño de un producto o servicio hasta su comercialización [13].

- ✓ Revisión del diseño, de los planes y de las especificaciones.
- ✓ Calificación del producto.
- ✓ Orientación de la ingeniería en función de la calidad.
- ✓ Programas y planes de aseguramiento de la calidad.
- ✓ Evaluación y capacitación a proveedores sobre calidad.
- ✓ Entrenamiento y capacitación para la operación con calidad.

Costos de evaluación: En esencia son los costos vinculados a las mediciones, evaluaciones o auditorías de productos o servicios en la búsqueda y detección de imperfecciones en los mismos para determinar el cumplimiento de los requisitos establecidos.

- ✓ Inspección y prueba de prototipos.
- ✓ Análisis del cumplimiento de las especificaciones.
- ✓ Inspecciones y pruebas de aceptación y recepción de los productos.
- ✓ Control del proceso e inspección de embarque.

Costos de fallas internas: Defectos (no conformidades) detectados antes de la entrega o expedición del producto o servicio al cliente [16].

- ✓ Retrabajos por inspecciones internas.
- ✓ Reinspección y ensayos.
- ✓ Rediseños.
- ✓ Inspección 100% por rechazos del muestreo.
- ✓ Pérdidas evitables de producción.
- ✓ Reclasificación de productos.
- ✓ Reparación de equipos.
- ✓ Modificación de precios (disminución de valor).
- ✓ Costos adicionales de manejo de documentación e inventarios.

Costos de fallas externas: Defectos detectados luego de la entrega o expedición del producto o servicio al cliente.

- ✓ Reclamos.
- ✓ Devoluciones y retiro de productos.

- ✓ Garantías.
- ✓ Concesiones, rebajas de precios (por causa de la calidad).
- ✓ Daños y perjuicios.
- ✓ Pérdida de ventas por devoluciones [16].

Los Costos de la Calidad representan un elemento clave en el funcionamiento exitoso de una empresa, debido a que sus beneficios podrán incrementarse en la misma medida en que disminuyen los costos.

1.5.1 Procedimiento para determinar los Costos Totales de la Calidad en los proyectos de software del Centro de Telemática

El cálculo de los Costos Totales de la Calidad representa una herramienta para los proyectos, con el propósito fundamental de mejorar la gestión de la calidad y por consiguiente reducir los costos de la calidad en los proyectos, lo cual contribuye a la toma de decisiones acertadas. Cada procedimiento para determinar costos de la calidad va a ser específico de la organización que lo implante adecuándose a sus necesidades.

Con el propósito de posteriormente proponer acciones y medidas a tomar para disminuir y mejorar los costos de calidad dentro de un proyecto de software, se realizó un estudio de la investigación titulada: “Propuesta de Procedimiento para determinar los Costos Totales de la Calidad en los proyectos de software del Centro de Telemática de la UCI” de los autores Yeisis Pérez y Jorge Félix Jiménez, investigación tomada como cimiento fundamental para el desarrollo del presente trabajo de diploma. El procedimiento existente para el cálculo de los costos de calidad cuenta con cinco etapas, las cuales se muestran a continuación:

Etapas 1: Diagnóstico de la situación actual del proyecto.

Etapas 2: Identificación y clasificación de los Costos Totales de la Calidad.

Etapas 3: Análisis y cálculo de los Costos Totales de la Calidad.

Etapas 4: Reporte de mejoramiento.

Etapas 5: Presentación de los resultados de los Costos Totales de la Calidad a la dirección del proyecto.

En la **Etapas 3** es donde se procede al cálculo de los Costos Totales de la Calidad por cada una de las categorías de costos antes mencionadas. Con las actividades por cada categoría de costos, se procede a realizar los cálculos de los costos de la calidad y los de la mala calidad. Dentro de los cálculos de la calidad se determinará el Esfuerzo Absoluto de la Calidad y el Esfuerzo Relativo de la Calidad, para ello

se utilizan métricas. Una métrica es una forma de medir, una escala, definida para realizar mediciones de uno o varios atributos.

Los cálculos que incluye esta etapa por cada categoría de costos, más los costos de calidad y mala calidad, son los cálculos a informatizar con el sistema que se propone desarrollar.

En este caso para calcular estos últimos se utilizan las siguientes métricas:

Donde:

P: Prevención, **E:** Evaluación, **Fi:** Fallas internas, **Fe:** Fallas externas, **C:** Creación.

Estos costos se expresan en horas para poderlos comparar fundamentalmente en el tiempo y analizar tendencias.

Esfuerzo de Calidad (EoQ)

$$EoQ = (P + E + Fi + Fe)$$

El Costo Absoluto de la Calidad o Esfuerzo de la Calidad (**EoQ**) no es más que la sumatoria de los costos de las actividades de la calidad.

Costo de Calidad (CoQ)

$$CoQ = (P + E + Fi + Fe) / (P + E + Fi + Fe + C)$$

El costo relativo de la calidad o costo de calidad (**CoQ**) no es más que el porcentaje que representa este costo absoluto de la calidad del total de horas dedicadas a la calidad, más las horas dedicadas a la creación y es aquí donde entran los costos por creación. Los costos de creación son aquellas actividades de creación que aportan valor al producto por ejemplo: Entrevistas con el cliente que aporten valor al producto, levantamiento de requisitos y actividades de implementación.

Esfuerzo de Mala Calidad (EoPQ)

$$EoPQ = Fi + Fe$$

El esfuerzo de la mala calidad (**EoPQ**) no es más que la sumatoria de los costos de las actividades incurridas por costos de fallas internas y externas.

Costo de Mala Calidad (CoPQ)

$$\text{CoPQ} = (F_i + F_e) / (P + E + F_i + F_e + C)$$

El costo de la mala calidad (**CoPQ**) no es más que el porcentaje que representa el esfuerzo de la mala calidad del total de horas dedicadas al esfuerzo de la calidad más los costos de creación.

Una vez calculados los costos por cada una de las categorías correspondientes y los costos de calidad y mala calidad, se procede a la **Etapa 4** del procedimiento. En dicha etapa se emite un reporte a la alta dirección del proyecto que encierra un resumen de los valores por cada categoría de costos, indicando donde se encuentran los mayores problemas sobre los que se debe actuar al hacer un análisis de los mismos en correspondencia con las características del proyecto.

Posteriormente se lleva a cabo la **Etapa 5** donde se realiza un análisis de los resultados o indicadores obtenidos en el reporte, permitiéndole a la alta dirección del proyecto identificar y evaluar los puntos críticos donde son más elevados los costos tanto de calidad como de mala calidad, a fin de poder actuar sobre aquellos que más le impacten y se puedan reducir sustancialmente. Por tanto, es indispensable realizar un análisis de dichos indicadores.

1.5.1.1 Análisis de indicadores que permiten conocer el estado de los costos

Una vez que los costos de calidad han sido identificados y estructurados, es necesario analizarlos con la intención de que sirvan de base para el desarrollo de acciones futuras. El propósito principal de este análisis es determinar lo que cuesta mantener determinado nivel de calidad. Tal acción es necesaria para informar a la gerencia cómo está funcionando el programa de gestión de calidad y poder ayudar a identificar las oportunidades que permitan mejorarla, así como reducir los costos.

El análisis de costos de calidad puede ser utilizado de diversas formas para tomar decisiones, se puede usar como herramienta de medición del informe de funcionamiento en un sentido estrictamente contable, también para planear y presupuestar, para el establecimiento de estrategias de competencia o para evaluar metas estratégicas, sin embargo, la aplicación más importante de este análisis es identificar problemas de calidad y utilizar los resultados para convencer a la gerencia que se necesitan y se justifican

los cambios. De tal manera que, los informes que arrojan estos análisis constituyen un indicador importantísimo para evaluar el progreso de los programas de mejoramiento de la calidad.

Dentro de los indicadores anteriormente mencionados se encuentran: **Costo de Calidad (CoQ) y Costo de Mala Calidad (CoPQ)**. Estos constituyen indicadores que muestran el comportamiento, magnitud en % y estado de los costos de calidad en un proyecto de software, teniendo en cuenta las características y particularidades del proyecto en cuestión, a fin de saber dónde son más importantes las acciones correctivas y se actúe sobre ellos lo antes posible en caso de resultar desfavorables para la organización. En el procedimiento especificado en el epígrafe anterior, una vez que se obtenga el reporte de los resultados de los cálculos, se procede a realizar la interpretación de los mismos. A continuación se muestra dicha interpretación por cada uno de los indicadores:

Interpretación del CoQ.

- Si el valor de CoQ es $\geq 50\%$, los Costos de la Calidad se consideran No Adecuado.
- Si el valor de $25\% \leq \text{CoQ} < 50\%$, los Costos de la Calidad se consideran Adecuado.
- Si $\text{CoQ} < 25\%$ los Costos de la Calidad se consideran Bastante Adecuado.

Interpretación del CoPQ.

- Si el $\text{CoPQ} \leq 5\%$, los Costos de la Mala Calidad se consideran Adecuado.
- Si el $\text{CoPQ} > 5\%$, los Costos de la Mala Calidad se consideran No Adecuado.

Una vez que se realice el análisis de estos indicadores luego de aplicado el procedimiento a un proyecto de software se tendrá una visión sobre los principales problemas identificados, de tal manera que facilite a la alta dirección del proyecto la decisión y la programación de las acciones de mejoras para reducirlos y/o eliminarlos, con el fin de mejorar la gestión de la calidad del proyecto y con ello incrementar la productividad y competitividad de la organización.

1.6 Metodología de Desarrollo

1.6.1 Metodología de Desarrollo de Software

Una metodología de desarrollo de software es un conjunto de técnicas, herramientas, procedimientos y soporte documental que permite a los desarrolladores definir los elementos necesarios para la

construcción de un nuevo producto de software. Es la que durante el proceso de desarrollo del software define “quién está haciendo qué, cuándo y cómo para alcanzar un determinado objetivo” [17].

1.6.1.1 Proceso Unificado de Desarrollo de Software

El Proceso Unificado de Software (RUP) no es más que un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones y diferentes tamaños de proyectos. Su objetivo es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible.

Para desarrollar la propuesta que presenta este trabajo, se ha decidido utilizar RUP como metodología de desarrollo, puesto que puede adaptarse tanto a proyectos grandes como a pequeños y no necesita de la interacción física del cliente para entregar un producto con calidad, costes y en el tiempo requerido. Además el uso de esta metodología asegura que se produzca desde sus primeras fases de desarrollo, un producto de calidad que cumpla con las características de funcionalidad, usabilidad y fiabilidad, mitigando desde horas tempranas los posibles riesgos que puedan aparecer durante el proceso de desarrollo del software, agregar además que debido a la estandarización de RUP como guía en el proceso de desarrollo de software, práctica a la que se suma nuestro país, se decide considerarlo como la metodología a utilizar. Como parte del proceso de desarrollo utilizando esta metodología se hace necesario el uso de un lenguaje de modelado para la modelación de los artefactos generados, en este caso se emplea el lenguaje UML, utilizado por RUP.

1.7 Lenguaje de Modelado UML

UML (por sus siglas en inglés *Unified Modeling Language*) se selecciona como lenguaje de modelado del sistema propuesto para modelar, construir, visualizar y documentar los elementos que lo conforman. Una particularidad de UML es que describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema. Ha sido adoptado por muchos proveedores de herramientas CASE y es el lenguaje de modelado soportado por RUP, metodología escogida para el desarrollo del sistema [18].

1.8 Herramienta CASE

Las herramientas CASE (Computer Aided Software Engineering), son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del

ciclo de vida de desarrollo de un software, permiten incrementar la productividad y el control de la calidad en cualquier proceso de elaboración de software, pues transforman la actividad de desarrollar software en un proceso automatizado. Facilitan la realización de prototipos, el desarrollo conjunto de aplicaciones y la reutilización de componentes de software [19]

1.8.1 Visual Paradigm

Visual Paradigm es la herramienta CASE utilizada para el modelado de artefactos del sistema que se propone desarrollar, debido a que es multiplataforma (incluyendo Linux), amigable en su uso y soporta además ingeniería directa e inversa, todos los diagramas UML y entidad-relación y el ciclo de vida del desarrollo de software completo: análisis, diseño, implementación, pruebas y despliegue.

