

**IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL PARA LA
EVALUACIÓN DE LA CALIDAD DE PRODUCTOS SOFTWARE DE LOS
GRUPOS DE INVESTIGACIÓN DE LA UNIVERSIDAD INDUSTRIAL DE
SANTANDER.**

NELSON PINTO CHACÓN

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍA FÍSICO MECÁNICA
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2011

**IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL PARA LA
EVALUACIÓN DE LA CALIDAD DE PRODUCTOS SOFTWARE DE LOS
GRUPOS DE INVESTIGACIÓN DE LA UNIVERSIDAD INDUSTRIAL DE
SANTANDER.**

NELSON PINTO CHACÓN

**PROYECTO DE INVESTIGACIÓN PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS**

DIRECTOR

Luis Carlos Gómez Florez. MSc.

Profesor Titular

CODIRECTOR

Ing. Nelson Enrique León Martínez

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍA FÍSICO MECÁNICA
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2011

DEDICATORIA

El autor dedica este logro a:

Primeramente a Dios por guiarme hasta esta instancia de mi vida, porque me ha dado la sabiduría y el entendimiento para alcanzar esta tan anhelada meta.

A mi padre que fue mi apoyo moral, mi maestro, mi amigo, mi guía, y aunque no pudo ver este momento, sé que donde esta se siente orgulloso de la persona que ha formado.

A mi madre por su apoyo incondicional, por su paciencia y buenos consejos.

A mis hermanos y familiares que con su apoyo no me dejaron desfallecer.

Nelson

AGRADECIMIENTOS

El autor expresa su agradecimiento a:

A la universidad Industrial de Santander por poner a disposición sus instalaciones y sus recursos a toda la comunidad.

A mis amigos que me acompañaron e hicieron más ameno el transitar que me ha traído hasta aquí.

Al profesor Luis Carlos Gómez Flórez, por la oportunidad que me dio de participar en este proyecto abriéndome las puertas de su grupo STI.

Al Ingeniero y candidato a magister Nelson Enrique León Martínez por su apoyo y comprensión, y por compartir su amplio conocimiento y su tiempo en el desarrollo de este proyecto.

TABLA DE CONTENIDO

INTRODUCCIÓN	14
1 GENERALIDADES	17
1.1 CONTEXTUALIZACIÓN DEL PROYECTO	17
1.1.1 <i>Grupos de investigación</i>	17
1.2 COMPOSICIÓN DEL PROYECTO	19
1.3 DEFINICIÓN DEL PROBLEMA	20
1.4 JUSTIFICACIÓN	22
1.5 OBJETIVOS	24
1.5.1 <i>Objetivo general.</i>	24
2 MARCO METODOLÓGICO Y TEÓRICO	26
2.1 MARCO METODOLÓGICO	26
2.2 MARCO TEÓRICO	27
2.2.1 <i>Modelos de evaluación</i>	27
2.2.1.1 Estándar de calidad ISO/IEC 14598: Evaluación del producto software	30
2.2.1.2 ISO / IEC 25000:2005 - Software product Quality Requirements and Evaluation (SQuaRE) 73	
2.2.2 <i>Herramientas de evaluación</i>	74
2.2.2.1 Comparación de herramientas para la evaluación de la calidad de productos software. 75	
3 RESULTADOS DEL PROYECTO DE INVESTIGACIÓN	82
3.1 DESARROLLO DE LA HERRAMIENTA PROPUESTA	82
3.1.1 <i>Herramientas de desarrollo</i>	82
3.1.2 <i>Metodología de desarrollo</i>	84
3.1.3 <i>Implementación de la herramienta</i>	87
3.1.3.1 Modelo de calidad	89
3.1.3.2 Evaluación	91
3.2 CASOS DE USO	92
3.3 DIAGRAMA DE CLASES	93
3.4 DESCRIPCIÓN DE LAS CLASES	95
3.5 ESTRUCTURA DE ARCHIVOS	107
3.6 OTROS PRODUCTOS	108
4 ILUSTRACIÓN DEL USO DE LA HERRAMIENTA	110
4.1 ILUSTRACIÓN	110
4.1.1 <i>Desarrollo de la evaluación de EOGEST 1.0 con QUIS.</i>	111
5 CONCLUSIONES Y RECOMENDACIONES	118
5.1 CONCLUSIONES	118
5.2 RECOMENDACIONES	119
BIBLIOGRAFIA	120

LISTA DE TABLAS

Tabla 1: Cumplimiento de los objetivos propuestos.....	25
Tabla 2: Modelos de calidad del producto software	28
Tabla 3: Descripción de las características y sub características de calidad según la norma ISO/IEC 9126-1. 2001	45
Tabla 4: Métricas de idoneidad	50
Tabla 5: Características de la calidad de uso	52
Tabla 6: Herramientas para la evaluación de la calidad del producto software	74
Tabla 7: Comparación de herramientas para la evaluación.....	79
Tabla 8: Descripción Modelo de calidad	90
Tabla 9: Descripción de la parte de evaluación	91
Tabla 10: Otros elementos.....	92
Tabla 11: Descripción de clases del sistema	95
Tabla 12: Descripción de formularios.....	101

LISTA DE FIGURAS

Figura 1: Composición de los grupos de investigación	18
Figura 2: Estructura del proyecto	19
Figura 3: Marco metodológico.....	26
Figura 4: Relación entre las normas ISO/IEC 9126 y 14598	31
Figura 5: Proceso de evaluación de un producto software	32
Figura 6: Calidad en el ciclo de vida del software	35
Figura 7: Relación entre medidas y características	36
Figura 8: Calidad del modelo durante el ciclo de vida.....	38
Figura 9: Calidad en el ciclo de vida del software	41
Figura 10: modelo de calidad para la calidad interna y externa	44
Figura 11: Esquema de la calidad de uso	52
Figura 12: Características, sub característica y atributos de calidad	55
Figura 13: Rangos de una escala de medición	60
Figura 14: Plan de evaluación.....	61
Figura 15: Relación entre las normas ISO 25000 y 9126 - 14598	73
Figura 16: Arquitectura de KEMIS	77
Figura 17: Arquitectura de MOSCA	79
Figura 18: Herramientas de desarrollo.....	83
Figura 19: Modelo evolutivo de desarrollo de software.....	85
Figura 20: Estructura general de QUIS.....	88
Figura 21: Pantallazo inicial para la parte de producto QUIS.	89
Figura 22: Esquema general de un modelo de calidad.....	90
Figura 23: Diagrama de casos de uso.	93
Figura 24: Diagrama de clases.	94
Figura 25: Estructura de archivos.	108
Figura 26: Presentación EOGEST 1.0.....	110
Figura 27: Definición de la evaluación.	112
Figura 28: Selección de productos a evaluar.....	112
Figura 29: Definición de la evaluación.	113
Figura 30: Ejecutar la evaluación.....	113
Figura 31: Resultados de la evaluación a EOGEST 1.0.	114
Figura 32: Reporte de evaluación (1).....	115
Figura 33: Reporte de evaluación (2).....	115

GLOSARIO

Calidad: Propiedad o conjunto de propiedades inherentes a algo, que permiten juzgar su valor. Características propias del software aquellas que tu quieres controlar y asegurar, el software es un producto inmaterial que no se fabrica, tampoco se degradan físicamente, sino que se desarrolla; El software puede tener errores, incidencias pero no son similares a lo que cualquier equipo de carácter físico.

Calidad del software: La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándar para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, en aras de lograr una mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que eleven la productividad, tanto para la labor de desarrollo como para el control de la calidad del software.

Gestión de la calidad de software (ISO 9000): Conjunto de actividades de la función general de la dirección que determina la calidad, los objetivos y las responsabilidades y se implanta por medios tales como la planificación de la calidad, el control de la calidad, el aseguramiento (garantía) de la calidad y la mejora de la calidad, en el marco del sistema de calidad.

Aseguramiento de la calidad de software: Conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto (software) satisfará los requisitos dados de calidad. El aseguramiento de calidad del software se diseña para cada aplicación antes de comenzar a desarrollarla. Hay quienes prefieren decir garantía de calidad en vez de aseguramiento.

Métricas: Una forma de medir y una escala, definidas para realizar mediciones de uno o varios atributos.

Producto Software: Conjunto de programas de computadora, procedimientos, posible documentación y datos asociados. (ISO/IEC 14598).

RESUMEN

TÍTULO

IMPLEMENTACIÓN DE UNA HERRAMIENTA COMPUTACIONAL PARA LA EVALUACIÓN DE LA CALIDAD DE PRODUCTOS SOFTWARE DE LOS GRUPOS DE INVESTIGACIÓN DE LA UNIVERSIDAD INDUSTRIAL DE SANTANDER.*

AUTOR

Nelson Pinto Chacón. **

PALABRAS CLAVE

Calidad del producto software, normas de calidad, ISO 14598, ISO 9126, métricas de software.

DESCRIPCIÓN

La calidad de los productos software se ha convertido en uno de los aspectos más importante en la organización, lo que implica que requiere una evaluación, la cual permita decidir si el producto que se desea o se desarrolla cumple o no con las expectativas que se desean, por lo cual se planteó en este proyecto hacer un estudio de cómo evaluar la calidad de los productos que se desarrollan en el interior de la universidad y cómo por medio de una herramienta apoyar esta evaluación.

Este proyecto tiene como objetivo principal presentar una herramienta software que permita la evaluación de la calidad de productos del mismo tipo, soportada en normas internacionales elaboradas para tal fin; esta herramienta está orientada a apoyar la metodología de evaluación en los grupos de investigación de la Universidad Industrial de Santander que desarrollan software como resultado de sus investigaciones.

Este proyecto generó como resultado de una investigación amplia acerca de calidad del software, de cómo se está evaluando la calidad de estos productos, herramientas que existen en el mercado para este fin y los modelos de calidad del software. El resultado obtenido es una herramienta computacional basada en normas internacionales (ISO 14598, 9126), para la evaluación de la calidad de los productos software que se genera en los grupos de investigación de la Universidad Industrial de Santander.

* Trabajo de grado

** Facultad de ingenierías fisicomecánicas. Escuela de ingeniería de sistemas e informática. Director: Luis Carlos Gómez Florez. Codirector: Nelson Enrique León Martínez .

SUMMARY

TITLE

IMPLEMENTATION OF A COMPUTATIONAL TOOL FOR EVALUATING THE QUALITY OF SOFTWARE PRODUCTS THE INVESTIGATION GROUPS THE UNIVERSIDAD INDUSTRIAL DE SANTANDER.*

AUTHOR

Nelson Pinto Chacón.**

KEY WORDS

Software product quality, quality standards, ISO 14598, ISO 9126, software metrics.

DESCRIPTION

The quality of the software products has become one of the most important aspects in the organization, it requires an assessment to decide whether the product being developed or is complying with the expectations placed, therefore this project was raised in a study of how to assess the quality of the products developed within the university as a tool by supporting this assessment.

This project has as main objective to present a software tool that allows the evaluation of the quality products of the same type, supported on international standards developed for this purpose, this tool is aimed at supporting the evaluation methodology in the investigation groups of Universidad Industrial de Santander software developers as a result of their investigations.

This project generated as a result of extensive research on software quality, how it is evaluating the software quality tools available in the market for this purpose and software quality models The result is a computational tool based on international standards (ISO 14598, 9126),for assessing the quality of software products that are generated in the research groups of the Universidad Industrial de Santander.

*Work degree

**Physical-engineering faculty. School systems engineering and computer. Directress: Luis Carlos Gómez Florez. Codirectress: Nelson Enrique León Martínez.

INTRODUCCIÓN

Los programas de computadora cada vez toman más relevancia en el mundo moderno, éstos se están volviendo indispensables en cualquier institución sea cual sea su actividad; éstos han pasado a ser el alma y vida de las mismas, por lo cual con el transcurso del tiempo se ha venido manejando un gran interés por darle a los productos software la importancia que estos merecen.

Uno de los principales problemas que actualmente se presenta en la industria del software es la ausencia de calidad en los productos, lo cual fue el motivo del desarrollo de este proyecto de investigación. Este proyecto tiene como objetivo el estudio y el posterior desarrollo de una herramienta computacional que permita la evaluación de productos software, utilizando normas y estándares internacionales para su medición.

En la parte 1 de este documento se hará una descripción general. En el capítulo 1 se mostrará las generalidades del proyecto, primero se hará la contextualización del proyecto presentado, seguidamente se definirá la problemática que intenta resolver el desarrollo de este trabajo y la justificación del mismo.

En el capítulo 2 se hará un recuento del diseño metodológico y referencial que precedió el desarrollo de este proyecto, en él se encuentra una breve descripción de las normas existentes para la evaluación de la calidad de los productos software, en este caso se hará énfasis en las normas internacionales ISO/IEC 14598 e ISO/IEC 9126, debido a que estas dos normas fueron la base teórica de este proyecto.

El capítulo 3 describe el desarrollo de la herramienta basada en el marco teórico seleccionado para su desarrollo. Allí se detalla las herramientas de desarrollo, la metodología y los procedimientos que se siguieron para su realización.

El capítulo 4 muestra la ilustración de utilización de la herramienta desarrollada. Esta se llevó a cabo con el software del grupo de investigación en ingeniería biomédica (GIIB) denominado EOGEST 1.0.

En el capítulo 5 se presentan conclusiones y recomendaciones que se proponen a partir de este proyecto.

PARTE I. PRESENTACIÓN DEL TRABAJO DE INVESTIGACIÓN

1 GENERALIDADES

1.1 Contextualización del proyecto

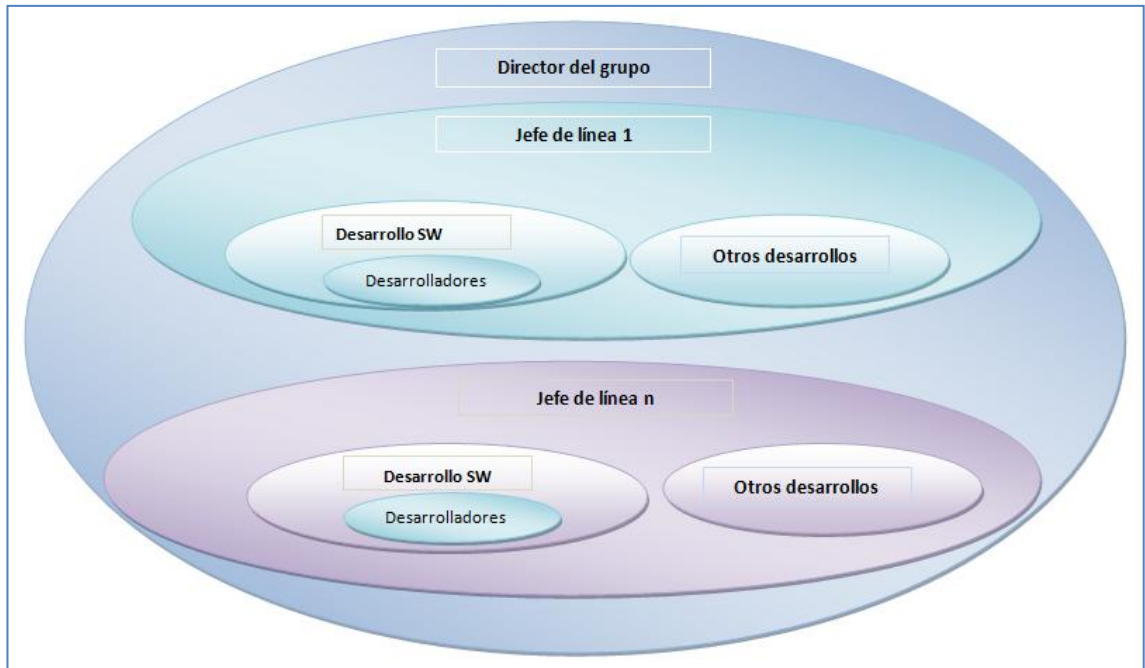
El presente proyecto se desarrolló en el interior del Grupo de Investigación en Sistemas y Tecnologías de la Información STI. Este proyecto está dirigido a los grupos de investigación de esta institución, los cuales desarrollan o producen software dentro de sus actividades y consiste en la implementación de un software soportado en normas internacionales elaboradas para la evaluación de calidad de productos software.

1.1.1 Grupos de investigación

En la Universidad Industrial de Santander existen cerca 113 grupos de investigación, de los cuales un gran porcentaje produce software, y la totalidad de ellos usan productos de este tipo en sus investigaciones. Por lo cual surgió el interés de desarrollar una herramienta que apoye la metodología para la evaluación de la calidad de estos productos.

En la figura 1 se describe la composición de un grupo de investigación en términos generales. En un grupo de investigación siempre hay un líder que es el director del grupo, luego se descomponen en sub-grupos llamados líneas de investigación, estos sub-grupos tienen sus jefes de línea o directores de línea estos responden ante el director y son los encargados de buscar las personas para el proyecto a desarrollar y acompañar al desarrollador en este proceso. Otro rol importante dentro de un grupo de investigación es el desarrollador, estas personas en su mayoría son estudiantes de pregrado, de últimos semestres los cuales se involucran en el proyecto el cual les sirve como requisito de grado.

Figura 1: Composición de los grupos de investigación



Fuente: Autor

Teniendo una idea de cómo funciona un grupo de investigación se hará una breve explicación de cómo se dan los desarrollos en un grupo de investigación.

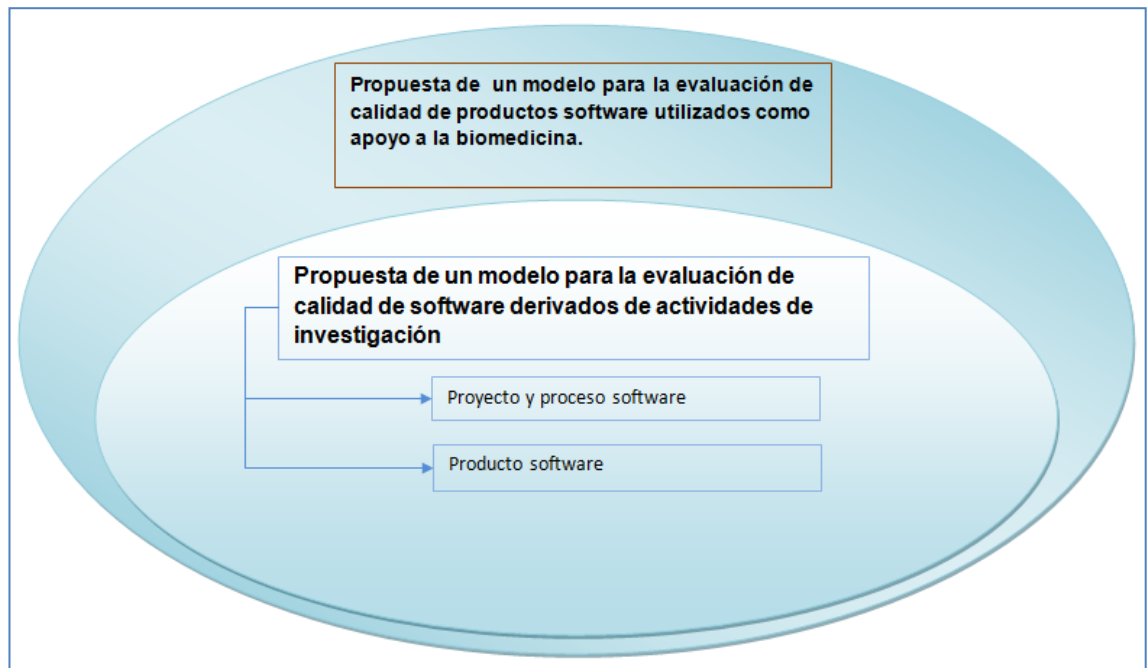
Los proyectos nacen de una idea en una línea de investigación, el jefe de línea busca personal para su desarrollo, este les plantea el problema y con esta idea se empieza a elaborar el plan para la realización del proyecto propuesto. En la mayoría de casos sucede que el desarrollador define la metodología, el lenguaje, la estructura para su plan de trabajo, es decir hace su desarrollo dependiendo de sus habilidades sin tener en cuenta estudios de viabilidad de desarrollo y la compatibilidad de herramientas para tal fin, generando en algunas ocasiones inconvenientes tales como incumplimiento en los cronogramas por cambios debido a falta de planeación, problemas de compatibilidad entre herramientas entre otros.

Otro inconveniente que se presenta es que a la hora de terminar el proyecto no se le hace una evaluación exhaustiva de calidad al software desarrollado, lo cual conlleva a que el proyecto desarrollado se utilice como se planificó y termine siendo solo otro producto más archivado en biblioteca.

1.2 Composición del proyecto

Este proyecto se desarrollo en La universidad Industrial de Santander a través de un proyecto de maestría avalado por la vicerrectoria de investigación y extensión propuesto por el ingeniero Nelson Enrique León Martínez aspirante a magister proyecto denominado: **propuesta de un modelo para la evaluación de calidad de software derivados de actividades de investigación**. El cual generó dos proyectos, este que se describe en este libro y otro proyecto denominado: **implementación de una herramienta computacional para la gestión y evaluación de proyectos y procesos de desarrollo de software de los grupos de investigación de la universidad industrial de Santander**, desarrollado por el estudiante de ingeniería de sistemas la estructura de este proyecto se muestra en la figura.

Figura 2: Estructura del proyecto



Fuente: Autor.

1.3 Definición del problema

¿Existe alguna herramienta para la evaluación de calidad de productos software utilizados como apoyo en los grupos de investigación, definido en términos de la identificación de características, atributos y métricas de calidad para la evaluación del producto software?

La utilización de productos software se ha vuelto una necesidad básica en cualquier tipo de institución que implique manejo de información y flujo de datos, lo cual nos conduce a una preocupación, y esta a un afán por mejorar este tipo de productos debido a la gran importancia que tiene la información en el mundo actual. Cabe recalcar que en la actualidad la gran mayoría de productos diferentes al software se rigen por normas de calidad que permiten al usuario final tener confianza a la hora de adquirirlo, por lo cual nos lleva a pensar que siendo el software un soporte el manejo de la información, deberíamos preocuparnos por evaluar y mejorar su calidad en todas sus etapas.

La Universidad Industrial de Santander es una de las universidades más reconocidas a nivel nacional por su calidad educativa y su buen desarrollo a nivel de grupos de investigación, por lo tanto parece ilógico pensar que una universidad de esta categoría desarrolla proyectos de investigación, no tenga como regla general la aplicación de normas de calidad en los proyectos que tienen como resultados productos software.

