



Universidad de las Ciencias Informáticas

Facultad 4

Trabajo de Diploma para optar por el título de Ingeniero en

Ciencias Informáticas

Título: Desarrollo de una herramienta para el cálculo de las métricas del modelo Calidad en el Uso, según las nuevas características de la ISO/IEC 25010.

Autor(es): Dariel Rubén Rodríguez del Portillo
Elvis Guillermo Mena Moya

Tutor(es): Ing. José Manuel Santos Alonso
Ing. Yurisbel Hernández Bernal

Pensamiento

Los hombres más exitosos son aquellos cuyo éxito es el resultado de una acumulación constante. Es el hombre que avanza paso a paso, con su mente cada vez más amplia y progresivamente más capaz para comprender cualquier tema o situación.

Alexander Graham Bell



DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Dariel Rodríguez del Portillo

Firma del autor

Elvis Guillermo Mena Moya

Firma del autor

Ing. Yurisbel Hernández Bernal

Firma del tutor

Ing. José Manuel Santos Alonso

Firma del tutor

DATOS DE CONTACTO

Dariel Rodríguez del Portillo

Email: drdelportillo@estudiantes.uci.cu

Elvis Guillermo Mena Moya

Email: egmena@estudiantes.uci.cu

José Manuel Santos Alonso

Profesor Asistente

Email: jmsantos@uci.cu

Yurisbel Hernández Bernal

Especialista "B" en Ciencias Informáticas

Email: ybernal@uci.cu

RESUMEN

El proceso de evaluación de la calidad presenta problemas tanto en su ciclo de desarrollo como al concluir el mismo, uno de estos problemas es que no se mide la Calidad en el Uso, lo que trae como consecuencia que no se obtenga la calidad que se requiere de los productos de software, además el proceso de evaluación está guiado por la norma ISO/IEC 9126-1 la cual se deroga actualmente y provoca que las herramientas que automatizan el proceso de medición de la calidad existentes pasen a ser obsoletas.

La presente investigación describe el desarrollo de una herramienta que implementa las métricas del modelo Calidad en el Uso de la norma ISO/IEC 25010 la cual sustituye a la ISO/IEC 9126-1, esta herramienta permitirá realizar evaluaciones de calidad que darán un veredicto conclusivo acerca de la calidad de los productos de software, facilitándole a los usuarios finales la realización de evaluaciones ágiles y de alto grado de confiabilidad debido a que se utilizan elementos de última generación como es la norma ISO/IEC 25010.

Como resultado se obtiene un producto funcional que evaluará la calidad de cualquier producto de software utilizando criterios de evaluación que se encuentran estandarizados internacionalmente en la norma ISO/IEC 25010 el cual traerá diferentes beneficios a las instituciones que decidan emplearlo para el aumento del número de clientes, mejora de las relaciones con los clientes y reducción de costos debido a la mala calidad.

Palabras claves: Calidad, Evaluación, Métricas, Norma, Producto

Índice

Introducción.....	1
Capítulo 1. Fundamentación Teórica.....	4
1.1 Introducción.....	4
1.2 Normas de calidad.....	4
1.2.1 ISO/IEC 9126.....	5
1.2.2 ISO/IEC 25000.....	5
1.2.3 ISO/IEC 25010:2011.....	6
1.2.4 Modelo de Calidad en el Uso.....	7
1.3 Herramientas para medir la calidad.....	8
1.3.1 Software Aided Quality in Use Assessment (SW-AQUA).....	9
1.3.2 WebQEM_Tool.....	9
1.3.3 Cálculo de las métricas externas.....	10
1.4 Metodología de desarrollo de software.....	13
1.4.1 Programación Extrema (XP).....	13
1.5 Lenguaje de programación.....	14
1.6 Herramientas.....	14
1.6.1 Entorno de desarrollo Integrado (IDE). NetBeans7.4.....	14
1.6.2 BizAgi 2.3.0.5.....	15
1.6.3 Visual Paradigm 8.0.....	15
1.6.4 IReport 5.1.0.....	15
1.6.5 SQLite 3.5.1.....	16
1.7 Conclusiones.....	16
Capítulo 2 Características del sistema.....	17
2.1 Introducción.....	17
2.2 Descripción del proceso.....	17
2.3 Objeto de automatización.....	18

2.4 Propuesta del modelo y el sistema.....	18
2.4.1 Modelo para la Calidad en el Uso	18
2.4.2 Métricas seleccionadas	18
2.4.3 Propuesta del sistema	24
2.4.4 Modelación del proceso del sistema	29
2.5 Características del Sistema.....	30
2.6 Fase de Exploración	30
2.6.1 Historia de Usuarios (HU)	31
2.7 Fase de planificación	37
2.7.1 Estimación de esfuerzo por Historias de Usuario.....	38
2.7.2 Plan de Iteraciones	38
2.7.3 Plan de duración de las iteraciones	39
2.7.4 Plan de entregas	39
2.7.5 Conclusiones	40
Capítulo 3 Implementación del Sistema y Prueba.....	41
3.1 Introducción	41
3.2 Patrón de Arquitectura	41
3.3 Arquitectura N-Capa	41
3.4 Patrones de diseño	44
3.5 Diseño físico de la base de datos.....	46
3.6 Tarjetas Clase – Responsabilidad – Colaborador (CRC).....	46
3.7 Tareas de ingeniería	50
3.8 Pruebas	55
3.8.1 Pruebas Unitarias	55
3.8.2 Pruebas de aceptación	56
3.8.3 Prueba para medir la Calidad en el Uso	62
3.9 Conclusiones.....	63

Conclusiones Generales..... 64

Recomendaciones..... 65

Referencias y Bibliografía..... 66

Índice de Tablas

Tabla 1: Tabla comparativa de Herramientas	13
Tabla 2 Realizar una evaluación de calidad.....	32
Tabla 3 Continuar una evaluación de calidad	33
Tabla 4 Visualizar las evaluaciones de calidad.....	33
Tabla 5 Insertar Evaluador	34
Tabla 6 Insertar Software	34
Tabla 7 Guardar los resultados de las evaluaciones de calidad.....	35
Tabla 8 Exportar reportes de la evaluación en formato pdf.....	35
Tabla 9 Deshabilitar Software.....	36
Tabla 10 Modificar Software	36
Tabla 11 Deshabilitar Evaluador.....	37
Tabla 12 Modificar Evaluador	37
Tabla 13 Estimación de esfuerzo por HU.....	38
Tabla 14 Plan de duración por Iteraciones.....	39
Tabla 15 Plan de entregas.....	40
Tabla 16 Tarjeta CRC Evaluador.....	47
Tabla 17 Tarjeta CRC Software.....	47
Tabla 18 Tarjeta CRC Evaluación	48
Tabla 19 Tarjeta CRC CoberturaContexto.....	48
Tabla 20 Tarjeta CRC Eficacia	49
Tabla 21 Tarjeta CRC Eficiencia.....	49
Tabla 22 Tarjeta CRC LibertadRiesgo	49
Tabla 23 Tarjeta CRC Satisfacción.....	50
Tabla 24 Tarea de ingeniería Realizar una evaluación de calidad	50
Tabla 25 Tarea de ingeniería Continuar una evaluación de calidad.....	51
Tabla 26 Tarea de ingeniería Visualizar evaluación de calidad.....	51

Tabla 27 Tarea de ingeniería Insertar evaluador	52
Tabla 28 Tarea de ingeniería Insertar software	52
Tabla 29 Tarea de ingeniería Guardar resultados de las evaluaciones de calidad.....	53
Tabla 30 Tarea de ingeniería Exportar reporte general de la evaluación en formato pdf	53
Tabla 31 Tarea de ingeniería Exportar reporte del veredicto conclusivo por característica en formato pdf	53
Tabla 32 Tarea de ingeniería Deshabilitar software	54
Tabla 33 Tarea de ingeniería Modificar software	54
Tabla 34 Tarea de ingeniería Deshabilitar evaluador.....	54
Tabla 35 Tarea de ingeniería Modificar evaluador	55
Tabla 36 Prueba de aceptación Realizar una evaluación de calidad	56
Tabla 37 Prueba de aceptación Continuar una evaluación de calidad	57
Tabla 38 Prueba de aceptación Visualizar Evaluaciones de calidad.....	57
Tabla 39 Prueba de aceptación Insertar evaluador.....	58
Tabla 40 Prueba de aceptación Insertar software.....	58
Tabla 41 Prueba de aceptación Guardar los resultados de evaluaciones de calidad.....	59
Tabla 42 Prueba de aceptación Exportar reporte de la evaluación en formato pdf.	59
Tabla 43 Prueba de aceptación Exportar reportes del veredicto conclusivo por característica formato pdf.....	60
Tabla 44 Prueba de aceptación Deshabilitar software	60
Tabla 45 Prueba de aceptación Modificar software	61
Tabla 46 Prueba de aceptación Deshabilitar evaluador	61
Tabla 47 Prueba de aceptación Modificar evaluador.	62

Índice de Figuras

Figura 1: Modelo Calidad en el Uso.....	7
Figura 2 Tabla de Característica Funcionalidad de Cálculo de métricas.....	11
Figura 3 Interpretación de resultado según la escala.....	11
Figura 4 Tabla resumen de Evaluación	12
Figura 5: Fases del proceso de evaluación.....	17
Figura 6 Interfaz Principal del sistema	24
Figura 7 Interfaz de Selección de Evaluador y Software.....	24
Figura 8: Estructura para el peso.....	25
Figura 9 Interfaz de Características y Sub-Características del modelo Calidad en el Uso	26
Figura 10 Interfaz de Veredicto conclusivo	27
Figura 11 Interfaz de Evaluaciones no terminadas	28
Figura 12 Interfaz de Contraseña	28
Figura 13: Procesos de nueva evaluación.	29
Figura 14: Estructura de Capas.....	42
Figura 15: Capa Interfaz de usuario.....	43
Figura 16: Capa del Negocio	43
Figura 17 : Capa de datos	44
Figura 18: Diseño físico de la base de datos	46

Introducción

En la actualidad son cada vez más las organizaciones que se dedican a la producción de software esto provoca que exista una alta competencia para vender y posicionar un producto de software determinado en el mercado mundial. Cuba a pesar de ser una industria emergente en la producción de software cuenta con varias empresas como Desoft, Softel y la UCI que se centran en dar soluciones informáticas con alto nivel de acabado. Estas organizaciones tienen la necesidad primaria de garantizar la calidad de sus productos porque los mismos fueron desarrollados por seres humanos y pueden presentar fallos debido a diferentes causas como análisis equivocados, diseños con fallos, programación con errores, especificaciones incorrectas, etc. La forma óptima para disminuirlos es asegurando la calidad de software que es un concepto que ha variado con los años y la experiencia, por lo que existen diferentes maneras de concebirlo. Entre los estudiosos que se han encargado de formular este concepto, se puede destacar a Roger Pressman que la define como la: *“Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario”*. (1)

Para lograr un software de buena calidad es preciso realizar varias actividades, entre las que se pueden mencionar el establecimiento de las características del software y las mediciones que se definen como: *“Conjunto de operaciones que tiene el objeto de determinar un valor de una medida”* (2); estas mediciones suministran una valiosa información, permitiendo desarrollar proyectos más acertados, mejorar costos y satisfacer en gran medida las necesidades del cliente. El proceso de medición persigue tres objetivos fundamentales: entender qué ocurre durante el desarrollo y el mantenimiento, controlar qué es lo que ocurre en los proyectos y mejorar los procesos y los productos. Sin embargo no se puede medir al software como tal, sino a las características que lo conforman.

En la actualidad existen métricas de calidad estandarizadas internacionalmente en normas o estándares y son definidas como *“Variable a la cual es asignada un valor como el resultado de la medición”* (2), estas métricas permiten medir la calidad del software en base a ciertas características; ya sea una vez finalizado el software, lo cual sería más trabajoso, o llevando un control durante cada etapa del ciclo de vida del mismo. Estas métricas proporcionan una indicación de cómo se ajusta el software a los requisitos implícitos y explícitos del producto.

En Cuba principalmente en la UCI se ha ido tratando el tema de la calidad llegando a crear CALISOFT que es el centro nacional de calidad de software que presta servicios como pruebas de aceptación con

el cliente, auditorias, revisiones y evaluación de productos que facilitan la implementación, la satisfacción del cliente y la productividad de una organización, además para la gestión de requisitos y procesos cuentan con nivel 2 de CMMI que le permite a la organización mejorar la calidad de los proyectos y sus productos. No obstante la calidad del software no se mide adecuadamente en el proceso de desarrollo, lo que ha dificultado la confección del producto final afectando el tiempo de entrega, el costo total y la relación con el cliente. Aunque se ha establecido en tesis y metodologías anteriores el modelo de calidad intrínseco del producto y su medición siguiendo las características de la ISO/IEC 9126-1 la cual se deroga actualmente; no se ha profundizado adecuadamente en el tema de la Calidad en el Uso que tiene importantes características que miden el impacto del software en los usuarios y que en el año 2011 fue definido por la nueva norma ISO/IEC 25010 como: “... *el grado en que un producto o sistema puede ser utilizado por usuarios específicos para satisfacer sus necesidades y alcanzar sus objetivos específicos con efectividad, eficiencia, independencia de riesgo y satisfacción en contextos de uso específicos* ” (3). Las organizaciones de desarrollo de software están adoptando el modelo de Calidad en el Uso de esta nueva norma, provocando que las herramientas existentes que miden la calidad del software a través de métricas pasen a ser obsoletas porque utilizan la ISO/IEC 9126-1 como modelo para calcular las métricas de calidad, y Cuba al producir software forma parte del desarrollo tecnológico que vive la sociedad mundial por lo que necesita implementar los cambios surgidos en el mundo fomentando la calidad y el desarrollo de nuestra industria, así como el prestigio en el intercambio con los clientes nacionales y extranjeros.

A raíz del análisis de todo lo anterior surge la interrogante: ¿Cómo automatizar la medición de la Calidad en el Uso de los productos de software según la norma internacional ISO/IEC 25010? Ello constituye el **Problema científico a resolver**.

Para proporcionarle una solución al problema analizado se define como **Objetivo General**: Concebir un modelo particular de la Calidad en el Uso y desarrollar una herramienta automatizada para medir la Calidad en el Uso de los productos de software según la norma internacional ISO/IEC 25010.