Cabe destacar su robustez, usabilidad y portabilidad [20].

1.9 Herramientas de desarrollo y tecnologías

Para el sistema que se propone desarrollar se decide adoptar como estandarización de la universidad para conseguir aplicaciones Web más confiables, el marco de trabajo Sauxe. El mismo ha sido definido por el proyecto ERP-Cuba² y constituye un paquete tecnológico para el desarrollo de soluciones. A continuación, de las herramientas que incluye el Sauxe se detallan las utilizadas en el presente trabajo de diploma que ayudarán al cumplimiento del objetivo general del mismo.

1.9.1 Marcos de trabajo

ExtJS Framework: ExtJS es un framework JavaScript utilizado para crear las interfaces de usuarios agradables del sistema propuesto. El mismo se utiliza en la capa de presentación del marco de trabajo Sauxe. Posee una gran cantidad de componentes para la realización de aplicaciones Web, entre los cuales se encuentran: los objetos de tipo Grid, Windows, Form, Panel, Menús entre otros. ExtJS posibilita

² ERP-Cuba: es un proyecto productivo de la Universidad de Ciencias Informáticas, dedicado a la producción de software de gestión para las empresas cubanas. El mismo consta de un equipo multidisciplinario integrado por estudiantes, profesores y otros profesionales que trabajan en su conjunto para la obtención del producto final (CedruX).

un intercambio de datos con el servidor Web a través de la Tecnología Ajax³ utilizando JSON⁴ para la transmisión de datos, logrando una mayor velocidad en dicho proceso.

UCID Framework: UCID Framework abarca la integración del Framework ExtJS con el sistema, incluyendo el integrador de interfaz, el generador de interfaz dinámica y la impresión de documentos, integra la iconografía, los diferentes temas de escritorio de una aplicación y el multilinguaje. Es utilizado en la capa de presentación del marco de trabajo Sauxe.

Zend Framework: Zend Framework se trata de un framework adoptado por el proyecto ERP para el desarrollo de aplicaciones Web y servicios Web con PHP, el mismo brinda soluciones para construir sitios Web modernos, robustos y seguros. Además es Open Source y trabaja con PHP 5. Zend Framework es un marco de desarrollo libre, orientado a objetos que brinda una arquitectura flexible. Provee un núcleo para el manejo del patrón de diseño MVC⁵. Dentro del marco de trabajo Sauxe el Modelo Vista Controlador se utiliza para integrar la capa de presentación con las capas de negocio y servicio y utiliza entre otras tecnologías el Zend Framework.

Doctrine Framework: Framework utilizado para el desarrollo de aplicaciones PHP 5.2.3 o superior que utilicen bases de datos, este framework implementa el patrón ORM⁶ (siglas en inglés de “mapeador

³ Acrónimo de *Asynchronous JavaScript And XML (JavaScript asíncrono y XML)*, es una técnica de desarrollo web para crear aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

⁴ Acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos.

⁵ MVC (Modelo Vista Controlador): es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

⁶ ORM (del inglés, Object-Relational Mapping): es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional. (Chidamber, 1994).

relacional de objetos”) para desarrollar el dominio de una aplicación y cuenta con un DBAL (siglas en inglés de “capa de abstracción de bases de datos”) incorporado que permite exportar una base de datos a sus clases correspondientes y viceversa, o sea, a partir de las clases creadas siguiendo las especificaciones de ORM, generar las tablas de la base de datos, cuenta además con un lenguaje de consulta propio que te abstrae del gestor que se está utilizando. Se utiliza en la capa de dominio y acceso a datos del marco de trabajo Sauxe.

1.9.2 Lenguajes de programación

PHP 5.2: Es un lenguaje interpretado de alto nivel, diseñado originalmente para la creación de páginas web dinámicas de manera rápida y fácil. Es un lenguaje de programación usado principalmente en interpretación del lado del servidor. Esta versión sigue mejorando el soporte para la Programación Orientada a Objeto así como los temas de seguridad, el rendimiento y manejo de excepciones [21].

JavaScript: Es un lenguaje de programación interactivo del lado del cliente que se utiliza principalmente para crear páginas Web dinámicas. Una página Web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario [22]. JavaScript es un lenguaje de programación interpretado, es decir, que no requiere compilación.

HTML: Es un Lenguaje de marcas de Hipertexto (Hipertexto Markup Language). Es muy sencillo y permite describir textos de forma estructurada y agradable, con enlaces, los cuales permiten conectar dos elementos entre sí o fuentes de información relacionadas. HTML es un sistema de etiquetas el cual está basado en especificar la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, entre otras.) así como los diferentes efectos que se quieren dar (cursiva, negrita o un gráfico determinado), los mismos indican cómo debe verse la información [21].

1.9.3 Servidor Web

WampServer: Wamp es una aplicación sobre Windows que dentro del marco de trabajo de Sauxe es utilizado para instalar automáticamente los servicios de Apache como servidor web y PHP como lenguaje

de programación También viene con SQLiteManager PHPMyAdmin para que administre sus bases de datos más fácilmente. WampServer tiene funcionalidades que lo hacen muy completo y fácil de usar:

- ✓ Gestionar sus servicios de Apache y MySQL.
- ✓ Instalar y cambiar de Apache, MySQL y PHP emisiones.
- ✓ Gestión de la configuración de sus servidores.
- ✓ Acceder a sus registros.
- ✓ Acceder a sus archivos de configuración.

1.9.4 Sistema gestor de Base de Datos

PostgreSQL 8.3: Es un poderoso sistema de gestión de base de datos relacional orientada a objetos de software libre. Se utiliza en la capa de datos del marco de trabajo Sauxe. Tiene una arquitectura probada por lo que ha ganado una sólida reputación para la fiabilidad, integridad y exactitud de los datos. Funciona en todos los principales sistemas operativos. Incluye los mejores tipos de datos. También soporta el almacenamiento de grandes objetos binarios, incluyendo imágenes, sonidos o vídeo [23].

Conclusiones parciales

En este capítulo se analizaron conceptos significativos, necesarios para la realización adecuada del presente trabajo de diploma, todo lo relacionado con los costos totales de la calidad que indican dónde será más provechosa una acción correctiva para una empresa y el mejoramiento continuo como la base para preservar la esencia de la gestión de la calidad en una organización. Se analizó el procedimiento existente que se utiliza de forma manual para los cálculos de costos de calidad en los proyectos del Centro de Telemática, así como los indicadores que permiten conocer el estado de los costos para posteriormente proponer acciones y medidas a tomar para disminuir y mejorar dichos costos. Por otra parte, se seleccionaron las herramientas y tecnologías adecuadas para el desarrollo del sistema que se propone.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática. Características del Sistema.

2.1 Introducción

En el presente capítulo se exponen las acciones propuestas para mejorar y disminuir los costos de calidad en el proceso de desarrollo de software que se desempeña en el Centro de Telemática de la UCI. Se abordan las principales características del sistema, se describen los procesos básicos que influyen en el objeto de estudio y hacen válido el problema a resolver anteriormente puntualizado. Se identifican los principales conceptos mediante un Modelo de Dominio necesario para un buen entendimiento del sistema y los requisitos que debe cumplir el sistema una vez concluido.

2.2 Centro de Telemática de la UCI

Con la informatización del procedimiento que permite calcular los costos totales de la calidad y la propuesta de medidas y acciones a tomar para disminuir estos, el Centro de Telemática de la UCI logrará una mejora en la gestión de la calidad y con la misma una optimización de los costos en los que incurren sus proyectos. El Centro tiene como misión ser un centro de desarrollo de sistemas y servicios informáticos en las ramas de las Telecomunicaciones y la Seguridad Informática [24].

A su vez tiene como visión ser:

- ✓ Centro de referencia nacional y latinoamericano que integra la producción e investigación en el desarrollo de sus productos y servicios.
- ✓ Desarrolla proyectos de Investigación más desarrollo (I+D) con entidades y organismos nacionales e internacionales.
- ✓ Desarrolla programas de maestrías y doctorados en el campo de las Telecomunicaciones y la Seguridad informática.
- ✓ Se asocia a empresas líderes en las Telecomunicaciones y la Seguridad Informática y amplía sus mercados.
- ✓ Desarrolla servicios y/o productos horizontales para otros centros de desarrollo.
- ✓ Desarrolla productos y servicios que posibilitan un ingreso importante a la economía nacional.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- ✓ Cuenta con todos sus procesos debidamente definidos e implementados.

El Centro de Telemática de la UCI cuenta con diez proyectos y se divide en dos departamentos, según se muestra en el organigrama (**Anexo 1**).

1. Departamento de Telecomunicaciones, el cual tiene como cliente a la Empresa de Telecomunicaciones de Cuba SA (ETECSA) y la empresa española Teltronic, cuenta con los siguientes proyectos:

- Call Center.
- SGIC – PABX (Sistema de Gestión Integral de Costo para PABX).
- Cubacel.
- SER - WAP (Servidor de Aplicaciones WAP).
- Montetra.
- TIP (Tele Identificador Personal).

2. Departamento de Seguridad Informática el cual tiene como clientes a ETECSA, MININT, y la Contraloría General de la República, cuenta con los siguientes proyectos:

- SIAI (Sistema Integral de Análisis e Información).
- Audis BD (Auditoría de BD).
- Servicios Telemáticos.
- ICON (Informatización de la Contraloría General de la República).

La Propuesta de acciones para disminuir y mejorar los costos de calidad en los proyectos, una vez que sean tomadas en consideración y con una posterior integración al sistema que se propone desarrollar como base de conocimientos, permitirá mejorar la gestión de la calidad del Centro y así tributar al logro de la visión, al pretender ser un centro de referencia nacional y latinoamericano capaz de poder desarrollar productos y servicios con la calidad requerida que posibiliten un ingreso importante a la economía nacional; facilitando la inserción en nuevos mercados.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

2.3 Objeto de estudio

2.3.1 Problema y situación problemática

En el Centro de Telemática de la UCI al cual pertenece el presente trabajo de diploma, los costos totales de la calidad constituyen una poderosa herramienta para mejorar la gestión de la calidad del mismo. Recientemente se ha definido un procedimiento para calcular los costos de calidad de los proyectos de software, el cual permite tener una visión objetiva sobre el desempeño del proyecto en cuanto a la Gestión de la Calidad de manera que se contribuya a reducir el tiempo invertido en corregir fallas internas y externas detectadas en los diferentes momentos destinados a evaluar el software. El mismo se basa en el cálculo de las actividades por categorías de costos, teniendo en cuenta las características y particularidades del proyecto. Además, se dificulta la necesidad de disminuir dichos costos ya que éste solo brinda un indicador que permite conocer el estado de estos, es decir no tiene precisadas las acciones y medidas a tomar que permitan disminuir y mejorar los costos de calidad en caso de que los resultados al aplicar el mismo sean desfavorables. El hecho de no contar con un sistema que informatice el procedimiento para calcular los costos de calidad en los proyectos del Centro y la propuesta de acciones y medidas a tomar que permitan disminuir y mejorar estos, incide en la toma de decisiones erróneas en el proceso de desarrollo de software de un proyecto; esto a su vez puede repercutir en los niveles de motivación de los integrantes del mismo. Los cálculos de todas las actividades por cada categoría de costos, más los costos de calidad y mala calidad, son los cálculos a informatizar con el sistema que se propone desarrollar.

2.3.2 Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática

La inconformidad con el resultado de los costos de calidad en un proyecto lleva a considerar varias alternativas de tipo correctivas, las que buscan informar sobre las acciones que se deben tomar en el momento de tener un resultado desfavorable. A continuación se exponen acciones que se pueden considerar para mejorar la calidad en un proyecto de software.