En la actualidad los grupos de investigación de la universidad dedicados a generar proyectos de software no siguen una norma de calidad y más aún, no sienten la necesidad de regirse por las normas que existen para tal uso. Esto se ha venido omitiendo y dejando a un lado sin darle la importancia y el valor que tiene, creando una cultura que ignora dichas normas y lo más grave, las personas encargadas de la formación del profesional no lo están inculcando en los futuros ingenieros. Tener este tipo de formación es muy importante ya que

permite dar más valor a sus desarrollos y en últimas a sus grupos de investigación, permitiéndoles ser mejores en lo que hacen y así recibir mayores reconocimientos a través de sus productos.

Este proyecto, aparte de crear una herramienta para gestionar las evaluaciones de calidad del producto software, pretende generar en el estudiante desarrollador un primer acercamiento a la cultura de calidad de software, brindándole los instrumentos necesarios que les permita evaluar en términos de calidad sus productos, descubriendo en qué están fallando y corregir sus falencias, permitiendo así una mejora progresiva en cada proyecto que se genere. Además, con este proyecto se quiere que a medida que sea utilizada la herramienta para evaluar calidad de productos software, ésta evolucione en términos de calidad y nuevas funcionalidades, de la misma manera que los proyectos que evalúa.

La propuesta que se plantea en el presente proyecto es diseñar una herramienta informática para la evaluación de la calidad de productos software utilizados como apoyo a los desarrollos que se llevan a cabo en los diferentes grupos de investigación de la universidad que generen este tipo de productos, definido en términos de la identificación de características, atributos y métricas para evaluar este tipo de software, de tal manera que las herramientas finales se conviertan en un apoyo al diagnóstico, seguimiento y control del software, para que puedan ser catalogados como productos de calidad optima, e ir mejorando cada vez más en este aspecto. Esta herramienta se basó en normas y estándares internacionales de calidad que existen para evaluar este tipo de productos.

El presente proyecto hace parte de la metodología para evaluar la calidad del software de los grupos de investigación de la Universidad Industrial de Santander que se está desarrollando actualmente en la DIF, este grupo en la actualidad viene desarrollando proyectos de este tipo, se destacan entre ellos

la evaluación de procesos y proyectos del software, formación en propiedad intelectual entre otros.

1.4 Justificación

Hoy en día existen diversas metodologías y estándares para evaluar la calidad del software como las Normas ISO y el CMMI que están orientados tanto a medir la calidad de desarrollo de los procesos como de los productos finales. Además de estas metodologías se necesita de una herramienta de apoyo que sea capaz de agrupar toda esta información de una manera eficaz.

En el mercado actual hay pocas herramientas que se enfocan en la evaluación de determinados aspectos de calidad en forma individual, entre los cuales se encuentran: Kemis² que está enfocado a la evaluación del producto software, Mosca³ basado en la evaluación de procesos de desarrollo software, Mosca+ desarrollado para la evaluación de procesos de desarrollo de software de simulación, basado en Mosca.

Debido a que no hay una herramienta capaz de evaluar los diferentes aspectos de calidad en conjunto, y sirva de apoyo a la metodología, que se desea implementar en la Universidad Industrial de Santander para evaluar la calidad del software desarrollado en los grupos de investigación, de tal manera que cumplan los estándares de calidad y puedan acceder fácilmente al mercado, cumpliendo así el objetivo por el cual se creó. A raíz de esta necesidad decidió implementar una herramienta de apoyo a esta metodología.

Dicha herramienta permite evaluar diferentes aspectos del desarrollo de software tales como la gestión de proyectos, así como la evaluación de

²Kybele Environment Measurement Information System. Es un entorno desarrollado por KybeleConsulting cuyo objetivo es facilitar la medición automatizada de la calidad de los productos software.

³Modelo Sistémico de Calidad de Software.

productos y procesos de desarrollo, con el fin de valorar y estimar la calidad de software que se implementa en la actualidad en la universidad, de tal manera que permita tanto el mejoramiento de herramientas desarrolladas así como el de desarrollos futuros.

Debido a que la evaluación de la calidad del software es un tema muy extenso, la investigación se dividió en dos partes, es por eso que el desarrollo de este proyecto se basó en la planeación de la evaluación y la evaluación propiamente dicha del producto software, paralelamente a este se desarrolló el proyecto mencionado anteriormente denominado: *implementación de una herramienta computacional para la gestión y evaluación de proyectos y procesos de desarrollo de software de los grupos de investigación de la universidad industrial de Santander*⁴, encargado de evaluar el proyecto y el proceso.

Esta herramienta tiene la capacidad de evaluar la calidad de un producto software, además permitirá que los grupos de investigación de la Universidad tengan una visión más exacta de la calidad de sus desarrollos, es decir, las características de las herramientas que han desarrollado respecto de las especificaciones dadas por el cliente, bien sean explícitas o implícitas.

Al evaluar la calidad de las herramientas finales desarrolladas en los grupos de investigación les permitirá detectar las fallas, para corregirlas antes de entregarlas al usuario final y no cometerlas en futuros proyectos. Estas prácticas le permitirán a la Universidad posicionarse como una entidad que produce software con niveles cuando menos aceptables de calidad, para ser comercializados o utilizados con mayor confianza en otras investigaciones.

Finalmente, la universidad no posee una herramienta que le permita hacer este tipo de evaluaciones, lo cual implica que tanto para los registros de autoría como para la evaluación de los productos de una investigación, que incluyan

⁴ Facultad de ingenierías Físico-mecánicas. Escuela de ingeniería de sistemas e informática. Universidad Industrial de Santander. Tesis para el título de ingeniero de sistemas. Autor: Jorge Iván Pimentel Ravelo

herramientas software, no se hace un control de los desarrollos realizados y no se sabe a ciencia cierta cuál es la calidad de estos.

1.5 Objetivos

1.5.1 Objetivo general.

Para el desarrollo de este proyecto se propuso como objetivo general:

Implementar una herramienta computacional que permita la evaluación de la calidad de productos software, que sirva como apoyo a la metodología de evaluación de calidad de herramientas software desarrolladas en los grupos de investigación de la Universidad Industrial de Santander.

Para el cumplimiento de este objetivo, se propusieron cuatro objetivos específicos, la tabla 1 describe como se dio cumplimiento a cada objetivo propuesto.

Tabla 1: Cumplimiento de los objetivos propuestos.

Objetivo	Cumplimiento
Estudio del estado del arte, pertinente a la evaluación de productos software.	<p>En esta parte se realizó la búsqueda de modelos y metodologías para la evaluación de la calidad de productos software. Al final se seleccionaron dos normas internacionales para tal fin estas fueron:</p> <ul style="list-style-type: none"> • ISO 14598 • ISO 9126
Estudio comparativo de herramientas, para la evaluación de productos software.	<p>En este ítem se realizó una búsqueda de herramientas para la evaluación, los resultados se muestran en el numeral 2.2.2 la tabla 7.</p>
Implementación una herramienta, para la evaluación de productos Software.	<p>Se dio cumplimiento a este objetivo con el desarrollo de la herramienta denominada QUIS, además de esto se desarrolló un artículo, el cual en el momento de escribir este documento se encontraba en evaluación.</p>
Ilustración de la herramienta.	<p>Este objetivo se cumplió con la evaluación de un software biomédico denominado EOGEST 1.0, los detalles de esta evaluación se describen en el capítulo 4 de este documento.</p>

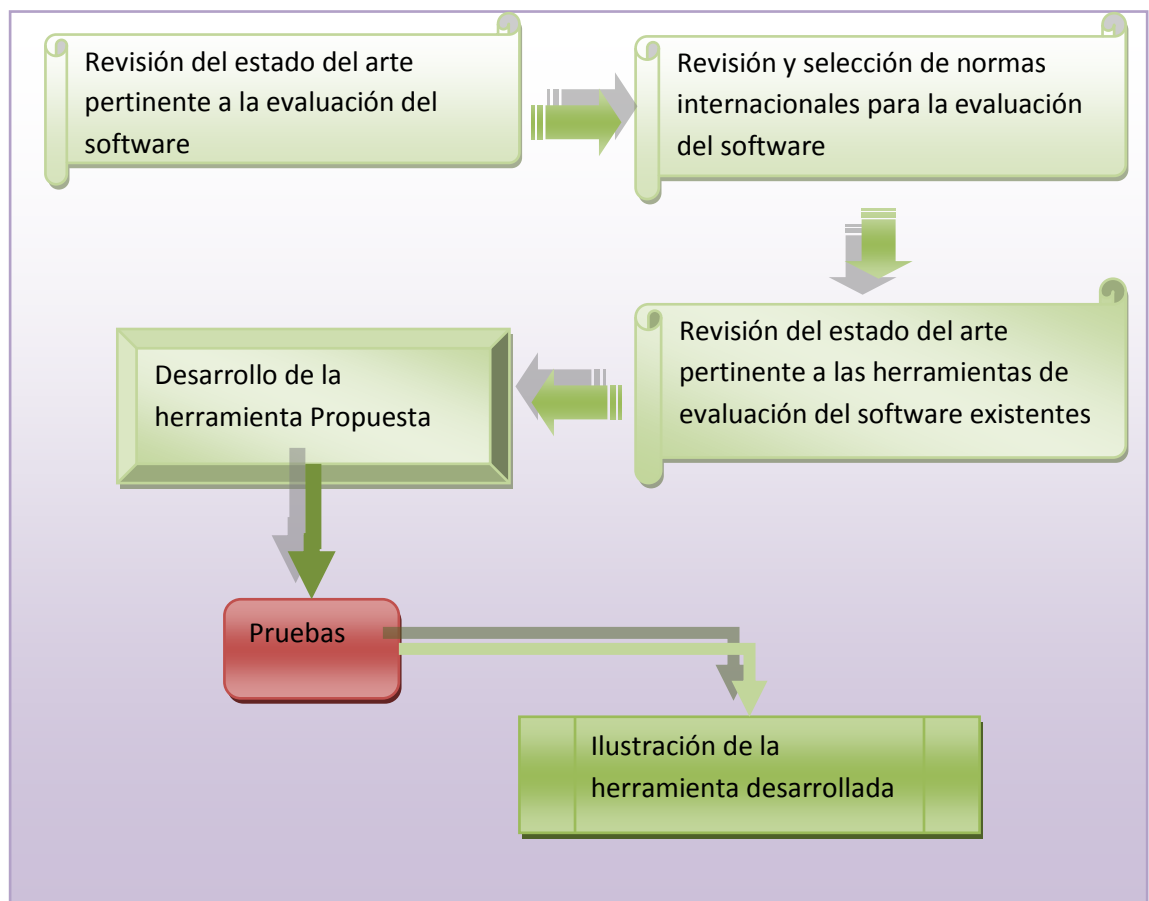
Fuente: Autor.

2 MARCO METODOLÓGICO Y TEÓRICO

2.1 Marco metodológico

El desarrollo metodológico de este trabajo se basó en el cumplimiento de cada uno de los objetivos específicos propuestos para el desarrollo del mismo, se propuso ir desarrollando cada objetivo en la medida que el anterior lo permitiera. La figura 3 muestra de forma gráfica cómo se definió la metodología.

Figura 3: Marco metodológico



Fuente: Autor.

Como se observa en la figura 3 el desarrollo del proyecto tuvo seis fases las cuales se desarrollaron secuencialmente. En la primera fase se realizó el

estudio generalizado de la evaluación de la calidad del software. La segunda fase consistió en seleccionar las normas en las cuales se iba a soportar para el desarrollo del presente trabajo, esta fueron la ISO/IEC 14598 (Evaluación del producto software) y la ISO/IEC 9126 (Calidad del producto software), las cuales se describirán más detalladamente en secciones posteriores. En la tercera fase se realizó una búsqueda y posterior estudio de las herramientas existentes en el mercado, realizando una evaluación comparativa para ver cómo estaba el estado del arte al respecto. La cuarta fase que consistió en la implementación del producto que se entrega soportando la investigación, en esta se realizó el desarrollo de la herramienta software planteada para la evaluación de la calidad de productos software. La fase cinco consistió en la ilustración de la herramienta desarrollada, en la cual se evaluó el software llamado EOGEST⁵ los resultados de esta evaluación se mostraran en el capítulo 4. La fase seis consistió en la detección de fallas que se presentaron en la evaluación de EOGEST, y la posterior corrección de éstas, allí se dio un ciclo hasta que la herramienta respondió correctamente.

2.2 Marco teórico

En esta parte se hará una descripción de la investigación que precedió el desarrollo del proyecto y las pautas que se siguieron para su logro.

2.2.1 Modelos de evaluación

Los programas de computadora que facilitan los procesos de innumerables entidades y diferentes actividades de desempeño, se han venido utilizando cada vez más ya es normal y necesario que cada institución tenga un sistema que apoye y facilite los procesos dentro de ella; por esto durante el proceso de evolución de estos sistemas también han tomado gran relevancia los modelos de calidad para este tipo de productos.

⁵ Software para la caracterización y análisis estadístico de señales electrográficas

En cualquier proceso que implique la elaboración de un producto, la evaluación de la calidad se ha vuelto una condición necesaria para la aprobación del mismo; dependiendo del nivel de calidad el producto adquiere un valor agregado y una mejor aceptación por parte del cliente que desea adquirir un producto de este tipo; por lo tanto los modelos de calidad de la evaluación del software, al igual que el mismo software, han venido evolucionando con el tiempo debido a que los productos de este tipo evolucionan rápidamente, lo cual implica nuevos requerimientos y nuevas necesidades.

Con el transcurso del tiempo se han venido desarrollando modelos que pretenden caracterizar la calidad del software diferenciando entre modelos de calidad del proceso y modelos de calidad del producto, los modelos del producto pretenden definir las características que debe satisfacer un producto software para cumplir con determinados criterios de calidad, de forma tal que su calidad se pueda cuantificar a través de atributos medibles (métricas), en la Tabla 2 se hace un recorrido por los diferentes modelos para la evaluación del producto software.

Tabla 2: Modelos de calidad del producto software

Modelo	Descripción	Año
Boehm	Principales criterios de calidad: Facilidad de uso y mantenibilidad.	1976
McCall	Principales criterios de calidad: operación, revisión y transición.	1977
Arthur	Modelo para evaluar la calidad de la documentación.	1985
Gilb	Especificación de requisitos de calidad establecida entre el analista y el usuario.	1988
Deutsch	Define características para evaluar	1988

	atributos de calidad en diferentes fases del proceso de software como parte de la calidad.	
Schulmeyer	Enfoque similar al del modelo de Deutsch, en el que se abarca también el proceso de desarrollo.	1990
Gillies	Modelo que evalúa aspectos internos y externos desde el punto de vista del producto y del proceso.	1992
Dromey	Modelo que evalúa la calidad del producto en función de estándares de código, clasificación de defectos y el desarrollo de herramientas de auditoría.	1995
ISO 9126	Estándar para la evaluación de la calidad del software con base en criterios de: Funcionalidad, Rendimiento, Usabilidad, Eficiencia, Mantenibilidad y Portabilidad.	2000

Fuente: CABALLERO RÚA Edgar Henry. Mejora de la calidad del software en el entorno de microempresas de TI. Universidad Politécnica de Madrid. <http://is.ls.fi.upm.es/doctorado/Trabajos20062007/Caballero.pdf>

El presente trabajo tomó como referencia el modelo de calidad establecido por la ISO/IEC 9126, debido a que es un modelo definido internacionalmente para la evaluación del software; existe un estándar que se está desarrollando actualmente a través del proyecto denominado SQuaRe⁶(ISO/IEC 25000:200), este unifica los estándares ISO/IEC 9026 e ISO/IEC 14598. En el momento que se inició esta investigación no se había completado esta norma por lo que se tomó como referencia las dos normas por separado, en la sección 2.2.1.2 se hará una breve descripción de esta norma. A continuación se describirán más detalladamente cada una de estas normas.

⁶Software product Quality Requirements and Evaluation

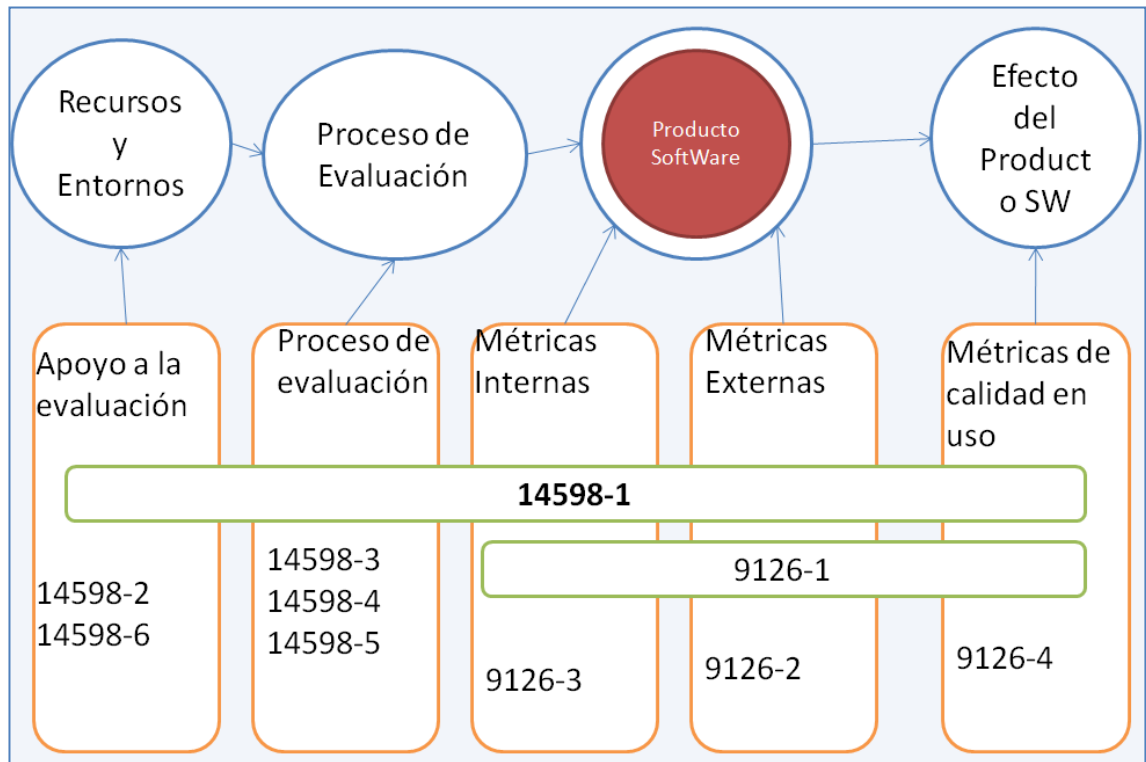
2.2.1.1 Estándar de calidad ISO/IEC 14598: Evaluación del producto software

En el proceso de ciclo de vida del software es necesario considerar mediciones en el proceso usado para diseñar, desarrollar, probar y controlar el producto. La norma ISO/IEC 14598 nos ofrece una visión general que explica una relación entre su serie y el modelo de calidad de la ISO/IEC 9126, define los términos técnicos utilizados, contiene requisitos generales para la especificación y la evaluación de calidad del software, y clarifica los conceptos generales. Además, provee un marco de trabajo para evaluar la calidad de todos los tipos de productos software y establece requisitos para métodos de medición y evaluación de los productos software⁷.

La Figura 4 nos muestra de manera más clara la relación entre las normas ISO/IEC 14598 y 9126 en el proceso de evaluación de la calidad del producto software.

⁷ Tomado de: ICONTEC: Estándar Internacional ISO/IEC 14598-1 Evaluación del producto software (1999). Pag 1.

Figura 4: Relación entre las normas ISO/IEC 9126 y 14598



Fuente: International Standard ISO/IEC 9126-1: 2001

El conjunto de normas de la ISO/IEC 14598 nos proporciona un marco de trabajo para la evaluación de la calidad de todos los tipos de productos software e indica los requisitos para los métodos de medición, y para el proceso de evaluación⁸.

La parte uno de la norma ISO/IEC 14598 explica la relación entre la evaluación del producto software y el modelo de calidad definido en la ISO/IEC 9126. Igualmente, presenta el proceso de evaluación dividido en varios pasos:

- Establecer los requerimientos de evaluación.
- Especificar la evaluación.
- Planear la evaluación.
- Ejecutar la evaluación.

⁸ Ibid. P. 1.

La Figura 5 nos hace una ilustración de manera más clara de cómo se da este proceso.

Figura 5: Proceso de evaluación de un producto software



FUENTE: CALIDAD DE SISTEMAS INFORMÁTICOS CAPITULO 5 PÁGINA 90

Establecer los requisitos de evaluación

✚ Establecer el propósito de la evaluación

El objetivo de la evaluación de la calidad del software consiste en apoyar directamente la producción y adquisición del software que satisfaga las necesidades de los usuarios. El objetivo final es garantizar que el producto proporcione la calidad necesaria para su utilización y que éste satisfaga las necesidades establecidas e implícitas de todos los usuarios involucrados durante el ciclo de vida del software⁹. La evaluación se debe aplicar tanto en productos intermedios como en los finales, esto con el fin de garantizar una calidad general en el producto final. Algunos de los objetivos que se quieren alcanzar para cada tipo de producto se listan a continuación.

⁹ Ibid p. 7.

- **Productos intermedios**
 - Decidir sobre la aceptación de un producto intermedio de un subcontratista.
 - Decidir cuándo un proceso está completo y cuando remitir los productos al siguiente proceso.
 - Predecir o estimar la calidad del producto final.
 - Recoger información con objeto de controlar y gestionar el proceso.

- **Producto final**
 - Decidir sobre la aceptación del producto.
 - Decidir cuándo publicar el producto.
 - Comparar el producto con otros competitivos.
 - Seleccionar un producto entre productos alternativos.
 - Valorar aspectos negativos tanto como positivos.
 - Decidir cuándo es conveniente descargar un producto

Identificar los tipos de productos

El tipo de producto a evaluar dependerá de la etapa del ciclo de vida y del propósito de la evaluación.