Objetivos específicos:

- Realizar el marco teórico referencial del tema.
- Conceptualizar el modelo a implantar a partir de la norma Internacional ISO/IEC 25010
- Desarrollar las funcionalidades de la herramienta.
- Realizar la validación de las funcionalidades del sistema.

El **Objeto de estudio** de la investigación es por tanto el modelo Calidad en el Uso de las normas internacionales y el **Campo de acción** son las características y sub-características de los productos de software, correspondientes al modelo Calidad en el Uso del estándar internacional ISO/IEC 25010

Para el desarrollo de la investigación se propone la siguiente **Idea a Defender**: Si se utiliza una herramienta automatizada basada en el estándar internacional ISO/IEC 25010 para medir la Calidad en el Uso se agilizará el proceso de medición de la calidad de un producto de software.

Tareas de Investigación:

- Estudio y análisis del modelo Calidad en el Uso del estándar internacional ISO/IEC 9126-1 e ISO/IEC 25010 así como herramientas que automatizan el proceso de medición.
- Selección de las herramientas y la metodología para el desarrollo de la herramienta automatizada.
- Estudio del estándar internacional ISO/IEC 25022 y selección de las métricas a emplear en el proceso de medición.
- Elaboración del diagrama de entidad-relación de la base de datos.
- Definición de las funcionalidades de la herramienta automatizada.
- Descripción de las funcionalidades de la herramienta automatizada.
- Implementación de las funcionalidades de la herramienta automatizada.
- Realización de pruebas para determinar anomalías en la herramienta automatizada.

Estructuración del trabajo de diploma

El presente trabajo investigativo está estructurado en 3 capítulos:

Capítulo 1: "Fundamentación teórica". Incluye el estado del arte del tema, se describe además la metodología, herramientas utilizadas para el desarrollo de la solución.

Capítulo 2: "Características del sistema". Se incluye en este capítulo la propuesta del modelo y el sistema, se especifican las Historias de Usuario que se encuentran dentro de la Fase de Exploración así como las diferentes características del sistema, además se realiza la Fase de Planificación donde se concretó la estimación por esfuerzo de las Historias de Usuario, plan de iteraciones así como su duración y el plan de entregas.

Capítulo 3: "Implementación del Sistema". Se realiza la implementación de la solución propuesta. Se realizan pruebas unitarias, de aceptación y se mide la Calidad en el Uso al producto para garantizar la calidad del mismo.

Capítulo 1. Fundamentación Teórica

1.1 Introducción.

En este capítulo se definen los términos y conceptos más importantes para lograr un mayor entendimiento de la investigación. Se manifiestan los resultados del estado del arte de las diferentes normas de calidad en el mundo y se especifican las particularidades que identifican las herramientas estudiadas. Además, se incluyen las principales características de las herramientas, así como de la metodología a utilizar para la implementación de la solución.

1.2 Normas de calidad

Para lograr un proceso de desarrollo con eficiencia han surgido una serie de herramientas, técnicas y modelos que facilitan a las organizaciones encargadas de la producción de software generar productos que cumplan las expectativas del cliente, herramientas que prometen ser la solución a los problemas de calidad, costos y tiempos de desarrollo; entre éstas podemos mencionar a los modelos de calidad.

(4)

Entre los muchos beneficios que aportan podemos destacar: (5)

Empresas que desarrollan software

- Controlar la calidad del producto.
- Mejorar las características del producto.
- Asegurar a sus clientes un nivel de calidad.
- Comparar con productos de la competencia.
- Posicionar su producto en el mercado.
- Aumentar las ventas del producto.

Empresas que adquieren software

- Conocer la calidad del producto que compran.
- Comparar entre distintas alternativas.
- Establecer acuerdos de nivel de servicio.
- Conseguir independencia del proveedor.
- Minimizar fallos en la producción.
- Reducir costos finales al comprar software.

1.2.1 ISO/IEC 9126

Esta norma internacional fue publicada en 1991 para evaluar la calidad del software.

El estándar se divide en cuatro partes que trata, respectivamente, los temas siguientes:

- 9126-1 Modelo de la calidad.
- 9126-2 Métricas externas.
- 9126-3 Métricas internas.
- 9126-4 Métricas de calidad en uso.

Modelo de calidad

El modelo de calidad que describe para la calidad de los productos software se divide en dos partes:

- Calidad externa y calidad interna
- Calidad durante el uso.

Se especifican seis características para la calidad interna y externa, que son además divididas en sub-características que se manifiestan externamente cuando el software se usa como una parte del sistema computarizado, y son un resultado de los atributos o cualidades internas del software. Para la calidad durante el uso se definen cuatro características. La calidad durante el uso es el efecto combinado que percibe el usuario de la calidad interna y externa del software. (6)

Las características definidas son aplicables a todo tipo de software, incluidos los programas de computación. Las características y sub-características proveen una consistente terminología sobre la calidad del software y además proveen un marco para especificar los requisitos de calidad para el software y permitiendo el intercambio entre las diversas capacidades del producto de software.

1.2.2 ISO/IEC 25000

ISO/IEC 25000, conocida como SQuaRE (Software Product Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. (3)

La familia ISO/IEC 25000 es el resultado de la evolución de otras normas anteriores, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones.

- ISO / IEC 2500n - División de Gestión de la Calidad. Las normas que conforman esta división definen todos los modelos comunes, términos y definiciones que se refieren, aún más por el resto de las normas de las series. La división también proporciona requisitos y lineamientos para una función de apoyo que se encarga de la gestión de los requisitos de software de calidad del producto, la especificación y evaluación.
- ISO / IEC 2501n - División Modelo de Calidad. Las normas que conforman esta división son actuales modelos detallados de calidad para los sistemas informáticos y de productos de software, la calidad del sistema en uso, y los datos. También proporciona una guía práctica sobre el uso de los modelos de calidad.
- ISO / IEC 2502n - División de Medición de la Calidad. Las normas que conforman esta división incluyen un modelo de software de calidad de productos de referencia de medición, definiciones matemáticas de las medidas de calidad, y una guía práctica para su aplicación. Se dan ejemplos de medidas internas y externas de la calidad del software, así como medidas para la calidad en uso. Elementos de Medida de la Calidad (QME) que forman bases para estas medidas.
- ISO / IEC 2503n – División de los Requisitos de calidad. Las normas que conforman esta división ayudan a especificar los requisitos de calidad, el uso de modelos de calidad y medidas de calidad.
- ISO / IEC 2504n - División de Evaluación de la Calidad. Las normas que conforman esta división proporcionan requisitos, recomendaciones y directrices para la evaluación de productos de software, realizado por los evaluadores, clientes y productores. El apoyo para la documentación de una medida como un módulo de la evaluación también está presente.

1.2.3 ISO/IEC 25010:2011

Esta norma se encuentra dentro de la División del Modelo de Calidad perteneciente a la familia de normas ISO/IEC 25000 .El modelo de calidad es el pilar fundamental para establecer un sistema de evaluación de la calidad del producto. En este modelo que reemplaza el estándar 9126-1 se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado y define sus modelos como:

- Un modelo de calidad en uso que se compone de cinco características (algunas de las cuales se subdividen en sub-características). Se relacionan con el resultado de la interacción cuando un producto se utiliza en un contexto particular de uso.
- Un modelo de calidad del producto que se compone de ocho características (que se subdividen en sub-características). Se refieren a propiedades estáticas de software y las propiedades

dinámicas del sistema informático. El modelo es aplicable a los productos de software y sistemas informáticos.

Las características definidas por ambos modelos son relevantes para todos los productos de software y sistemas informáticos. Estas características y sub-características proporcionan coherencia terminológica para especificar, medir y evaluar la calidad del producto software y sistemas informáticos.

Los modelos de calidad en esta norma internacional se centran en un equipo de destino que incluye el producto de software (aunque la Calidad en el Uso del modelo tiene en cuenta todo el sistema hombre-máquina). El equipo de destino también incluye hardware, productos de software no objetivo, los datos no objetivo, y los datos de destino, que es el objeto del modelo de calidad de datos (ISO / IEC 25012). (3)

1.2.4 Modelo de Calidad en el Uso

Este modelo de la norma ISO/IEC 25010 define cinco características relacionadas con los resultados de la interacción del usuario con el sistema: la eficacia, la eficiencia, la satisfacción, independencia de riesgo y cobertura de contexto.

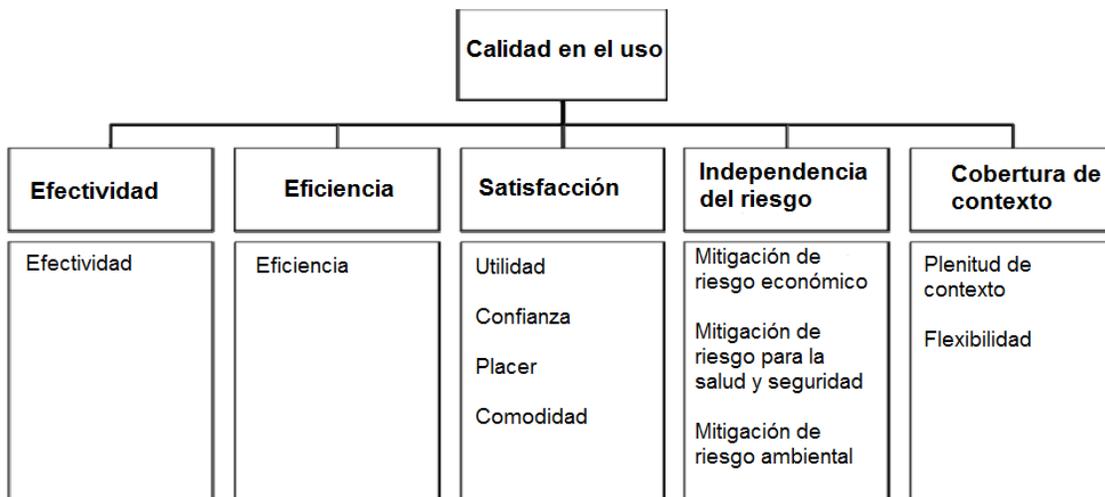


Figura 1: Modelo Calidad en el Uso

La Calidad en el Uso de un sistema caracteriza el impacto que el producto (sistema o software del producto) cuenta sobre los interesados. Está determinada por la calidad del software, el hardware y el entorno operativo, y las características de los usuarios, las tareas y el entorno social. Todos estos factores contribuyen a la Calidad en el Uso del sistema. (3)

Características y sub-características de la Calidad en el Uso (3)

Efectividad. La exactitud y la exhaustividad con la que los usuarios logran los objetivos específicos.

Eficiencia. Los recursos empleados en relación con la exactitud y la exhaustividad con la que los usuarios logran sus metas.

Satisfacción. El grado de satisfacción de las necesidades de los usuarios cuando un producto se utiliza en un contexto de uso especificado.

- **Utilidad.** El grado en que la parte interesada está satisfecho con su rendimiento percibido de objetivos pragmáticos, incluyendo los resultados aceptables percibidos de uso y consecuencias del consumo.
- **Confianza.** El grado en que el usuario está convencido de que el producto se comportará según lo previsto.
- **Placer.** El grado de placer que siente el usuario al ejecutar el sistema.
- **Comodidad.** El grado en que el usuario está satisfecho con la comodidad física.

Independencia del riesgo. El grado en que un producto o sistema mitiga el riesgo potencial para el estatus económico, la vida humana, la salud o el medio ambiente.

- **Mitigación de riesgo económico.** El grado en que un producto o sistema mitiga el riesgo potencial para el estatus financiero, la operación eficiente, la propiedad comercial, la reputación u otros recursos en los contextos de uso pretendidos.
- **Mitigación de riesgo para la salud y la seguridad.** El grado en que un producto o sistema mitiga el riesgo potencial para las personas en los contextos de uso pretendidos.
- **Mitigación de riesgo ambiental.** El grado en que un producto o sistema mitiga el riesgo potencial para la propiedad o el medio ambiente en los contextos de uso pretendidos.

Cobertura de contexto. El grado en que un producto puede ser utilizado con eficacia, eficiencia, independencia de riesgo y satisfacción en los contextos de uso especificados y en contextos más allá de los identificados inicialmente.

- **Plenitud de contexto.** El grado en que un producto puede ser utilizado con eficacia, eficiencia, independencia de riesgo y satisfacción en todos los contextos de uso especificados.
- **Flexibilidad.** El grado en que un producto puede ser utilizado con eficacia, eficiencia, independencia de riesgo y satisfacción en los contextos más allá de los especificados.

1.3 Herramientas para medir la calidad

El proceso de medir la calidad de software es difícil y engorroso, no es algo que se pueda determinar por una escala que brinde una evaluación, es un proceso que demanda tiempo y trabajo documentado

para analizar información y corregir errores. Con el uso de una herramienta automatizada basada en un modelo de calidad internacional que aporta características y sub-características por las cuales se debe regir la calidad de un software nos permitirá minimizar los errores de cálculo en el resultado de las métricas y obtener los valores de las métricas con el mínimo esfuerzo posible asegurando así el nivel de satisfacción del cliente.

1.3.1 Software Aided Quality in Use Assessment (SW-AQUA)

Es una herramienta para medir la Calidad en el Uso de sitios web basada en el estándar internacional 9126-1 que implementa las métricas de las características de Efectividad, Productividad y Satisfacción que a su vez cuenta con tres secciones: Evaluador, Usuarios y Evaluación.

La sección de Usuarios:

- Permite realizar el registro de los usuarios que participarán en las evaluaciones y eliminar los registros de los usuarios que no tiene participación en las evaluaciones. También ofrece opciones de búsqueda y modificación de registros de los usuarios.

La sección de Evaluador:

- Implementa características similares a los usuarios.

La sección de Evaluación:

- En esta sección se encuentra el control de las evaluaciones, es decir, su creación, eliminación, búsqueda, modificación y aplicación. Además implementa la opción de generación de resultado, en la cual se obtienen las métricas y se presentan los resultados para su posterior análisis.

Funcionamiento

La aplicación de la evaluación consiste en una prueba de usuario. El usuario prueba el sitio web en cuestión y realiza las tareas que previamente el evaluador le haya indicado. Mientras el usuario realiza las tareas propuestas, el evaluador observa su comportamiento y va registrando los acontecimientos con la ayuda de SW-AQUA.

Al final de la aplicación se pueden obtener de manera instantánea los resultados.