Acciones para disminuir costos de prevención:

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- ✓ **En cuanto a las actividades relacionadas con la planeación de la calidad:** Desde un inicio, describir y documentar las actividades del proyecto, sus tareas asociadas y el propósito de cada una de ellas, así como su orden, identificando cuándo ocurren las revisiones técnicas, las auditorías, de acuerdo al calendario del proyecto e identificando los estándares que guiarán dichas actividades. Por otra parte, establecer y documentar los requisitos con el cliente para que sirva como base para el trabajo, gestionar los cambios que pueda realizar el cliente con el objetivo de determinar el impacto de dichos cambios e iniciar acciones apropiadas a tiempo. Es por ello que se debe mantener un seguimiento de los requisitos durante todo el proyecto.
- ✓ **En cuanto a las actividades relacionadas con el mejoramiento de la calidad:** Evaluar desde horas tempranas y de forma continua los sistemas utilizados que contribuyan al aseguramiento de la calidad y sus resultados. Asimismo, reorientar los esfuerzos de prevención, adoptando las acciones correctivas para eliminar las no conformidades que puedan detectarse en el momento de la evaluación. Esto conseguirá una potente mejora desde el inicio del proceso de desarrollo de software.
- ✓ **En cuanto al adiestramiento y capacitación del personal del proyecto:** Hacer énfasis en una adecuada capacitación del personal de manera que cada uno de los procesos que realiza sean eficaces y eficientes, logrando así disciplinar al personal en que el próximo paso del proceso es el cliente. Con considerable frecuencia los trabajadores han aprendido sus labores de otro trabajador que nunca fue entrenado apropiadamente, es poca o ninguna la capacitación que han recibido, por lo tanto se ven obligados a seguir instrucciones imposibles de entender o no saben si han hecho correctamente su trabajo y por consiguiente no pueden desempeñar eficientemente su trabajo debido a que ninguna persona le dice cómo hacerlo. Por ende, es necesario crear nuevos equipos de trabajo y emplear nuevos métodos de reentrenamientos.
- ✓ **En cuanto al mantenimiento preventivo de los equipos de procesos:** Determinar cuál o cuáles son los equipos más críticos del proyecto para realizar reemplazos, restauraciones, evaluaciones, inspecciones en un período de tiempo que no afecte el calendario del proyecto o el uso de los equipos.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- ✓ **En cuanto a la gestión de riesgos:** Identificar y reducir continuamente los riesgos del proyecto a través del ciclo de vida del proyecto, valorando la probabilidad de ocurrencia, el impacto, las causas, para establecer la prioridad con la cual emplear recursos para mitigar estos riesgos.

Acciones para disminuir costos de evaluación:

- ✓ **En cuanto a la auditoría de calidad del producto:** Detectar, prevenir y eliminar el uso excesivo de recursos destinados a esta actividad. Esta flexibilidad permite buscar nuevas oportunidades de mercado u optimizar los recursos para producir lo mismo a menor costo. Por otra parte, a medida que la prevención va mostrando resultados positivos, ir reduciendo paulatinamente los costos de evaluación. Es decir, en caso de poseer pocos problemas de calidad, es necesario hacer menos auditoría de calidad al producto, porque los mismos se están haciendo bien desde la primera vez, es decir, se obtienen pocos productos defectuosos. Esto trae consigo que se reduzcan los costos por fallas internas y externas, aparejado a esto las reparaciones, los retrabajos y la reinspección disminuyen.
- ✓ **En cuanto al refinamiento de los requisitos del producto:** Revisar los estándares de calidad para ver si son realistas en relación con la adecuación para el uso. Los costos de evaluación aseguran que los productos tengan conformidad con estándares de calidad y requerimientos de desempeño.
- ✓ **En cuanto a las pruebas realizadas al producto:** Reducir al mínimo las actividades de inspección, ensayos o pruebas realizadas al producto mediante una adecuada planeación de las mismas. Ciertamente la inspección reduce los fallos, pero excesivo trabajo de inspección puede disparar el costo total de la calidad. A fin de evitarlo, se recomienda siempre que sea posible, mantener el autocontrol del inspector, y en todo caso, evitar la inspección 100%.
- ✓ **En cuanto a la verificación del nivel por cada empleado:** Realizar estudios de precisión, para medir la eficiencia de los empleados. Asimismo, se pueden realizar muestreos de trabajo del personal para comprobar el conocimiento y habilidad alcanzada en el trabajo que realiza.
- ✓ **En cuanto a la inspección final del producto:** Mejorar los equipos y métodos de inspección, con el fin de reducir el costo.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- ✓ **En cuanto a revisiones de la documentación:** Lograr que el costo de revisión represente un valor mínimo y que la persona que ejecute esta actividad tenga habilidad en la misma.

Acciones para disminuir costos de fallas internas y externas:

- ✓ **En cuanto a los reprocesos:** Reducir los desperdicios y reprocesos originados por productos defectuosos. Por otra parte, mejorar la calidad de los procesos de trabajo en todo el ciclo de vida del proyecto, los costos provocados por fallas internas generan pérdidas de calidad en cualquier parte del proceso, falta de productividad y como consecuencia, también de competitividad al generar rechazos, retrabajos, reprocesos, riesgos, ideas no aprovechadas por falta de preparación, tecnología, etc. que también implican un costo de calidad, por tanto, todos los esfuerzos deben estar dirigidos a reducir estos reprocesos.
- ✓ **En cuanto a maquinaria mal mantenida:** Las máquinas en mal estado crean tiempos inactivos en el proyecto que dan lugar al incumplimiento de los plazos de entrega, defectos en el producto, gastos en reparación y disminución de la productividad y la calidad. Para disminuir estos tiempos, es importante disponer de un buen servicio de mantenimiento o incluso disponer de máquinas de repuesto si es posible.
- ✓ **En cuanto a insuficiente capacitación:** La carencia de experiencia en la actividad que se ha de desarrollar permite un elevadísimo riesgo para el proyecto. Carecer de experiencia constituye en sí la base fundamental de todas las demás causas que llevan al fracaso. Por esta razón, se propone establecer un plan de capacitación del personal, fijando responsables y fecha, de manera que se certifique su cumplimiento.
- ✓ **En cuanto a errores de diseño, impresión y procesamiento de datos:** Se deben considerar algunos aspectos como son: conseguir que todas las personas sean conscientes del problema y tengan el deseo de resolverlo, realizar la investigación de las causas del problema junto al personal afectado, planificar las acciones correctoras pertinentes, fijando plazos y responsables y realizar un seguimiento de las acciones correctoras adoptadas.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- ✓ **En cuanto a reclamaciones del cliente:** Lógicamente, cuando se está fabricando un producto, si un fallo se detecta en las primeras fases del ciclo de producción, el costo es menor que si se detecta ese fallo cuando ya se ha entregado el producto al cliente. Como consecuencia de esto se le debe dar tratamiento a las reclamaciones del mismo y a sus consecuencias, por ejemplo, los reembolsos al cliente o anulación de facturas, gastos por desplazamiento si es necesario visitar al cliente, etc. Por tanto, es significativo desde el inicio del proceso de desarrollo del software evitar errores, trazar las metas para alcanzar el producto deseado. Esto implica, trabajar conjuntamente para que todos asuman su parte de responsabilidad en la consecución del fin común de todos: la satisfacción final del cliente.
- ✓ **En cuanto a descuentos:** Asegurar que la entrega del producto esté libre de defectos, es decir, se encuentre en buen estado antes de la entrega al cliente, con el propósito de evitar hacer un descuento al mismo para compensarle por la entrega de un producto en mal estado. El valor de ese descuento es un costo absurdo.

2.3.3 Objeto de Automatización

Se quiere automatizar el cálculo de los costos de calidad en los proyectos del Centro de Telemática mediante una aplicación web. Mediante la aplicación se podrán realizar los cálculos correspondientes para cada una de las categorías de costos permitiéndole al encargado de realizar dichos cálculos obtener información consistente sobre el estado de los costos del proyecto y mostrar reportes de los mismos.

2.3.4 Información que se maneja

Se cuenta con amplia información sobre el procedimiento que se realiza de forma manual para el cálculo de los costos de calidad, detalles de cómo realizar el cálculo para cada categoría de costos dependiendo de las características y particularidades del proyecto.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

2.3.5 Propuesta del sistema

Se ha decidido darle solución al problema mediante una aplicación web que facilite el cálculo de los costos de calidad en los proyectos del Centro de Telemática, permitiendo mostrar un reporte de cada categoría de costos para un proyecto específico.

2.3.6 Modelo de dominio

Cuando se hace uso de la ingeniería de software y se alude al empleo de alguna metodología de desarrollo, en este caso RUP, en lo primero que se piensa es en realizar un Modelo de Negocio, pero para ello se hace necesario la identificación de los procesos del mismo, siendo este un conjunto de actividades que responden directamente a un proceso en sí. En el estudio realizado del problema que se plantea, no fue posible identificar claramente los procesos que se llevan a cabo en el negocio, dado el escaso balance de la estructura de estos, por tanto se decidió la realización de un Modelo de Dominio. El Modelo de Dominio ayuda a la interpretación precisa del sistema, posibilita obtener correctamente los requerimientos y entender el objeto de estudio para darle una solución adecuada al problema a resolver. En esencia y en el caso que se observa, el Modelo de Dominio tiene como objetivo entender y describir los objetos más importantes dentro del contexto del sistema y las relaciones entre ellos.

A continuación aparecen varios conceptos que serán de utilidad para entender la lógica del modelado.

Costos de calidad: Se entiende como costos de calidad, aquellos incurrido en el diseño, implementación, operación y mantenimiento de los sistemas de calidad de una organización.

Costos de prevención: Aquellos en el que se incurre buscando que la fabricación de productos esté apegada a las especificaciones.

Costos de evaluación: En esencia es el costo vinculado a las mediciones, evaluaciones o auditorías de productos o servicios en la búsqueda y detección de imperfecciones en los mismos para determinar el cumplimiento de los requisitos establecidos.

Costos de creación: Son aquellas actividades de creación que aportan valor al producto.

Costos de fallas internas: Defectos (no conformidades) detectados antes de la entrega o expedición del producto o servicio al cliente.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Costos de fallas externas: Defectos detectados luego de la entrega o expedición del producto o servicio al cliente.

Métrica: Es una forma de medir, una escala, definida para realizar mediciones de uno o varios atributos.

Costos de Calidad y Costos de Mala Calidad: Son indicadores que muestran el comportamiento, magnitud en % y estado de los costos de calidad en un proyecto de software.

Indicador: Puede definirse como “un instrumento de medida, cuantitativo o cualitativo, que refleja la cantidad de calidad que posee una actividad o un servicio cualquiera”.

Proyecto: Actividad que involucra a un grupo seleccionado de estudiantes y profesores con un objetivo único, el cual será la obtención de un producto de software final y al cual se le realiza el cálculo de los costos de calidad.

Reporte: Muestra el resultado de los costos de calidad de un proyecto por cada categoría, además de los costos de calidad y mala calidad.