El objetivo es que cuando el producto es realmente utilizado por el usuario, y se ajusta a las necesidades que declaró, implica que tiene calidad de uso. La calidad externa sólo puede ser evaluada para un sistema completo de hardware / software en el que el producto de software es solo una parte. Las características externas son aquellas que se evalúan al momento de ejecutar el software. Los valores de las acciones externas dependen necesariamente y en gran medida del software, por lo que este tiene que ser evaluado como parte de un sistema de trabajo. El software debe tener una calidad en uso que ha de satisfacer las necesidades de los usuarios, para llevar a cabo las tareas correspondientes en particular. Un software que funciona

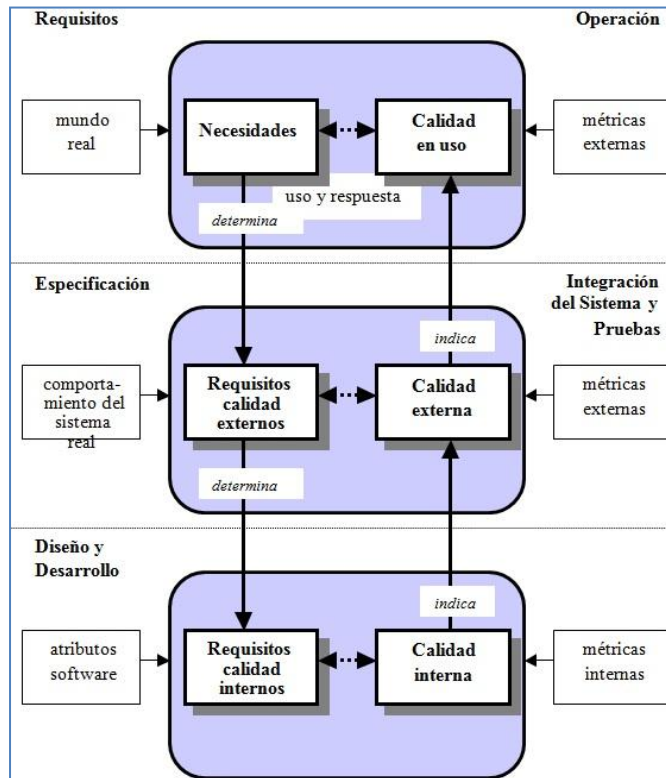
satisfactoriamente en un ambiente, en ocasiones puede mostrar defectos de calidad en otro entorno¹⁰.

Por tanto, las características de la calidad externa son evaluadas bajo condiciones que imitan lo más fielmente las posibles condiciones previstas de uso. Las características externas se miden cuando el código está completo, aunque como no puede ser posible emular las condiciones exactas de uso (por ejemplo, entorno de red y características de los usuarios), las medidas externas son sólo indicadores a menudo de la calidad real de uso.

Si los requisitos de calidad externa no se alcanzan los resultados de la evaluación pueden ser utilizados como comentarios para modificar las características de software con el fin de mejorar ésta, apoyando así un proceso de mejora continua. En la Figura 6 se ve de manera gráfica como se presenta este proceso en la calidad del ciclo de vida del software.

¹⁰ Ibid. P.9.

Figura 6: Calidad en el ciclo de vida del software



Fuente: INTERNATIONAL STANDARD ISO/IEC 14598-1 1999(E)

En los aspectos del desarrollo, se definen los requisitos internos de calidad que le permitan a los productos intermedios ser verificados en términos de calidad. Los desarrolladores y los encargados del mantenimiento normalmente son los únicos interesados en las características internas. La modularidad y la trazabilidad son ejemplos de características internas. Las medidas internas directas pueden ser atributos y pueden ser indicadores de otras características internas. El logro de la calidad interna necesaria contribuirá a satisfacer las necesidades externas del software en uso. Por lo tanto las características internas de calidad del software pueden ser utilizadas como indicadores para estimar la calidad del software final, ver Figura 7.

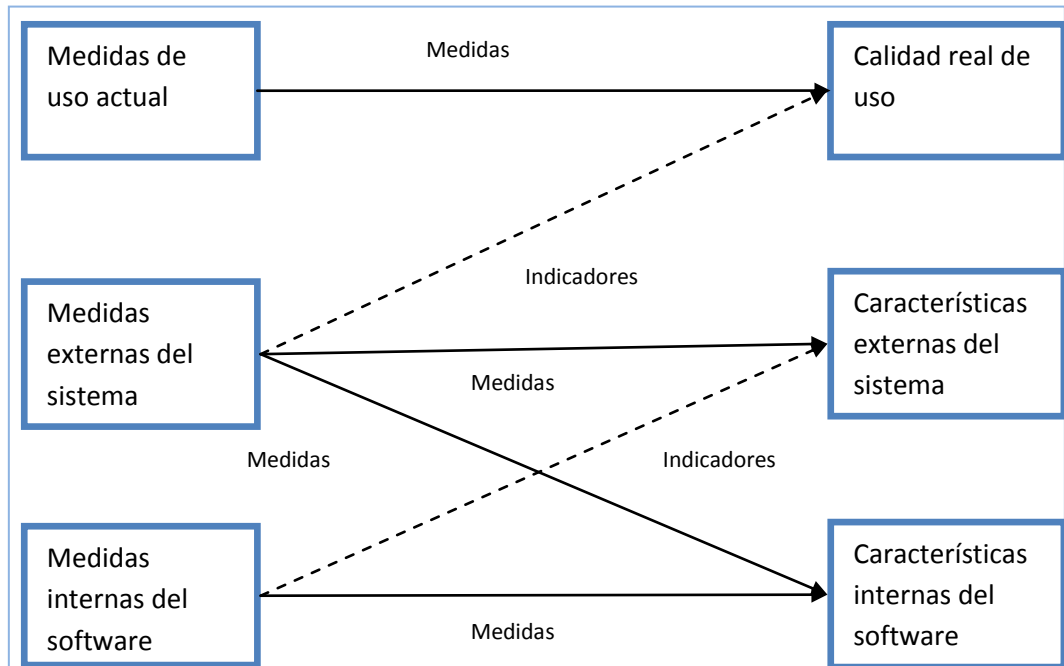
Por ejemplo, el tiempo de respuesta es una medida importante para evaluar la usabilidad y la eficiencia del software, pero el tiempo de

respuesta no se puede medir durante el desarrollo. Con el fin de evaluar la eficacia del producto durante el desarrollo, la longitud del camino se podría medir en función del producto intermedio o sus especificaciones. Estos podrían ser utilizados como indicadores que proporcionan estimaciones aproximadas de tiempo de respuesta en determinadas condiciones.

Es muy importante que los atributos internos de calidad del software estén directamente relacionados con las necesidades de calidad externa, de manera que las características de calidad de los productos software en fase de desarrollo (tanto externo del producto software y productos intermedios) se puedan evaluar con respecto a las necesidades finales del sistema de calidad en uso. Las métricas internas son de poco valor a menos que haya pruebas de que están relacionados con la calidad externa¹¹.

Figura 7: Relación entre medidas y características

¹¹ Ibid. P. 10



Fuente: INTERNAL STANDARD ISO/IEC 14598-1 1999(E)

🚦 Especificar el modelo de calidad

La norma 14598-1 establece que el paso a seguir es elegir un modelo de calidad para la evaluación, la norma ISO/IEC 9126 contiene el modelo de evaluación permite que la calidad del producto software se especifique y se evalúe a partir de diferentes perspectivas asociadas con la adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, control de calidad y auditoría del software. Este puede ser usado por los desarrolladores, compradores, el personal de garantía y de calidad y por los evaluadores independientes, particularmente aquellos responsables de especificar y evaluar la calidad del producto software¹².

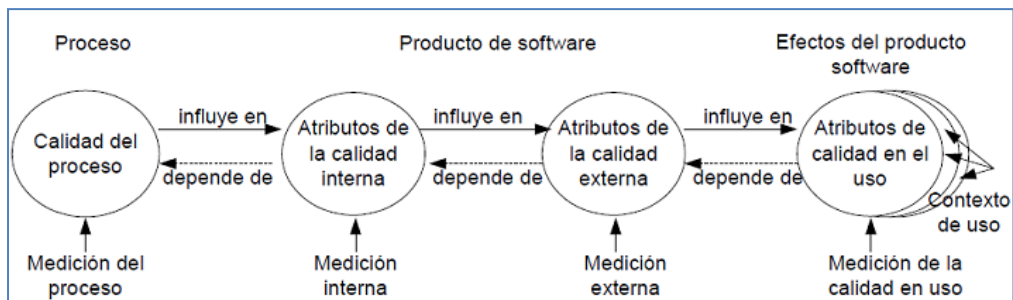
El estándar ISO-9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de seis características básicas, cada una de ellas se detalla a través de un

¹² Ibid. P. 11.

conjunto de sub características que permiten profundizar en la evaluación de la calidad de productos de software.

La figura 8 nos muestra cómo se comporta el modelo, las necesidades de calidad del usuario las cuales incluyen requerimientos para la calidad en las especificaciones de uso. Las necesidades identificadas pueden ser usadas cuando las especificaciones externas e internas de calidad usan características y sub características de la calidad de productos software.

Figura 8: Calidad del modelo durante el ciclo de vida



Fuente: STANDARD ISO/IEC 9126 -1:2001 Página 3

La calidad del proceso contribuye a demostrar la calidad del producto, y a demostrar la calidad en el uso. Por lo tanto la Evaluación de la calidad y la mejora de un proceso es un medio para mejorar la calidad del producto, la evaluación y la mejora de la calidad del producto es uno de los medios de mejora en la calidad de uso. De igual manera, la evaluación de la calidad de uso puede proporcionar realimentación para mejorar un producto, de la misma manera la evaluación de un producto puede proveer realimentación para mejorar un proceso.

Como podemos ver en la figura 8 los atributos internos apropiados del software son un pre-requisito para su logro requiere un comportamiento externo, un apropiado requerimiento externo es un pre-requisito para lograr calidad de uso, esto es el ciclo de vida del software.

ISO/IEC 9126 nos proporciona las normas para evaluar la calidad del producto. La parte uno de esta norma describe un modelo que se divide en dos partes para la evaluación de la calidad de los productos software: a) calidad interna y calidad externa y b) la calidad de uso. La primera parte del modelo especifica seis características de las calidades externas e internas, que se subdividen en sub características. Estas sub características se manifiestan en el exterior cuando el software se utiliza como parte de un sistema informático, y son el resultado de los atributos internos del software. Esta parte de la ISO/IEC 9126 no entra en detalles en el modelo de calidad interna y externa por debajo del nivel de las sub características¹³.

La segunda parte del modelo especifica cuatro características de la calidad de uso, pero no elabora el modelo de calidad en el uso a un nivel inferior de características. La calidad de uso es el efecto combinado del usuario del producto y de las seis características de calidad de software mencionadas anteriormente. Las características y sub características proporcionan una terminología coherente para la calidad de productos software. Igualmente proporcionan un marco para la especificación de requisitos de calidad y para lograr que las ventajas y desventajas del software tengan un equilibrio en las capacidades del producto.

Los requerimientos en calidad de los productos software por lo general incluyen valoración de criterios de calidades internas, y externas de uso, para satisfacer las necesidades de los desarrolladores, los encargados del mantenimiento, compradores y usuarios finales¹⁴

¹³ ICONTEC: Estándar Internacional ISO/IEC 9126-1 Modelo de calidad. En Calidad del producto software. (2000). Pag. 1-3.

¹⁴ Ibid. P. 3.

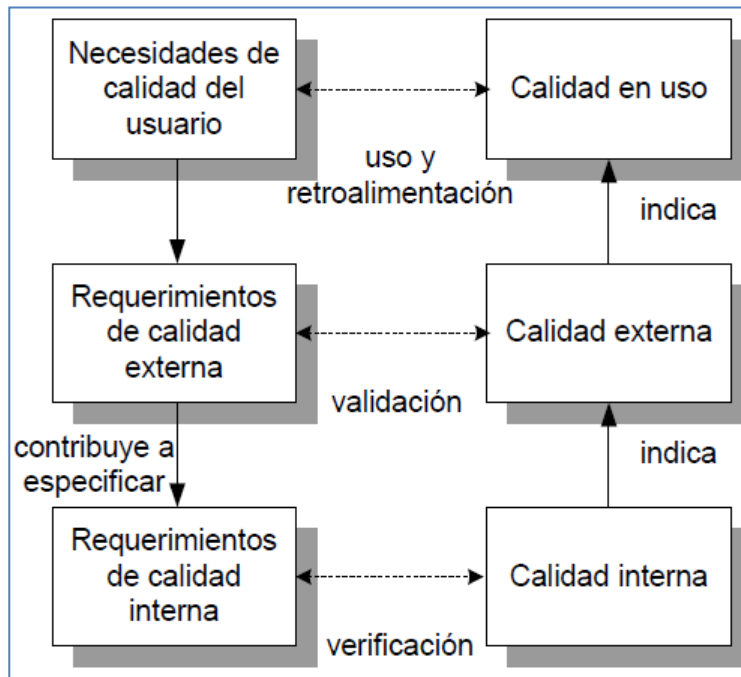
Calidad en el ciclo de vida del software

Durante el ciclo de vida del software es muy normal que los puntos de vista de la calidad interna y externa y la calidad de uso cambien. Por ejemplo especificaciones en la calidad como: en los requerimientos de calidad o en el inicio del ciclo de vida, es sobre todo visto desde el exterior por los usuarios, esta difiere de la calidad de los productos intermedios, tales como diseño de calidad, que se considera, sobre todo la forma interna y de ver a los desarrolladores. Las tecnologías utilizadas para lograr el nivel necesario de calidad, tales como la elaboración y evaluación de la calidad, la necesidad de apoyar a estos puntos de vista diversos, por lo cual es necesario definir estas perspectivas y las tecnologías asociadas a la calidad, con el fin de gestionar la calidad adecuada en cada etapa del ciclo de vida.

El objetivo es lograr la calidad necesaria y suficiente para satisfacer las necesidades reales de los usuarios, las necesidades presupuestadas por el usuario no siempre reflejan las reales, porque: (1) los usuario no suelen ser consistentes con sus necesidades reales, (2) Las necesidades pueden cambiar después de su declaración, (3) los usuarios pueden tener diferentes entornos y (4) puede ser imposible a todos los posibles usuarios, particularmente para el software fuera observación. Los requisitos de calidad, no pueden definirse completamente antes de iniciar la representación de las necesidades, y los objetivos, no necesariamente para lograr la calidad perfecta, pero si una calidad necesaria y suficiente para cada contexto de uso específico, cuando se entrega el producto y realmente es utilizado por los usuarios. El objetivo para que este producto tenga el efecto requerido en un contexto particular de uso se diagrama en la Figura 9¹⁵.

¹⁵ Ibid. P. 3.

Figura 9: Calidad en el ciclo de vida del software



Fuente: ISO/IEC 9126-1:2001. Página 4

Como se observa en la figura 9, existen varias etapas en el ciclo de vida, a continuación se definirán cada una de ellas.

- **Necesidades de calidad del usuario:** se puede especificar los requisitos de calidad a través de la calidad en el uso de las métricas, generalmente por las métricas externas, y algunas veces por las métricas internas. Estos requerimientos especificados por las métricas deben ser usados como criterios cuando se valida un producto.
- **Requerimientos de calidad internos:** especifica el nivel de calidad requerido desde el punto de vista interno del producto. Los requerimientos de calidad internos son usados para especificar propiedades de los productos intermedios. Estos pueden incluir modelos estáticos y dinámicos, otros documentos y códigos fuente. Los requerimientos de calidad interna pueden ser usados como objetivos de validación en varios estados del desarrollo. Ellos pueden también ser usados para definir

estrategias de desarrollo y criterios de evaluación y verificación durante el desarrollo. Esto puede incluir el uso de métricas adicionales que están fuera del alcance de la ISO/IEC 9126. Los requisitos específicos de calidad interno deben ser definidos cuantitativamente usando métricas internas.

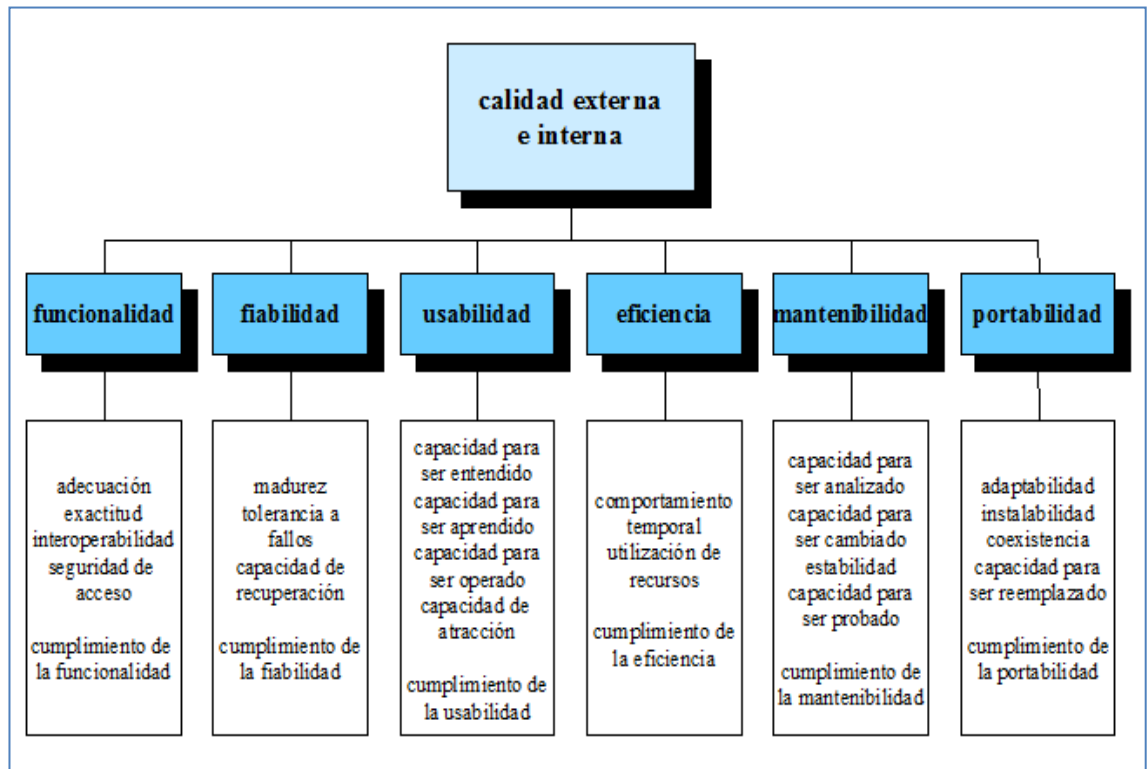
- **Calidad interna:** Es la totalidad de las características del producto software visto internamente. La calidad interna es una forma de medir y evaluar las exigencias de la calidad interna. Detalles en la calidad del producto software pueden ser mejorados durante la implementación del código, la revisión y el control, pero la naturaleza fundamental de la calidad del producto software representada por la calidad interna se mantiene sin cambios a menos que sea rediseñada.
- **Calidad externa estimada:** esta calidad es estimada o prevista por el producto final en cada uno de los estados de desarrollo por cada característica, basado en el conocimiento de las métricas internas.
- **Calidad Externa:** Son todas las características del producto software visto desde el exterior. Esta es la calidad vista cuando el software es ejecutado, que es comúnmente medido y evaluado, mientras que las se realizan pruebas en un entorno simulado con pocos datos usando métricas externas. Durante la prueba de los fallos deben ser descubiertos y eliminados. Sin embargo algunas fallas podrían aparecer o quedar después de las pruebas ya que es difícil corregir la arquitectura del software otros aspectos fundamentales del diseño del software, el diseño general se mantiene sin cambios fundamentales durante el periodo de pruebas.
- **Calidad estimada en uso:** esta cualidad es estimada por el producto final de software, en cada etapa del desarrollo, para cada característica de calidad en el uso, y basado en el conocimiento de la calidad interna y externa.

- **Calidad de uso:** se basa en la opinión del usuario sobre la calidad del producto de software cuando se usa en un entorno específico y un contexto específico de uso. Esta mide el grado en que los usuarios pueden alcanzar sus metas en un ambiente particular en vez de medir del software mismo.

Modelo de calidad para las métricas internas y externas

La calidad de un producto software debería ser evaluado usando un modelo de calidad. ISO 9126-1 propone un modelo de calidad categorizando la calidad de los atributos software en seis características (Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Mantenibilidad y Portabilidad), estas características responden a una pregunta central, igualmente son subdivididas en sub características. Las sub características pueden ser medidas con métricas internas o externas como se ve en la figura 10..

Figura 10: modelo de calidad para la calidad interna y externa



Fuente: International standard ISO/IEC 9126-1 página 7

En la tabla 3 se resume la definición de cada característica y sub característica del modelo, conjuntamente con una pregunta que debe responder cada característica.

Tabla 3: Descripción de las características y sub características de calidad según la norma ISO/IEC 9126-1. 2001

Características	Atributos claves	
<p>Funcionalidad: Se define como un conjunto de atributos que atañen a la existencia de un conjunto de funciones y sus propiedades específicas. Estas funciones son las que satisfacen las necesidades implícitas y establecidas</p> <p>Pregunta clave: ¿Las funciones y propiedades satisfacen las necesidades explícitas e implícitas; esto es, el qué, el cómo,...?</p> <p>Fiabilidad: Conjunto de atributos que atañen a la</p>	<p>Adecuación</p>	<p>Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.</p>
	<p>Exactitud</p>	<p>Capacidad del producto software para proporcionar los resultados o efectos correctos o acordados, con el grado necesario de precisión.</p>
	<p>Interoperabilidad</p>	<p>Capacidad del producto software para interactuar con uno o más sistemas especificados.</p>
	<p>Seguridad de acceso</p>	<p>Capacidad del producto software para proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados.</p>
	<p>Cumplimiento funcional</p>	<p>Capacidad del producto software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con funcionalidad.</p>
	<p>Madurez</p>	<p>Capacidad del producto software para evitar fallar como resultado de fallos en el software.</p>

<p>capacidad del software para mantener su nivel de prestación bajo condiciones establecidas durante un tiempo establecido.</p> <p>Pregunta clave: Puede mantener el nivel de rendimiento, bajo ciertas condiciones y por cierto tiempo?</p> <p>Usabilidad: Capacidad del producto software de ser entendido, aprendido, usado y atraer al usuario, cuando es utilizado bajo</p>	<p>Tolerancia a fallos</p>	<p>Capacidad del software para mantener un nivel especificado de prestaciones en caso de fallos software o de infringir sus interfaces especificados.</p>
	<p>Capacidad de recuperación</p>	<p>Capacidad del producto software para restablecer un nivel de prestaciones especificado y de recuperar los datos directamente afectados en caso de fallo.</p>
	<p>Cumplimiento de la fiabilidad</p>	<p>Capacidad del producto software para adherirse a normas, convenciones o regulaciones relacionadas con la fiabilidad.</p>
	<p>Capacidad para ser entendido</p>	<p>Capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares.</p>
	<p>Capacidad para ser aprendido</p>	<p>Capacidad del producto software que permite al usuario aprender sobre su aplicación.</p>

<p>ciertas condiciones específicas.</p>	<p>Capacidad para ser operado</p>	<p>Capacidad del producto software que permite al usuario operarlo y controlarlo</p>
	<p>Capacidad de atracción</p>	<p>Capacidad del producto software para ser atractivo al usuario.</p>
<p>Pregunta clave: ¿El software es fácil de usar y de aprender?</p>	<p>Cumplimiento de la usabilidad</p>	<p>Capacidad del producto software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.</p>
<p>Eficiencia: Capacidad del producto software para proporcionar un rendimiento apropiado relacionado con el total de recursos utilizados bajo condiciones establecidas.</p>	<p>Comportamiento en el tiempo</p>	<p>La capacidad del producto software para proporcionar adecuados tiempos de respuesta, de procesamiento y de tasas de eficiencia en el desempeño de su función, bajo condiciones establecidas.</p>
	<p>Utilización de los recursos</p>	<p>La capacidad del producto software para utilizar una apropiada cantidad y tipos de recursos cuando el software desempeña su función bajo condiciones establecidas. El Talento Humano está incluido como parte de la productividad.</p>
<p>Pregunta clave: ¿Es rápido y minimalista en cuanto al uso de recursos?</p>	<p>Conformidad con la eficiencia</p>	<p>La capacidad del producto software para adherirse a estándares o convenciones relacionadas con la eficiencia.</p>

<p>Mantenibilidad: Capacidad del producto software para ser modificado.</p> <p>Pregunta clave: ¿Es fácil de modificar y verificar?</p>	<p>Capacidad para ser analizado</p>	<p>Es la capacidad del producto software para serle diagnosticadas deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas.</p>
	<p>Capacidad para ser cambiado</p>	<p>Capacidad del producto software que permite que una determinada modificación sea implementada.</p>
	<p>Estabilidad</p>	<p>Capacidad del producto software para evitar efectos inesperados debidos a modificaciones del software.</p>
	<p>Capacidad para ser probado</p>	<p>Capacidad del producto software que permite que el software modificado sea validado.</p>
	<p>Cumplimiento de la mantenibilidad</p>	<p>Capacidad del producto software para adherirse a normas o convenciones relacionadas con la mantenibilidad.</p>
	<p>Portabilidad: Capacidad del producto software para ser transferido de un entorno a otro. El entorno</p>	<p>Adaptabilidad</p>

se interpreta tanto a nivel software y hardware, como aquel entorno relacionado con la organización.