1.3.2 WebQEM_Tool

Esta es una herramienta privada perteneciente a la Universidad Nacional de la Plata de Argentina, está basada en el modelo Web QEM (Quality Evaluation Method) el cual es un sub modelo de la norma

internacional 9126-1. La principal función de esta herramienta es medir la calidad de los sitios web para esto implementa las métricas pertenecientes a las características de usabilidad, funcionalidad, confiabilidad, eficiencia, mantenibilidad y portabilidad. Esta herramienta permite a los evaluadores la administración de proyectos, pudiendo crear, abrir y modificar proyectos, los cuales contienen información de sitios Web a ser evaluados. Permite el ingreso y edición de indicadores de preferencia de calidad elementales (es decir, valores numéricos asignados a atributos) tomados de uno o varios sitios Web, lo cuales serán calculados brindando un indicador de calidad global para cada sitio dado un proyecto de evaluación. Esto habilita a evaluar y/o comparar por ejemplo la calidad de productos Web y dar recomendaciones. De este modo, se muestran los resultados a través de documentos con información textual, tabular y gráfica, generados dinámicamente a partir de las tablas almacenadas en la capa de datos.

1.3.3 Cálculo de las métricas externas

Es una herramienta realizada en formato Excel por el profesor José Manuel Santos Alonso basada en el estándar 9126-1 para medir la calidad externa de cualquier producto de software. Esta herramienta cuenta con una ayuda donde se describe brevemente el proceso de evaluación de la norma 9126-1, además tiene una tabla referente al resumen de la medición, ahí se ven los valores obtenidos por cada característica así como el grado de conformidad que tiene cada una. Presenta varias tablas referentes al proceso de evaluación, cada tabla tiene una sección para la medición de la calidad y el resumen de la medición. La sección de medición cuenta con las características Funcionalidad que a su vez tiene las sub-características de Idoneidad, Exactitud e Interoperatividad. La segunda característica es Fiabilidad que tiene las sub-características de Madurez, Tolerancia ante fallos y Recuperabilidad. La última que trata la herramienta es Usabilidad que tiene sub-características como Comprensibilidad y Atracción, además tiene campos para la recolección de datos referentes a los atributos de las métricas y el peso de cada sub-característica. Esta sección también cuenta con las tablas de interpretación de resultados de las métricas según la escala de nivel requerido que tiene cuatro valoraciones (Mal, Regular, Bien y Muy Bien). En las siguientes imágenes se puede ver la estructura de la característica de Funcionalidad y las tablas de interpretación de resultado de las métricas.

Programa Componente No.		n								
Tabla 1 Medición de la Calidad del Componente										
Subcaracterísticas	Denominación de la Métrica	Variables	Descripción	Datos (entrar)	Peso (entrar)	Medición, fórmula	Interpretación del valor obtenido	Resultados calculados		
								De la medición	Puntos asignados	Puntos asignados a la Subcaracterística (Peso*Promedio Puntos/función)
Métricas de Funcionalidad										
Idoneidad	Adecuación Funcional	A	Número de funciones en las cuales se detectaron problemas en la evaluación.	1	1	$X = 1 - A/B$	1 ($0 \leq X \leq 1$)	0,9846154	3	3
		B	Número de funciones evaluadas.	65						
	Completitud de la Implementación Funcional	A	Número de funciones perdidas detectadas en la evaluación.	0		$X = 1 - A/B$	1 ($0 \leq X \leq 1$)	1	3	
		B	Número de funciones descritas en especificación de requisitos.	65						
	Cobertura de la Implementación Funcional	A	Número de funciones incorrectamente implementadas o funciones perdidas detectadas.	13		$X = 1 - A/B$	1 ($0 \leq X \leq 1$)	0,8	3	
		B	Número de funciones descritas en la especificación de requisitos.	65						

Figura 2 Tabla de Característica Funcionalidad de Cálculo de métricas

Escala para nivel requerido 1			Escala para nivel requerido 0		
Rango del dato o nivel obtenido	Valoración	Puntos	Rango del dato o nivel obtenido	Valoración	Puntos
0 - 0,2	Mal	0	0 - 0,2	Muy Bien	3
0,3 - 0,5	Regular	1	0,3 - 0,5	Bien	2
0,6 - 0,7	Bien	2	0,6 - 0,7	Regular	1
0,8- 1	Muy Bien	3	0,8- 1	Mal	0

Figura 3 Interpretación de resultado según la escala

La sección resumen de la medición tiene campos referentes a los valores obtenidos por característica los cuales se determinan realizando el promedio de los valores de las sub-característica analizada una vez realizado el cálculo de las métricas y determinado el valor de las sub-características, además presentan los campos de grado de conformidad y criterio de evaluación que determinan si el software evaluado es: Conforme, suficientemente conforme, medianamente conforme, no conforme y si necesita pequeñas modificaciones , grandes modificaciones o ninguna.

Tabla 2. Resumen de la medición

RESUMEN DEL CUESTIONARIO COMPONENTE INDIVIDUAL				
Característica	Promedio	Valor (Redondeado)	GRADO DE CONFORMIDAD	CRITERIO DE EVALUACIÓN
1	2	3	4	5
Funcionalidad	1,72	2	Suficientemente Conforme	Pequeñas Modificaciones
Confiabilidad	1,46	1	Medianamente Conforme	Grandes Modificaciones
Usabilidad	2,25	2	Suficientemente Conforme	Pequeñas Modificaciones

Figura 4 Tabla resumen de Evaluación

Una vez finalizado el análisis de estas herramientas se puede concluir que se hace necesario la implementación de una nueva herramienta debido a las siguientes razones:

- Las herramientas utilizan el estándar internacional 9126-1 el cual deroga actualmente provocando que sean obsoletas.
- La herramienta Cálculo de métricas no implementa el modelo Calidad en el Uso del cual no se tiene mucho conocimiento.
- La herramienta SW-AQUA nada más mide la Calidad en el Uso de los sitios web porque no emplean todas las características y métricas pertenecientes al modelo que implementan.
- La herramienta SW-AQUA no da una valoración en cuanto al nivel de calidad que presenta el software evaluado.
- La herramienta WebQEM_Tool es privativa por lo que su uso está limitado.

Tabla 1: Tabla comparativa de Herramientas

Nombre	Norma que Utilizan	Modelo de Calidad	Valoración resultados	Productos que evalúan
SW-AQUA	9126-1	Calidad en el Uso	NO	Aplicaciones web
WebQEM_Tool	9126-1(Web QEM)	Mixto	Si	Aplicaciones web
Cálculo de las métricas externas	9126-1	Calidad Externa	Si	Cualquier producto de software
QU-SWMeter(Herramienta desarrollada)	25010	Calidad en el Uso	Si	Cualquier producto de software

1.4 Metodología de desarrollo de software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Estas sirven de guía para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben tener. Detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

1.4.1 Programación Extrema (XP)

La Programación Extrema es una metodología ágil de desarrollo de software que se basa en los valores de sencillez, comunicación, valor y el respeto. Desarrollada por Kent Beck, XP surge como respuesta y posible solución a los problemas derivados del cambio en los requerimientos. Básicamente plantea que se trabaje directamente con el cliente haciendo pequeñas iteraciones y como resultado mini entregas cada un período determinado. Se utilizan las historias de usuario con el objetivo que cada una no puede demorarse más de una semana, permitiendo el trabajo en parejas, donde los programadores menos experimentados aprenden de los más experimentados.

Justificación de la metodología seleccionada

Para guiar el desarrollo de esta aplicación se decide emplear una metodología ágil específicamente XP. A continuación se exponen las razones de su elección:

- Es un proyecto pequeño donde todo el trabajo se realiza por una pareja de programadores.
- El cliente forma parte del equipo de desarrollo, logrando una mejor retroalimentación, corrección de errores y así un producto que satisfaga todas las necesidades del mismo.
- No se necesita la generación de tantos artefactos y roles pues como se había planteado es un proyecto pequeño que está centrado en ser desarrollado en el menor tiempo posible.
- Los requisitos tienden a cambiar frecuentemente, así, conforme avanza el trabajo, el cliente puede agregar nuevas Historias de Usuario, dividir las o simplemente eliminarlas. XP permite al equipo la modificación de sus planes conforme a todo lo anterior.

1.5 Lenguaje de programación

JAVA

Lenguaje de programación orientado a objetos con una sintaxis fácilmente accesible y un entorno robusto creado por la compañía Sun Microsystems en 1995. La plataforma se adquiere de forma gratuita y hoy en día se ejecuta en cientos de computadoras personales en todo el mundo y en diferentes dispositivos, como lo son los móviles, aparatos de televisión, aparatos especializados y consolas de juegos.

Ventajas (7)

- Es un lenguaje simple, elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos.
- Un lenguaje seguro, tiene ciertas políticas que evitan que se puedan codificar virus con este lenguaje.

Desventajas (7)

- Los programas hechos en Java no tienden a ser muy rápidos.

1.6 Herramientas

1.6.1 Entorno de desarrollo Integrado (IDE). NetBeans7.4

Este IDE fue seleccionado porque el equipo de desarrollo cuenta con un conocimiento previo del mismo lo que facilita la implementación de la herramienta. Además es libre y gratuito sin restricciones de uso garantizando su disponibilidad y acceso por parte del equipo de desarrollo, otra razón por la

que se seleccionó fue por las facilidades que le ofrece al programador para escribir programas y compilarlos como él autocompletado de código y búsquedas avanzadas de errores.

1.6.2 BizAgi 2.3.0.5

“BizAgi es un modelador de procesos que permite representar de forma esquemática todas las actividades y decisiones que se toman en el negocio. Con una interfaz que recuerda a Microsoft Office, BizAgi Process Modeler cumple con el estándar BPMN (Business Process Management Notation)”. (8)

El Modelador de Procesos BizAgi permite crear diagramas de procesos de forma sencilla y muy rápida gracias a su interfaz amigable muy parecida a la de Microsoft Office. Este modelador permite validar los diagramas sin necesidad de esperar a que estén terminados completamente.

Se utiliza esta herramienta porque permite modelar y visualizar los procesos, lo cual evita muchos inconvenientes a la hora de implementarlos.

1.6.3 Visual Paradigm 8.0

La herramienta seleccionada para el modelaje de los diagramas que representarán gráficamente el ciclo de vida del desarrollo del producto es el Visual Paradigm, porque esta es una herramienta utilizada para modelar siguiendo el estándar UML. *“Esta herramienta favorece un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación”* (9). A continuación se listan las ventajas que se tuvieron en cuenta para la selección de esta herramienta:

- Disponibilidad en múltiples plataformas (Windows, Linux).
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Licencia: gratuita y comercial.
- Posee características gráficas muy cómodas que permite modelar todos los diagramas.
- Permite la integración con el IDE de desarrollo NetBeans y con el gestor de base de datos SQLite.

1.6.4 IReport 5.1.0

Es una aplicación gratuita y de código abierto escrito en java que permite modelar y diseñar reportes de manera visual, soporta jdbc para los orígenes de datos, es fácil de instalar y tiene una alta usabilidad debido a que tiene asistentes de configuración de plantillas con opciones de pre visualización de la exportación de los reportes en formatos PDF, HTML, TXT y XML. (10)

Esta herramienta fue seleccionada porque está escrita en Java eliminando cualquier problema de compatibilidad, además es de fácil utilización, porque el diseño de los reportes esta guiado por asistentes de configuración de plantillas.

1.6.5 SQLite 3.5.1

Es un proyecto de dominio público que implementa una pequeña librería de aproximadamente 500KB programada en lenguaje C, que funciona como un sistema de gestión de base de datos relacionales. Es un sistema completo de bases de datos que soporta múltiples tablas, índices, triggers y vistas. No necesita un proceso separado funcionando como servidor porque lee y escribe directamente sobre archivos que se encuentran en el disco duro. El formato de la base de datos es multiplataforma e indistintamente se puede utilizar el mismo archivo en sistemas de 32 y 64 bits.

Este sistema gestor de base de datos fue seleccionado porque permite que la base de datos se encuentre junto con la herramienta garantizando la portabilidad de la misma.

1.7 Conclusiones

El estudio realizado les permitió a los autores definir las diferentes características y sub-características del modelo Calidad en el Uso de la norma internacional ISO/IEC25010 así como la metodología y las herramientas necesarias para el desarrollo de la aplicación.

Capítulo 2 Características del sistema

2.1 Introducción

En este capítulo se describen las características del sistema. Se identifican los procesos existentes con el objetivo de comprender aquellos que intervienen en la evaluación de la calidad, para un correcto diseño e implementación de los mismos. También se hace alusión a las fases de Exploración y Planificación propias de la metodología de desarrollo utilizada, donde se confeccionan las Historias de Usuario importantes para cada iteración definida por el equipo de desarrollo.

2.2 Descripción del proceso

El proceso de evaluación de la calidad de software consta de 5 fases.

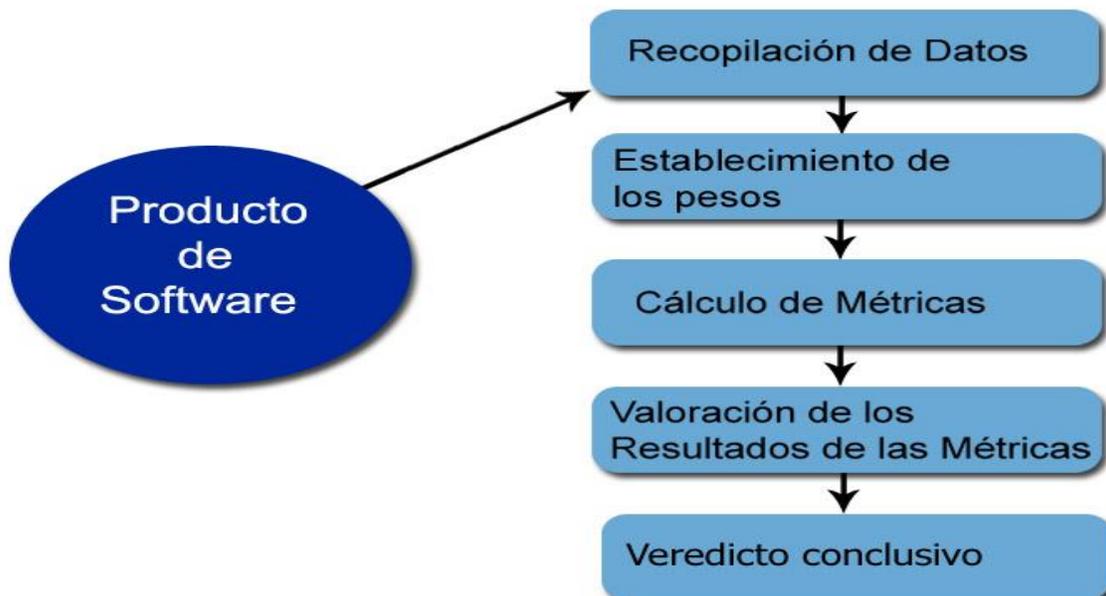


Figura 5: Fases del proceso de evaluación

Comienza con la recopilación de los datos sobre las variables de las métricas por el encargado de realizar la evaluación; después de haber recopilado los datos de las variables se procede al establecimiento de los pesos que posee cada sub-característica en el proyecto de software sometido a evaluación. Una vez recopilados los datos y establecido el peso se prosigue con el cálculo de las métricas. Después de haberse hecho el cálculo de las métricas se hace una valoración de los resultados de las pruebas y el comportamiento de las características de calidad. Es decir, se le da una puntuación a cada una de las sub-características según el resultado del cálculo de las métricas de esa sub-característica y el peso que tenga. Por último se llega a una conclusión final sobre el grado de

aceptación que tiene el producto que se está evaluando. Este grado puede ser aceptado, diferido o no aceptado.