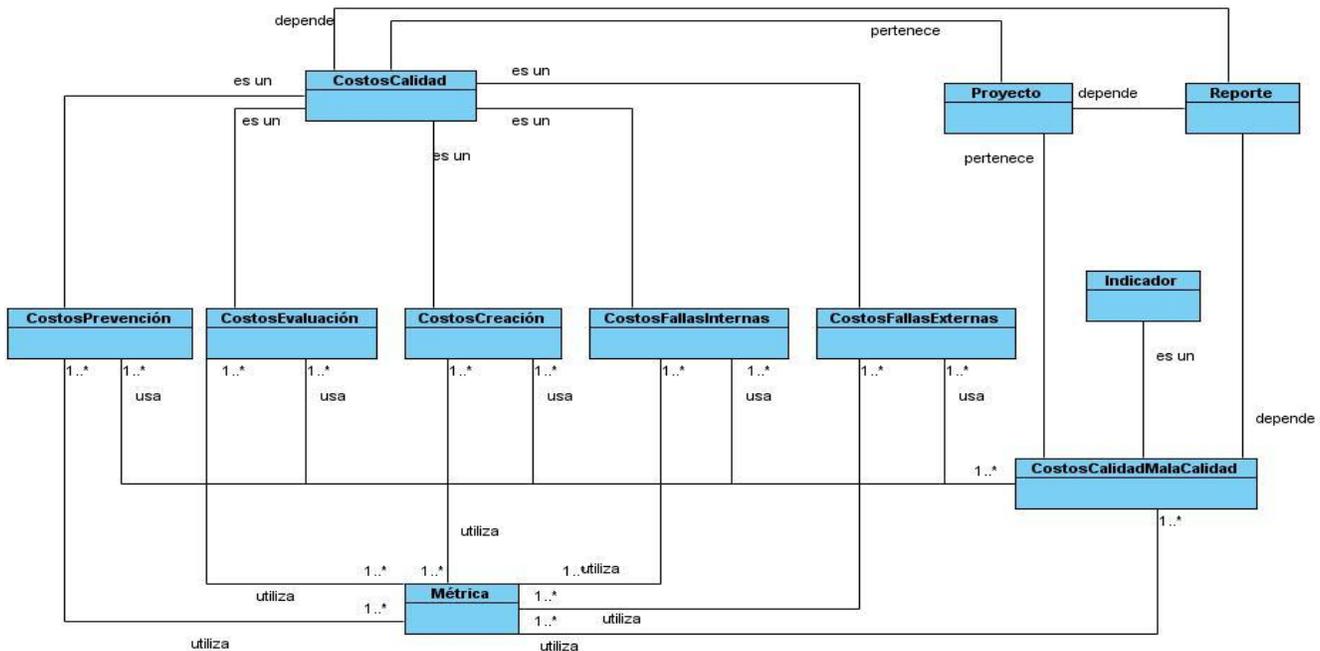


Fig. 7. Modelo de Dominio.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

2.3.7 Especificación de los requerimientos del software

Los requerimientos del software son las características y cualidades que el sistema debe tener. Estos se agrupan en dos grupos, los funcionales y los no funcionales. Una vez definidos e identificados los mismos, se puede transitar al modelamiento del sistema. Además es posible afirmar que es uno de los flujos más relevantes en la fase de inicio definida por RUP, ya que aporta la visión general de lo que se quiere hacer en el sistema.

Listado de los Requisitos Funcionales (RF)

Los requisitos funcionales se definen como las condiciones o capacidades que el sistema debe cumplir. Estos requisitos son actividades de la etapa 3 del procedimiento definido para el cálculo de los costos de calidad.

Se desea que el sistema permita:

RF 1. Registrar proyecto.

El sistema debe permitir al Administrador de la calidad registrar el proyecto al cual se le van a realizar los cálculos de costos de calidad, introduciendo el siguiente dato:

- Nombre del proyecto.

RF 2. Calcular el tiempo total por revisiones y documentación.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo total en realizar búsquedas con el propósito de documentarse respecto a lo que se pretende hacer, insertando los siguientes datos:

- TRDPP: Tiempo en revisiones y documentación por persona.
- (n): Cantidad de personas.

Datos a mostrar

- TTRD: Tiempo total por revisiones y documentación.

$$TTRD = \sum_{i=1}^n TRDPP$$

RF 3. Calcular el tiempo total por planeación de la calidad.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo empleado a planificar la calidad, comprende todas las actividades realizadas en el proyecto para planificar la misma, como por ejemplo, definición y establecimiento de políticas y objetivos de la calidad, planificación de actividades de aseguramiento de la calidad (revisiones, auditorias, pruebas), requisitos de la calidad relacionados con el producto, entre otras, insertando los siguientes datos:

- HDPPC: Horas dedicadas por el personal para planificar la calidad.
- (n): Cantidad de personas.

Datos a mostrar

- TTPPC: Tiempo total por planeación de la calidad.
$$TTPPC = \sum_{i=1}^n HDPPC$$

RF 4. Calcular el tiempo total por mejoramiento de la calidad.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo total de todas las actividades relacionadas con el mejoramiento de la calidad que el proyecto estime conveniente hacer, insertando los siguientes datos:

- TPPMC: Tiempo dedicado por persona al mejoramiento de la calidad.
- (n): Cantidad de personas.

Datos a mostrar

- TTMC: Tiempo total por mejoramiento de la calidad.
$$TTMC = \sum_{i=1}^n TPPMC$$

RF 5. Calcular el tiempo de adiestramiento y capacitación del personal del proyecto.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el total de tiempo invertido por el personal del proyecto en capacitar sus trabajadores para un mejor desempeño de sus responsabilidades, insertando los siguientes datos:

- PMC: Personal matriculado en curso.
- CHC: Cantidad de horas del curso.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- TEPC: Tiempo en elaboración de un plan de proyecto.
- TAH: Tiempo en adiestramiento de las herramientas.
- (n): Cantidad de cursos.

Datos a mostrar

TTAC: Tiempo total por mejoramiento de la calidad.
$$TTAC = \sum_{i=1}^n (PMC * CHC) + TEPC + TAH$$

RF 6. Calcular el tiempo total por planificación de medidas y programas de seguridad.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el total de tiempo invertido en medidas y programas de seguridad para lograr que el proyecto esté protegido, insertando los siguientes datos:

- TPP: Tiempo en planificar las medidas por persona.
- (n): Cantidad de personas.

Datos a mostrar

- TTMPs: Tiempo total por planificación de medidas y programas de seguridad.

$$TTMPs = \sum_{i=1}^n TPP$$

RF 7. Calcular el tiempo total por mantenimiento preventivo.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el total de tiempo empleado en mantener las máquinas en perfecto estado, así como también el tiempo invertido cuando alguna deja de funcionar, insertando los siguientes datos:

- TRPE: Tiempo en reparaciones por equipo.
- (n): Cantidad de equipos.

Datos a mostrar

TTMP: Tiempo total por mantenimiento preventivo.
$$TTMP = \sum_{i=1}^n TRPE$$

RF 8. Calcular el tiempo total por gestión de riesgos.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido por el proyecto en identificar y clasificar los riesgos, insertando los siguientes datos:

- TAGR: Tiempo en actividades para la gestión de riesgos.
- (n): Cantidad de actividades.

Datos a mostrar

- TTGR: Tiempo total por gestión de riesgos. $TTGR = \sum_{i=1}^n TAGR$

RF 9. Calcular el tiempo total de auditorías de calidad.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en realizar auditorías de calidad al producto, insertando los siguientes datos:

- TA: Tiempo de la auditoría.
- (n): Cantidad de auditorías.

Datos a mostrar

- TTAC: Tiempo total por auditorías de calidad. $TTAC = \sum_{i=1}^n TA$

RF 10. Calcular el tiempo total en revisiones de documentación.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en revisiones a documentos que tienen que ver con el tema que se trata en el proyecto, insertando los siguientes datos:

- TRD: Tiempo en revisiones y documentación.
- (n): cantidad de revisiones.

Datos a mostrar

- TTR: Tiempo total de revisiones. $TTR = \sum_{i=1}^n TRD$

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

RF 11. Calcular el tiempo total de pruebas al producto.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo incurrido en realizar pruebas al producto en cada fase de su desarrollo, insertando los siguientes datos:

- TP: Tiempo en planificación de prueba.
- TAP: Tiempo en aplicación de prueba.
- TVRP: Tiempo en verificar resultado de la prueba.
- (n): Cantidad de pruebas.

Datos a mostrar

- TTP: Tiempo total por prueba.

$$TTP = \sum_{i=1}^n TP + \sum_{i=1}^n TAP + \sum_{i=1}^n TVRP$$

RF 12. Calcular el tiempo total en verificar el nivel de cada empleado.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo incurrido en verificar el nivel de cada empleado, se realizaría esta actividad de ser necesaria por incumplimientos de los mismos en sus tareas, insertando los siguientes datos:

- TRPE: Tiempo en realizar pruebas al empleado.
- (n): Cantidad de pruebas a empleados.

Datos a mostrar

- TTNE: Tiempo total de nivelación del empleado.

$$TTNE = \sum_{i=1}^n TRPE$$

RF 13. Calcular el tiempo total de inspección del producto.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en una inspección final del producto ya sea al final de una fase o al terminar el producto completo, insertando los siguientes datos:

- TRIG: Tiempo en realizar la inspección general.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- (n): cantidad de personas que realizan esta operación.

Datos a mostrar

- TTIF: Tiempo total de la inspección final. $TTIF = \sum_{i=1}^n TRIG$

RF 14. Calcular el tiempo total de refinamientos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en realizar refinamientos, insertando los siguientes datos:

- TR: Tiempo en refinamientos.
- (n): Cantidad de refinamientos.

Datos a mostrar

- TTR: Tiempo total de refinamiento. $TTR = \sum_{i=1}^n TR$

RF 15. Calcular el tiempo total en reparaciones de máquinas.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en reparar máquinas que dejen de funcionar en el periodo del proyecto, insertando los siguientes datos:

- TRM: Tiempo en reparaciones de maquinarias.
- (n): Cantidad de máquinas.

Datos a mostrar

- TTRM: Tiempo total en reparaciones de maquinaria. $TTRM = \sum_{i=1}^n TRM$

RF 16. Calcular el tiempo total de capacitación.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en volver a capacitar un personal que ya fue capacitado, o que no se le capacitó de la manera debida, insertando los siguientes datos:

- TCC: Tiempo del curso de capacitación.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- MC: Matrícula del curso.
- (n): Cantidad de cursos.

Datos a mostrar

- TTIC: Tiempo total por insuficiente capacitación. $TTIC = \sum_{i=1}^n (TCC * MC)$

RF 17. Calcular el tiempo total por descuido de seguridad.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido por descuidos de seguridad ya sea del local o de los equipos de desarrollo, insertando los siguientes datos:

- TGS: Tiempo en garantizar la seguridad.
- (n): Cantidad de actividades.

Datos a mostrar

- TTDS: Tiempo total por descuido de seguridad. $TTDS = \sum_{i=1}^n TGS$

RF 18. Calcular el tiempo total por reproceso.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en realizar nuevamente un proceso que ya se realizó pero no tuvo la calidad requerida, insertando los siguientes datos:

- TER: Tiempo en eliminar reproceso.
- (n): Cantidad de reprocesos.

Datos a mostrar

- TTPR: Tiempo total por presencia de reproceso. $TTPR = \sum_{i=1}^n TER$

RF 19. Calcular el tiempo total en análisis de ausencias.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en analizar ausencias así como corregir las mismas, insertando los siguientes datos:

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- TCA: Tiempo en corregir la ausencia.
- (n): Cantidad de personas.

Datos a mostrar

- TTPA: Tiempo total por ausencias.

$$TTPA = \sum_{i=1}^n TCA$$

RF 20. Calcular el tiempo total invertido en corregir errores de diseño, impresión y procesamiento de datos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en corregir errores de diseño, impresión y procesamiento de datos que puedan presentarse en el transcurso del proyecto, insertando los siguientes datos:

- TCED: Tiempo en corregir errores de diseño.
- TCEI: Tiempo en corregir errores de impresión.
- TCEPD: Tiempo en corregir errores de procesamiento de datos.
- (n): Cantidad de errores.