Pregunta clave: ¿Es fácil de transferir de un ambiente a otro?

Instalabilidad

Capacidad del producto software para ser instalado en un entorno especificado.

Coexistencia

Capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.

Capacidad para ser reemplazado

Capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno.

Cumplimiento de la portabilidad

Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.

Fuente: Resumen de características de calidad ISO/IEC 9126-1: 2001 pag7-11

En la tabla 4 podemos ver como es la estructura una característica en este caso Funcionalidad y la sub característica Idoneidad.

Tabla 4: Métricas de idoneidad

Métricas de idoneidad externas									
Nombre de la métrica	Propósito de las métricas	Método de aplicación	Medida, formula y cálculos de elementos de datos	Interpretación del valor de la medida	Tipo de escala de la métrica	Tipo de medida	Entrada a la medida	Referencia ISO/IEC 12207 SLCP	Público objetivo
Adecuación funcional	Qué tan adecuadas son las funciones evaluadas?	Número de funciones que son adecuadas para realizar tareas específicas comparadas con el número de funciones evaluadas	$X = 1 - A / B$ A = número de funciones en las cuales se detectaron problemas en la evaluación B = número de funciones evaluadas	$0 \leq X \leq 1$ A mayor cercanía a 1.0, mayor adecuación	Absoluta	X = contador/contador A = contador B = contador	Especificación de requerimientos (Esp. Req.) Reporte de evaluación	6.5 Validación 6.3 Garantía de seguridad 5.3 Calificación de las	Desarrollador. SQA

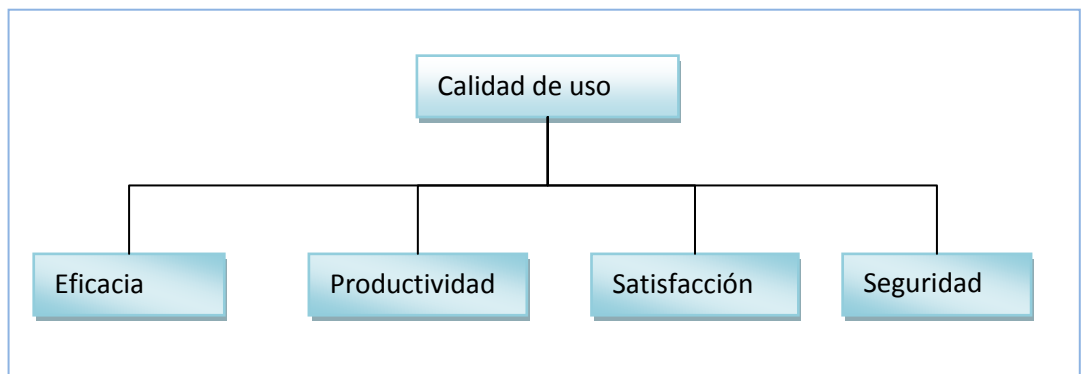
Integridad de implementación funcional	Qué tan completa es la implementación de acuerdo a la especificación de requerimientos?	Hace pruebas funcionales (pruebas de baja negra) del sistema de acuerdo a la especificación de requerimientos. Contar el número de funciones ausentes detectadas en la evaluación y compararlas con el número de funciones descritas en la especificación de requerimientos	$X = 1 - A / B$	$0 \leq X \leq 1$	Absoluta	A = contador X = contador X = contador/contador	Esp. Req. Reporte de evaluación	pruebas	Desarrollador. SQA
								6.5 Validación	

Fuente: ISO/IEC 9126-2: 2001

Modelo para la calidad en el uso

La capacidad de uso es la capacidad del producto software para permitir a determinados usuarios conseguir determinados objetivos con efectividad, productividad, seguridad de acceso y satisfacción en contextos de uso específico¹⁶, ver figura 11.

Figura 11: Esquema de la calidad de uso



Fuente: International standard ISO/IEC 9126-1 página 7

La tabla 5 describe las características de la calidad de uso.

Tabla 5: Características de la calidad de uso

Característica	Descripción
<i>Eficacia</i>	Capacidad del software para permitir a los usuarios alcanzar objetivos específicos con precisión y completamente en un contexto específico de uso.
<i>Productividad</i>	Capacidad del producto software para permitir a los usuarios emplear recursos apropiados con relación a la eficacia alcanzada en un contexto específico de uso.
<i>Satisfacción</i>	Capacidad del producto software para satisfacer al usuario en un contexto específico de uso.
<i>Seguridad</i>	Capacidad del producto software para alcanzar niveles aceptables de riesgo hacia la gente, negocio, software,

¹⁶ Op. cit. ISO 9126-1. P. 12.

propiedad o medio ambiente, en un contexto específico de uso.

Fuente: International standard ISO/IEC 9126-1

Especificar la evaluación.

Seleccionar métricas

La informática ha evolucionado desde un enfoque puramente artesanal hasta llegar a uno más científico, donde cada vez es mayor la necesidad de justificar de una manera razonada tanto los diseños como la propia programación. Es, por tanto, cada vez más importante disponer de un método y de un sistema de *MÉTRICAS* adecuados a nuestras necesidades¹⁷.

Una **métrica** es una medida efectuada sobre algún aspecto del sistema en desarrollo o del proceso empleado que permite, previa comparación con unas medidas de referencia, obtener conclusiones sobre el aspecto medido con el fin de adoptar las decisiones necesarias. Con esta definición, la definición y aplicación de una métrica no es un objetivo en sí mismo sino un medio para controlar el desarrollo de un sistema de software¹⁸

La medición del software es muy importante ya que permite a los administradores y desarrolladores entender mejor el proceso de desarrollo, así como la calidad del software que se produce.

¹⁷ Necesidad de sistemas formales de métricas para proyectos de software OO. Sevilla-JJOO-1997.pdf disponible en: <http://www.di.uniovi.es/~aquilino/ficheros/articulos/Metricas/Sevilla-JJOO-1997.pdf>

¹⁸ Op.cit. ISO 9126-1. P. 14.

Características de las métricas

Se afirma que para que las métricas sean útiles éstas deben tener las siguientes cuatro características:

- Cuantificables, deben basarse en hechos, no en opiniones.
- Independientes, los recursos no deben poder ser alterados por los miembros que las apliquen o utilicen.
- Explicable, debe documentarse información acerca de la métrica y de su uso.
- Precisas, debe de conocerse un nivel de tolerancia permitido cuando se mide.

En general, la medición persigue tres objetivos fundamentales [1]:

- Entender qué ocurre durante el desarrollo y el mantenimiento
- Controlar qué es lo que ocurre en nuestros proyectos
- Mejorar nuestros procesos y nuestros productos

Beneficio de las métricas

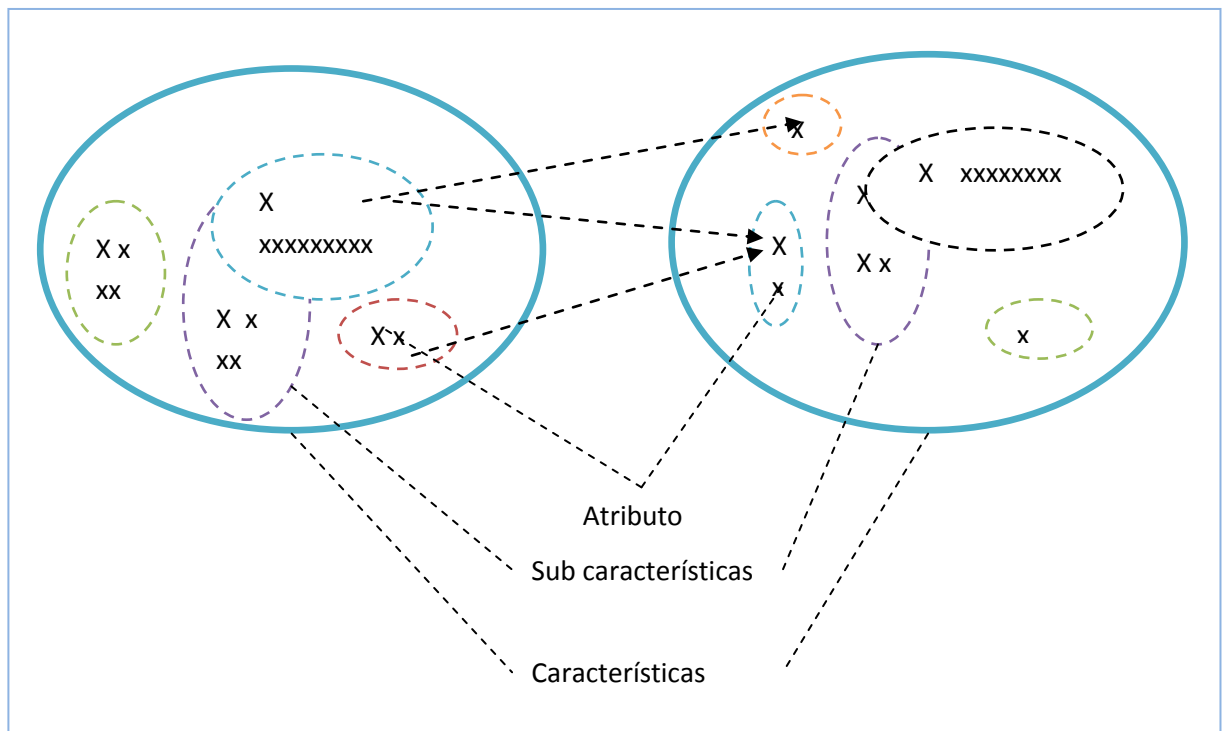
- El proceso del software (para mejorarlo).
- El proyecto del software (para ayudar a estimar, control de calidad, evaluación de productividad, control de proyectos).
- Calidad del producto (para ayudarse en la toma de decisiones tácticas a medida que el proyecto evoluciona).

Atributos internos y externos

Las características de calidad en las que la norma ISO 9126-1 se descompone, por atributos internos y externos propios de dichas características. Los atributos internos son indicadores de los atributos

externos. Un atributo interno puede influir a una o más características y una característica puede verse influida por uno o más atributos. Las características y sub características son medidas, por tanto, a través de sus correspondientes atributos ver figura 12.

Figura 12: Características, sub característica y atributos de calidad



Fuente: International standard ISO/IEC 9126-1. Pag 14

Como se observa en la grafica 11la totalidad de atributos de la calidad del software son clasificados jerárquicamente en una estructura definida en características y sub características. El nivel más alto de esta estructura consiste en las características de calidad y el nivel más bajo en los atributos de calidad del software. La jerarquía no es perfecta debido a que algunos atributos pueden contribuir a más de una sub característica¹⁹.

Las sub características pueden ser medidas por métricas internas o por métricas externas. La interrelación entre las los atributos internos y las medidas externas nunca es perfecta, y el efecto que los atributos

¹⁹ Op.Cit. ISO 9126-1. Pag. 14.

internos tienen sobre las medidas externas serán determinados por la experiencia, y dependerá del contexto particular en el cual el software es usado.

De la misma manera, las propiedades externas (como: idoneidad, precisión, tolerancia a fallos, comportamiento en el tiempo), influenciara en las cualidades observadas. Un fallo en la calidad de uso (Eje: el usuario no puede completar una tarea), se puede remontar a los atributos de cualidades externas (Idoneidad u Operabilidad).

Métricas internas

La norma define las métricas internas como aquellas medidas que se realizan sobre un producto software no ejecutable, tal como la norma indica “un producto software intermedio debería ser evaluado usando métricas internas”²⁰.

Métricas externas

Las métricas externas son medidas del producto software obtenidas del comportamiento del sistema en la fase de ejecución del mismo. Las métricas de la calidad del uso, como tercer gran concepto propuesto por la norma, miden la extensión en la que un producto alcanza las necesidades expuestas por el usuario de forma específica en relación a los objetivos de efectividad, seguridad, productividad y satisfacción [3].

Relación entre métricas internas y externas

Cuando son definidos los requerimientos de calidad del producto software, las características o sub características de calidad del producto contribuye a los requerimientos de calidad enumerados. Entonces, las

²⁰ Ipcid. P. 15.

métricas apropiadas y los rango aceptables son especificados para cuantificar los criterios de calidad los cuales validaran que cumpla con las necesidades de los usuarios. Los atributos de calidad interna del software son luego definidos y especificados para lograr finalmente el plan que requieren la calidad externa y la calidad de uso para construir dentro del producto durante su desarrollo²¹.

Es recomendable que las métricas internas tengan fuerte relación con las métricas externas para que puedan ser usadas para predecir los valores de las métricas externas.

La interpretación de las medidas se puede realizar de tres formas:

- Medida directa. Una medida directa es una medida de un atributo que no depende de las medidas de otros atributos.
- Medida indirecta. Una medida indirecta es derivada de medidas de uno o más atributos.
- Indicadores. Son aquellas medidas que pueden ser estimadas o predichas desde otras medidas.

Las métricas tienen ciertas propiedades deseables que se detallan a continuación:

- Fiabilidad.
- Indicabilidad.
- Disponibilidad.
- Corrección.
- Imparcialidad.

²¹ Op.Cit ISO/IEC 9126-1. P. 15.

El conjunto de métricas que contiene están organizadas por características y sub características, donde cada métrica contiene:

- Nombre.
- Propósito.
- Método de aplicación.
- Medida, fórmula y cómputo de datos.
- Interpretación del valor medido.
- Tipo de escala.
- Tipo de medida.
- Fuente de medida.
- Audiencia.

La norma 9126-3 proporciona métricas internas para medir los atributos de las características de calidad definidas en la norma 9126-1. Con las siguientes cualidades:

- Se aplican a un producto de software no ejecutable.
- Se aplican durante las etapas de desarrollo.
- Permiten medir la calidad de los entregables intermedios.
- Permiten predecir la calidad del producto final.
- Permiten al usuario iniciar acciones correctivas temprano en el ciclo de desarrollo.

El conjunto de métricas están organizadas igualmente por características y sub características, donde tiene los mismos campos que la norma ISO 9126-2. Por lo tanto existirán métricas de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. Las propiedades ISO 9126 deseables son las siguientes: confiable, repetible, reproducible, disponible, indicable, correcta y con significado. Los pasos que se sugieren son los siguientes:

- Identificación de los requisitos de calidad.
- Especificación de la evaluación.
- Diseño de la evaluación.
- Ejecución de la evaluación.
- Retroalimentación a la organización.

Establecer niveles para las métricas

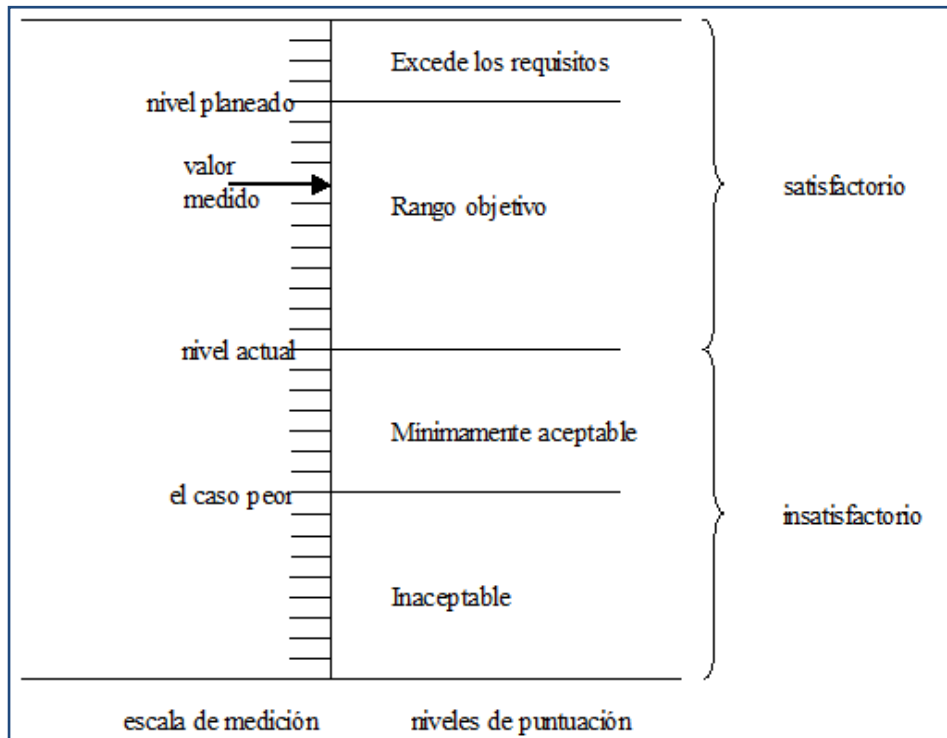
Los valores que arrojan las métricas son ubicados en una escala, esta escala mide el nivel de satisfacción de los requisitos y está dividida en rangos en donde un rango alto es de mejor calidad que un rango bajo, es decir:

- La escala se divide en dos categorías: Satisfactorio e insatisfactorio.
- La escala se divide en cuatro categorías, limitadas por el nivel actual de un producto existente o alternativo, el peor caso y el nivel proyectado.

El nivel actual se declara con el fin de controlar que el nuevo sistema no suponga un deterioro en la relación a la situación actual. El nivel proyectado es lo que se considera alcanzable con los recursos disponibles. El peor caso es la frontera para la aceptación del usuario por si acaso el sistema no cumple el nivel proyectado [1].

La Figura 13 describe mejor la división de los rangos en la escala de medición.

Figura 13: Rangos de una escala de medición



Fuente: ISO/IEC 14598-1: 1999. (Figura 7)

✚ Establecer criterios de valoración

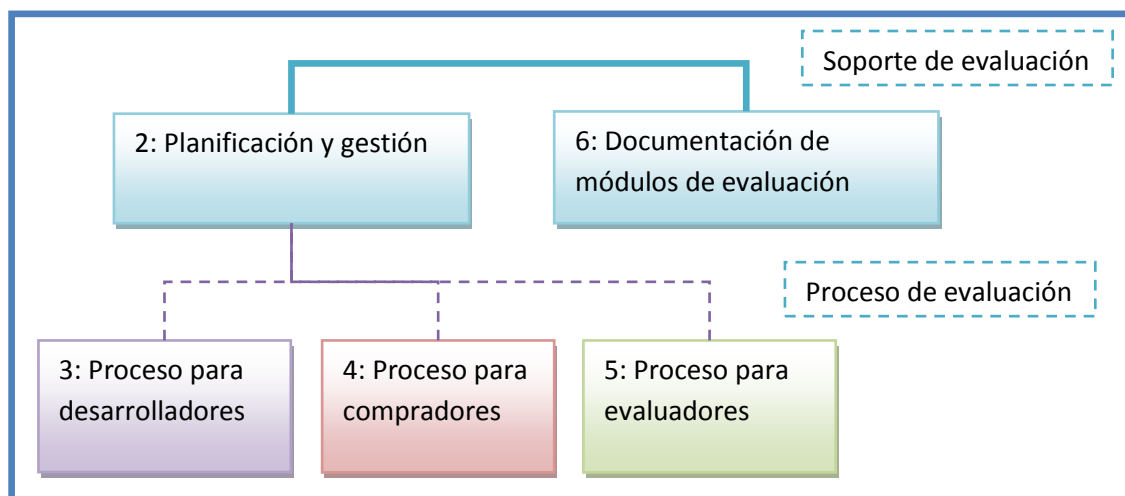
Para hacer una buena valoración de la calidad del software se deben establecer criterios para diferentes características de calidad, los cuales deben estar en términos de sub características individuales, o una combinación de sub características ponderadas, el procedimiento generalmente incluye otros aspectos como: tiempo y costo esto contribuye a la evaluación de la calidad del producto software en un entorno particular.

Diseñar la evaluación

✚ Producir un plan de evaluación

El plan de evaluación describe los métodos de evaluación y el programa de acciones del evaluador (ISO/IEC 14598-3, ISO/IEC 14598-5 o ISO/IEC 14598-5). Debe ser consistente con el plan de mediciones (ISO/IEC 14598-2), ver figura 14.

Figura 14: Plan de evaluación



Fuente: Tomado de *International Estándar ISO/IEC 14598-1: 1999(E)*

14598-2 Planificación y gestión.