2.3 Objeto de automatización

El subproceso de evaluación de la Calidad en el Uso de los productos de software y obtener reportes acerca de los resultados de la evaluación.

2.4 Propuesta del modelo y el sistema

2.4.1 Modelo para la Calidad en el Uso

Para seleccionar el modelo se efectuaron varios pasos, los cuales se especifican a continuación:

- Seleccionar las características: Todas las características del modelo Calidad en el Uso fueron seleccionadas.
- Seleccionar las sub-características: La única sub-característica no seleccionada fue la de Mitigación de riesgo ambiental debido a que la norma ISO/IEC 25022 se encuentra en proyecto no tiene bien definida la métrica que la compone causando incomprensión de la misma en los autores.
- Seleccionar las métricas: De las sub-características seleccionadas se escogieron la gran mayoría de las métricas, algunas métricas no fueron elegidas por su complejidad e incomprensión debido a que la norma ISO/IEC 25022 se encuentra en fase de desarrollo y no se encuentran definidas adecuadamente.

2.4.2 Métricas seleccionadas

A continuación se detallan las métricas pertenecientes al modelo calidad en uso que se encuentran estandarizadas internacionalmente y definidas en la ISO/IEC 25022.

Métricas de efectividad.

1.1 Efectividad.

a) Tarea completada.

Se propone la métrica **(1.1.a)** que permite hacer un análisis de las tareas que se realizaron.

$$X = A/B$$

A -Número de tareas completadas.

B -Total de tareas.

b) Frecuencia de error.

Se propone la métrica **(1.1.b)** que permite hacer un análisis de la frecuencia de errores por tarea.

$$X = A/B$$

A -Número de errores.

B–Número de tareas.

Métricas de Eficiencia.

2.1 Eficiencia.

a) Tiempo.

Se propone la métrica **(2.1.a)** que permite hacer un análisis sobre el tiempo que demora en completar X.

$$X = Tt/Ta$$

A–Tiempo objetivo.

B–Tiempo real.

b) Tiempo relativo de la tarea.

Se propone la métrica **(2.1.b)** que permite hacer un análisis sobre el tiempo que demora un usuario normal en realizar una tarea en comparación con un experto.

$$X = B/A$$

A–Tiempo de la tarea del usuario normal.

B–Tiempo de la tarea del usuario experto.

c) Tiempo relativo de acciones del usuario.

Se propone la métrica **(2.1.c)** que permite hacer un análisis sobre si el usuario realiza el número de acciones necesarias.

$$X = A/B$$

A–Número de acciones realizadas por el usuario.

B–Número de acciones realmente necesarias.

Métricas de Satisfacción.

3.1 Utilidad.

a) Proporción de quejas del cliente.

Se propone la métrica **(3.1.a)** que permite hacer un análisis sobre el número de clientes que presentan quejas sobre el sistema.

$$X = A/B$$

A—Número de clientes que se quejan.

B—Número total de clientes.

b) Uso discrecional.

Se propone la métrica **(3.1.b)** que permite hacer un análisis sobre los usuarios que usan el sistema.

$$X = A/B$$

A—Número de veces que se utiliza el software.

B—Número de veces que se pretende usar.

3.2 Confianza.

a) Escala de confianza.

Se propone la métrica **(3.2.a)** que permite hacer un análisis sobre si el usuario confía que el sistema se comportará de la manera adecuada.

$$X = A/B$$

A—Cuestionario psicométrico.

B—Promedio de la población.

3.3 Placer.

a) Escala de placer.

Se propone la métrica **(3.3.a)** que permite hacer un análisis sobre si el usuario siente placer al satisfacer sus necesidades personales.

$$X = A/B$$

A—Cuestionario psicométrico.

B—Promedio de la población.

3.4 Comodidad.

a) Escala de comodidad.

Se propone la métrica **(3.4.a)** que permite hacer un análisis sobre si el usuario está satisfecho con su comodidad física.

$$X = A/B$$

A–Cuestionario psicométrico.

B–Promedio de la población.

Métricas de Independencia de riesgo.

4.1 Mitigación de riesgo económico.

a) Retorno de la inversión.

Se propone la métrica **(4.1.a)** que permite hacer un análisis sobre los beneficios obtenidos después de una inversión.

$$X = A/B$$

A–Beneficios obtenidos.

B–Cantidad invertida.

b) Tiempo para lograr el retorno de la inversión.

Se propone la métrica **(4.1.b)** que permite hacer un análisis sobre si los beneficios obtenidos después de una inversión fueron en un tiempo aceptable.

$$X = A/B$$

A–Tiempo para alcanzar el ROI.

B–Tiempo aceptable para lograr ROI.

c) Tiempo de entrega.

Se propone la métrica **(4.1.c)** que permite hacer un análisis acerca del tiempo de las entregas que se realizan.

$$X = A/B$$

A–Tiempo de entrega real.

B–Tiempo de entrega objetivo.

d) Artículos faltantes.

Se propone la métrica **(4.1.d)** que permite hacer un análisis acerca de los artículos faltantes.

$$X = A/B$$

A–Número real de elementos que faltan.

B–Objetivo de elementos faltantes.

e) Ingresos de cada cliente.

Se propone la métrica **(4.1.e)** que permite hacer un análisis de los ingresos de cada cliente.

$$X = A/B$$

A–Ingresos reales de un cliente.

B–Objetivo de los ingresos de un cliente.

f) Errores con consecuencias económicas.

Se propone la métrica **(4.1.f)** que permite hacer un análisis de la frecuencia de los errores humanos o del software que causan daños económicos.

$$X = A/B$$

A– Número de errores con consecuencias económicas.

B–Número total de situaciones de uso.

g) Software corrupto.

Se propone la métrica **(4.1.g)** que permite hacer un análisis de la frecuencia de la corrupción de software ocasionada por los errores humanos o del software.

$$X = A/B$$

A–Número de ocurrencias de corrupción del software.

B–Número total de situaciones de uso.

4.2 Mitigación de riesgo para la salud y la seguridad.

a) Frecuencia de problema de salud y seguridad de los usuarios.

Se propone la métrica **(4.2.a)** que permite hacer un análisis acerca de la frecuencia de los usuarios que reportan problemas en cuanto al uso del producto.

$$X = A/B$$

A–Número de usuarios con problemas de salud.

B–Número total de usuarios.

b) Impacto de problemas de salud y seguridad en los usuarios.

Se propone la métrica **(4.2.b)** que permite hacer un análisis acerca del impacto de los problemas de salud y seguridad en los usuarios.

$$X = N * T * S$$

N = Número de personas afectadas.

T = Tiempo.

S = Grado de importancia.

c) Seguridad de las personas afectadas por el uso del sistema.

Se propone la métrica **(4.2.c)** que permite hacer un análisis acerca del impacto de los problemas de salud y seguridad en los usuarios.

$$X = A/B$$

A–Número de personas que ponen en peligro.

B–Número total de personas potencialmente afectadas por el sistema.

Métricas de Cobertura de contexto.

Contexto Exhaustivo.

a) Contexto Exhaustivo.

Se propone la métrica **(5.1.a)** que permite hacer un análisis acerca de los contextos en que el producto puede ser usado.

$$X = A/B$$

A–Número de contextos con usabilidad inaceptable.

B–Número total de contextos de uso.

Flexibilidad.

a) Contexto de uso flexible.

Se propone la métrica **(5.2.a)** que permite hacer un análisis acerca de los contextos en que el producto puede ser usado.

$$X = A/B$$

A–Número de contextos adicionales en los cuales el producto sería utilizable.

B–Número total de contextos adicionales en el que se podría utilizar el producto.

2.4.3 Propuesta del sistema

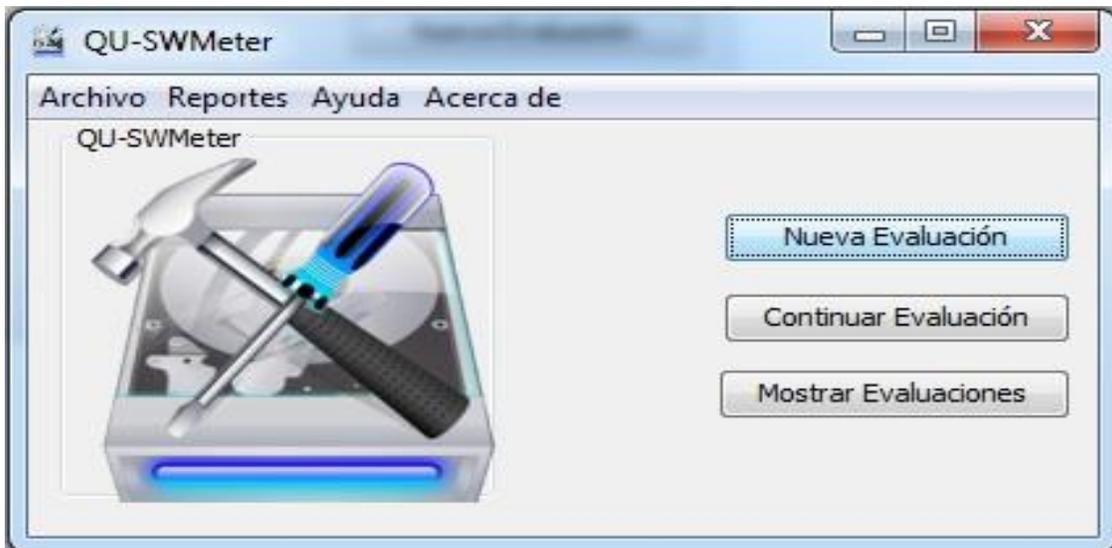


Figura 6 Interfaz Principal del sistema

Para realizar una evaluación el evaluador deberá seleccionar en la interfaz principal de la herramienta la opción del menú “Nueva Evaluación”, esta lo conducirá a una nueva interfaz donde se registrará los datos del software a evaluar y los del evaluador o si el software y el evaluador ya se encuentran en el sistema simplemente se seleccionan de una lista.

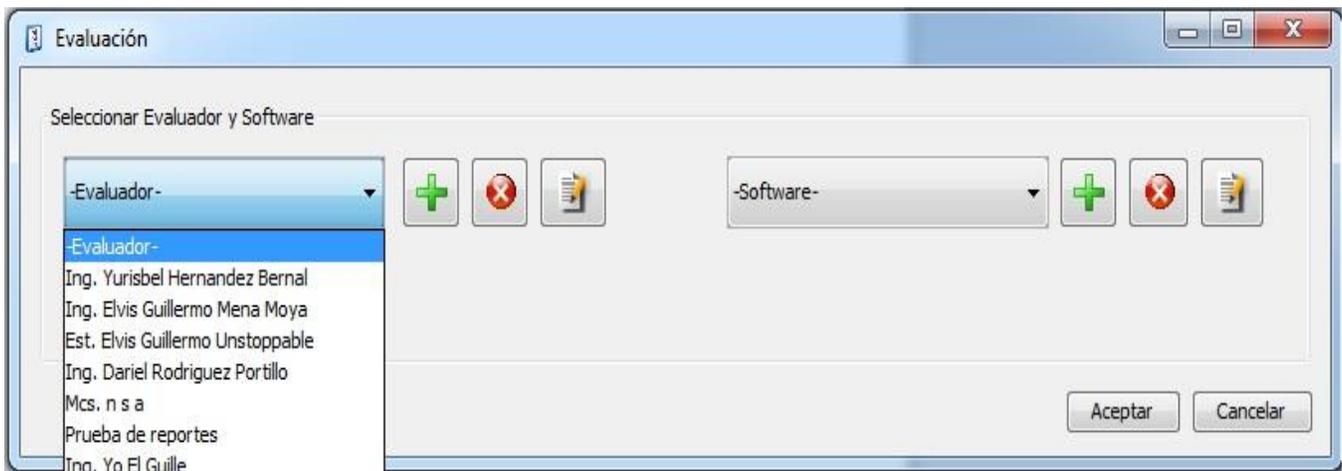


Figura 7 Interfaz de Selección de Evaluador y Software

Una vez confirmados los datos se abrirá un nuevo formulario con las métricas pertenecientes a las características y sub-características del modelo Calidad en el Uso; en esta interfaz el usuario seleccionará los pesos correspondientes a cada sub-característica del modelo. Estos pesos

determinan el grado de importancia de cada sub-característica sobre el producto que se está evaluando, se sugiere para el peso la siguiente estructura: 2 si la relevancia es alta, 1 si es media y 0 si es baja.