Datos a mostrar

- TTEDIPD: Tiempo total por errores de diseño, impresión y procesamiento de datos.

$$TTEDIPD = \sum_{i=1}^n TCED + \sum_{i=1}^n TCEI + \sum_{i=1}^n TCEPD$$

RF 21. Calcular el tiempo total por cambio de ingeniería.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido al hacer algún cambio de ingeniería que no estaba previsto, insertando los siguientes datos:

- TCI: Tiempo en cambio de ingeniería.
- (n): Cantidad de cambios de ingeniería.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Datos a mostrar

- TTCI: Tiempo total por cambio de ingeniería.

$$TTCI = \sum_{i=1}^n TCI$$

RF 22. Calcular el tiempo total por reclamaciones del cliente.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo en reclamaciones que pueda hacer el cliente al no estar satisfecho con lo que se está haciendo o con el producto terminado, insertando los siguientes datos:

- TAR: Tiempo en atender la reclamación.
- (n): Cantidad de reclamaciones.

Datos a mostrar

TTRC: Tiempo total por reclamaciones del cliente.

$$TTRC = \sum_{i=1}^n TAR$$

RF 23. Calcular el tiempo total en materiales devueltos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en devoluciones de procesos por insatisfacciones, insertando los siguientes datos:

- TCMD: Tiempo en corregir el material devuelto.
- (n): Cantidad de correcciones al material devuelto.

Datos a mostrar

- TTCMD: Tiempo total en corregir el material devuelto.

$$TTCMD = \sum_{i=1}^n TCMD$$

RF 24. Calcular el tiempo total por descuentos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo por descuentos hechos por insatisfacciones, insertando los siguientes datos:

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- TRD: Tiempo en realizar descuento.
- (n): Cantidad de descuentos.

Datos a mostrar

- TTD: Tiempo total por descuento.

$$TTD = \sum_{i=1}^n TRD$$

RF 25. Calcular el tiempo total de demandas por incumplimientos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en demandas del cliente por insatisfacciones, insertando los siguientes datos:

- TADI: Tiempo en atender demandas por incumplimiento.
- (n): Cantidad de demandas.

Datos a mostrar

- TTDI: Tiempo total de demandas por incumplimiento.

$$TTDI = \sum_{i=1}^n TADI$$

RF 26. Calcular el tiempo total en garantías.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido en garantías, insertando los siguientes datos:

- TPG: Tiempo por garantías.
- (n): Cantidad de garantías.

Datos a mostrar

- TTG: Tiempo total por garantías.

$$TTG = \sum_{i=1}^n TPG$$

RF 27. Calcular el tiempo total por entrevistas.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido por entrevistas, insertando los siguientes datos:

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

- TE: Tiempo por entrevistas.
- (n): Cantidad de entrevistas.

Datos a mostrar

- TTE: Tiempo total por entrevistas.
$$TTE = \sum_{i=1}^n TE$$

RF 28. Calcular el tiempo total por levantamientos de requisitos.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido por levantamiento de requisitos, insertando los siguientes datos:

- TLR: Tiempo levantamiento de requisitos.
- (n): Cantidad de requisitos.

Datos a mostrar

- TTLR: Tiempo total por levantamiento de requisitos.
$$TTLR = \sum_{i=1}^n TLR$$

RF 29. Calcular el tiempo total por implementación.

El sistema debe permitir al Administrador de la calidad calcular y mostrar el tiempo invertido por implementación, insertando los siguientes datos:

- TI: Tiempo por implementación.
- (n): Cantidad de actividades de implementación.

Datos a mostrar

- TTI: Tiempo total por implementación.
$$TTI = \sum_{i=1}^n TI$$

RF 30. Calcular el esfuerzo de la calidad.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

El sistema debe permitir al Administrador de la calidad calcular el esfuerzo de la calidad, mostrando el resultado en un mensaje.

- EoQ: Esfuerzo de la calidad.

RF 31. Calcular el costo de calidad.

El sistema debe permitir al Administrador de la calidad calcular el costo de calidad, mostrando el resultado en un mensaje.

Dato a mostrar

- CoQ: Costo de la calidad. $CoQ = (P + E + Fi + Fe) / (P + E + Fi + Fe + C)$

RF 32. Calcular el esfuerzo de la mala calidad.

El sistema debe permitir al Administrador de la calidad calcular el esfuerzo de la mala calidad, mostrando el resultado en un mensaje.

Dato a mostrar:

- EoPQ: Esfuerzo de la mala calidad. $EoPQ = Fi + Fe$

RF 33. Calcular el costo de la mala calidad.

El sistema debe permitir al Administrador de la calidad calcular el costo de la mala calidad, mostrando el resultado en un mensaje.

Dato a mostrar:

- CoPQ: Costo de la mala calidad. $CoPQ = (Fi + Fe) / (P + E + Fi + Fe + C)$

RF 34. Mostrar reporte de costos.

El sistema debe permitir al Administrador de la calidad mostrar un reporte de los costos calculados.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Listado de los Requisitos no Funcionales (RNF)

Los requisitos no funcionales se definen como las cualidades o propiedades que el producto o sea el software debe tener, estos representan las características del producto.

Usabilidad

RNF1. Debe consentir el manejo de cualquier Administrador de la calidad posibilitando confianza y seguridad en cuanto a su uso.

Interfaz

RNF2. La aplicación debe ser amigable y muy profesional permitiendo una interacción amena.

RNF3. El diseño de la interfaz de Administrador de la calidad debe llamar la atención al mismo.

Confidencialidad

RNF4. La información debe estar protegida contra accesos no autorizados.

RNF5. La información debe ser publicada por una persona con los privilegios asignados, o sea el Administrador de la calidad.

Soporte

RNF6. Debe ofrecer la posibilidad de un fácil mantenimiento.

RNF7. Se hallará implementado sobre tecnología Web lo que proporcionará su uso mediante la red.

Seguridad

RNF 8. Tener protección contra acciones no autorizadas y que puedan afectar la integridad, confidencialidad y seguridad de los datos.

RNF 9. Asegurar que el Administrador de la calidad sea autenticado antes de realizar cualquier operación sobre la aplicación.

RNF10. El sistema deberá estar disponible brindando una posibilidad de uso de 24 horas.

Software

La aplicación debe correr en una estación de trabajo que conste con lo siguiente:

RNF11. PostgreSQL 8.2.

RNF12. PHP 5.2.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática. Características del Sistema

2.3.8 Definición de los Casos de Uso

2.3.8.1 Definición de los actores

Actor	Descripción
Administrador de la calidad	Persona encargada de realizarle los cálculos de costos de calidad a un proyecto específico y emitir reportes de dichos costos.

2.3.8.2 Diagrama de Paquetes del Sistema

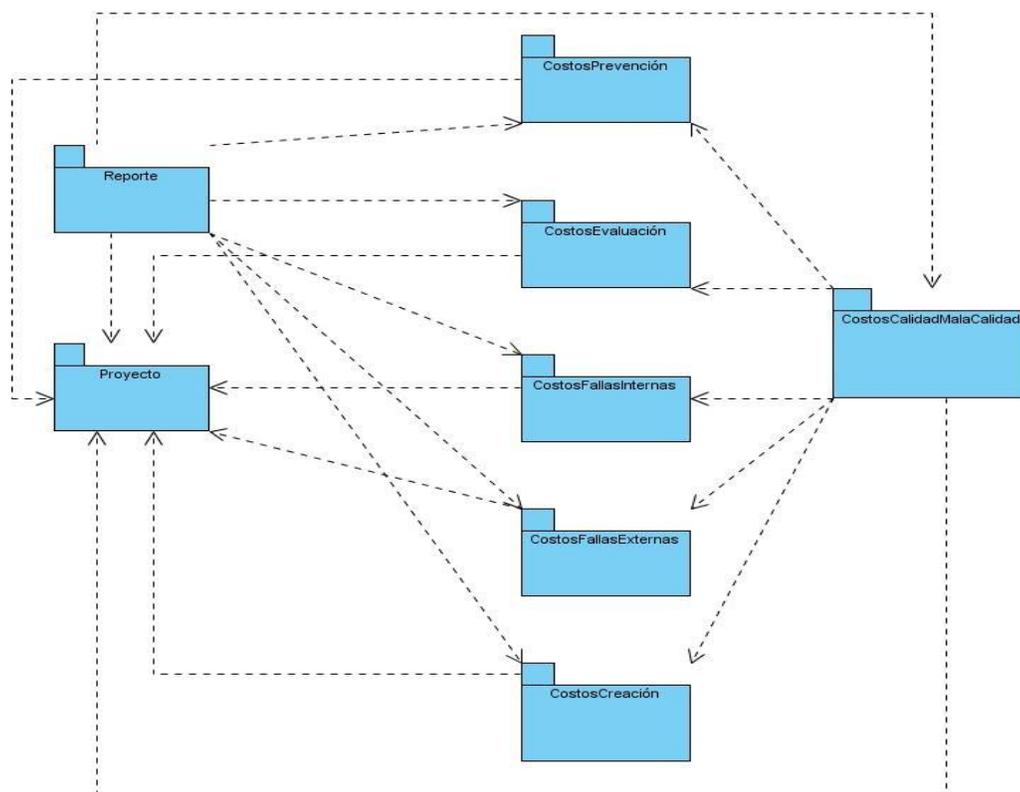


Fig. 8. Diagrama de Paquetes.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

2.3.8.3 Diagrama de Casos de Uso del Sistema

A continuación se muestra el DCUS del paquete más significativo Costos Evaluación. Los demás diagramas se encuentran detallados en el **Anexo 2**.

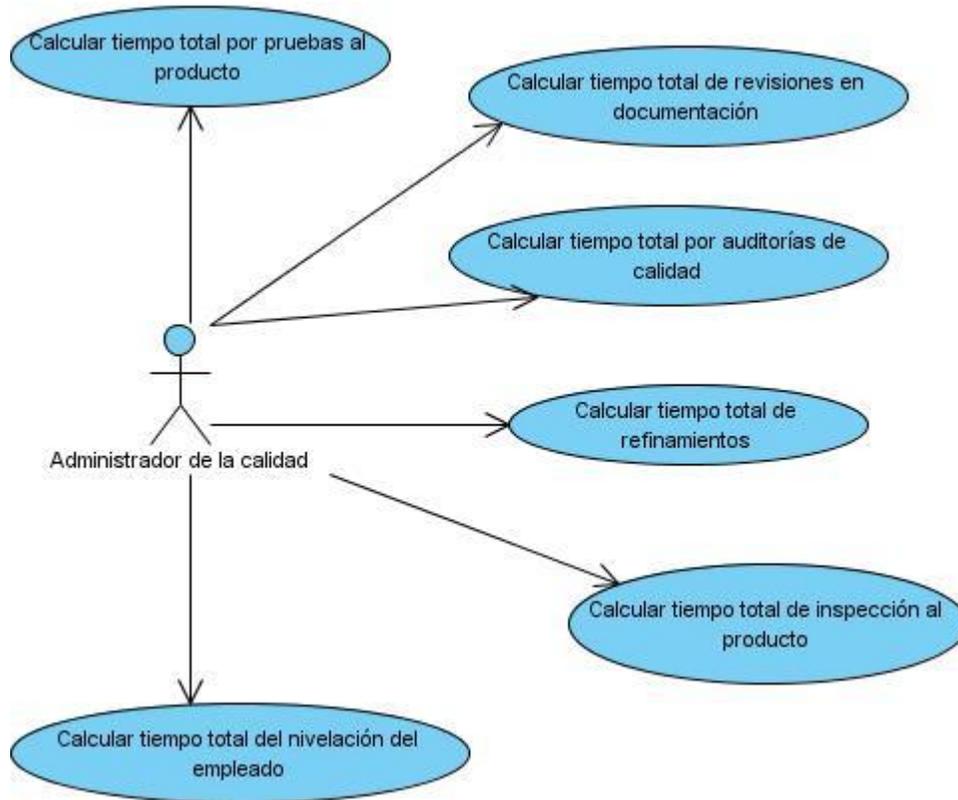


Fig. 9. DCUS del Paquete Costos Evaluación.

2.3.8.4 Descripción de los Casos de Uso del Sistema

A continuación se muestra como ejemplo la descripción del CU Calcular tiempo total de auditorías de calidad del paquete Costos Evaluación, las restantes descripciones se encuentran descritas en el **Anexo 3**.