Esta parte contiene los requerimientos y las guías para las funciones de soporte tales como el planeamiento y gestión para la evaluación del producto del software. Fundamentalmente, en esta parte, se planifican las mediciones y las actividades de evaluación. Específicamente²², se incluye:

- Preparación de las políticas.
- Definición de objetivos organizacionales y de mejora.
- Identificación de la tecnología.
- Asignación de responsabilidades.
- Identificación e implementación de técnicas de evaluación para software desarrollado y adquirido.
- Entrenamiento en tecnología, recopilación de datos y herramientas.
- Comparación y administración de mejoras dentro la organización.

²² ICONTEC: Estándar Internacional ISO/IEC 14598 2 Evaluación del producto software (1999). Pag 1.

La evaluación cuantitativa de la planificación debe incluir los siguientes ítems, si alguno de ellos no se aplica debe indicarse.

- **Introducción:** Esta debe contener:
 - El propósito del plan
 - La audiencia del plan
 - La intención del plan

- **Evaluación de los objetivos:** En este capítulo se debe proporcionar una declaración clara sobre los objetivos de la evaluación y la aplicación prevista del software. Deben ser utilizables con el fin de fijar los objetivos de calidad y respectivos criterios.

- **Características aplicables de calidad:** En este ítem se deben proporcionar declaraciones de las características de calidad, en la que se apoyan los objetivos. Los objetivos de calidad establecidos pueden ser a la vez productos y procesos orientados.

- **Lista de prioridades:** En esta parte se debería dar prioridad a las características del estudio y se debe proporcionar un sustento racional para estas prioridades.

NOTA - Las prioridades pueden cubrir los requerimientos del proceso, (por ejemplo, desde el punto de vista del negocio, formación del personal puede ser considerado una prioridad más alta).

- **Objetivos de Calidad (Características):** En esta parte se deben proporcionar objetivos cuantificables de calidad que se pueden medir en las fases intermedia o final del proyecto desarrollo. Se podrá indicar el máximo tiempo de inactividad de un producto instalado o del sistema.

- **Planificación:** En este ítem se debe proporcionar un plan claro con objetivos, metas y resultados establecidos.
- **Definición de responsabilidades:** Aquí se deben definir todas las responsabilidades previstas relacionadas con la aplicación del Plan. Esto incluye todos los datos recogida, análisis de tareas, la aplicación de otros requisitos de apoyo, información, seguimiento y requisitos similares.
- **Categorías de medición:** En esta parte deben definir las diferentes medidas que se prevé que se llevaran a cabo. Si las medidas se toman en los productos y procesos se deben clasificar y llevar a cabo de la siguiente manera:
 - **Las mediciones de calidad del producto:**
Incluyen medidas tales como rendimiento, fiabilidad y portabilidad. Estado en qué esta la etapa de desarrollo de estas mediciones, con qué frecuencia se repiten, ¿qué técnicas o herramientas se utilizan para ayudar a la captura de datos y análisis, y las acciones previstas cuando hay divergencias de los objetivos establecidos. Estas mediciones pueden ser en el producto final o en productos intermedios, que se puede obtener mediante el uso de sub-características.
 - **Las mediciones de procesos de calidad:**
Describir la forma de medidas de proceso que se llevó a cabo. Incluye el seguimiento de la eficacia de las normas, herramientas de uso, gestión de proyectos, etc.
Estas medidas también pueden ser llevadas a cabo en varias fases del ciclo de vida del proyecto. Definir y documentar la

manera en que los procesos relacionados y las acciones correctivas se llevarán a cabo, si se considera necesario.

- **Uso y análisis de los datos:** Se debe definir la cantidad de datos que van a ser analizados, en este caso que métodos estadísticos se van a emplear y qué presentación técnicas se debe utilizar.
Se debe hacer referencia a las responsabilidades que se han dicho, las herramientas de apoyo. También se debe indicar cómo la información
Se prevé que se integrará en el seguimiento del progreso, del proceso de aceptación del producto o dentro de él.
- **Reportes:** En esta parte se debe definir si los resultados analizados se habrán de comunicar dentro o fuera del proyecto o en la evaluación de productos y también debe definir la manera de resolver todos los temas pendientes.
- **Otros requerimientos:** Esta parte se puede utilizar para incluir los requisitos no cubiertos con anterioridad, por ejemplo, puede incluir la siguiente información.
 - **Técnicas y métodos empleados:** Proporciona una descripción completa de las técnicas y métodos utilizados (por ejemplo, el método para determinar el tamaño; el de evaluación del desarrollo de vencimiento, método de inspección para la detección de errores; modelo de corrección de defectos para predecir los índices de error). La sección o el material que se refiere debe ser clara y completa para que pueda ser fácilmente comprendido y utilizado por todo el personal.

- **Apoyar las herramientas:** Describir o proporcionar referencias - como en la parte de técnicas y métodos, pero cubriendo las necesidades de apoyo de la herramienta. Esto puede incluir guías para el uso de bases de datos, hojas de cálculo y paquetes estadísticos.
- **Las normas pertinentes y guías:** Describir el uso y los beneficios pertinentes a la compra y adquisición de normas y modelos para la evaluación.
- **Evaluación de proveedores:** Incluir la evaluación y medición de los procedimientos para la efectiva evaluación cuantitativa para proveedores de los productos software. Esto puede incluir el número de ejemplares en el mercado, el estado actual de error, las encuestas sobre el rendimiento posterior instalación de apoyo, las estadísticas, sobre el pasado y los usuarios satisfechos actualmente, la gestión del rendimiento y la estabilidad financiera. Los parámetros relacionados de interés para la aplicación, que hayan sido obtenidos de otros proveedores, esto se puede incorporar en el plan de evaluación de los proveedores.

14598 – 3 Proceso para desarrolladores

Esta parte provee los requerimientos y las recomendaciones para la evaluación del producto software cuando la evaluación es conducida en paralelo con el desarrollo y llevada a cabo por el desarrollador. Se enfoca en el uso de indicadores que pueden predecir la calidad final del producto midiendo los productos intermedios que se desarrollan durante el ciclo de vida. Esta parte cubre el planeamiento y evaluación de mediciones

internas y externas con el fin de asegurar de que la calidad del producto sea incorporada en la fase de desarrollo²³.

Una vez identificadas las características fundamentales de calidad y el marco de trabajo de mediciones, deben ser definidas las etapas siguientes:

- **Organización**

Los aspectos organizacionales de desarrollo y de soporte deben formar parte de todo el sistema de calidad y del plan de mediciones. Véase la ISO/IEC 14598 - Parte 2.

- **Planeamiento del Proyecto y Requerimientos de Calidad**

El desarrollo y el ciclo de vida de soporte deben ser establecidos y documentados durante el plan de calidad o en otros documentos. Es de vital importancia verificar que el productor y las medidas de control requeridas sean técnicamente factibles, razonables y alcanzables (dentro de los límites de tiempo).

- **Especificaciones**

En esta fase, el desarrollador realiza un mapeo de los requerimientos internos y externos de calidad, con relación a las especificaciones. Los requerimientos de mediciones resultantes de esta fase deben ser un tipo de mapeo entre las especificaciones de requerimientos, requerimientos externos de calidad, requerimientos internos correspondientes de calidad y atributos especificados junto a sus escalas de medición y valores objetivos que contribuyan a la cuantificación de la calidad del software. Todo esto puede enfocarse por proyecto o por producto.

²³ ICONTEC: Estándar Internacional ISO/IEC 14598-3 Evaluación del producto software (1999). Pag. 1

- ***Diseño y Planeamiento***

Los procedimientos requeridos para el análisis y recopilación de datos necesitan ser definidos. De esta manera, el plan incluirá: cronogramas, designación de responsabilidades, uso de herramientas, bases de datos y entrenamiento especializado requerido. La precisión de las mediciones y técnicas estadísticas deben ser especificadas (véase la ISO/IEC 14598 - Parte6). En esta fase también deberá considerarse cómo los resultados de las mediciones impactarán en el desarrollo; por lo tanto, acciones de contingencia y de mejora, deben ser consideradas.

- ***Montaje y Pruebas***

Durante la etapa de montaje y pruebas, las mediciones actuales son recolectadas, se realizan análisis apropiados y se toman acciones necesarias. En cada fase del desarrollo debe procurarse lograr un montaje primeramente enfocado a las características internas y externas de calidad que definan la calidad global del producto y que puedan ser validadas por los resultados de las pruebas y la experiencia del usuario.

14598 - 4 Proceso para compradores.

Esta parte provee los requerimientos y las recomendaciones para que la evaluación del producto software sea conducida en función a los compradores que planean adquirir o re-usar un producto de software existente o pre-desarrollado²⁴.

Los que adquieren el producto pueden comprar paquetes completos ya sea desarrollados según ciertas especificaciones o pre-desarrollados para un mercado más general. Los compradores también podrían ser desarrolladores que desean integrar productos estándar en sus propios

²⁴ ICONTEC: Estándar Internacional ISO/IEC 14598-4. Evaluación del producto software (1999). Pag. 1

diseños de software, o tratarse de desarrolladores buscando herramientas específicas de software. Al respecto, cuatro etapas son necesarias:

- ***Establecimiento de los Requerimientos***

El alcance de la evaluación necesita ser establecido. Los requerimientos para la calidad del software definidos en la ISO/IEC 9126 pueden ser usados como punto de partida pero otros aspectos como el costo y el de cumplimiento a regulaciones deberán ser también considerados. El tiempo de la evaluación necesita ser consistente con los objetivos; enfoques muy tempranos podrían no proporcionar una figura adecuada de la situación mientras que enfoques muy tardíos podrían ser muy limitados en su uso.

- ***Especificación de la Evaluación***

Durante la redacción de las especificaciones, debe considerarse:

- Los requerimientos de calidad a ser evaluados correlacionados con la calidad en uso y métricas externas con prioridades además de un umbral de aceptación definido.
- El alcance y lo que cubren los casos de prueba donde sean aplicables referencias a módulos de evaluación.
- Métodos de recolección de mediciones, información requerida y métodos de análisis.

- ***Diseño de la Evaluación***

El tipo de evaluación depende del tipo de software que está siendo evaluado. Software bajo desarrollo puede ser abordado en puntos discretos durante el desarrollo o cuando esté completo. Un plan de evaluación necesita considerar:

- Necesidades de acceso a la documentación del producto, herramientas de desarrollo y personal.
- Requerimientos en costos y conocimientos.

- Cronograma de evaluación y arreglos de contingencia, hitos claves y criterio para decisiones de evaluación.
 - Métodos y herramientas de reporte, procedimientos para la validación y estandarización sobre proyectos futuros.
- ***Ejecución de la Evaluación***
Aunque esta etapa podría ser simplemente un registro en un libro de seguimiento, podría tenerse la necesidad de incluir:
 - Los resultados mismos y la trazabilidad del producto así como información de configuración.
 - Registros de análisis, resultados y decisiones.
 - Problemas, limitaciones en las mediciones y cualquier compromiso con relación a los objetivos originales
 - Conclusiones sobre los resultados de la evaluación pero también sobre los métodos empleados

14598 – 5 Proceso para evaluadores.

Esta parte provee los requerimientos y recomendaciones para la evaluación del producto software cuando la evaluación es conducida por evaluadores independientes. En esta parte, tienen un rol importante los requerimientos de evaluación, las especificaciones de evaluación, el diseño de la evaluación²⁵, las actividades de evaluación y el reporte de evaluación. Estas etapas son resumidas a continuación:

- ***Requerimientos de Evaluación***

Los requerimientos deberían adicionalmente definir:

- La extensión de la cobertura (o el alcance).
- Los objetivos de evaluación y métodos de reporte.
- Las calificaciones e independencia requeridas de un evaluador.

²⁵ ICONTEC: Estándar Internacional ISO/IEC 14598-5. Evaluación del producto software (1999). Pag 1.

- ***Especificación de la Evaluación***

Las especificaciones adicionalmente deberían cubrir:

- Definición del alcance y formato en las métricas empleadas identificando como deberán ser derivadas a partir de los requerimientos del producto.
- La identificación de mediciones no determinadas para asegurar que ciertos niveles de frecuencia y objetividad requeridos sean obtenidos.
- La identificación de métodos de correlación con relación a los resultados de las mediciones.

Se tienen identificadas tres sub-actividades con relación a la especificación de la evaluación:

- El análisis de la descripción del producto.
- La especificación de las mediciones a ser realizadas.
- La verificación de la especificación resultante frente a los requerimientos de evaluación.

14598 – 6 Documentación de los módulos de evaluación.

Esta parte provee las guías para la documentación del módulo de evaluación. Estos módulos representan la especificación del modelo de calidad y las correspondientes métricas internas y externas que serán aplicadas a una evaluación en particular. Incluye métodos y técnicas de evaluación más las mediciones actuales resultantes de su aplicación. En esta parte también se considera la administración efectiva de complejidades inherentes a las cuestiones de medición.

Las actividades de medición coordinadas son una característica para una evaluación efectiva y un plan necesita proveer un cronograma de evaluación que provea al mismo tiempo información óptima cuando la

evaluación sea conducida durante el desarrollo. Los módulos de la evaluación son componentes claves de la ISO/IEC 14598-6 y son usados para proveer un formato consistente y repetible de reporte. Dichos módulos proveen:

- Visibilidad de la información necesitada para cuadrar con requerimientos específicos de calidad.
- Documentación de las interfaces necesarias con herramientas de medición.

La ISO/IEC 14598-6 trata también sobre los requerimientos de la documentación y divide a los módulos de evaluación en los seis componentes siguientes:

- Introducción – Cubre el control del documento, las relaciones con otros documentos, los requerimientos técnicos y una razón para el módulo.
 - Alcance – Se relaciona con la características de calidad o sub-características que deberán ser alcanzadas, el nivel de la evaluación (tomando en cuenta la importancia de la característica, la técnica de evaluación usada incluyendo cualquier teoría necesaria) y la aplicabilidad del módulo.
 - Referencias.
 - Definiciones requeridas.
 - Entradas requeridas – Datos a ser recopilados y métricas a ser calculadas.
 - Información sobre la interpretación de los resultados
- **Resultados de la Evaluación**

En esta etapa se tiene la generación del reporte de evaluación incluyendo una revisión independiente de los resultados de la evaluación. Normalmente, el reporte final será precedido por un

borrador de tal manera que el personal involucrado con el producto pueda proveer una retroalimentación sobre la evaluación.

Ejecutar la evaluación

Tomar medidas.

Para la medición, las métricas seleccionadas se aplicaran a los productos software. Los resultados son los valores en las escalas de medición de las de las métricas.

Comparar con criterios.

Los resultados de las medidas se comparan con los criterios establecidos, para poder establecer el estado real del producto.

Valorar resultados.

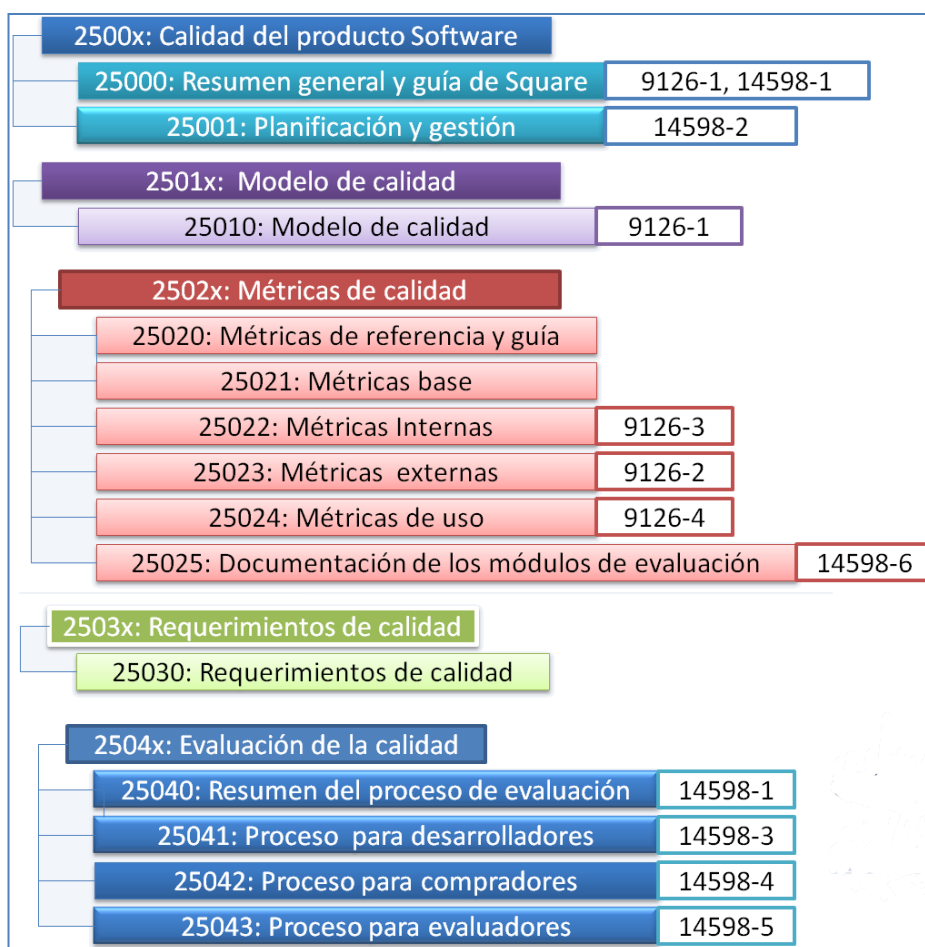
La evaluación es el último paso en el proceso de evaluación del producto software. Los resultados son las declaraciones de las medidas en que el producto software cumple con los requisitos de calidad, estos resultados se comparan con otros aspectos tales como el tiempo y los costos. Finalmente, una decisión de gestión se hará con base en los criterios de gestión. El resultado es una decisión administrativa sobre la aceptación o rechazo, o sobre la liberación o no del producto de software.

Los resultados de la evaluación son muy importantes para las decisiones respecto a las nuevas etapas en el desarrollo del ciclo de vida del producto software, por que en estas etapas se pueden prevenir fallas en el producto final.

2.2.1.2 ISO / IEC 25000:2005 - Software product Quality Requirements and Evaluation (SQuaRE)

ISO / IEC 25000:2005 proporciona una guía para el uso de la nueva serie de normas internacionales denominada SQuaRE. Esta norma nos proporciona una definición de modelos y referencias comunes, así como la relación entre los documentos, permitiendo a los usuarios de esta guía una buena comprensión de las series de normas internacionales, de acuerdo con su propósito de uso. Esta norma contiene una explicación del proceso de transición entre las ya mencionadas ISO / IEC 9126 y la 14598, la figura 15 muestra claramente la relación de estas normas.

Figura 15: Relación entre las normas ISO 25000 y 9126 - 14598



Fuente: <http://alarcos.inf-cr.uclm.es/doc/cmsi/trabajos/Joaquin%20Ruiz.pdf> pag 15-16

2.2.2 Herramientas de evaluación

“Para que las métricas puedan ser evaluadas de un modo práctico, eficiente y exacto es necesario contar con herramientas que permitan automatizar la adquisición, la presentación y el análisis de los valores obtenidos para dichas métricas”.²⁶

Existen en el mercado herramientas que fueron creadas con el fin de evaluar la calidad de los productos software, esto nació debido a la necesidad de hacer más práctica y más ágil la evaluación, en el desarrollo del estado del arte acerca de las herramientas para la evaluación de la calidad de software se evidenció que la gran mayoría de herramientas que existen se limitan a medir solo el código fuente, o solo una parte de la calidad, otras son creadas muy a la medida para evaluar cierto producto; en la tabla 6 se resume y se muestran algunas de estas herramientas y ciertas generalidades de ellas.

Existen dos clasificaciones generales para este tipo de herramientas estas son: **Herramientas de Análisis Dinámico:** aquellas herramientas que realizan el análisis del software ejecutando el código fuente de dicho software.

Herramientas de Análisis Estático: aquellas herramientas que llevan a cabo el análisis sin necesidad de ejecutar el software bajo estudio.

Tabla 6: Herramientas para la evaluación de la calidad del producto software

Nombre	Licencia	Interfaz	Lenguajes	Entrada	Código
PMD/CPD	BSD-style	Línea de comandos	Java	Código fuente	Texto plano xml
CheckStyle	LGPL	Línea de comandos	Java	Código fuente	Texto plano xml
Klocwork	Software	GUI Línea	C/C++/Java	Byte code	HTML

²⁶Giles A, Daich G. 1995. Metrics Tools. Crosstalk, the Journal of Defense Software Engineering.

k7	propietario	de comandos		Código fuente	
JDepend	BSD	Línea de comandos	Java	Byte code	Texto plano xml
JavaNCSS	GNU GPL	Línea de comandos	Java	Código fuente	Texto plano xml
McCabeIQ	Software propietario	GUI	Ada, ASM86, C, C#,C++, COBOL,RUBY, JSP, PERL, PL1,VB,VB.NET	Código fuente	Texto plano gráficos
Simian	Software propietario	Línea de comandos	Java, C#, C, C++, COBOL, RUBY, JSP, ASP	Código fuente	Texto plano
CodeSonar	Software propietario	Línea de comandos	C/C++ y ADA	Código fuente	HTML
CKJM	Open Source	Línea de comandos	Java	Byte code	Texto plano

Fuente:Rodríguez Monje Moisés. XI cursos de verano de Santander. Calidad de procesos y productos software. Calidad del producto software. ISO/IEC 25000. Julio 16 2010. <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/santander/mrodriguez-iso25000-update.pdf>

Estas herramientas son utilizadas para medir productos de tipo software, pero se quedan cortas si lo que se quiere es evaluar la calidad general. Como se observa en la tabla 6: solo evalúan el código fuente o Byte Code, es decir solo evalúan una parte del software, algunos de estos son utilizados por otros proyectos macros, para evaluar los aspectos que se necesitan.