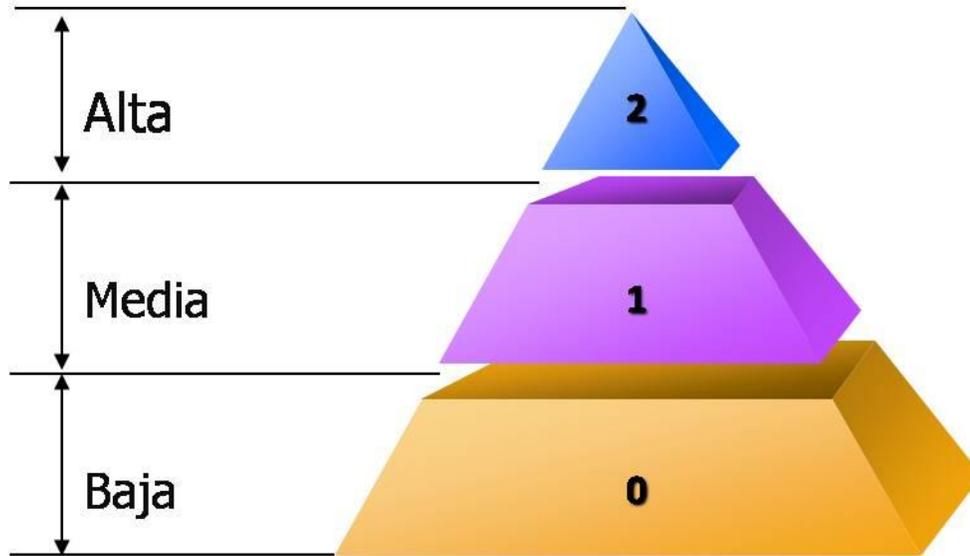


Figura 8: Estructura para el peso

Una vez confirmado los pesos 1 ó 2, se habilitarán los campos de recolección de datos que previamente fueron obtenidos por el evaluador y para esto se brinda la **plantilla # 1 (Ver anexo 1)**; una vez los valores son introducidos por el usuario y seleccionada la opción de “Medir” inmediatamente el sistema devolverá los resultados de las métricas, después si el usuario desea conocer la evaluación que propone el sistema acerca de la calidad del software evaluado deberá elegir la opción “Evaluar” que le brindará un veredicto conclusivo acerca del nivel de calidad del software. Una vez analizado el veredicto el usuario podrá salir del sistema o volver al menú principal.

The screenshot shows a software window titled "Medición" with a standard Windows-style title bar. The main content area is divided into several sections:

- Header:** "Evaluador: Mcs n s a", "Software: Testfgh 1.1", and "Versión: 1".
- Características:** A set of tabs including "Efectividad", "Eficiencia", "Satisfacción" (which is selected), "Cobertura de contexto", and "Independencia de riesgo".
- Subcaracterísticas de Satisfacción:** A large container with several sub-sections:
 - Métricas Pesos:** Includes dropdowns for "Utilidad" (values 0, 1, 2), "Placer" (value 0), and "Comodidad" (value 0).
 - Métricas de Utilidad:** Includes "Proporción de quejas del cliente" with input fields for "A: 0" and "B: 0 X:"; and "Uso discrecional" with input fields for "A: 0" and "B: 0 X:".
 - Métricas de Confianza:** "Escala de confianza" with a dropdown set to "0".
 - Métricas de Placer:** "Escala de placer" with a dropdown set to "0".
 - Métricas de Comodidad:** "Escala de comodidad" with a dropdown set to "0".
- Footer:** Three buttons: "Posponer", "Medir", and "Cancelar".

Figura 9 Interfaz de Características y Sub-Características del modelo Calidad en el Uso



Figura 10 Interfaz de Veredicto conclusivo

Si el usuario no ha terminado la evaluación porque no tiene los datos referentes a las métricas tiene la opción de guardar el progreso en esa misma interfaz a través de la opción "Posponer". Para continuar una evaluación el usuario deberá seleccionar en el menú de la interfaz principal "Continuar Evaluación" e inmediatamente le saldrá un formulario para seleccionar la evaluación que desea continuar, después se mostrará la interfaz de contraseña donde el evaluador introducirá la contraseña que le corresponde, una vez confirmada la misma la herramienta le brindará la interfaz de las características y sub-características del modelo Calidad en el Uso con el progreso hasta ese momento y se podrá continuar con la evaluación.

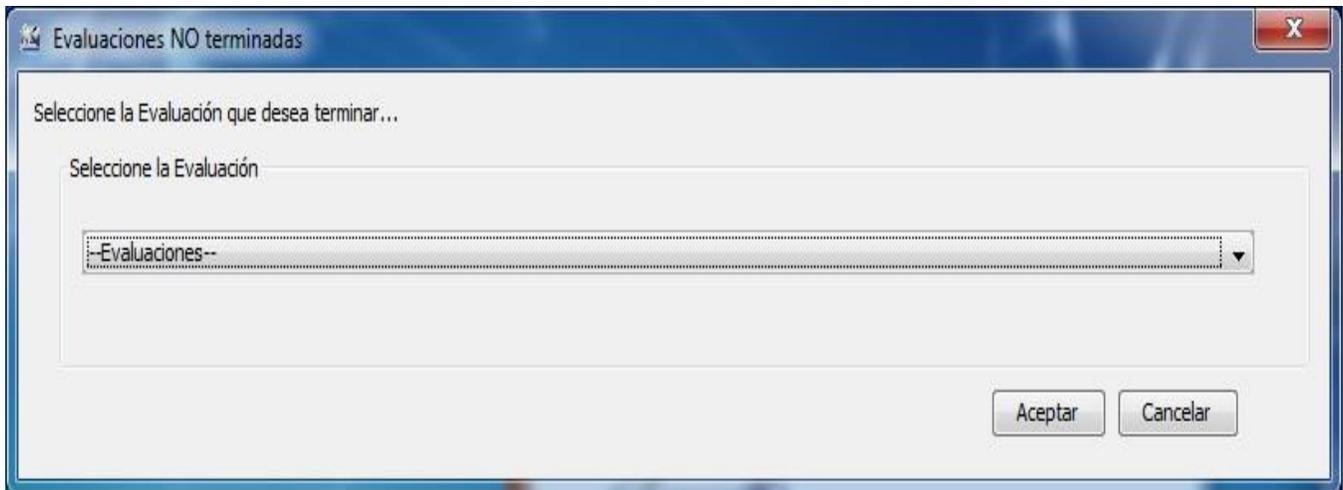


Figura 11 Interfaz de Evaluaciones no terminadas



Figura 12 Interfaz de Contraseña

La herramienta será capaz de exportar reportes de los proyectos de evaluación, ya sean resumidos por el resultado de las características (**Ver Anexo #2**), o el resultado de la evaluación en general (**Ver Anexo #3**), también tendrá la posibilidad de insertar, deshabilitar, modificar y listar tanto los software evaluados como los evaluadores y visualizar las evaluaciones realizadas. Las imágenes correspondientes a estas funcionalidades se encuentran en el **Anexo #4**. Además la herramienta contará con una ayuda que describirá los principales procesos para mayor entendimiento.

2.5 Características del Sistema

Para un correcto funcionamiento del sistema se deben tener en cuenta las siguientes características:

Usabilidad.

Se necesitará una preparación previa para operar con el sistema. Se requiere un nivel medio de conocimientos de computación, aunque el manejo de la aplicación es sencillo, permitiendo la fácil comprensión por el usuario. Además el sistema contará con documentos que explican los valores de las métricas como ayuda.

Disponibilidad.

La herramienta será portable, permitiendo que el usuario siempre pueda evaluar y guardar los resultados de un software determinado en cualquier momento porque la base de datos se encuentra integrada con la herramienta.

Rendimiento.

La eficiencia de la herramienta estará dada por la velocidad de las consultas a la base de datos así como por la simplicidad del cálculo de las métricas.

Hardware.

Para la ejecución de la herramienta se debe disponer de una computadora como mínimo de 256Mb de RAM.

Software.

La herramienta podrá ser ejecutada en los sistemas operativos Windows y Linux y necesitara la JRE instalada.

Personas relacionadas con el sistema.

Las personas relacionadas con la herramienta serán aquellas encargadas de evaluar la Calidad en el Uso del software.

2.6 Fase de Exploración

La fase de Exploración es la primera fase definida por la metodología XP. Es aquí donde se define el alcance real del sistema permitiendo una familiarización del equipo de desarrollo con las herramientas, tecnologías y procesos. Esta fase comienza por la creación de una serie de historias de usuario, las cuales definen mediante su redacción qué es lo que verdaderamente necesita el cliente.

2.6.1 Historia de Usuarios (HU)

Las Historias de Usuario son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Las HU serán representadas mediante tablas divididas por las siguientes secciones:

- **Número:** Número de la historia de usuario incremental en el tiempo.
- **Nombre de Historia de Usuario:** El nombre de la historia de usuario sería para identificarlas mejor entre los desarrolladores y el cliente.
- **Modificación de Historia de Usuario Número:** Si sufrió alguna modificación anterior.
- **Usuario:** Involucrados en el desarrollo de la HU.
- **Iteración Asignada:** Número de la iteración.
- **Prioridad en negocio:** Alta, Media o Baja.
- **Riesgo en Desarrollo:** Alta, Media o Baja.
- **Puntos estimados:** Tiempo estimado que se demorará el desarrollo de la HU.
- **Descripción:** Breve descripción de la HU.
- **Observaciones:** Señalamiento o advertencia del sistema.

El cliente y el equipo de desarrollo trabajan en conjunto para definir como agrupar las HU para su lanzamiento. A continuación se muestran las HU.

Tabla 2 Realizar una evaluación de calidad

Historia de Usuario	
Número: 1	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Realizar una evaluación de calidad	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema selecciona la opción “Nueva Evaluación”, alto seguido aparecerá la interfaz de “Evaluación” donde el evaluador selecciona su nombre dando la posibilidad de que si no se encuentra en el sistema ejecuta la HU “Insertar Usuario”, después se prosigue a seleccionar el software a evaluar que si no se encuentra en el sistema ejecuta HU “Insertar Software”. Una vez confirmados los datos aparecerá la interfaz de selección de pesos y de introducción de los valores de las métricas, después de confirmados estos datos el evaluador selecciona la opción “Evaluar” alto seguido aparecerá la interfaz con el resultado de la evaluación.	
Observaciones:	

Tabla 3 Continuar una evaluación de calidad

Historia de Usuario	
Número: 2	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Continuar una evaluación de calidad	
Prioridad en negocio: Alta	Riesgo en desarrollo: medio
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema selecciona la opción "Continuar Evaluación", alto seguido aparecerá la interfaz de "Evaluaciones no terminadas" donde el evaluador selecciona la evaluación que desea terminar. Una vez confirmados los datos aparecerá la interfaz de contraseña donde el evaluador introducirá la que le corresponde, alto seguido la herramienta le brindará la interfaz de selección de pesos y de introducción de los valores de las métricas, después de confirmados estos datos el evaluador selecciona la opción "Evaluar" alto seguido aparecerá la interfaz con el resultado de la evaluación.	
Observaciones:	

Tabla 4 Visualizar las evaluaciones de calidad

Historia de Usuario	
Número: 3	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Visualizar las evaluaciones de calidad	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema selecciona la opción "Visualizar Evaluación", alto seguido aparecerá la interfaz de "Evaluaciones terminadas" donde el evaluador selecciona la evaluación de la cual desea ver sus resultados una vez confirmados los datos se ejecuta la HU "Exportar reportes de las evaluaciones en pdf".	
Observaciones:	

Tabla 5 Insertar Evaluador

Historia de Usuario	
Número: 4	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Insertar Evaluador	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario insertar un nuevo evaluador que estará conformado por el nombre, primer y segundo apellido, contraseña, confirmar contraseña y la categoría científica.	
Observaciones:	

Tabla 6 Insertar Software

Historia de Usuario	
Número: 5	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Insertar Software	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario insertar un nuevo software que estará conformado por el nombre y la versión.	
Observaciones:	

Tabla 7 Guardar los resultados de las evaluaciones de calidad

Historia de Usuario	
Número: 6	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Guardar resultados de las evaluaciones de calidad.	
Prioridad en negocio: Alta	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: Cuando el evaluador seleccione la opción “Medir” o “Posponer” de la interfaz de selección de pesos y de la introducción de valores de las métricas se guardarán automáticamente las evaluaciones.	
Observaciones:	

Tabla 8 Exportar reportes de la evaluación en formato pdf

Historia de Usuario	
Número: 7	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Exportar reportes de la evaluación en formato pdf.	
Prioridad en negocio: Alta	Riesgo en desarrollo: medio
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al evaluador exportar reportes en formato pdf de las evaluaciones.	
Observaciones:	

Tabla 9 Deshabilitar Software

Historia de Usuario	
Número: 8	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Deshabilitar Software	
Prioridad en negocio: Media	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario deshabilitar un software ya insertado seleccionando primero el software y después la opción “Deshabilitar” donde el software como la opción se encuentran en la interfaz de “Evaluación”.	
Observaciones:	

Tabla 10 Modificar Software

Historia de Usuario	
Número: 9	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Modificar Software	
Prioridad en negocio: Media	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario modificar un software ya insertado, seleccionando primero el software y después la opción “Modificar” donde el software y la opción se encuentran en la interfaz de “Evaluación”.	
Observaciones:	

Tabla 11 Deshabilitar Evaluador

Historia de Usuario	
Número: 10	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Deshabilitar Evaluador	
Prioridad en negocio: Media	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario deshabilitar un evaluador ya insertado seleccionando primero el evaluador y después la opción “Deshabilitar” donde el evaluador y la opción se encuentran en la interfaz de “Evaluación”.	
Observaciones:	

Tabla 12 Modificar Evaluador

Historia de Usuario	
Número: 11	Usuario: Elvis Guillermo Mena Moya Dariel Rodríguez del Portillo
Nombre historia: Modificar Evaluador	
Prioridad en negocio: Media	Riesgo en desarrollo: bajo
Puntos estimados: 1	Iteración asignada: 1
Descripción: El sistema deberá permitir al usuario modificar un evaluador ya insertado seleccionando primero el evaluador y después la opción “Modificar” donde el evaluador y la opción se encuentran en la interfaz de “Evaluación”.	
Observaciones:	

2.7 Fase de planificación

La actividad de planeación en la metodología XP comienza con la creación de una serie de HU que describen las características y funcionalidades requeridas para el software a construir. Durante la fase

de planificación se realiza una estimación del esfuerzo que costará implementar cada HU. Este se expresa utilizando como medida el punto.

Un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. Esta estimación incluye todo el esfuerzo asociado a la implementación de la HU.