Descripción de Casos de Uso: Calcular tiempo total de auditorías de calidad.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Caso de Uso:	Calcular tiempo total por auditorías de calidad.	
Actores:	Administrador de la calidad	
Resumen:	El caso de uso se inicia cuando el Administrador de la calidad selecciona del menú principal la opción que le permite calcular los tiempos incurridos en costos de evaluación, seleccionando a continuación la opción que le permite calcular el tiempo total por auditorías de calidad, introduciendo posteriormente los datos para calcular este tiempo. El caso de uso concluye cuando el Administrador de la calidad selecciona la opción que le permite calcular el tiempo, el sistema almacena el resultado en la BD y muestra el dicho resultado en la interfaz.	
Precondiciones:	Proyecto registrado. Usuario autenticado.	
Referencias	RF9	
	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

1. Selecciona la opción que permite calcular los tiempos incurridos en costos de evaluación.	2. Muestra la interfaz correspondiente, en la cual se muestran todos los tiempos por costos de evaluación posibles a calcular.
3. Selecciona la opción que permite calcular el tiempo total por auditorías de calidad.	4. Muestra los datos que permiten calcular este tiempo. <ul style="list-style-type: none"> • TA: Tiempo de la auditoría. • (n): Cantidad de auditorías.
5. Introduce los datos solicitados y selecciona la opción que permite calcular dicho tiempo.	6. Valida que los datos introducidos sean correctos y no sean nulos.
	7. Calcula el tiempo total por auditorías de calidad.
	8. Almacena el resultado en la BD.
	9. Muestra el siguiente resultado obtenido en la interfaz: 10. TTAC: Tiempo total por auditorías de calidad.
Prototipo de Interfaz	

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

<input type="checkbox"/> CTT de audit. de calidad	<input type="checkbox"/> CTT de revisiones	<input type="checkbox"/> CTT por pruebas	<input type="checkbox"/> CTT de nivelación del empleado	<input type="checkbox"/> CTT de insp. del producto	<input type="checkbox"/> CTT de refinamiento
Cantidad de auditorías:	<input style="width: 100%;" type="text"/>				
Tiempo de la auditoría:	<input style="width: 100%;" type="text"/>				
Tiempo total por auditorías de calidad:	<input style="width: 100%;" type="text"/>				
<input type="button" value="Calcular"/>					

Flujo Alternativo 7^a Datos obligatorios nulos

Acción del Actor	Respuesta del Sistema
	6 ^a .1. Indica que existen datos obligatorios nulos. Ir a la acción 5.
Poscondiciones	Tiempo total por auditorías de calidad calculado.

Capítulo 2. Propuesta de acciones para disminuir los costos de calidad en los proyectos del Centro de Telemática.

Características del Sistema

Conclusiones parciales

En este capítulo se realizó el estudio de los procesos del negocio donde se pudo definir la no existencia de los mismos, por lo que fue necesario realizar un diagrama de clases del dominio para lograr un entendimiento preciso de lo que se requería. Se abordaron los principales problemas que existen en Centro de Telemática, demostrándose la necesidad de desarrollar una aplicación que informatice el cálculo de los costos de calidad. Se definieron los requisitos funcionales y no funcionales del sistema para el obtener los resultados más apropiados. Se especificaron las relaciones entre el actor y casos de usos, así como la descripción del caso de uso más significativo dentro del paquete crítico del sistema.

Capítulo 3. Diseño del Sistema

3.1 Introducción

El objetivo fundamental a tratar en el presente capítulo es la realización de un diseño consecuente que dé respuesta a los requerimientos planteados en la sección anterior. Se obtiene una visión general del sistema mediante los diagramas de clases del diseño, así como la especificación de los patrones de diseño.

3.2 Diseño del sistema

En el diseño se modela el sistema de manera que se cumplan los requisitos funcionales y no funcionales. El diagrama de clases del diseño es un modelo de objetos que describe la realización de los casos de uso, centrándose en los requisitos funcionales y no funcionales. Su propósito es adquirir una comprensión de los requisitos funcionales y las restricciones. Mediante él se crea una entrada apropiada y un punto de partida para la implementación de la aplicación. Es un modelo físico, dinámico y no genérico, es el que da forma al sistema mientras intenta preservar la estructura definida por el modelo de análisis.

3.2.1 Diagrama de clases del diseño

A continuación se muestra el diagrama de clases del diseño del paquete Costos Evaluación, los restantes se encuentran detallados en el **Anexo 4**.

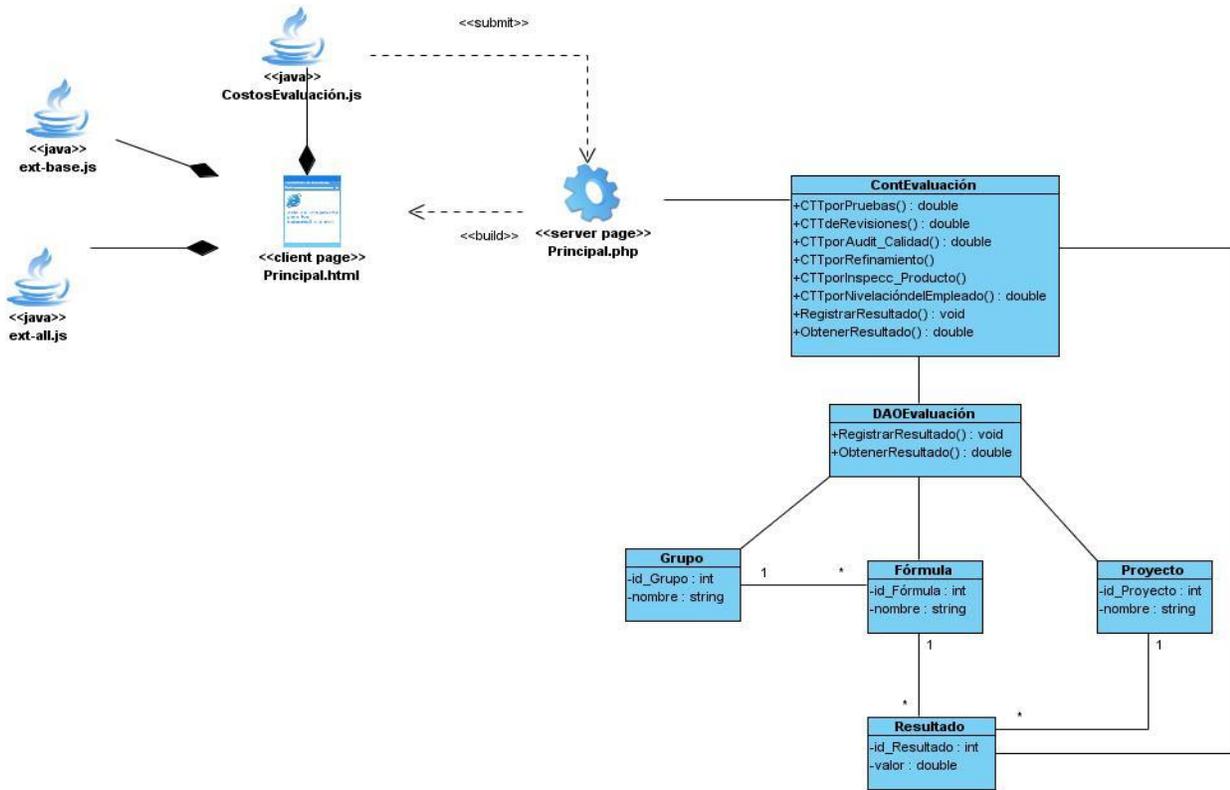


Fig.10. DCD del Paquete Costos Evaluación.

3.2.2 Diagramas de Interacción

UML como lenguaje de modelado define tres grandes grupos de diagramas: de estructuras estáticas o estáticas, de comportamiento o dinámicas y de implementación. Específicamente en los diagramas dinámicos se engloban los de actividades, los de estado y los de interacción, donde estos últimos se clasifican en diagramas de colaboración o secuencia, pero aunque difieren en su organización o distribución, coinciden en querer visualizar o mostrar la interacción entre objetos. Los diagramas de colaboración destacan la organización de los objetos que participan así como el paso de mensajes entre los mismos. Los diagramas de secuencia resaltan el tiempo de vida de cada uno de los objetos que intervienen en el proceso de interacción así como el orden temporal de los mensajes que se intercambian entre objetos lo que ofrece una visión clara del flujo a lo largo del tiempo. Cada uno de ellos posee gran

importancia para los desarrolladores porque mediante ellos se muestra una representación visual de lo que se quiere lograr una vez culminado el sistema o la aplicación Web. Por no ser imprescindible la presencia de ambos, se muestran solo los diagramas de colaboración.

3.2.2.1 Diagramas de Colaboración

A continuación solo se muestra un ejemplo de diagrama de colaboración contenido dentro del paquete más significativo, en este caso el paquete Costos Evaluación, los restantes son muy similares.

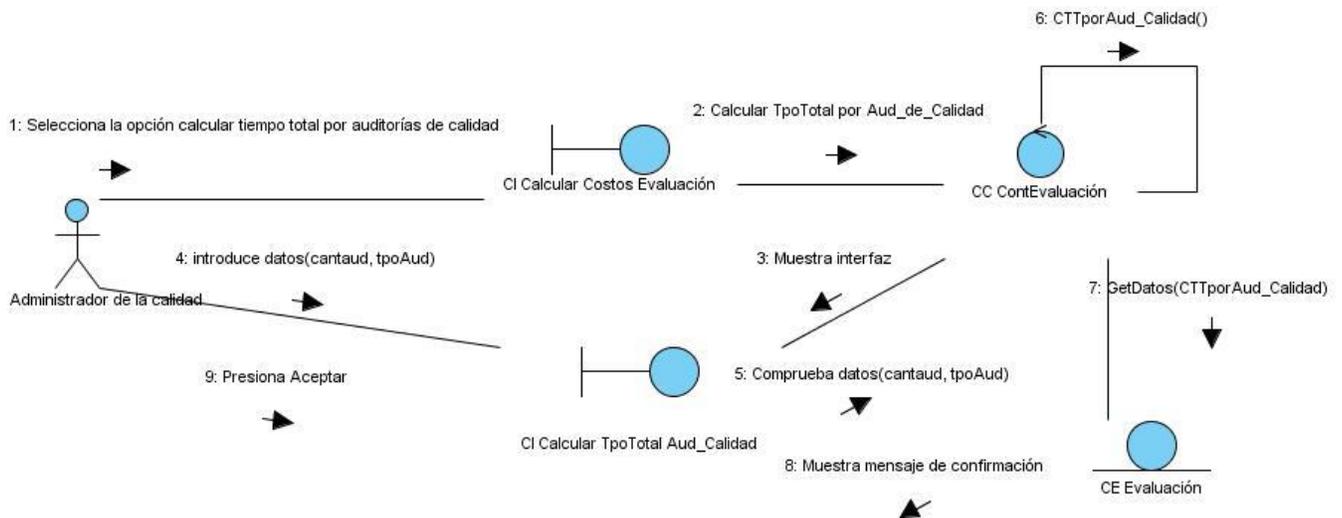


Fig. 11. Diagrama de Colaboración CU_Calcular tiempo total de auditorías de calidad

3.3 Arquitectura y Patrones de Diseño

Todo sistema o aplicación requiere de una visión futura, una arquitectura para tener una idea de cómo se va a implementar, para ver la forma que va a tener dicha aplicación o sistema. En este caso se usará la arquitectura n capas, específicamente la arquitectura en 3 capas y el MVC, utilizados para desarrollar aplicaciones web por el marco de trabajo Sauxe. A continuación se muestran algunas características de los mismos:

La arquitectura de tres capas se refiere a un diseño reciente que introduce una capa intermedia al proceso. Cada capa es un proceso separado y bien definido corriendo en plataformas separadas.