2.2.2.1 Comparación de herramientas para la evaluación de la calidad de productos software.

Como mencionamos anteriormente existen una cantidad de herramientas que pretenden evaluar la calidad del software, pero en la mayoría de ellas existen

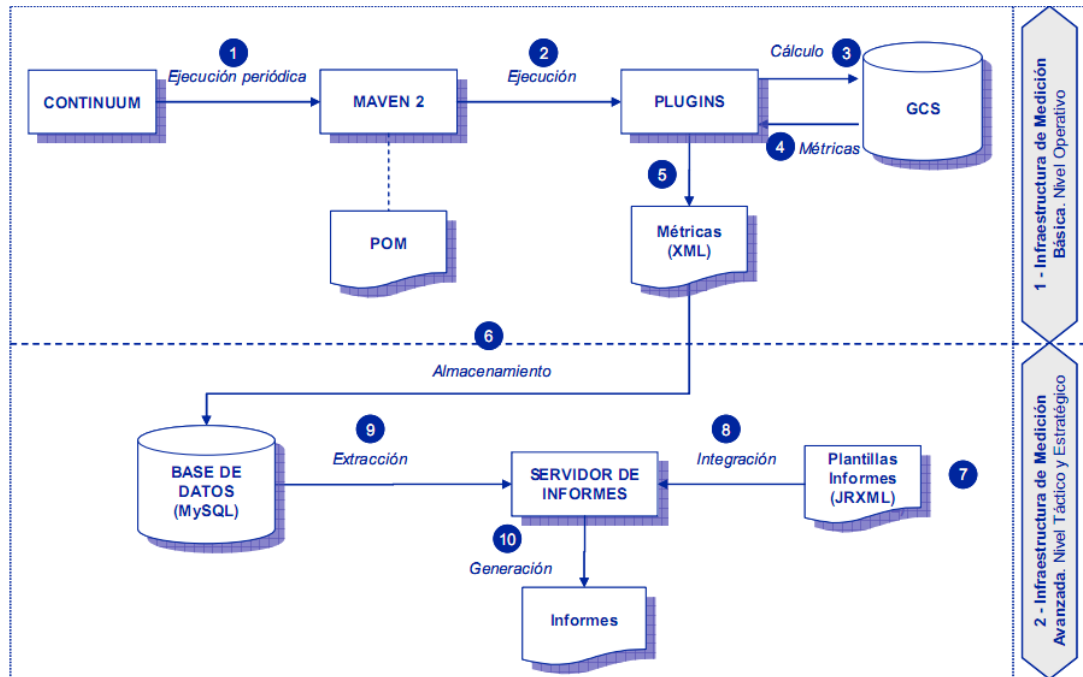
limitaciones ya sea de cubrimiento de la calidad total, pues están destinadas a evaluar cierto tipo de productos o están creadas para evaluar software de grandes dimensiones. En esta sección se hará una breve comparación con herramientas existentes para la evaluación de la calidad del software, estas herramientas en comparación con las de la Tabla 6 evalúan la calidad de una forma más amplia y se asemejan a la herramienta que se desarrollo en el presente proyecto.

KEMIS: “KybeleEnvironmentMesaurementInformationSystem” es un entorno desarrollado por KybeleConsulting que proporciona, por un lado, un conjunto predefinido de aplicaciones de software libre, junto con su configuración e instalación, que permiten implantar un sistema de medición de la calidad software a nivel operativo, táctico y estratégico , y por otro, un soporte metodológico basado en PSM “Practical Software and SystemMeasurement” para la evaluación de la calidad del producto software.

KEMIS se basa en el estándar ISO/IEC 9126, y utiliza herramientas de software libre para la obtención de métricas del producto entre estas herramientas se destacan, JavaNCSS, PMD, CheckStyle, Findbugs y JDepend que, junto con herramientas de construcción de software como Maven y Ant, y de integración continua como Cruise Control o Continuum, permiten automatizar la obtención de métricas, personalizar los resultados, y definir la periodicidad con la que realizan las mediciones.

KEMIS está orientado más a ofrecer indicadores de calidad pues asume que en esta parte es donde está más relacionado con los costos.

Figura 16: Arquitectura de KEMIS



Fuente: <http://www.kybeleconsulting.com/index.php/kemis-un-entorno-para-la-medicion-de-la-calidad-del-producto-software.html>

En esta arquitectura se destaca la división de la imagen en dos mitades separadas por una línea discontinua. Esta separación corresponde a las dos fases en las que se realiza la implantación del entorno:

- Infraestructura de medición básica (nivel operativo): La primera fase corresponde a la instalación y configuración de Maven 2, así como de los plugins de medición. Al terminar esta primera fase, el usuario dispone de un entorno que le permite obtener métricas de calidad de manera periódica y automática.
- Infraestructura de medición avanzada (niveles táctico y estratégico): La segunda fase corresponde a la instalación y configuración del entorno para la generación de informes. Al terminar esta segunda fase, el usuario dispone del entorno de medición KEMIS completo, lo que le permite, además de realizar las actividades de la primera fase, obtener

informes personalizados con los principales indicadores de calidad del producto²⁷ ;

MOSCA: “*Modelo Sistémico de Calidad*” propuesto por Mendoza. **MOSCA** s un modelo para la evaluación de Software educativo con un enfoque sistémico, cuyo propósito es evaluar la calidad sistémica dentro de una organización desarrolladora de Software de este tipo²⁸.

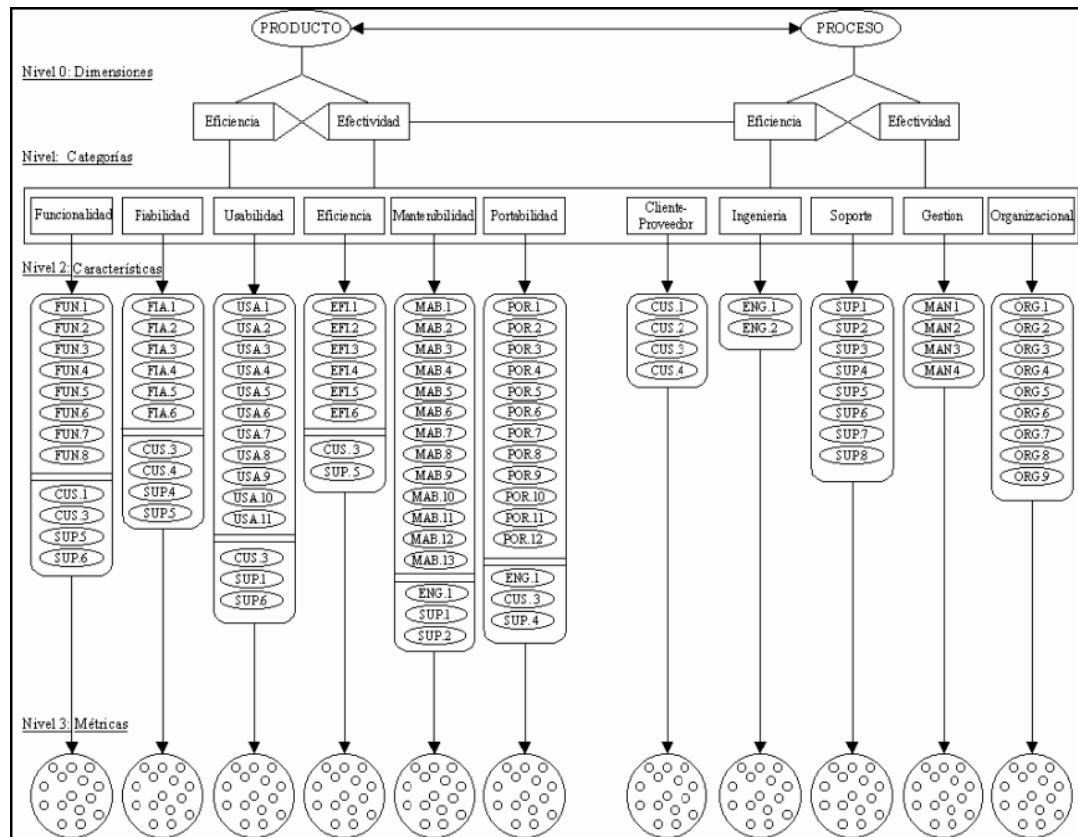
MOSCA al igual que KEMIS está basada en las 6 características que plantea la norma 9126, En la Figura 17 se muestra el diagrama del Modelo Sistémico de Calidad (MOSCA). En esta Figura se observa como el modelo MOSCA está constituido por dos sub modelos (el sub modelo del Producto y sub modelo del Proceso).

De acuerdo con los objetivos previsto en evaluación, del modelo MOSCA se puede tomar sub modelo del Producto, el sub modelo del Proceso, ambos inclusive. El primero de ellos, se utiliza para evaluar software ya elaborado, mientras que segundo se emplea cuando además, se requiere evaluar el proceso de su desarrollo.

²⁷ Rodríguez Moises, Garzás Javier, Piattin Mario, Kemis: Un entorno para la medición de la calidad del producto software. Disponible en: www.kybeleconsulting.com

²⁸ Mendoza Luis E. Pérez María. Grimán Anna C. Roja Teresita. Algoritmo para la Evaluación de la Calidad Sistémica del Software. Laboratorio de Investigación en Sistemas de Información (LISI) Departamento de Procesos y Sistemas, Universidad Simón Bolívar Caracas – Venezuela. Disponible en: http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_21.pdf.

Figura 17: Arquitectura de MOSCA



Fuente: [Mendoza et al., 2001].

A continuación la Tabla 7 Nos muestra de forma más detallada algunas características de estas herramientas de evaluación mencionadas.

Tabla 7: Comparación de herramientas para la evaluación

Producto	Modelo utilizado	Características
KEMIS	ISO/IEC 9126	<ul style="list-style-type: none"> • Un conjunto predefinido de aplicaciones de software libre. • Una configuración personalizada de dichas aplicaciones. • Un sistema de medición de la calidad software a nivel operativo, táctico y estratégico. • Un soporte metodológico basado en

	<p>psm para la evaluación de la calidad del producto software.</p> <ul style="list-style-type: none"> • Un cuadro de mando integrado que permite la visualización de los resultados obtenidos.
<p>MOSCA ISO/IEC 9126</p>	<ul style="list-style-type: none"> • Para evaluar la dimensión producto, se propone instanciar las categorías: funcionalidad, fiabilidad y utilidad, cada una de las cuales poseen características que permiten evaluar su eficiencia y efectividad. • Está dividido en dos sub-modelos (el sub-modelo del Producto y el sub-modelo del Proceso), y estos a su vez se dividen e 4 niveles (como se observa en la figura 17).

PARTE II. RESULTADOS Y DISCUSIÓN

3 RESULTADOS DEL PROYECTO DE INVESTIGACIÓN

El proyecto de investigación que se presenta dio como resultados finales una herramienta software la cual está en la capacidad de evaluar la calidad de productos del mismo tipo soportada en normas internacionales, conjuntamente con su respectivo manual de usuario, y un artículo el cual se describirá en una sección posterior.

3.1 Desarrollo de la herramienta propuesta

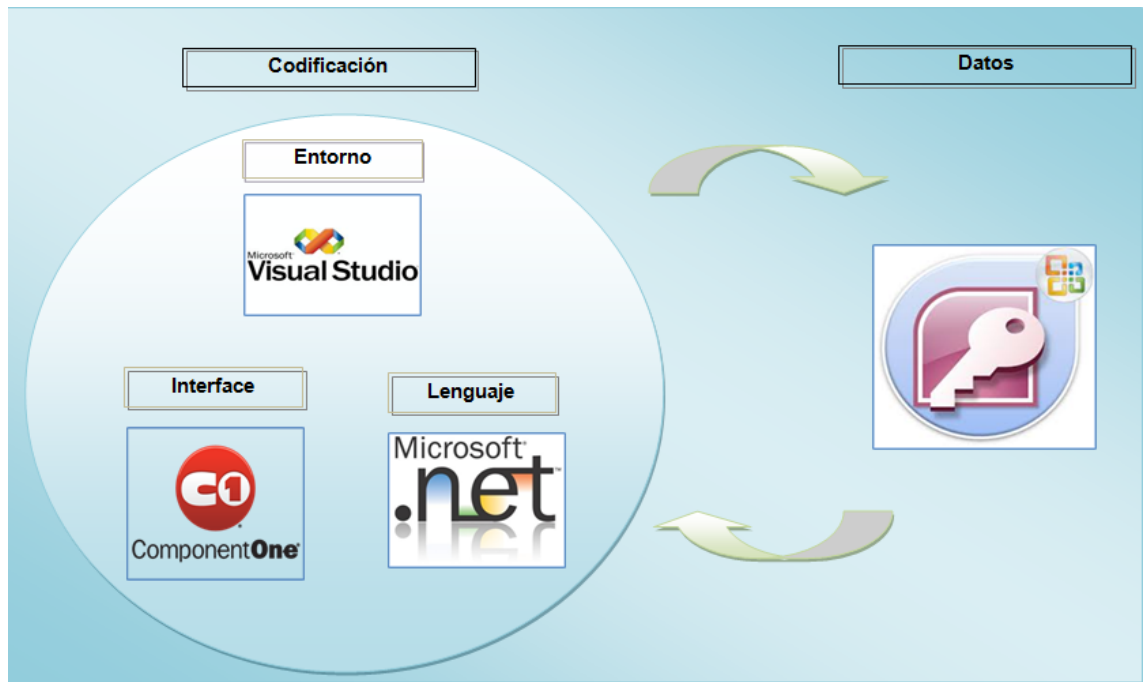
En esta sección se hará una descripción de cómo se generó la implementación de esta herramienta, y todo el entorno que apoyó el desarrollo.

3.1.1 Herramientas de desarrollo

Al momento de iniciar un proyecto software es de vital importancia saber elegir las herramientas de desarrollo que se van a utilizar, y cuáles son las que mejor rendimiento y adaptación tienen al tipo de proyecto que se desea desarrollar.

Para el desarrollo de la presente herramienta se utilizaron los siguientes componentes de desarrollo.

Figura 18: Herramientas de desarrollo.



Fuente: Autor.

📌 Entorno desarrollo

Como entorno de desarrollo se utilizó Microsoft visual estudio 2008, y el framework 3.5, por su interfaz gráfica y su amplio manejo de lenguajes de programación, además de esto en la universidad cuenta con el acuerdo de Campus Agreement y por lo cual no se necesita adquirir una licencia comercial.

📌 Lenguaje de programación

El lenguaje de programación utilizado fue Visual Basic .net, se escogió este lenguaje debido a que es un lenguaje de fácil manejo, y se contaba con una persona con una vasta experiencia en el manejo de estas herramientas.

📌 Componentes de diseño de interfaz

Para el diseño de interfaz como de reportes, diseño de tablas y toda la parte gráfica se utilizaron unos componentes para visual studio,

llamados ComponentOne, estos dan una apariencia más agradable que se familiariza más con el usuario.

Motor de base de datos

Como la herramienta requería una base de datos fue necesario añadir un motor de Base de datos. Para este caso y como se necesitaba una base de datos ágil y que fuera flexible se eligió Microsoft Acces.

3.1.2 Metodología de desarrollo

Más que una metodología de desarrollo, en la elaboración de la herramienta desarrollada se utilizó un modelo para el desarrollo software denominado el prototipado evolutivo.

El prototipado evolutivo desarrolla el concepto de sistema (producto tecnológico) a medida que avanza el proyecto. En este modelo se hace una implantación del sistema inicial, se expone a los comentarios del usuario, y se refina en N versiones hasta que se desarrolle el sistema adecuado tal como se describe en la figura 19. Esta metodología de desarrollo tiene como objetivo entender los requisitos del usuario y trabajar para mejorar la calidad de los requisitos, se comienza por definir los requisitos que no están claros para el usuario y se utiliza el prototipo para experimentar con ellos.

Figura 19: Modelo evolutivo de desarrollo de software.



Fuente: http://pinilla73.blogspot.com/2011_04_01_archive.html

Las razones fundamentales por las cuales se plantea esta metodología para el desarrollo del software son las siguientes:

- Es importante desarrollar los aspectos más visibles del sistema, para poder incorporar sugerencias de cambio por el usuario, en etapas tempranas del desarrollo.
- Es necesario saber si se han interpretado correctamente las especificaciones y necesidades del usuario.
- En muchos casos los usuarios no tienen una idea clara de lo que desean, por lo tanto se deben tomar decisiones y suponer que es lo que el usuario quiere. Por lo tanto la construcción de los prototipos brinda la posibilidad de hacer refinamientos en los requisitos en forma sucesiva a fin de acercarse al producto deseado.

- La decisión se fundamenta en la ventaja de la realización de los cambios en etapas tempranas y la posibilidad de emisión de varios prototipos evaluables durante el desarrollo, obteniéndose de este modo una metodología integral también para el proceso de evaluación del programa.
- Esta metodología favorece un intercambio de conocimientos y de autocrítica a la herramienta, lo que conlleva a que se produzcan muchas pruebas antes de liberar un nuevo prototipo así como mejoras rápidas a problemas que puedan surgir durante su uso.
- Se presentara parte del sistema al cliente y se continuara con el desarrollo del prototipo basándonos en la retroalimentación obtenida.
- Se llegara a un punto en el cual el usuario decidirá que el prototipo cumple con las expectativas deseadas, se completaran los trabajos pendientes y el prototipo se convertirá en el producto final.

El procedimiento a seguir para la metodología planteada es el siguiente:

- La construcción de prototipos comienza con la recolección de los requisitos.
- El desarrollador y usuario se reúnen y definen los objetivos globales para la aplicación, identifican todos los requisitos conocidos y perfilan las áreas en donde será necesaria una mayor definición.
- Luego se produce el diseño del prototipo que se enfoca sobre la representación de los aspectos del software más visibles al usuario (por

ejemplo, métodos de entrada y formatos de salida) y se continúa con su desarrollo.

- El prototipo es evaluado por el usuario y se utiliza para refinar los requisitos del software a desarrollar.

Se produce un proceso interactivo en el que el prototipo es “afinado” (Refinamiento del prototipo) para que satisfaga las necesidades del usuario, al mismo tiempo que facilita al desarrollador una mejor comprensión de lo que hay que hacer y poder entregar el producto final requerido o producto de ingeniería²⁹.

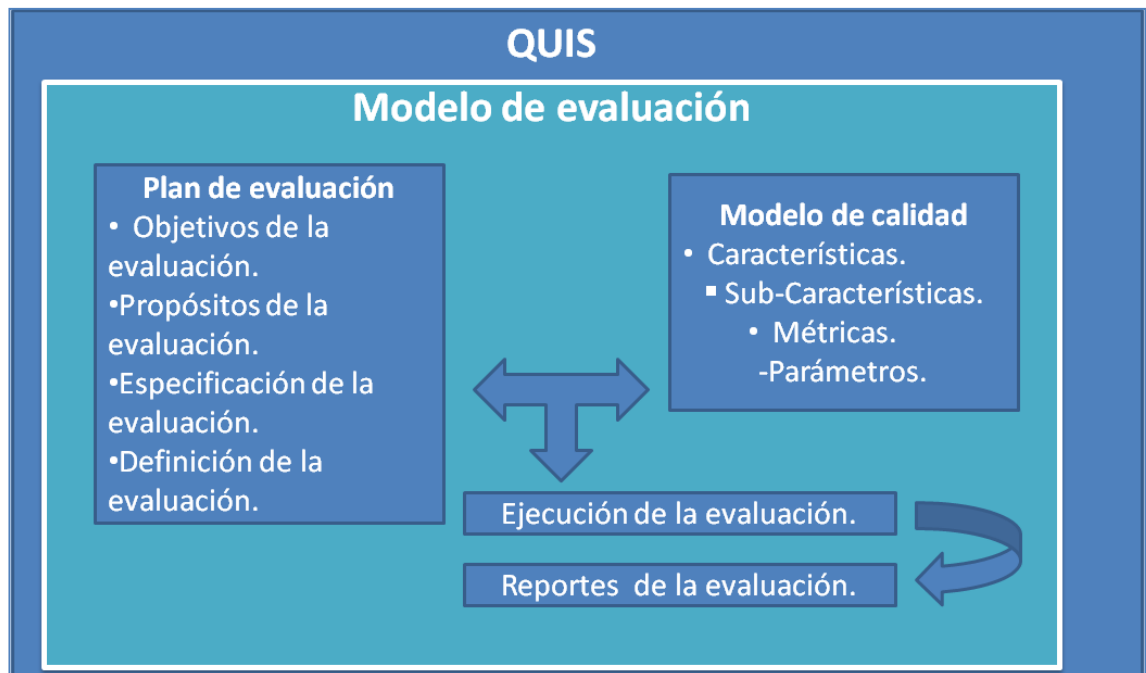
3.1.3 Implementación de la herramienta

Definidas las herramientas de desarrollo y la metodología se procedió a la implementación de la herramienta que se denominó QUIS “QualityUIS”, por ser un software para la evaluación de la calidad y fue desarrollado en la Universidad Industrial de Santander (UIS).

La estructura de QUIS se muestra en la figura 20.

²⁹ CICLOS DE VIDA DE UN SOFTWARE "MODELO EVOLUTIVO". Disponible en: http://pinilla73.blogspot.com/2011_04_01_archive.html

Figura 20: Estructura general de QUIS.

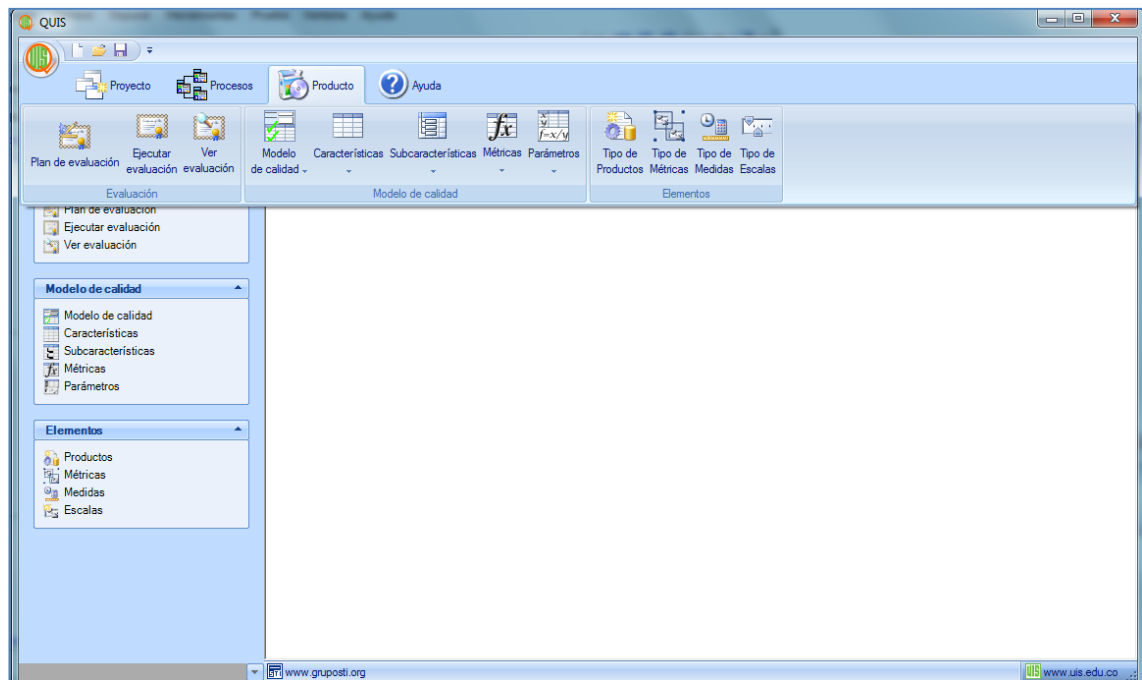


Fuente: Autor.