2.7.1 Estimación de esfuerzo por Historias de Usuario

A continuación se muestra la estimación del esfuerzo por cada HU propuesta para el desarrollo de la aplicación:

Tabla 13 Estimación de esfuerzo por HU

Historia de Usuario	Puntos de estimación
Realizar una evaluación de calidad	1
Continuar una evaluación de calidad	1
Visualizar las evaluaciones de calidad	1
Insertar evaluador	1
Insertar software	1
Exportar reportes de la evaluación formato pdf	1
Deshabilitar software	1
Modificar software	1
Deshabilitar evaluador	1
Modificar evaluador	1
Guardar los resultados de las evaluaciones de calidad	1

2.7.2 Plan de Iteraciones

Después de ser identificadas y descritas las HU y estimar el esfuerzo dedicado a la realización de cada una de ellas, se procede a la planificación de la fase de implementación estableciendo una división de tres iteraciones

Iteración 1

En la iteración 1 se llevará a cabo el desarrollo de las HU del número 1 hasta el número 6, pertenecientes al proceso principal de evaluación de la calidad del software.

Iteración 2

En la iteración 2 se llevará a cabo el desarrollo de las HU del número 8 hasta el 10, brindará funcionalidades complementarias.

Iteración 3

En la iteración 3 se llevará a cabo el desarrollo de las HU número 7 referente a los reportes y la 11 referente a la modificación de los datos de los evaluadores.

2.7.3 Plan de duración de las iteraciones

El plan de duración de las iteraciones se encarga de mostrar las HU en el orden en que se implementarán en cada iteración así como la duración estimada de las mismas.

Tabla 14 Plan de duración por Iteraciones

Iteración	Orden de la Historias de usuario a implementar.	Duración total
1	Realizar una evaluación de calidad Continuar una evaluación de calidad Insertar evaluador Insertar software Visualizar una evaluación de calidad Guardar los resultados de las evaluaciones de calidad	6 semanas
2	Deshabilitar software Modificar software Deshabilitar evaluador	3 semanas
3	Exportar reportes de la evaluación formato pdf Modificar evaluador	2 semanas

2.7.4 Plan de entregas

El plan de entrega detalla la fecha final de cada iteración.

HA: Herramienta que automatiza el proceso de medir la Calidad en el Uso.

Tabla 15 Plan de entregas

Final de la Iteración 1 (25 de abril del 2014)	Final de la Iteración 2 (15 de mayo del 2014)	Final de la Iteración 2 (31 de mayo del 2014)
HA	HA	HA

2.7.5 Conclusiones

Basado en la información obtenida acerca de la propuesta del modelo y sistema, las historias de usuarios recogidas así como el plan realizado para la entrega por iteraciones se puede decir que se obtuvo la información necesaria para la implementación de la solución que dará respuesta a la problemática existente.

Capítulo 3 Implementación del Sistema y Prueba

3.1 Introducción

En este capítulo se describen las fases implementación y prueba, propias de la metodología de desarrollo XP. Se identifican y organizan las clases relevantes para las funcionalidades del sistema así como el patrón arquitectónico utilizado para la aplicación web. Se procederá al diseño de la base de datos y abordar las tareas de la ingeniería definidas. Por último, y no menos importante se realizarán las pruebas al software las cuales se derivan de las Historias de Usuario y Tareas de la Ingeniería que se han implementado como parte del lanzamiento del software.

3.2 Patrón de Arquitectura

Los patrones arquitecturales expresan la estructura fundamental para el sistema de software a desarrollar. Para el desarrollo del presente sistema el patrón de arquitectura a seguir será el N-Capa o programación por capas.

3.3 Arquitectura N-Capa

Es una arquitectura usada para aplicaciones de escritorio como para aplicaciones web. Esta arquitectura tiene como objetivo principal separar el código de presentación, código de lógica del negocio y código de almacenamiento, su principal ventaja consiste en que las capas de una aplicación pueden residir en una misma máquina o separadas en diferentes computadoras, además si se necesita realizar algún cambio o actualización puede ser realizado sin afectar a la aplicación porque cada capa es independiente con respecto a las demás asegurando que no exista código mezclado.

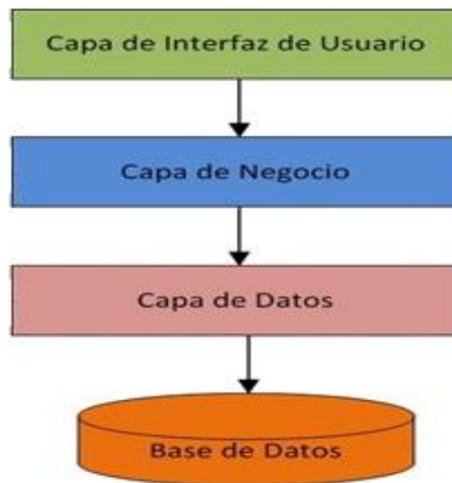


Figura 14: Estructura de Capas

Capa interfaz de usuario

En esta capa se presenta el sistema al usuario que tiene la característica de ser amigable y fácil de usar, es la que comunica información al usuario y captura su respuesta realizándole validaciones simples a los datos ingresados. Esta capa solo tiene comunicación con la capa de negocio.

Capa de negocio

Esta capa contiene el código de las funcionalidades de la aplicación que está dado por la información brindada por el usuario y los datos que se encuentran almacenados, esto es posible debido a la comunicación que existe con la capa de interfaz de usuario y la capa de datos.

Capa de datos

En esta capa es donde se encuentran almacenados los datos y es la que se encarga de acceder a los mismos, recibe solicitudes directamente de la capa de negocio para almacenar y extraer datos.

A continuación se presentan las clases de la herramienta divididas por las capas de la arquitectura:

Capa interfaz de usuario

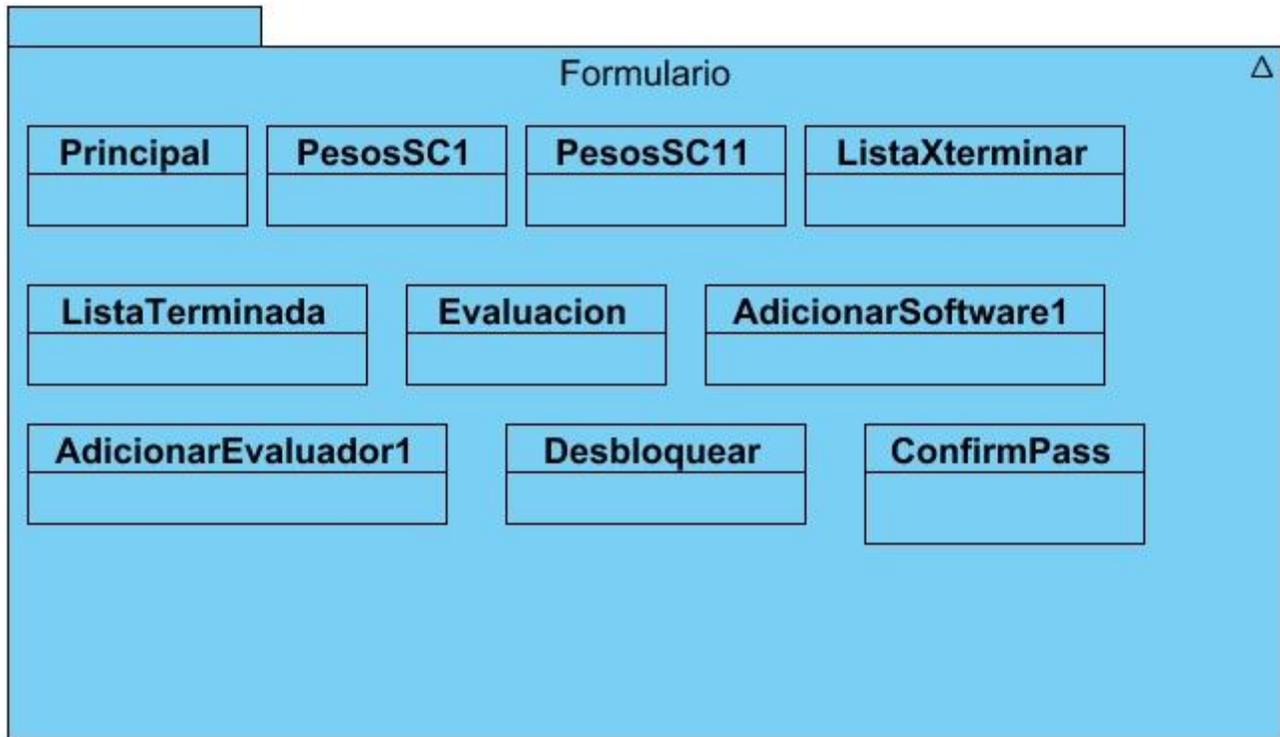


Figura 15: Capa Interfaz de usuario

Estas clases se encuentran dentro del paquete **Formulario** para representar la capa de interfaz de usuario.

Capa del Negocio

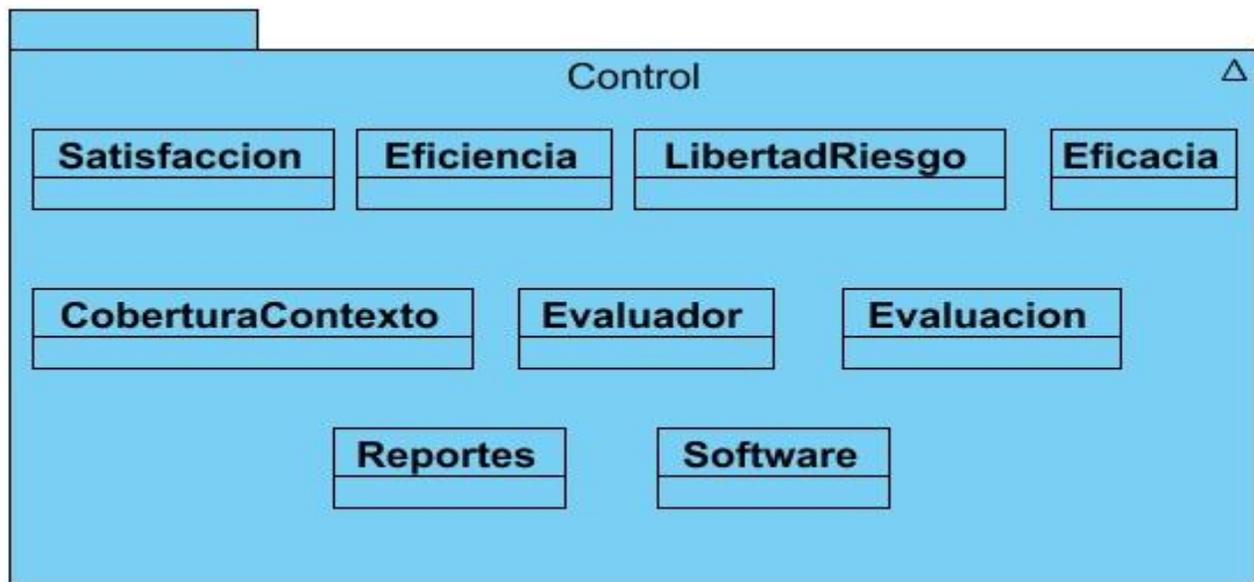


Figura 16: Capa del Negocio

Estas clases se encuentran dentro de la carpeta **Control** para representar la capa de negocio. Las mismas procesan las peticiones del usuario y son las encargadas de la gestión de cada una de las entidades como su nombre lo indica.

Capa de Datos

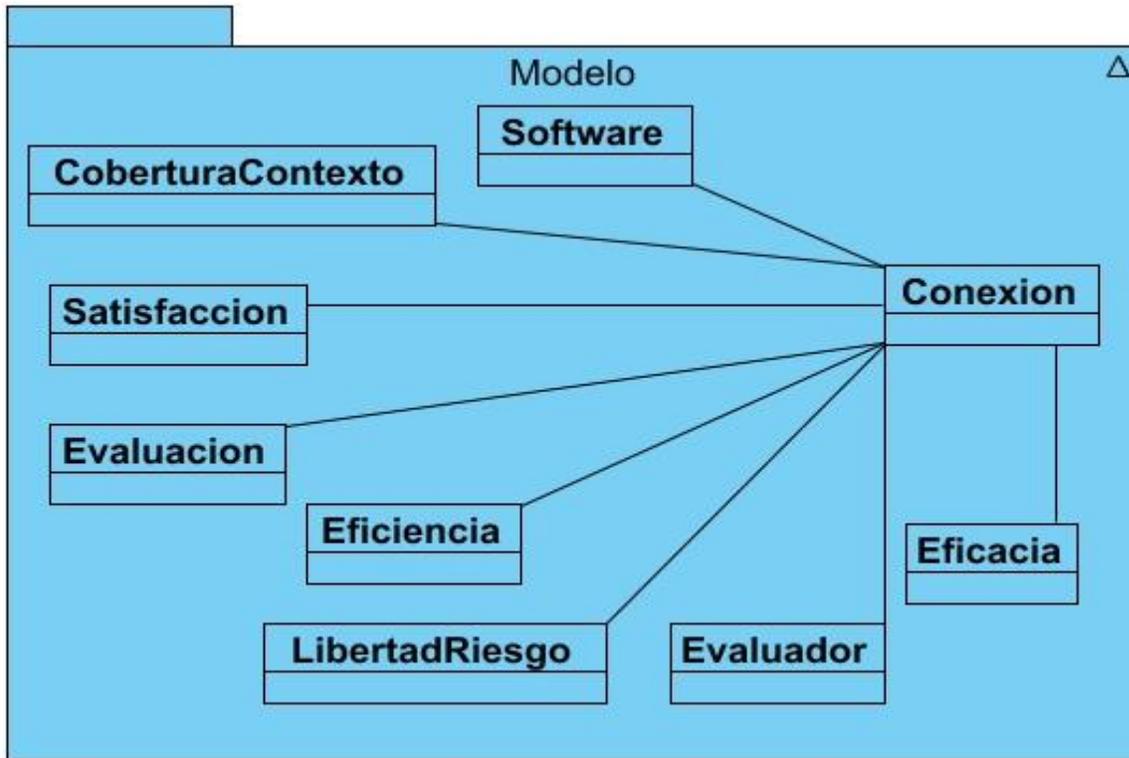


Figura 17 : Capa de datos

Estas clases se encuentran dentro de la carpeta **Modelo** para representar la capa de datos. Las mismas representan las entidades de la herramienta y Conexion es la clase encargada de la gestión de la conexión a la base de datos.

3.4 Patrones de diseño

Los patrones son parejas de problema/solución con un nombre, que codifican buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades. Estos patrones se dividen en patrones GRASP y GOF.