En la arquitectura tradicional de tres capas se instala una interfaz de usuario en la computadora del usuario final (el cliente). La arquitectura basada en web transforma la interfaz de búsqueda existente (el explorador de web) en la interfaz del usuario final. En esta capa, de las herramientas seleccionadas para el desarrollo de la aplicación, se utilizan el ExtJS Framework para crear las interfaces de usuario de la aplicación que se propone desarrollar y el UCID Framework que abarca la integración del Framework ExtJS con el sistema, incluyendo el integrador de interfaz, el generador de interfaz dinámica, etc.

La parte funcional de la arquitectura de tres capas, generalmente es conocida como la capa intermedia o el servidor de aplicaciones. Este es donde la mayoría de los procesos ocurren. En esta capa, de las herramientas seleccionadas para el desarrollo de la aplicación, se utiliza el Zend Framework, encargado del desarrollo de la aplicación web.

La tercera capa comúnmente es el sistema de administración de la base de datos, donde los datos requeridos por la capa intermedia son almacenados. La tercera capa se localiza en un servidor separado conocido como el servidor de base de datos. En esta capa, de las herramientas seleccionadas para el desarrollo de la aplicación, se utiliza el Doctrine Framework para mapear la base de datos y exportar una base de datos a sus clases correspondientes y viceversa. A continuación se muestra la figura que presenta la arquitectura en 3 capas.

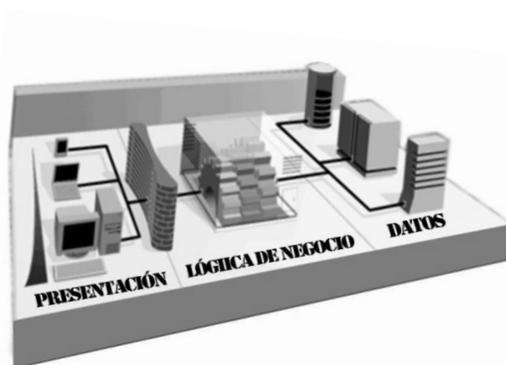


Fig. 12. Arquitectura de 3 capas.

Modelo-Vista-Controlador (MVC)

El patrón conocido como Modelo-Vista-Controlador (MVC) separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes:

- ✓ El Modelo, representa los datos de la aplicación.
- ✓ La Vista, maneja la presentación visual de los datos del modelo.
- ✓ El Controlador, interactúa con el modelo actualizando la vista.

En el marco de trabajo Sauxe, este patrón de arquitectura se emplea en la capa de negocio de la arquitectura 3 capas del Sauxe, con el propósito de integrar dicha capa con la capa de presentación. MVC separa la lógica de la presentación de los datos por lo que las aplicaciones son más fáciles de mantener.

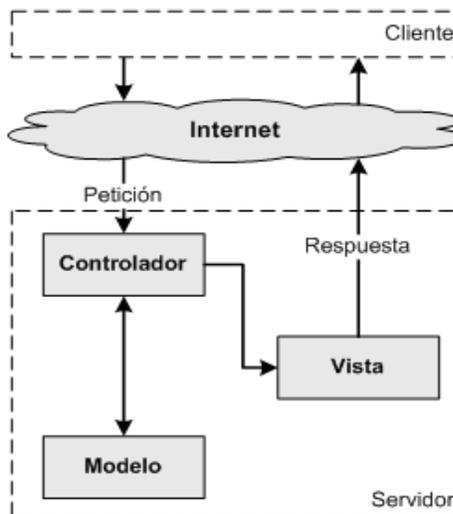


Fig. 13. Patrón Modelo-Vista-Controlador.

Patrones de Diseño

Fachada

Es un patrón de diseño que proporciona una interfaz de alto nivel unificada para facilitar el acceso a los métodos. Se utiliza para simplificar los métodos expuestos de una clase u objeto facilitando considerablemente su uso. En la creación de las clases que administrarán las distintas fases del proceso completo que realizará la herramienta, se diseña una interfaz sencilla de cada una para simplificar y facilitar el uso de las mismas por las clases de las capas superiores. En resumen, el patrón fachada consiste en crear una única clase de manejo más fácil que nos permita acceder a un conjunto más o menos numeroso y complicado de clases que necesitan una inicialización compleja.

Problemas que soluciona

- Un cliente necesita acceder a parte de la funcionalidad de un sistema más complejo.
Solución: Define una interfaz que permita acceder solamente a esa funcionalidad.
- Existen grupos de tareas muy frecuentes para las que se puede crear código más sencillo y legible.
Solución: Define funcionalidades que agrupan estas tareas en funciones o métodos sencillos y claros.

3.4 Diseño de la Base de Datos

El modelo de datos describe de una forma abstracta cómo se representan los datos en un sistema de gestión de base de datos. Es una herramienta para especificar los tipos de datos y la organización de los mismos que son permisibles. Este modelo es una guía para el diseño de la base de datos. En la siguiente figura se muestra el modelo de datos del sistema y las relaciones existentes entre sus tablas.

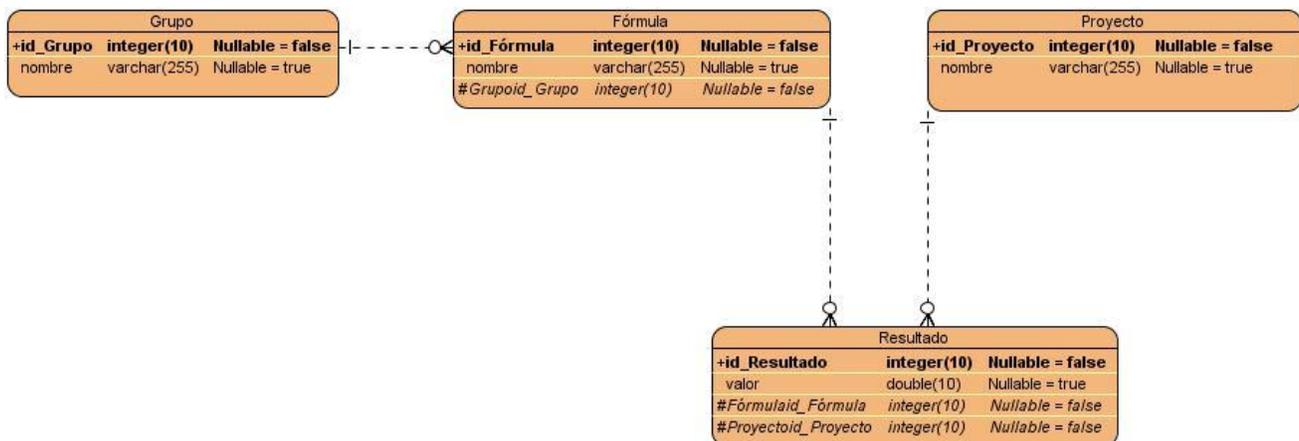


Fig. 14. Modelo de Datos.

Conclusiones parciales

En el presente capítulo se mostró una representación gráfica de los diagramas de clases del diseño e interacción. Se definió el modelo de datos de la aplicación, además de la arquitectura candidata del sistema, analizándose el patrón de diseño para su correcta implementación.

Capítulo 4. Pruebas

4.1 Introducción

En el presente capítulo se detalla la propuesta de diseño de Casos de Prueba basados en Casos de Uso para la aplicación que se propone desarrollar.

4.2 Propuesta de diseño de Casos de Pruebas.

A partir del momento en que el equipo de desarrollo ha elaborado los casos de usos del sistema se pueden elaborar los casos de pruebas que no son más que un conjunto de entradas con datos de prueba, unas condiciones de ejecución, y unos resultados esperados con el propósito de identificar y comunicar las condiciones que se llevarán a cabo en la prueba. Los casos de pruebas son necesarios para verificar la aplicación exitosa y aceptable de los requisitos del producto (que se especifican en los casos de uso).

En la elaboración de los casos de prueba se utilizó un método que es una variante del de particiones equivalentes, propuesto por RUP, basado totalmente en los casos de uso y consta para su elaboración de tres pasos fundamentales: identificación de escenarios de prueba, identificación de los casos de prueba (identificación de las variables) y por último identificación de valores de datos para las pruebas.

A continuación se muestra un ejemplo de diseño de caso de prueba para el paquete más significativo: Costos Evaluación, los restantes diseños de casos de prueba se encuentran en el **Anexo 5**.

Caso de Prueba Calcular tiempo total de auditorías de calidad

Descripción General

El caso de uso se inicia cuando el Administrador de la calidad selecciona del menú principal la opción que le permite calcular los tiempos incurridos en costos de evaluación, seleccionando luego la opción que le permite calcular el tiempo total por auditorías de calidad, introduciendo posteriormente los datos para calcular este tiempo. El caso de uso concluye cuando el Administrador de la calidad selecciona la opción que le permite calcular el tiempo, el sistema almacena el resultado en la BD y muestra dicho resultado en la interfaz.

Condiciones de Ejecución

Aplicación disponible.

Proyecto registrado

SC Calcular tiempo total por auditorías de calidad

ID del escenario	Escenario	Cantidad de auditorías	Tiempo de la auditoría	Respuesta del sistema	Flujo central
EC 1 "Calcular tiempo total de auditorías de calidad".	Calcula exitosamente el tiempo total por auditorías de calidad.	V	V	El sistema muestra el tiempo total por auditorías de calidad.	<ul style="list-style-type: none"> - El actor selecciona la opción "Costos de evaluación". - El sistema muestra la interfaz donde se muestran todos los tiempos por costos de evaluación posibles a calcular. - El actor selecciona la opción "Calcular tiempo total de auditorías de calidad".
EC 2 "Campos obligatorios incorrectos o nulos".	Se introducen datos incorrectos o nulos.	I	V	El sistema muestra un mensaje de información, alertando que existen campos incorrectos o nulos.	<ul style="list-style-type: none"> - El sistema muestra los datos que permiten calcular este tiempo. - El actor introduce los datos y selecciona el botón "Calcular". - El sistema valida que los datos introducidos sean correctos y no sean nulos. - El sistema calcula el tiempo total por auditorías de calidad. - El sistema almacena el resultado en la BD. -El sistema muestra el siguiente resultado obtenido en la interfaz: <p>TTAC: Tiempo total por auditorías de calidad.</p>
		V	I		

Descripción de las variables

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Cantidad de auditorías	Campo numérico	No	Los datos a introducir puede contener ["0"- "9" caracteres]
2	Tiempo de la auditoría	Campo numérico	No	Los datos a introducir puede contener ["0"- "9" caracteres]

Conclusiones parciales

En este capítulo se validaron los requisitos mediante el diseño de casos de prueba basados en casos de uso correspondiente a la aplicación que se propone desarrollar. Para ello se hizo uso del método de prueba Caja Negra específicamente utilizando la técnica Partición de Equivalencia.

Conclusiones

Una vez realizada la investigación y concluido el diseño de la aplicación Web para el cálculo de los Costos Totales de la Calidad y cumplidos los objetivos planteados así como las tareas de investigación, se pueden proyectar los resultados alcanzados:

- ✓ Se elaboró la propuesta de acciones y medidas a tomar para mejorar los costos de calidad en los proyectos de software del Centro de Telemática.
- ✓ Se seleccionaron las herramientas y tecnologías necesarias a utilizar para un buen modelado y desarrollo de la aplicación Web.
- ✓ Se diseñó una aplicación Web que garantiza el cálculo de los costos de calidad en los proyectos de investigación y desarrollo del Centro de Telemática.

Recomendaciones

- ✓ Implementar la aplicación Web para el cálculo de los Costos Totales de la Calidad con el fin de mejorar la deficiente gestión de la calidad en el Centro de Telemática.
- ✓ Aplicar las acciones propuestas para mejorar costos de calidad en el Centro de Telemática de la UCI, poniéndose a disposición de cada uno de los proyectos productivos que existen en la comunidad universitaria.
- ✓ Integrar como base de conocimientos la propuesta de acciones para mejorar y disminuir costos de calidad en los proyectos de software a la aplicación que se piensa desarrollar.
- ✓ Ofertar cursos referidos a los Costos Totales de la Calidad expresados en horas en el Centro de Telemática de la UCI.