Como se ve en la figura 20 la estructura de la herramienta consiste en: un marco general denominado modelo de evaluación, el modelo se basó en la norma ya mencionada 14598 de la ISO, la cual dice que se debe llevar un plan y un modelo de evaluación, en este caso QUIS tomo como modelo de calidad el proporcionado por la mencionada norma 9126 de la ISO. Para el plan de evaluación se realizó siguiendo el esquema propuesto por la ISO 14598. Una vez reunidos los parámetros de entrada se ejecuta la evaluación y posterior a esto se procede a la generación de reportes.

En la figura 21 veremos el pantallazo inicial para la sección de producto.

Figura 21: Pantallazo inicial para la parte de producto QUIS.

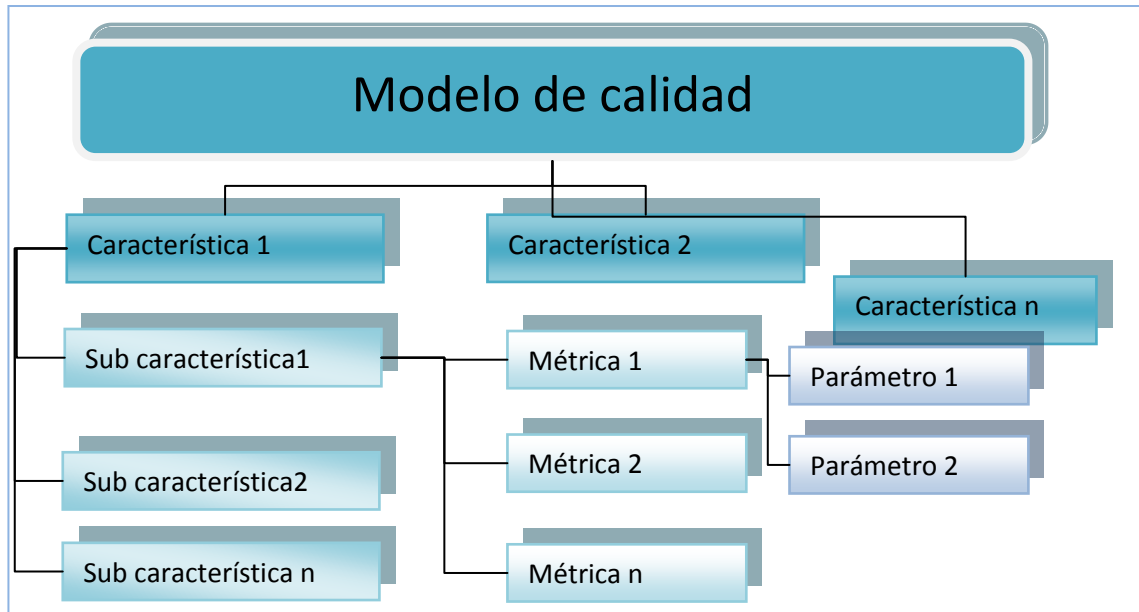


Fuente: Autor.

3.1.3.1 Modelo de calidad

En esta sección se hace la definición del modelo de calidad que se desea agregar a la herramienta. Para que el modelo sea aceptado por esta, debe tener ciertas condiciones ver figura 22; la figura muestra la jerarquía que debe tener un modelo de calidad según la norma ISO 9126.

Figura 22: Esquema general de un modelo de calidad.






Fuente: Autor.



Esta parte se implemento en la herramienta de la siguiente manera:

Modelo de calidad

En esta parte se puede gestionar toda la parte relacionado con el modelo de calidad y todo lo que comprende ver 23; la tabla 8 muestra de manera más ordenada lo mencionado.

Tabla 8: Descripción Modelo de calidad

NOMBRE	DESCRIPCIÓN	FIGURA
Modelo de calidad	Esta parte tiene dos opciones, nuevo y editar, en nuevo se crea el nuevo modelo con los respectivos datos asociados, y editar se despliega el formulario donde se puede realizar cambios al modelo seleccionado o eliminarlo.	 Modelo de calidad ▾
Características	Al seleccionar este ítem se despliegan dos opciones al igual que modelo de calidad, en esta parte se gestiona todo lo relacionado con características.	 Características
Sub características	En esta opción se hace gestión de lo referente a las sub características de calidad, y sea nueva editar alguna o eliminar.	




Subcaracterísticas		
Métricas	Esta parte es una de las más importantes ya que en esta se gestiona el corazón del modelo de calidad que son las métricas, descripción, fórmula, tipo de medida entre otros más.	 Métricas
Parámetros	Esa parte se encarga de la gestión de los parámetros que son los que al final se seleccionan para la medición.	 Parámetros

Fuente: Autor.

3.1.3.2 Evaluación

Esta parte es la encargada de la gestión de la evaluación, esto incluye todos los parámetros que menciona la norma 14598, se puede ejecutar la evaluación y generar reportes de esta. La tabla 9 describe de manera más detallada cada ítem de esta sección.

Tabla 9: Descripción de la parte de evaluación





NOMBRE	DESCRIPCIÓN	FIGURA
Plan de evaluación	En esta sección muestra el formulario con los requisitos exigidos para realizar una evaluación, al aceptar se carga la información en la base de datos para luego poder ejecutarla.	 Plan de evaluación
Ejecutar la evaluación	En esta sección al dar clic se carga en un formulario la evaluación que se cargo anteriormente, en la tabla del formulario se debe llenar con los datos que se seleccionaron con los correspondientes valores medidos.	 Ejecutar evaluación
Ver la evaluación	En este ítem se puede ver detalladamente cada evaluación que se ha hecho, igualmente se puede generar un reporte de todas las evaluaciones que se han hecho a determinado software.	 Ver evaluación

Fuente: Autor.

Elementos

Esta parte gestiona elementos que son comunes a varios ítems del modelo, tales como tipo de medida, tipo de producto entre otros ver tabla 10.

Tabla 10: Otros elementos

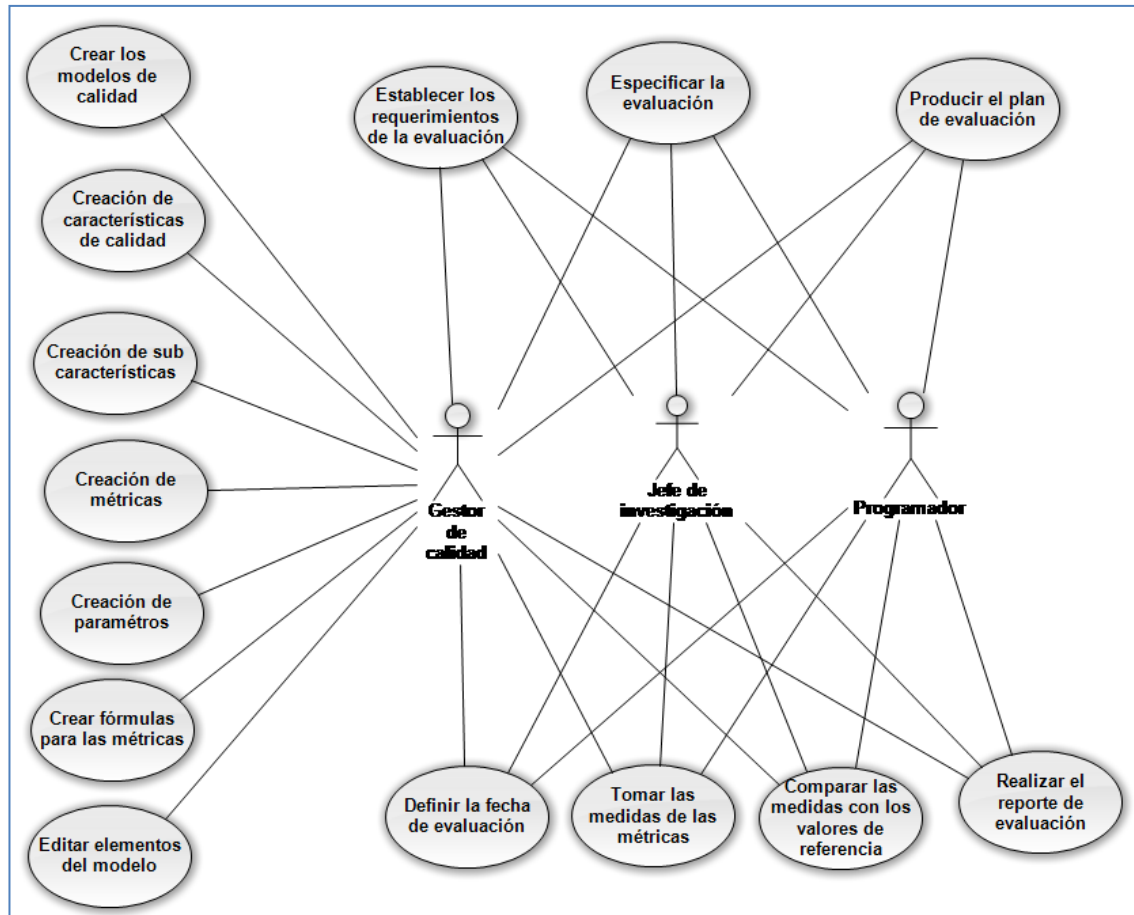
NOMBRE	DESCRIPCIÓN	FIGURA
Tipos de productos	En esta sección se gestionan los tipos de productos que puedan haber estos se guardan en una base de datos.	 Tipo de Productos
Tipos de métricas	En esta sección se gestiona el tipo de métricas las cuales se relacionan con las características para describir de qué tipo son.	 Tipo de Métricas
Tipos de medidas	En esta sección se refiere a los tipos de medidas que puede identificar un parámetro.	 Tipo de Medidas
Tipo de escala	En esta sección se gestiona el tipo de escala relacionada con las métricas.	 Tipo de Escala

3.2 Casos de uso

Los casos de uso que corresponde al presente proyecto se detallan en la figura 24, allí se pueden ver detalladamente cómo se relacionan los usuarios con las actividades que se generen, para el buen uso de la herramienta.

Estos casos de uso se basaron en la ya mencionada tesis de maestría “*propuesta de un modelo para la evaluación de calidad de software derivados de actividades de investigación*”.

Figura 23: Diagrama de casos de uso.



Fuente: Autor.

3.3 Diagrama de clases

Las clases que componen la herramienta implementada se muestran en la figura 25, estas clases se explicaran más detalladamente en la parte inferior de la grafica.

Figura 24: Diagrama de clases.



Fuente: Autor.

3.4 Descripción de las clases

La tabla 11 hace una descripción un poco más amplia de los componentes del proyecto (clases y formularios), y se hace una relación con las actividades de los casos de uso.

Tabla 11: Descripción de clases del sistema

Nombre	Descripción	Actividad
ClaEvaion	Esta clase es la encargada de almacenar los atributos de medición seleccionados, en esta clase también se hace el ensamblado de la formula con los parámetros escogidos y se evalúan devolviendo el valor de cada métrica.	Evaluar el producto Sub actividad <ul style="list-style-type: none"> ➤ Almacenar atributos seleccionados. ➤ Realizar la evaluación de los parámetros seleccionados en la formula de la métrica.
ClaProcto	Esta clase es la principal e inicia al ejecutar el programa es la encargada de gestionar los directorios utilizados y el enlace con la base de datos, el cual es gestionado por una función dentro de esta clase.	Está asociada con todas las actividades de la herramienta.
ClaCarcas	Esta Clase es la encargada de gestionar todo lo relacionado con las	Gestionar el modelo de calidad.

	características del modelo de calidad, a la hora de crear una nueva.	Sub actividad ➤ Crear sub características.
ClaCarEvaion	Esta clase se encarga de gestionar los criterios que se tienen en cuenta para realizar la evaluación relacionada con la norma 14598.	Evaluar el producto Sub actividad ➤ Producir el plan de evaluación.
ClaCarSubCarcas	Esta clase es la encargada de gestionar la relación entre Características y sub características.	Gestionar el modelo de calidad. Sub actividad ➤ Crear relaciones entre características y sub características.

	<p>Esta clase es una de las más importantes ya que en ella se crean todos los atributos de una métrica, entre ellos la fórmula allí se realiza una validación de la misma para comprobar que su estructura este correcta.</p>	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Crear métricas.
ClaMetEvaion	<p>Esta clase está encargada de almacenar los criterios de medición seleccionados, a la hora de realizar la gestión de la evaluación.</p>	<p>Definir la evaluación.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Establecer los requerimientos de la evaluación.
ClaMetParros	<p>Esta Clase se encarga de crear las relaciones entre métricas y los parámetros asociados a estas.</p>	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p>

		<ul style="list-style-type: none"> ➤ Crear relaciones entre métricas y parámetros.
ClaModCaldad	Esta clase es la encargada de almacenar los parámetros que componen los modelos de calidad.	Gestionar el modelo de calidad. Sub actividades. <ul style="list-style-type: none"> ➤ Crear el modelo de calidad.
ClaModCarcas	Esta clase es la encargada de enlazar el modelo de calidad con las características que se encuentran dentro de la base de datos.	Gestionar el modelo de calidad. Sub actividades. <ul style="list-style-type: none"> ➤ Asociar el modelo de calidad con las características.
ClaObjEvaion	Esta clase se encarga de guardar los objetivos de la evaluación.	Definir la evaluación. Sub actividades <ul style="list-style-type: none"> ➤ Establecer los requerimientos de la evaluación.
ClaParEvaion	Esta clase se encarga de guardar los parámetros generales del modelo que son escogidos para la evaluación.	Definir la evaluación. Sub actividades <ul style="list-style-type: none"> ➤ Establecer los

		requerimientos de la evaluación.
ClaParros	Esta clase se encarga de guardar los criterios que debe tener un parámetro del modelo de calidad.	Especificar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Crear los parámetros de las métricas.
ClaPernal	Esta clase es la encargada de guardar los datos requeridos del personal involucrado en la evaluación.	Definir la evaluación. Sub actividad <ul style="list-style-type: none"> ➤ Especificar la evaluación.
CalPerProcto	Esta clase es la encargada de guardar los datos requeridos del personal involucrado en la evaluación.	Definir la evaluación. Sub actividad <ul style="list-style-type: none"> ➤ Especificar la evaluación.
ClaResEvaion	Esta clase se encarga de guardar parámetros de la de la evaluación, tales como la fecha en que se realizada cada evaluación, tipo de producto a ser	Evaluar el producto. Sub actividad. <ul style="list-style-type: none"> ➤ Definir fecha y

	evaluado entre otros.	otros aspectos.
ClaResEval	Esta clase guarda los datos referentes a los responsables de realizar la evaluación.	Definir la evaluación. Sub actividad <ul style="list-style-type: none"> ➤ Establecer los requerimientos de la evaluación.
ClaResMetrica	Esta clase recibe los valores de las métricas una vez sean tomados, conjuntamente con las características asociadas.	Evaluar el producto. Sub actividad <ul style="list-style-type: none"> ➤ Tomar las medidas de las métricas.
ClaSubCarcas	Esta clase almacena los datos pertinentes a las sub características.	Gestionar el modelo de calidad. Sub actividad. <ul style="list-style-type: none"> ➤ Guardar sub características.

ClaSubMetcas	Esta clase se encarga de gestionar la relación entre sub características y métricas	Gestionar el modelo de calidad. Sub actividad. <ul style="list-style-type: none"> ➤ Relacionar sub características y métricas.
ClaValParEvaion	Esta clase se encarga de guardar los valores individuales de cada parámetro conjuntamente con su descripción.	Evaluar el producto. Sub actividad <ul style="list-style-type: none"> ➤ Tomar medidas.

Descripción de los formularios

Tabla 12: Descripción de formularios.

ForAcerca	Formulario de inicio, presentación del software.	
ForCreCarcas	Este formulario se encarga de recoger la	Gestionar el modelo

	<p>información a la hora de crear una característica, está compuesto de:</p> <ul style="list-style-type: none"> • Una caja de texto, en la cual se digita el nombre. • Un combo box, el cual contiene la lista de los tipos de métricas a las cuales puede pertenecer una característica. • Una caja de texto, en la cual se escribe una descripción de la característica a crear. • Una grilla, que contiene todas las sub características las cuales se pueden asociar a esta nueva característica. 	<p>de calidad.</p> <p>Sub actividad.</p> <ul style="list-style-type: none"> ➤ Crear características.
ForCreMetcas	<p>Formulario el cual recoge los datos para crear una nueva métrica este está compuesto por dos pestañas estas son:</p> <ul style="list-style-type: none"> ✚ Los datos para las métricas. <ul style="list-style-type: none"> • Nombre • Nombre corto • Propósito de la métrica. • Método de aplicación. • Escala de medición. • Origen de los datos. • Valor optimo. • Valor mínimo. • Valor máximo. ✚ Creación de la fórmula <ul style="list-style-type: none"> • Grilla de parámetros. • Caja para la fórmula. • Elementos para agregar a la fórmula, números, signos, entre otros. 	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad.</p> <ul style="list-style-type: none"> ➤ Crear las características.
ForCreParos	Este formulario recoge la información pertinente a los parámetros del modelo	Gestionar el modelo de calidad.

	<p>de calidad, este se compone de:</p> <ul style="list-style-type: none"> • Caja de texto, donde va el nombre del parámetro. • Caja de texto, para el nombre cortó. • Caja de texto, para el tipo de medida del parámetro. 	<p>Sub actividad.</p> <ul style="list-style-type: none"> ➤ Crear los parámetros.
ForCreSubCarcas	<p>En este formulario se encarga de recolectar la información de las sub características que se vayan a crear está compuesto por:</p> <ul style="list-style-type: none"> • Caja de texto, para el nombre. • Caja de texto, para la descripción. • Grilla, que almacena todas las métricas que están en la base de datos. 	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Crear sub características.
ForCreTipEscala	<p>Formulario que gestiona los tipos de escala de las métricas.</p>	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Crear elementos del modelo de calidad.
ForCreTipMedida	<p>Formulario que gestiona los tipos de medida de las métricas.</p>	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Crear elementos del modelo de calidad.
ForCreTipMetrica	<p>Formulario que gestiona los posibles tipos de métricas que puedan existir.</p>	<p>Gestionar el modelo de calidad.</p> <p>Sub actividad</p> <ul style="list-style-type: none"> ➤ Crear elementos del modelo de calidad.

ForCreTipProcto	Formulario que gestiona los posibles tipos de productos que puedan existir.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Crear elementos del modelo de calidad.
ForEdiCarcas	Este formulario recibe el nombre y el tipo de una característica la cual se desee editar, allí se carga toda la información de esta, la cual se puede editar o eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar características.
ForEdiMetcas	Este formulario recibe el nombre de una métrica la cual se desee editar, allí se carga toda la información de esta, la cual se puede editar o eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar métricas.
ForEdiModCaldad	Este formulario recibe el nombre de un modelo de calidad el cual se desee editar, allí se carga toda la información de este, la cual se puede editar o eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar el modelo de calidad.
ForEdiParros	Este formulario recibe el nombre de un parámetro el cual se desee editar, allí se carga toda la información de este, la cual se puede editar o eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar los parámetros de calidad.
ForEdiSubCarcas	Este formulario recibe el nombre de una Sub características la cual se desee editar, allí se carga toda la información de esta, la cual se puede editar o eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar sub características.
ForEdiTipMedida	En este formulario se realiza la edición de los tipos de medida existentes en la	Gestionar el modelo de calidad.

	base de datos, allí también está la opción de eliminar.	Sub actividad <ul style="list-style-type: none"> ➤ Editar elementos del modelo.
ForEdiTipMetrica	En este formulario se realiza la edición de los tipos de métricas existentes en la base de datos, allí también está la opción de eliminar si se desea.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar elementos del modelo.
ForEvaion	<p>En este formulario se realizan las evaluaciones, este consta de:</p> <ul style="list-style-type: none"> • Un combo box, el cual contiene los tipos de productos escogidos para ser evaluados. • Una grilla, una vez seleccionado el tipo de producto la grilla carga los elementos seleccionados que se van a evaluar a este producto, a esta grilla se le pueden agregar más columnas y es allí donde se colocan los valores a cada parámetro cargado. 	Evaluar el producto. Sub actividad. <ul style="list-style-type: none"> • Tomar las medidas de las métricas.
ForModCaldad	<p>En este formulario se reciben los datos para el ingreso de un nuevo modelo de calidad, este consta de:</p> <ul style="list-style-type: none"> • Un tex box, para el nombre del modelo. • Un tex box, para la descripción del modelo. • Una grilla, en la cual se cargan todas las características disponibles para asociar al modelo que se está creando. 	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Guardar el modelo de calidad.
ForModEvaion	Este formulario es el encargado de recoger los datos para que se pueda hacer efectiva la evaluación, estos datos tales como el plan de evaluación, criterios, encargados, tipo de producto,	Definir la evaluación Sub actividades. <ul style="list-style-type: none"> ➤ Establecer los requerimiento

	etc.	s de la evaluación. ➤ Especificar la evaluación. ➤ Producir el plan de evaluación.
ForPreion	Formulario de inicio al ejecutar el software.	
ForPripal	Formulario principal, es el que contiene todos el menú de la herramienta.	
ForProtos	Este formulario contiene una grilla con todos los posibles productos que se puedan evaluar, de allí se seleccionan y al dar aceptar se cargan en el formulario de carga de la evaluación.	Definir la evaluación. Sub actividad. ➤ Seleccionar los productos a evaluar.
ForReportes	Este formulario se encarga de mostrar los reportes que general las evaluaciones, estos se pueden guardar en formatos como pdf, Excel entre otros, o imprimir directamente.	Evaluación del producto. Sub actividad. ➤ Generar reportes.
ForSelCarica	Este formulario se encarga de seleccionar una característica y el tipo que pertenece a la cual se le desea realizar algún cambio o simplemente eliminar, acción que se realiza a través de dos combo box.	Gestionar el modelo de calidad. Sub actividad ➤ Editar características.
ForSelMetcas	Este formulario se encarga de seleccionar una métrica a la cual se le desea realizar algún cambio o simplemente eliminar, acción que se realiza a través de un combo box.	Gestionar el modelo de calidad. Sub actividad ➤ Editar métricas.
ForSelModCaldad	Este formulario se encarga de seleccionar un modelo de calidad al cual	Gestionar el modelo de calidad.