“Los patrones GRASP (patrones generales de software para asignar responsabilidad) describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones”. (11)

Experto: Asigna una responsabilidad al experto en información: la clase que tiene la información necesaria para realizar la responsabilidad. Brinda soporte al bajo acoplamiento porque los objetos se valen de su propia información para hacer lo que se les pide y a la alta cohesión porque el comportamiento se distribuye entre las clases que cuentan con la información requerida, lo que permite tener clases sencillas.

Alta cohesión: Asignar una responsabilidad de modo que la cohesión siga siendo alta. La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Una baja cohesión hace muchas cosas no afines o realiza trabajo excesivo.

3.5 Diseño físico de la base de datos

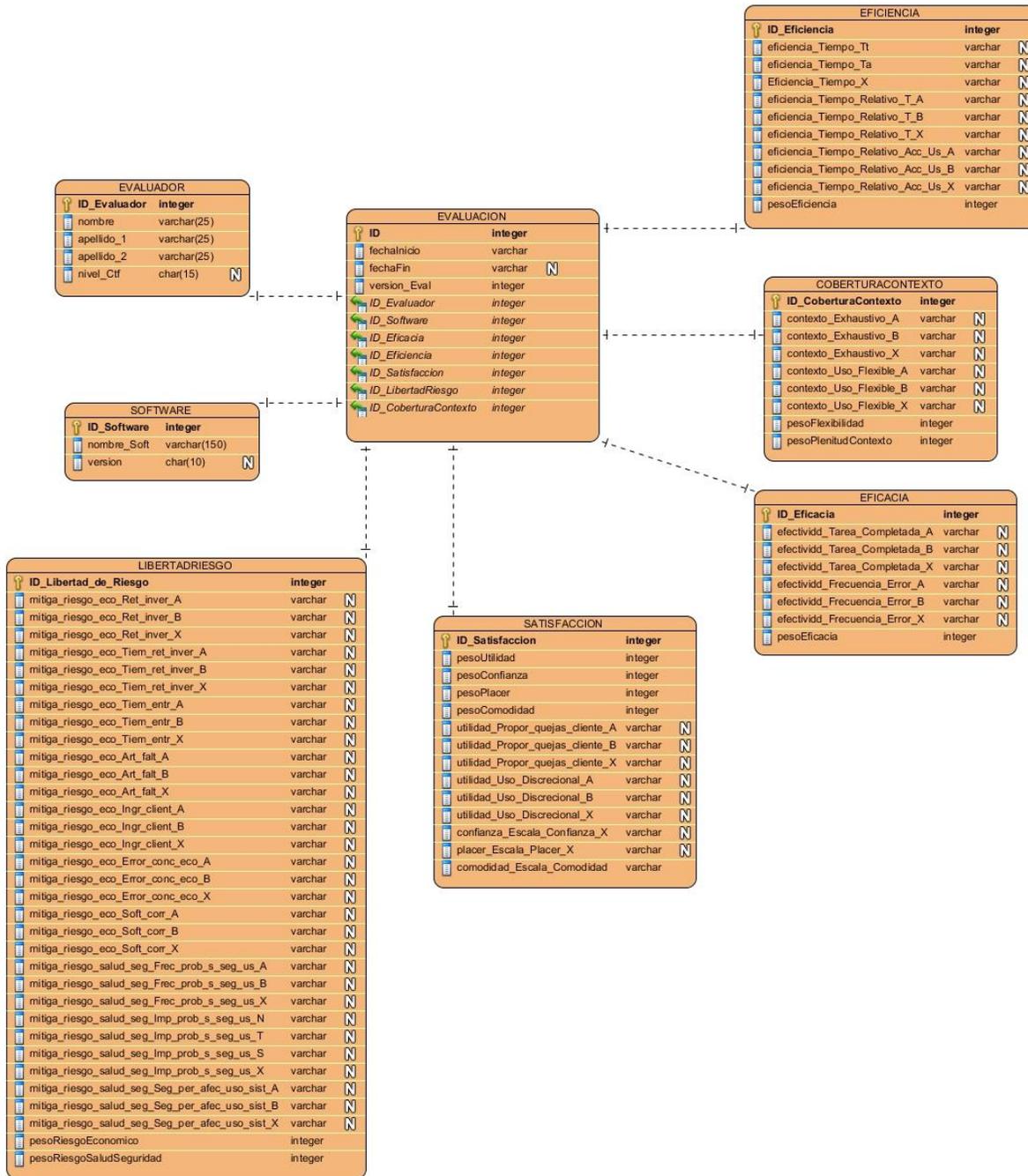


Figura 18: Diseño físico de la base de datos

3.6 Tarjetas Clase – Responsabilidad – Colaborador (CRC)

Las tarjetas Clase Responsabilidad Colaborador (CRC) proponen una forma de trabajo, preferentemente grupal, para encontrar los objetos del dominio de la aplicación, sus responsabilidades

y cómo colaboran con otros para realizar tareas. Estas registran el nombre de las clases, sus responsabilidades y las otras clases con la que colaboran.

Tabla 16 Tarjeta CRC Evaluador

Clase Evaluador	
Descripción: Clase que gestiona los evaluadores	
Responsabilidad	Colaboradores
Ejecutar la función adicionar y eliminar evaluador	Formulario Evaluación AdicionarEvaluador1 ConfirmPass Modelo Evaluador
Listar los evaluadores	
Ejecuta la función Modifica y actualiza los evaluadores	
Obtener el evaluador	

Tabla 17 Tarjeta CRC Software

Clase Software	
Descripción: Clase que gestiona los software	
Responsabilidad	Colaboradores
Ejecutar la función adicionar y eliminar software	Formulario Evaluación AdicionarSoftware1 Modelo Software
Listar los software	
Ejecuta la función Modifica y actualiza los software	
Obtener el software	

Tabla 18 Tarjeta CRC Evaluación

Clase Evaluación	
Descripción: Clase que gestiona las evaluaciones	
Responsabilidad	Colaboradores
Ejecutar la función adicionar y eliminar evaluación	Formulario Evaluación PesosSC1 PesosSC11 EvaluacionTerminada EvaluacionXTerminar Desbloquear Modelo Software Evaluador CoberturaContexto Eficacia Eficiencia LibertadRiesgo Satisfacción Reportes
Listar las evaluaciones	
Ejecuta la función de modificar y actualizar las evaluaciones	
Obtener evaluación Obtener valoración de la evaluación	

Tabla 19 Tarjeta CRC CoberturaContexto

Clase CoberturaContexto	
Descripción: Clase que gestiona los valores de las métricas de cobertura de contexto	
Responsabilidad	Colaboradores
Ejecutar la función adiciona los valores de las métricas	Formulario PesosSC1 PesosSC11 Modelo CoberturaContexto
Obtener los valores de cobertura de contexto Obtener valoración de sus métricas	

Tabla 20 Tarjeta CRC Eficacia

Clase Eficacia	
Descripción: Clase que gestiona los valores de las métricas de eficacia	
Responsabilidad	Colaboradores
Ejecutar la función adiciona los valores de las métricas	Formulario PesosSC1 PesosSC11 Modelo Eficacia
Obtener los valores de eficacia Obtener valoración de sus métricas	

Tabla 21 Tarjeta CRC Eficiencia

Clase Eficiencia	
Descripción: Clase que gestiona los valores de las métricas de eficiencia	
Responsabilidad	Colaboradores
Ejecutar la función adiciona los valores de las métricas	Formulario PesosSC1 PesosSC11 Modelo Eficiencia
Obtener los valores de eficiencia Obtener valoración de sus métricas	

Tabla 22 Tarjeta CRC LibertadRiesgo

Clase LibertadRiesgo	
Descripción: Clase que gestiona los valores de las métricas de libertad de riesgo	
Responsabilidad	Colaboradores
Ejecutar la función adiciona los valores de las métricas	Formulario PesosSC1 PesosSC11 Modelo LibertadRiesgo
Obtener los valores de libertad de riesgo Obtener valoración de sus métricas	

Tabla 23 Tarjeta CRC Satisfacción

Clase Satisfacción	
Descripción: Clase que gestiona los valores de las métricas de satisfacción	
Responsabilidad	Colaboradores
Ejecutar la función adiona los valores de las métricas	Formulario
Obtener los valores de satisfacción Obtener valoración de sus métricas	PesosSC1 PesosSC11 Modelo Satisfacción

3.7 Tareas de ingeniería

Las tareas de ingeniería son el perfeccionamiento constante de los procesos de producción, con el fin de que estos sean cada vez más productivos. En estas el cliente detalla brevemente aspectos relacionados con la tarea a realizar. Ellas al igual que las historias de usuario tienen un tratamiento muy dinámico y flexible, pueden ser cambiadas por tareas más específicas o generales y añadirse nuevas o ser modificadas. En estas se especifica la fecha de inicio y fin de la tarea, se nombra al programador responsable de cumplirla y describe que se tratara de hacer en la tarea. A continuación se muestran las tareas de ingeniería de las Historias de Usuarios.

Tabla 24 Tarea de ingeniería Realizar una evaluación de calidad

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 1
Nombre Tarea: Realizar una evaluación de calidad	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: Permite al usuario seleccionar un evaluador y un software a evaluar así como el peso de la sub-característica e introducir los valores correspondientes a los campos de las métricas para la ejecución de la evaluación.	

Tabla 25 Tarea de ingeniería Continuar una evaluación de calidad

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 2
Nombre Tarea: Continuar Evaluación de Calidad	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
<p>Descripción: Permite al evaluador recomenzar una evaluación de calidad previamente realizada, la herramienta le mostrará la interfaz de selección de pesos e introducción con los valores correspondientes a los campos de las métricas en el estado que el evaluador guardó el progreso.</p>	

Tabla 26 Tarea de ingeniería Visualizar evaluación de calidad

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 3
Nombre Tarea: Visualizar evaluación de calidad	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
<p>Descripción: Permite al evaluador ver las evaluaciones de calidad, aquí podrá ver el nombre del evaluador y el software que se evaluó así como la versión de la evaluación y sus fechas de inicio y fin.</p>	

Tabla 27 Tarea de ingeniería Insertar evaluador

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 4
Nombre Tarea: Insertar evaluador	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El sistema deberá permitir al usuario insertar un nuevo evaluador, al insertar el evaluador el sistema guardará los datos correspondientes en la base de datos, posibilitando que en otra ocasión ese mismo usuario seleccione su nombre de una lista y así realizar una evaluación.	

Tabla 28 Tarea de ingeniería Insertar software

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 5
Nombre Tarea: Insertar software	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El sistema deberá permitir al usuario insertar un nuevo software a evaluar, al insertar el software el sistema guardará los datos correspondientes en la base de datos posibilitando que se le pueda realizar una nueva evaluación a un software ya evaluado.	

Tabla 29 Tarea de ingeniería Guardar resultados de las evaluaciones de calidad

Tarea de Ingeniería	
Número Tarea: 6	Número Historia de Usuario: 6
Nombre Tarea: Guardar resultados de las evaluaciones de calidad.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El sistema deberá permitir al usuario guardar los resultados de la evaluación.	

Tabla 30 Tarea de ingeniería Exportar reporte general de la evaluación en formato pdf

Tarea de Ingeniería	
Número Tarea: 7	Número Historia de Usuario: 7
Nombre Tarea: Exportar reporte general de la evaluación en formato pdf.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El sistema deberá permitir al evaluador exportar reportes en formato pdf de las evaluaciones conformado por los resultados de las métricas como por los veredictos conclusivos.	

Tabla 31 Tarea de ingeniería Exportar reporte del veredicto conclusivo por característica en formato pdf

Tarea de Ingeniería	
Número Tarea: 8	Número Historia de Usuario: 7
Nombre Tarea: Exportar reportes del veredicto conclusivo por característica formato en pdf.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El sistema deberá permitir al evaluador exportar reportes en formato pdf del veredicto conclusivo acerca del nivel de calidad que tiene el software evaluado.	

Tabla 32 Tarea de ingeniería Deshabilitar software

Tarea de Ingeniería	
Número Tarea: 9	Número Historia de Usuario: 8
Nombre Tarea: Deshabilitar software.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El evaluador podrá deshabilitar un software registrado.	

Tabla 33 Tarea de ingeniería Modificar software

Tarea de Ingeniería	
Número Tarea: 10	Número Historia de Usuario: 9
Nombre Tarea: Modificar software.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El evaluador podrá modificar un software registrado. Pero no los valores de las evaluaciones echas a ese software.	

Tabla 34 Tarea de ingeniería Deshabilitar evaluador

Tarea de Ingeniería	
Número Tarea: 11	Número Historia de Usuario: 10
Nombre Tarea: Deshabilitar evaluador.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya, Dariel Rodríguez del Portillo	
Descripción: El evaluador podrá deshabilitar sus datos del sistema.	

Tabla 35 Tarea de ingeniería Modificar evaluador

Tarea de Ingeniería	
Número Tarea: 12	Número Historia de Usuario: 11
Nombre Tarea: Modificar evaluador.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1/5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Elvis Guillermo Mena Moya ,Dariel Rodríguez del Portillo	
Descripción: El evaluador podrá modificar sus datos	

3.8 Pruebas

Dentro de los instrumentos capaces de medir el estado de calidad de un producto se encuentran las pruebas. El proceso de pruebas se dirige fundamentalmente a componentes del software o al sistema de software en general, con el objetivo de medir hasta cuando el software cumple las funcionalidades establecidas por el cliente. Las pruebas del software verifican y revelan la calidad de un producto. Son utilizadas para identificar posibles errores en la implementación, calidad, o usabilidad de un programa de software.

Propio de la metodología XP, se lleva a cabo la Fase de Prueba. Durante el desarrollo de software, XP establece probar constantemente tanto como sea posible, esto permite un aumento de la calidad del sistema desarrollado reduciendo el número de errores no detectados.

3.8.1 Pruebas Unitarias

Las pruebas unitarias son las encargadas de verificar el código y son diseñadas por los programadores. Cada uno de los desarrolladores tiene que ir probando constantemente lo que va obteniendo en el transcurso de la implementación de un sistema, para garantizar que las funcionalidades exigidas por el cliente estén siendo implementadas correctamente.

Las pruebas unitarias fueron desarrolladas constantemente cada vez que se terminaba de implementar alguna funcionalidad probándola directamente en el entorno real.

3.8.2 Pruebas de aceptación

Las pruebas de aceptación también llamadas pruebas del cliente se enfocan en las características generales y las funcionalidades del sistema. En estas serán probadas las funcionalidades exigidas por el cliente, descritas en las Historias de Usuario que se han implementado.