Referencias Bibliográficas

1. **Mazorra Lopetey, Amilkar.** "Metodología simplificada para cálculo de costos de calidad " en Observatorio de la Economía Latinoamericana. 124, 2009. Disponible en:
<http://www.eumed.net/cursecon/ecolat/cu/2009/aml.htm>
2. **Segoviano Martínez, Zamira; Muñoz Roja, Maikel; Abreu Sáez, Ana Margarita; Granado Ibarra, Lilibeth; Trujillo Casañola.** REVISIONES DE SOFTWARE EN LA UNIVERSIDAD DE LAS CIENCIAS INFORMATICAS. UCI, Ciudad de La Habana. Septiembre 2010. Disponible en <http://semanatecnologica.fordes.co.cu/public/site/328.pdf>.
3. **Fernández Carrasco, Oscar M; García León, Delba; Beltrán Benavides, Alfa.** Un enfoque actual sobre la calidad del software. [Online] Septiembre-Diciembre 1995. [Cited: Noviembre 25, 2010.] http://www.bvs.sld.cu/revistas/aci/vol3_3_95/aci05395.htm. Pressman, R.S,"Ingeniería del software. Un enfoque práctico". 2ª ed. 1992: McGraw-Hill.
4. **Pressman, Rogers.** Ingeniería del Software. Un enfoque práctico. Quinta Edición.
5. **Valenciano Díaz, Raiza; Sánchez Batista, Ana María.** "Estrategia de aseguramiento de la calidad para los procesos de la Fábrica de Software para implementar Sistemas Integrales de Gestión de Documentos y Archivos". UCI. Ciudad de La Habana, 2009.98.
6. **Caballero Alcántara, Jose Luis.** Gestión de La Calidad. Fundamentos y Conceptos. Sistema de la Calidad. (2005).
Disponible en: http://www.elprisma.com/apuntes/administracion_de_empresas/gestiondelacalidad.
7. **Rico Menéndez, J.** Evolución del concepto de Calidad, 10, N.º 3 - 169. Disponible en:
<http://www.meditex.es/elmedico/publicaciones/trasplantes3/169-175.pdf>.
8. **Fernández Sanz, Luis.** **Necesidades de medición en la gestión y el aseguramiento de calidad del software.** 1999, Disponible en: <http://www.sc.ehu.es/jiwdcoj/remis/docs/aseguracal.htm>.
9. **Morera Cruz, Jose Orlando.** Mejoramiento Continuo. 2002. Disponible en:
<http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/meconti.htm>.
10. Mejoramiento Continuo. Disponible en:

- http://www.freelancecolombia.com/articulos/liderazgo-mejoramiento_continuo.php.
11. **Montañez Graterol, Luis Eduardo**. “Diseño de un modelo para el control de la calidad de los proyectos de desarrollo de software en las organizaciones del estado Venezolano”. 2006. Disponible en: <http://biblioteca2.ucab.edu.ve/anexos/biblioteca/marc/texto/AAQ9075.pdf>.
 12. Estudio del modelo de capacitación de Madurez y su aplicación en empresas guatemaltecas. Disponible en: http://biblioteca.usac.edu.gt/tesis/08/08_6152.pdf.
 13. **Echarte Mazorra, Yaima; Molina Flores, Fanny**. “Generalidades sobre los costos de calidad”. Disponible en: <http://www.monografias.com/trabajos-pdf2/generalidades-costos-calidad/generalidades-costos-calidad.pdf>
 14. STRAUB, P. El costo de la calidad y el costo de la mala calidad. 2006.
 15. Guía Básica para Monitorear los Costos de la Calidad. 2006. Disponible en: <http://www.updce.ipn.mx/ae/guiasem/costosdecalidad.pdf>.
 16. Economía de la Calidad. Disponible en: <http://www.fing.edu.uy/iimpi/academica/grado/ctrlcalidad/teorico/EconomiaDeLaCalidad.pdf>.
 17. **Addison-Wesley**. “El Proceso Unificado de Desarrollo de Software”. Madrid, 1999.
 18. **Larman, Craig**. UML y Patrones. La Habana: Editorial Félix Varela, 2004.
 19. **Londoño, M.Y.O.** Herramientas C.A.S.E.[En línea] [Citado el: 29 de noviembre de 2010] <http://personales.com/cuba/cardenas/internet/case.html>.
 20. **Pavón, Eduardo León**. Blog de Eduardo León. [En línea] 2008. [Citado el: 5 de diciembre de 2010.] <http://slion2000.blogspot.com/2007/04/visual-paradigm-una-herramienta-de-lo.html>.
 21. **González Ortega, Miguel; Domínguez Suárez, Lázaro Enrique**. Desarrollo del subsistema de Estructura y Composición versión 2.0 perteneciente al Sistema de Gestión de Entidades Cedrux. UCI. Ciudad de La Habana, 2010.80.
 22. **Pérez Valdés, Damián**. MAESTROS DEL WEB 2 de Noviembre de 2007, [Consultado el: 4 de enero de 2011]. Disponible en: <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programación-para-la-web>.
 23. LENGUAJE DE PROGRAMACIÓN HTML [En línea] [Citado el: 6 de enero de 2011.] Disponible en:

www.monografias.com/trabajos7/html/html.shtml

24. **Jiménez Valdés, Jorge Félix; Pérez Díaz, Yeisis.** “Propuesta de procedimiento para determinar los Costos Totales de la Calidad en los proyecto de software del Centro de Telemática de la UCI”. UCI. Ciudad de La Habana, 2009.74.

Bibliografía

1. Calisoft. [Online] Universidad de las Ciencias Informáticas, 2010. <http://calisoft.uci.cu/>.
2. **Salazar, Delia.** Joseph M. Juran y sus aportes a la calidad.
3. **Pressman, Roger S.** Ingeniería del Software. Un enfoque práctico. Quinta Edición.
4. Tutorial Mejora Continua. Parte del Sistema de Gestión de Calidad.
5. **Domínguez, L. C.** Multimedia Introducción al CMMI Nivel 2. Ciudad de la Habana, Universidad de las Ciencias Informáticas, 2008.
6. **Scalone, Fernanda.** 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>.
7. **Meneses, Nora.** Costos de Calidad. Gestión de Calidad 2010.
8. **Panusieri, Jorge.** Gestión de Calidad. 2010
<http://www.economicas.unsa.edu.ar/otros/tecnicatura/Clase2-Gesti%C3%B3n-de-Calidad.pdf>
9. **Jiménez Valdés, Jorge Félix; Pérez Díaz, Yeisis.** “Propuesta de procedimiento para determinar los Costos Totales de la Calidad en los proyecto de software del Centro de Telemática de la UCI”. UCI. Ciudad de La Habana, 2009.74.
10. **Bauta Camejo, René R; Torres Gálvez, Ariel.** Generación de ficheros de mapeo mediante el uso de una herramienta basada en el framework Doctrine. Ciudad de La Habana, 2010.
11. **Gillezeau B, Patricia; Romero, Samuel.** Sistema de Costos de Calidad como Proceso de Mejoramiento Continuo. Disponible en:
http://www.saber.ula.ve/bitstream/123456789/17582/2/articulo_1.pdf.
12. Capítulo II. Costos de Calidad. Disponible en: http://catarina.udlap.mx/_uploads/tales/documentos/lcp/zamacona_s_r/capitulo2.pdf.
13. **Jorge Cabreja, Liudmila; Ávila Pérez, Iraida.** Procedimiento para el cálculo de los costos de calidad en Contribuciones a la Economía. Abril 2009. Disponible en:
<http://www.eumed.net/ce/2009a/>.
14. Guía Básica para Monitorear los Costos de la Calidad. Marzo 2006. Disponible en:
<http://www.updce.ipn.mx/ae/quiasem/costosdecalidad.pdf>.

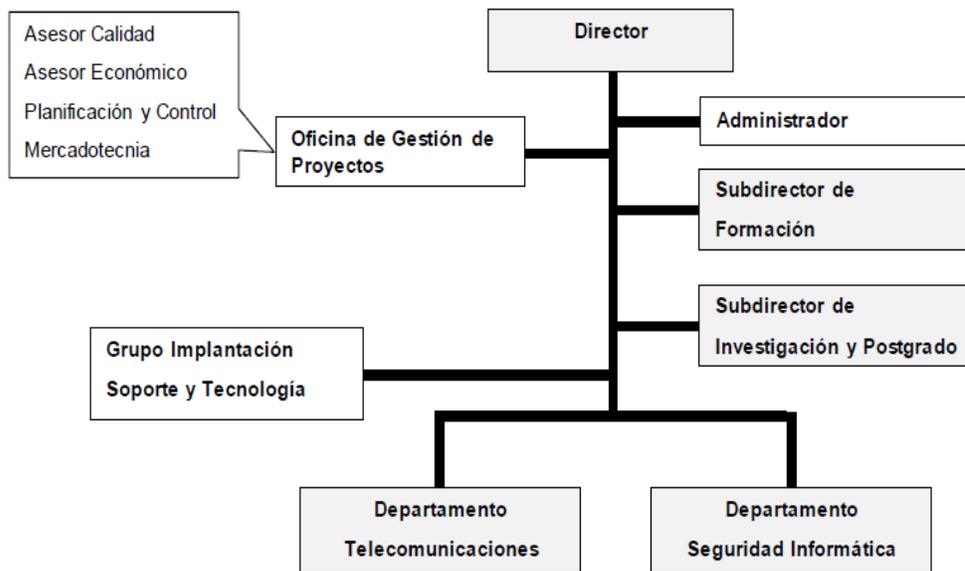
Glosario de Términos

- A Audis BD:** Auditoria de BD, nombre de un proyecto del Centro de Telemática de la UCI.
- C CASE:** Ingeniería de software asistida.
CMMI: Modelo de Madurez de Capacitación Integrada.
CoQ: Costos de calidad.
CoPQ: Costos de mala calidad.
- E ETECSA:** Empresa de Telecomunicaciones de Cuba SA.
EoQ: Esfuerzo de calidad.
EoPQ: Esfuerzo de mala calidad.
- I I+D:** Investigación más Desarrollo.
ICON: Informatización de la Contraloría General de la República, nombre de un proyecto del Centro de Telemática.
- M MA:** Medición y Análisis.
- P PPQA:** Aseguramiento de la Calidad a Procesos y Productos.
PP: Planeación del Proyecto.
PMC: Monitoreo y Control del Proyecto.
- R REQM:** Administración de Requisitos.
RUP: Proceso Unificado de Software.
- S SAM:** Administración de Acuerdos con Proveedores.
SIE Center: Centro de Excelencia para la Industria de Software.
SGIC-PABX: Sistema de Gestión Integral de Costo para PABX, nombre de un proyecto del Centro de Telemática.
SIAI: Sistema Integral de Análisis de Información, nombre de un proyecto del Centro de Telemática.
SER-WAP: Servidor de aplicaciones WAP, nombre de un proyecto del Centro de Telemática.
- T TLM:** Centro de Telemática.
TIP: Tele Identificador Personal, nombre de un proyecto del Centro de Telemática.
- U UCI:** Universidad de las Ciencias Informáticas.
UML: Lenguaje de Modelado Unificado.

Anexos

Anexo 1

Organigrama del Centro de Telemática de la UCI



Anexo 2

Documento Complementario Anexos/ Diagrama de Casos de Uso del Sistema.

Anexo 3

Documento Complementario Anexos/ Descripción de los Casos de Uso del Sistema.

Anexo 4

Documento Complementario Anexos/ Diagrama de clases del diseño.

Anexo 5

Documento Complementario Anexos/ Diseño de Casos de Pruebas.