Las pruebas de aceptación se llevarán a cabo redactando los casos de prueba, teniendo en cuenta el orden de las Historias de Usuario y la prioridad que ha sido asignada a las funcionalidades.

Tabla 36 Prueba de aceptación Realizar una evaluación de calidad

Caso de prueba de aceptación	
Código: HU1_T1	Historia de Usuario: Realizar una evaluación de calidad
Nombre: Realizar una evaluación de calidad	
Descripción: Permite Realizar una evaluación de calidad.	
Condición de ejecución: Que sea seleccionado un software y un evaluador	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Nueva evaluación. • El usuario selecciona un software y un evaluador. • El usuario introduce los pesos y valores correspondientes a las métricas que medirán la calidad del software. • El usuario selecciona la opción Medir. • El usuario selecciona la opción Evaluar. 	
Resultado esperado: La evaluación será insertada en la herramienta con sus respectivos valores además de dar nivel de calidad que tiene el software según sus datos.	
Evaluación de la prueba:	

Tabla 37 Prueba de aceptación Continuar una evaluación de calidad

Caso de prueba de aceptación	
Código: HU2_T2	Historia de Usuario: Continuar Evaluación
Nombre: Continuar una evaluación de calidad	
Descripción: Permite continuar una evaluación de calidad previamente realizada.	
Condición de ejecución: Que sea seleccionado una evaluación sin terminar.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Continuar evaluación. • El usuario introduce los pesos y valores correspondientes a las métricas que medirán la calidad del software. 	
Resultado esperado: En la interfaz de la evaluación aparecerán los últimos datos guardados por el usuario referente a esa evaluación sin terminar.	
Evaluación de la prueba:	

Tabla 38 Prueba de aceptación Visualizar Evaluaciones de calidad

Caso de prueba de aceptación	
Código: HU3_T3	Historia de Usuario: Visualizar Evaluaciones de calidad
Nombre: Visualizar Evaluaciones de calidad	
Descripción: Permite visualizar las evaluaciones de calidad terminadas.	
Condición de ejecución: Que sea seleccionada una evaluación terminada.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Visualizar evaluación. • El usuario selecciona la evaluación para ver sus resultados. 	
Resultado esperado: En la interfaz de la evaluación los resultados de la evaluación seleccionada.	
Evaluación de la prueba:	

Tabla 39 Prueba de aceptación Insertar evaluador

Caso de prueba de aceptación	
Código: HU4_T4	Historia de Usuario: Insertar evaluador
Nombre: Insertar evaluador	
Descripción: Permite insertar un evaluador.	
Condición de ejecución: Llenar todos los campos.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Nueva Evaluación. • El usuario selecciona la opción Insertar evaluador. • El usuario llena los campos correspondientes y selecciona la opción Aceptar. 	
Resultado esperado: Los datos serán registrados en la base de datos	
Evaluación de la prueba:	

Tabla 40 Prueba de aceptación Insertar software

Caso de prueba de aceptación	
Código: HU5_T5	Historia de Usuario: Insertar software
Nombre: Insertar software	
Descripción: Permite insertar un software.	
Condición de ejecución: Llenar todos los campos.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Nueva Evaluación. • El usuario selecciona la opción Insertar software. • El usuario llena los campos correspondientes y selecciona la opción Aceptar. 	
Resultado esperado: Los datos serán registrados en la base de datos	
Evaluación de la prueba:	

Tabla 41 Prueba de aceptación Guardar los resultados de evaluaciones de calidad.

Caso de prueba de aceptación	
Código: HU6_T6	Historia de Usuario: Guardar los resultados evaluaciones de calidad.
Nombre: Guardar los resultados evaluaciones de calidad.	
Descripción: Permite guardar los resultados evaluaciones de calidad. Guardar los resultados evaluaciones de calidad.	
Condición de ejecución: Llenar todos los campos.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal la opción Nueva Evaluación o Continuar Evaluación. • El usuario selecciona la opción Insertar software y evaluador. • El usuario llena los campos correspondientes y selecciona la opción Medir o Posponer. 	
Resultado esperado: Los datos serán registrados en la base de datos	
Evaluación de la prueba:	

Tabla 42 Prueba de aceptación Exportar reporte de la evaluación en formato pdf.

Caso de prueba de aceptación	
Código: HU7_T7	Historia de Usuario: Exportar reportes de la evaluación formato pdf.
Nombre: Exportar reportes de la evaluación formato pdf.	
Descripción: Permite exportar reportes de la evaluación formato pdf.	
Condición de ejecución: Las evaluaciones se deben encontrar registradas en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú Reporte o Mostrar evaluaciones. 	
Resultado esperado: El sistema exportará los resultados en archivos de formato pdf.	
Evaluación de la prueba:	

Tabla 43 Prueba de aceptación Exportar reportes del veredicto conclusivo por característica formato pdf.

Caso de prueba de aceptación	
Código: HU7_T8	Historia de Usuario: Exportar reportes del veredicto conclusivo por característica formato pdf.
Nombre: Exportar reportes del veredicto conclusivo por característica formato pdf.	
Descripción: Permite exportar reportes del veredicto conclusivo por característica formato pdf.	
Condición de ejecución: Las evaluaciones se deben encontrar registradas en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú Reporte o Mostrar evaluaciones. 	
Resultado esperado: El sistema exportará los resultados del veredicto conclusivo exclusivamente en archivos de formato pdf.	
Evaluación de la prueba:	

Tabla 44 Prueba de aceptación Deshabilitar software

Caso de prueba de aceptación	
Código: HU8_T9	Historia de Usuario: Deshabilitar Software
Nombre: Deshabilitar Software	
Descripción: Permite deshabilitar software	
Condición de ejecución: Los evaluadores deben encontrarse en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal Nueva Evaluación. • El usuario selecciona la opción deshabilitar software 	
Resultado esperado: El sistema deshabilitará ese software del sistema.	
Evaluación de la prueba:	

Tabla 45 Prueba de aceptación Modificar software

Caso de prueba de aceptación	
Código: HU9_T10	Historia de Usuario: Modificar Software
Nombre: Modificar Software	
Descripción: Permite modificar software	
Condición de ejecución: Los evaluadores deben encontrarse en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal Nueva Evaluación. • El usuario selecciona la opción modificar software. 	
Resultado esperado: El sistema cambiará los datos viejos por los actuales y los guardará en la base de datos.	
Evaluación de la prueba:	

Tabla 46 Prueba de aceptación Deshabilitar evaluador

Caso de prueba de aceptación	
Código: HU10_T11	Historia de Usuario: Deshabilitar evaluador
Nombre: Deshabilitar evaluador	
Descripción: Permite deshabilitar evaluador	
Condición de ejecución: Los evaluadores deben encontrarse en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal Nueva Evaluación. • El usuario selecciona la opción deshabilitar evaluador. 	
Resultado esperado: El sistema deshabilitará el evaluador del sistema.	
Evaluación de la prueba:	

Tabla 47 Prueba de aceptación Modificar evaluador.

Caso de prueba de aceptación	
Código: HU11_T12	Historia de Usuario: Modificar evaluador
Nombre: Modificar evaluador	
Descripción: Permite modificar un evaluador	
Condición de ejecución: Los evaluadores deben encontrarse en la base de datos de la herramienta.	
Entrada/Pasos de ejecución: <ul style="list-style-type: none"> • El usuario presiona del menú principal Nueva Evaluación. • El usuario selecciona la opción modificar evaluador. 	
Resultado esperado: El sistema cambiará los datos viejos por lo nuevos y los registrará en la base de datos.	
Evaluación de la prueba:	

3.8.3 Prueba para medir la Calidad en el Uso

Esta prueba será guiada por la norma internacional ISO/IEC 25010 la misma determinará si la herramienta automatizada se encuentra lista para ser usada por el usuario final. Esta prueba enfocará la evaluación de la Calidad en el Uso en la característica de Satisfacción específicamente las sub-características de Placer, Comodidad y Confianza para esto el equipo de desarrollo creó un cuestionario que atiende diferentes características de la herramienta como la interfaz visual y el formato y fuente de letra, este cuestionario fue respondido mientras se probaba la herramienta para así obtener respuestas verídicas y exactas, el mismo se puede ver en el **Anexo #5**. Las respuestas dadas al cuestionario por el cliente fueron satisfactorias por lo que los valores que se introdujeron en la herramienta correspondiente a las métricas de cada sub-característica y su peso fue de 2 y una vez realizado el proceso de evaluación automatizado por la herramienta se conoció el veredicto conclusivo en cuanto al nivel de calidad en el uso estos resultados se encuentran en el **Anexo #6**.

3.9 Conclusiones

En este capítulo se realizó un profundo análisis sobre el patrón arquitectónico utilizado en el que se desarrolla la herramienta automatizada. Se identificaron y describieron las tareas de la ingeniería listas para su implementación, correspondiente a las Historias de Usuario descritas en el capítulo anterior. Se obtuvo el modelo físico de la base de datos para el cual es necesario para guardar los datos. Por último, y no menos importante, se llevaron a cabo las pruebas de aceptación logrando la satisfacción del cliente con el software desarrollado.

Conclusiones Generales

Finalizado el proceso investigativo y el desarrollo de un software capaz de realizar el cálculo y valoración de las métricas del modelo Calidad en el Uso según las nuevas características de la norma ISO/IEC 25010, se puede arribar a una serie de conclusiones que a continuación se exponen:

- El estudio de los diferentes software similares existentes en el mercado, así como la sistematización y el análisis integral de los elementos referidos a un software para medir la Calidad en el Uso según las nuevas características de la ISO/IEC 25010, la metodología y herramientas seleccionadas, y las concepciones teóricas que la sustentan, constituyen un punto de partida para implementar un software capaz de medir la Calidad en el Uso según las nuevas características de la ISO/IEC 25010.
- Se describieron las características del sistema, se realizaron las fases de exploración y planificación correspondientes a la metodología seleccionada, lo que permite un correcto y organizado desarrollo posterior del proceso de implementación.
- Luego de realizarse el proceso de implementación se obtuvo un software capaz de medir la Calidad en el Uso según las nuevas características de la ISO/IEC 25010, y con la realización del proceso de pruebas se comprobó que se cumplieron con todas las necesidades especificadas por el cliente.

Recomendaciones

El desarrollo de la herramienta para medir la Calidad en el Uso según la norma ISO/IEC 25010 es satisfactorio, sin embargo, para alcanzar mejores resultados se recomienda:

- Implementar las métricas de la sub-característica Mitigación de riesgo ambiental una vez que la norma ISO/IEC 25022 quede definida adecuadamente.
- Emplear la técnica estadística de redes bayesianas de correlación para así medir la calidad de los software interactuando con el entorno en el cual se ejecuta.

Referencias y Bibliografía

1. **Pressman.** *Enfoque Practico.* Madrid : s.n., 1998. 84-481-3214-9.
2. **ISO/IEC 90003.** *Software engineering -- Guidelines for the application of ISO 9001:2000 to computer software.* 2004.
3. **ISO/IEC 25000.** ISO. [En línea] 4 de Enero de 2010. [Citado el: 11 de Noviembre de 2013.] http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=35733.
4. **Alarcón, Armando Silva.** Entérate en Línea. [En línea] 25 de enero de 2004. [Citado el: 12 de diciembre de 2013.] <http://www.enterate.unam.mx/Articulos/2004/Enero/modelos.htm>.
5. **ISO 25000.** *Evaluación de productos.* [En línea] 2014. [Citado el: 3 de febrero de 2014.] <http://iso25000.com/index.php/evaluacion-productos>.
6. Normalización, Oficina Nacional de. *NC ISO/IEC 9126-1.* La Habana : Cuban National Bureau of Standards, 2005.
7. Java. [En línea] [Citado el: 20 de Enero de 2014.] <http://www.infor.uva.es/~jmrr/tgp/java/JAVA.html>.
8. Rowman, Danny. Buenas tareas. *Manual de bizagi.* [En línea] 2009. [Citado el: 20 de Enero de 2014.] <http://www.buenastareas.com/ensayos/Manual-De-Bizagi/1362471.html>.
9. Paradigm, Visual. [En línea] [Citado el: 20 de abril de 2014.] <http://www.visual-paradigm.com>.
10. Laterza, Older Osvaldo Krause. Sistema avanzado de distribucion. *IReport características mas relevantes.* [En línea] 11 de Myo de 2011. [Citado el: 20 de Enero de 2014.] http://mistock.lcompras.biz/index.php?option=com_content&view=article&id=1249:ireport&catid=59:tallersoftware&Itemid=116.
11. Larman, Craig. *UML y Patrones 2da edición.*
12. *Revista Avances en Sistemas e Informática.* Mario R. Moreno S., Gabriel U. González C., Diana C. Echartea C. 1657-7663, 2008, Vol. 5.
13. Netbens, Comunidad del. netbens. [En línea] [Citado el: 23 de Enero de 2014.] https://netbeans.org/community/releases/61/index_es.html.

14. Montoya, Lizzet Cabrera y Hinojosa, Virgen Yuliet Acosta. *MÉTRICAS ESTANDARIZADAS INTERNACIONALMENTE, PROPUESTAS PARA EVALUAR LA CALIDAD DE LOS PRODUCTOS DE SOFTWARE*. La Habana : s.n., 2008.
15. Metodologías de Desarrollo de software. [En línea] 2010. [Citado el: 20 de 11 de 2013.] <http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/Tema04.pdf..>
16. Software y aplicaciones web. [En línea] [Citado el: 20 de abril de 2014.] <http://www.jtentor.com.ar/post/Arquitectura-de-N-Capas-y-N-Niveles.aspx>.
17. Perez, Lucas. Practicas de software. [En línea] [Citado el: 20 de abril de 2014.] <http://www.practicadesoftware.com.ar/2010/12/introduccion-a-arquitectura-en-capas/>.
18. XP, Metodologia. XP programming.com. [En línea] [Citado el: 20 de abril de 2014.] <http://xprogramming.com/index.php>.
19. Sqlite. <http://www.sqlite.org>. <http://www.sqlite.org/about.html>. [En línea] [Citado el: 20 de abril de 2014.] <http://www.sqlite.org/about.html>.
20. [En línea] [http://www.visual-paradigm.com/product/vpuml/..](http://www.visual-paradigm.com/product/vpuml/)
21. Luis Olsina, Guillermo Covella, Gustavo Rossi. *Web-Engineering_Mendes_C04WebQuality*.