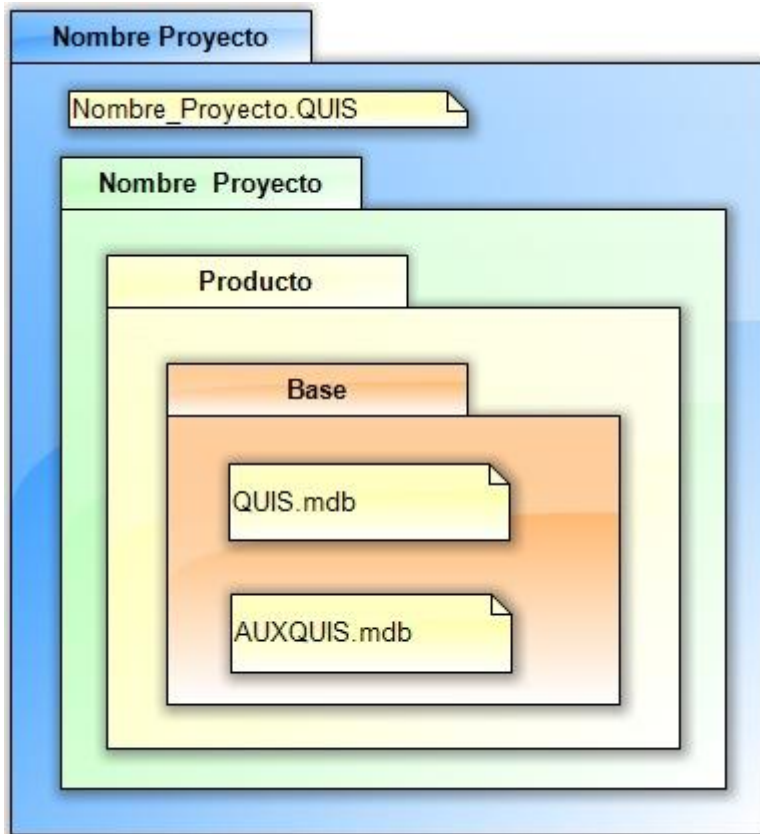
	se le desea realizar algún cambio o simplemente eliminar, acción que se realiza a través de un combo box.	Sub actividad <ul style="list-style-type: none"> ➤ Editar modelo de calidad.
ForSelParros	Este formulario se encarga de seleccionar un parámetro al cual se le desea realizar algún cambio o simplemente eliminar, acción que se realiza a través de un combo box.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar parámetros.
ForSelSubCarica	Este formulario se encarga de seleccionar una sub característica a la cual se le desea realizar algún cambio o simplemente eliminar, acción que se realiza a través de un combo box.	Gestionar el modelo de calidad. Sub actividad <ul style="list-style-type: none"> ➤ Editar sub características.
ForVerEvaion	<p>Este formulario se encarga de mostrar las evaluaciones que se han hecho, este consta de:</p> <ul style="list-style-type: none"> • Un combo box, en el cual se puede seleccionar que evaluación ver. • Una grilla, la cual se llena automáticamente al seleccionar una opción en el combo. • Botón ver reporte, al pulsar este botón se carga el reporte de todas las evaluaciones. 	Evaluar el producto Sub actividades. <ul style="list-style-type: none"> ➤ Realizar el reporte de evaluación.

Fuente: Autor.

3.5 Estructura de archivos

La estructura de archivos de la herramienta está diseñada como se muestra en la figura 26.

Figura 25: Estructura de archivos.



Fuente: Autor.

Esta estructura se compone de una carpeta principal que contiene todo el proyecto, este proyecto solo cuenta con dos archivos que son la base de datos que se denominó QUIS y una copia de esta llamada AUXQUIS que se creó como respaldo. Estas se encuentran dentro de una carpeta llamada Base, que a su vez está dentro de otra llamada Producto, así está distribuida la estructura de los archivos del proyecto QUIS.

3.6 Otros productos

El presente proyecto además de la herramienta mencionada se realizó un artículo con el título “*HERRAMIENTA COMPUTACIONAL PARA LA EVALUACIÓN DE CALIDAD DE PRODUCTOS SOFTWARE ENMARCADOS EN ACTIVIDADES DE INVESTIGACIÓN*”, este artículo se envió a la revista

Scientia et Technica, de la universidad tecnológica de Pereira (UTP)³⁰ para la edición 48, la cual aun se encuentra en evaluación; en este artículo hace una breve descripción de la herramienta desarrollada y la investigación que precedió este desarrollo.

Paralelamente a esto se está realizando el registro de software para QUIS, en el momento de la realización de este documento se encuentra en proceso.

³⁰ <http://www.utp.edu.co/revistaciencia/>

4 ILUSTRACIÓN DEL USO DE LA HERRAMIENTA

4.1 Ilustración

En esta sección se mostrara el resultado de una evaluación, realizada por la herramienta que se desarrollo en el presente proyecto (QUIS).

Se escogió como producto a evaluar un software desarrollado en el grupo de investigación en Ingeniería Biomédica (GIIB) denominado EOGEST 1.0, el cual está desarrollado totalmente en Matlab, su de datos está desarrollada en Mysql; este producto está enfocado en la caracterización y análisis estadístico de señales electrooculográficas³¹.

La figura 27 nos muestra la interfaz de EOGEST de la presentación inicial de este software.

Figura 26: Presentación EOGEST 1.0.



Fuente: Autor. Tomada de Software EOGEST 1.0 en ejecución.

Para la evaluación se asignaron los siguientes roles:

³¹ Señal que registra cambios en el potencial eléctrico entre la cornea y la retina

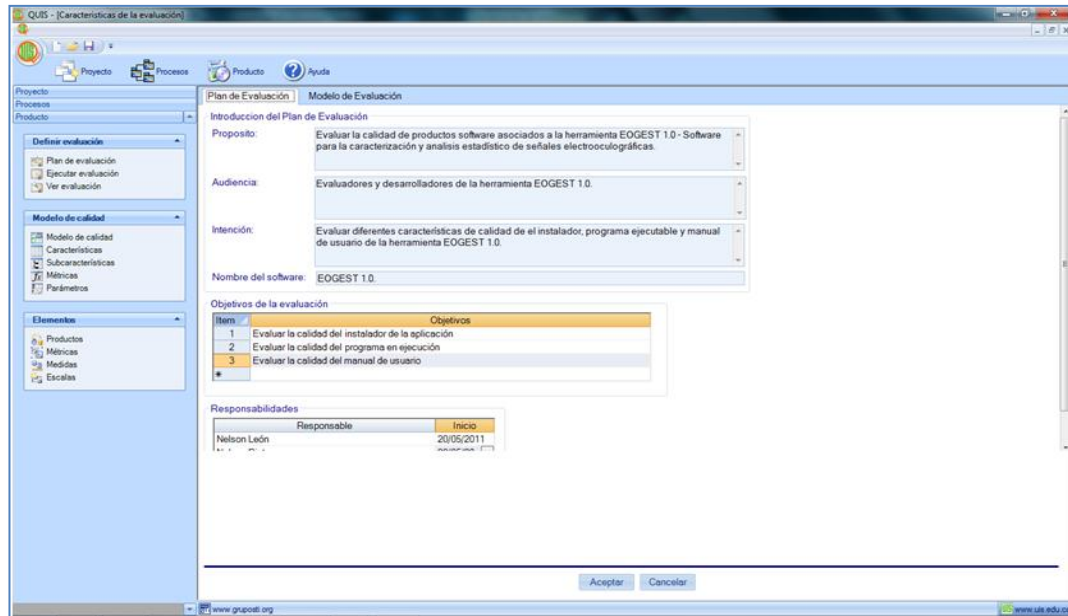
- ✚ **Programador:** Ingeniero Nelson Enrique León Martínez.
- ✚ **Jefe de investigación:** Ingeniero Nelson Enrique León Martínez.
- ✚ **Gestor de calidad:** Estudiante auxiliar Nelson Pinto Chacón.

4.1.1 Desarrollo de la evaluación de EOGEST 1.0 con QUIS.

Para iniciar la evaluación es necesario tener definidas las reglas de evaluación, el producto a evaluar y el modelo de evaluación que se va a utilizar. Como se comentó al inicio de este texto, QUIS proporciona un marco de evaluación basado en el modelo de calidad ISO 14598 y el modelo de calidad propuesto en la norma ISO 9126.

Teniendo definido el marco de la evaluación se procede a realizar la evaluación. Primero se define la evaluación, esto se hace en el menú producto, plan de evaluación, allí se cargara el formulario que se ve en la figura 30. En este formulario en la primera pestaña se recibió la información del plan de evaluación, como propósito de la evaluación, audiencia, intención, nombre del producto a evaluar, se definieron los objetivos de la evaluación y los responsables, en la segunda pestaña se hizo la definición de los elementos de evaluación que se utilizaron para evaluar a EOGEST 1.0 ver figura 28.

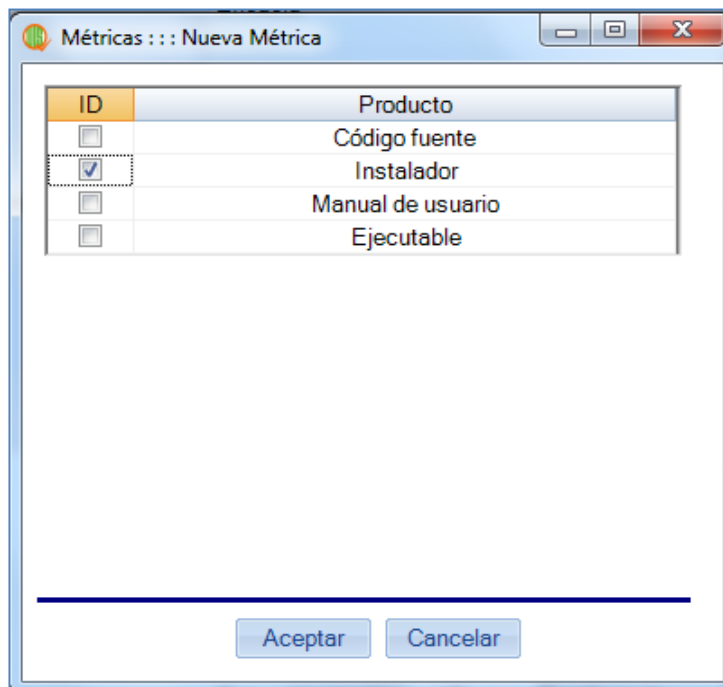
Figura 27: Definición de la evaluación.



Fuente: Autor.

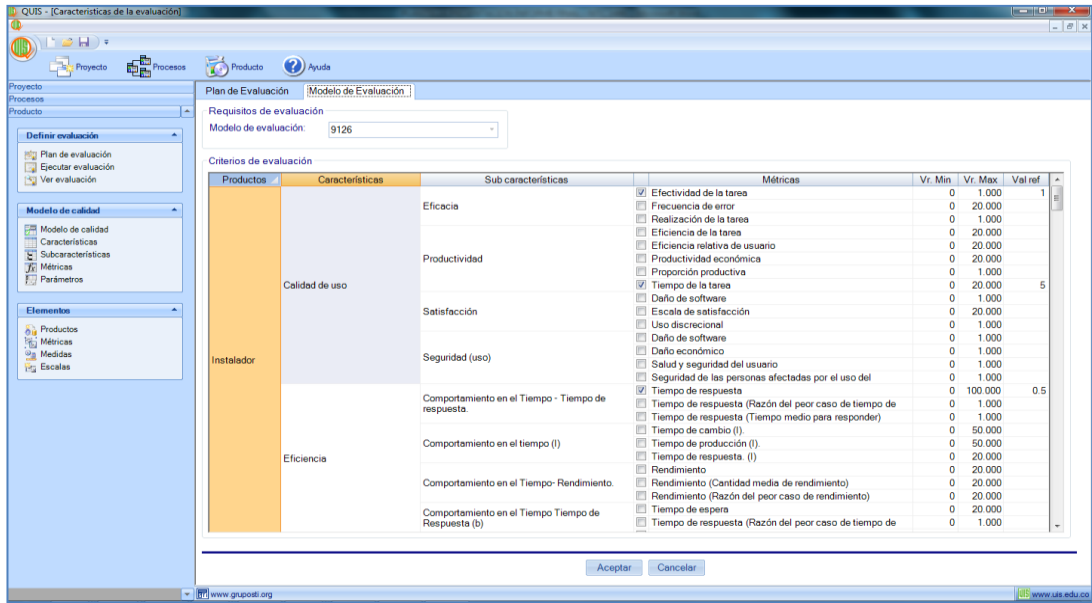
Ya definido el plan de evaluación se procedió a seleccionar los productos a evaluar, esto se hace como muestra la figura 29.

Figura 28: Selección de productos a evaluar.



Fuente: Autor.

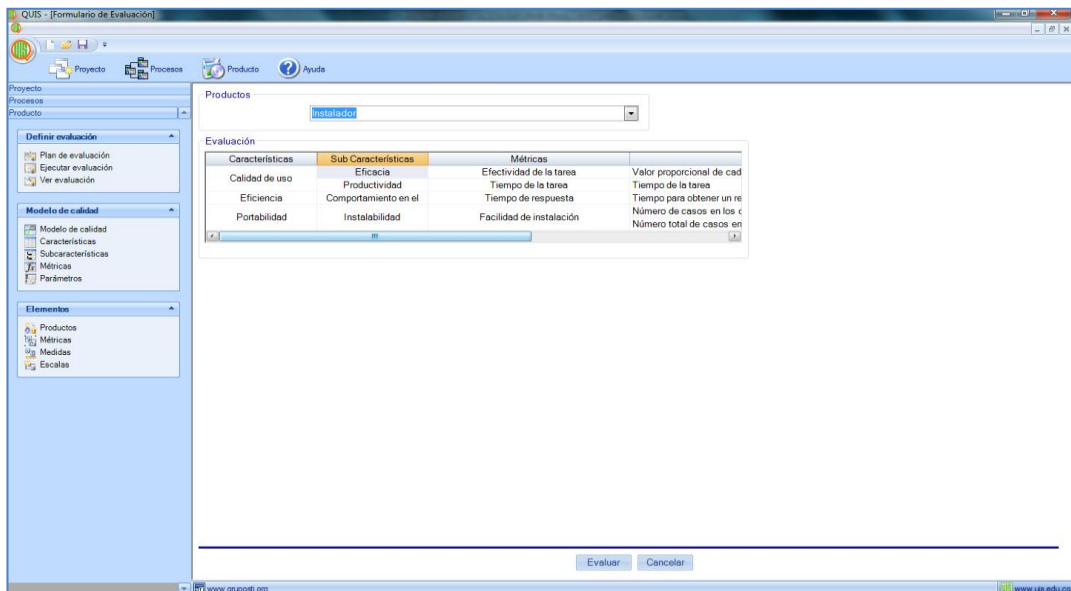
Figura 29: Definición de la evaluación.



Fuente: Autor.

Una vez definida la evaluación se guarda en la base de datos. Luego se ejecutó la evaluación de EOGEST 1.0, con los ítems seleccionados, ver figura 30.

Figura 30: Ejecutar la evaluación.



Fuente: Autor.

La figura 32 muestra los resultados de la evaluación realizada, en la tabla se ve que en la columna 6 existen filas de diferente color la explicación a esto es que los colores verdes indican que los valores cumplen con los rangos de valores propuestos, los otros no cumplieron y se deben corregir estos ítems para que el producto satisfaga las condiciones y criterios propuestos.

Figura 31: Resultados de la evaluación a EOGEST 1.0.

Evaluaciones
Seleccione la evaluación:

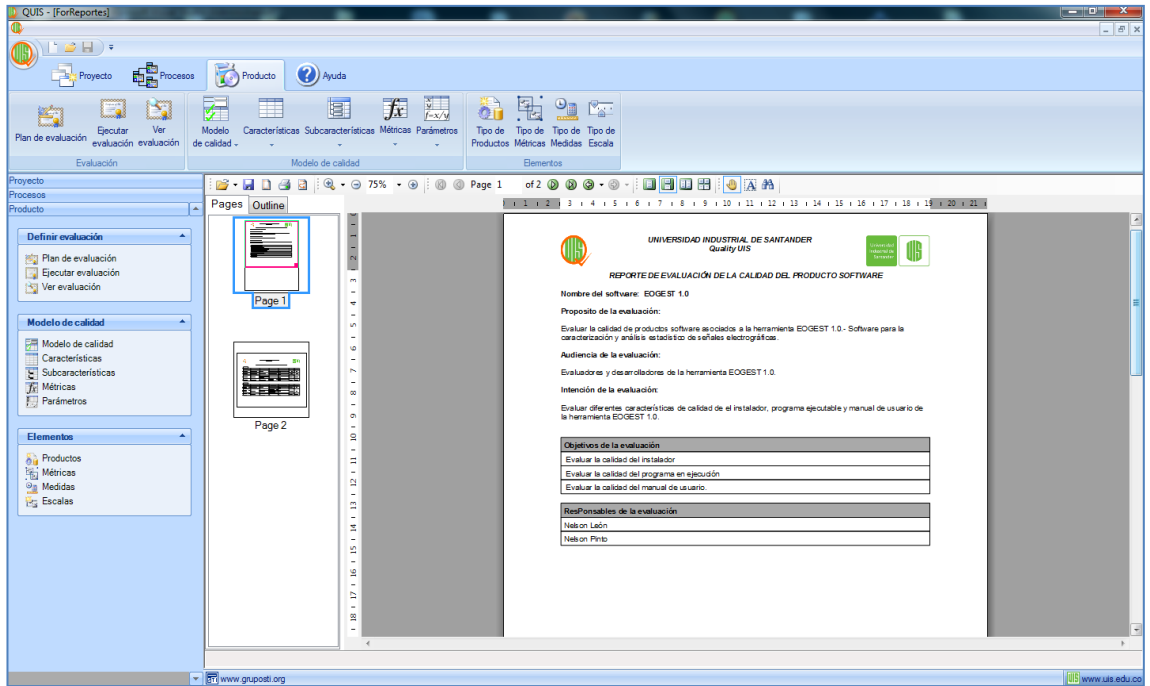
Evaluación
Fecha de la medida: 09/07/2011

Producto	Características	Sub características	Métricas	Permitido	Medido
Instalador	Calidad de uso	Eficacia	Efectividad de la tarea	1	1
		Productividad	Tiempo de la tarea	5	15
	Eficiencia	Comportamiento en el Ti	Tiempo de respuesta	0.5	0.5
		Instalabilidad	Facilidad de instalación	0.8	0.333
Calidad de uso	Eficacia	Efectividad de la tarea	0.9	1	
	Usabilidad	Capacidad para ser aprend	Complejidad de document	0.99	1
		Capacidad para ser opera	Claridad de mensaje (I)	0.9	0.926

Fuente: Autor.

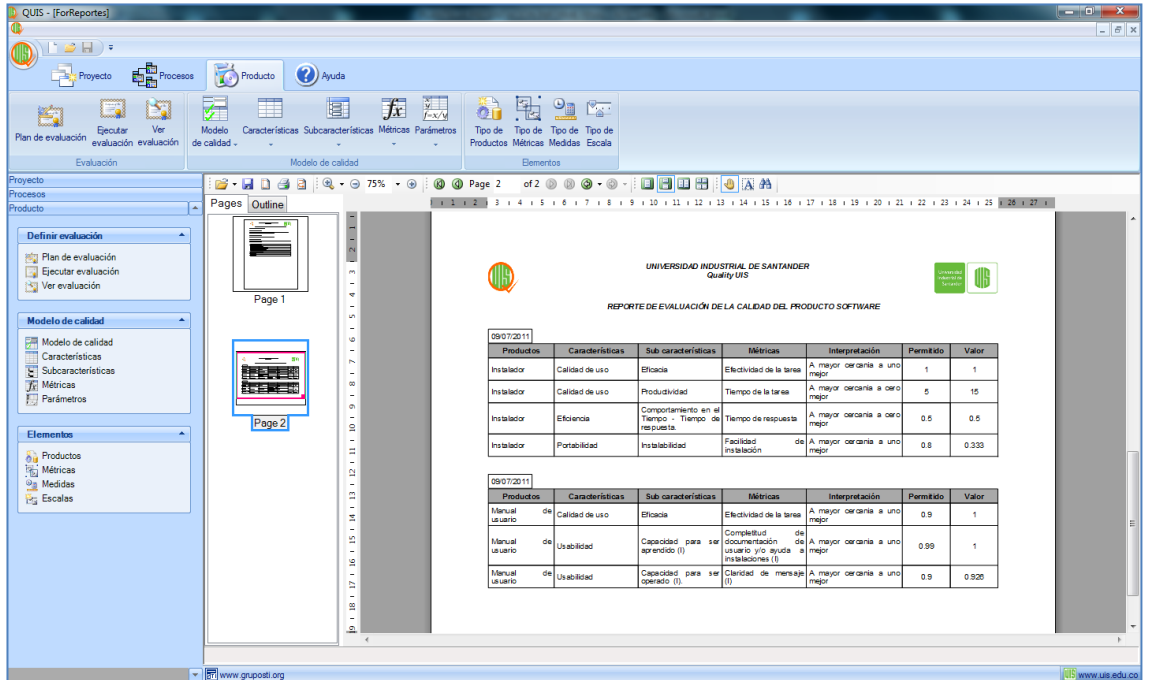
El botón ver reporte de la parte inferior genera un reporte de todas las evaluaciones realizadas; para EOGEST solo se realizaron dos, estas se categorizan por fechas ver figura 33 y 34.

Figura 32: Reporte de evaluación (1).



Fuente: Autor.

Figura 33: Reporte de evaluación (2).



Fuente: Autor.

Como podemos observar en la figura 32, el software EOGEST 1.0 en la pequeña evaluación que se realizó no cumplió los requerimientos en dos ítems del producto instalador, el ítem tiempo de la tarea se demoró más del propuesto y la facilidad de instalación no fue la esperada.

PARTE III. CONCLUSIONES Y
RECOMENDACIONES

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

La realización de este proyecto provee de una herramienta muy poderosa a los grupos de investigación de la Universidad Industrial de Santander, primero por el software desarrollado el cual está en capacidad de evaluar la calidad de los productos software generado por estos grupos y segundo por la investigación que lo precedió la cual provee de nuevos conocimientos y desarrollo de metodologías para la evaluación de la calidad de productos software.

La buena utilización de la herramienta, y el mantenimiento que se le haga se convertirá en un gran aliado para los grupos de investigación en sus desarrollos ya que el gran soporte que tiene permite las mejoras en los desarrollos, y ganando así reconocimiento a nivel interno y externo institucionalmente hablando.

La herramienta desarrollada, se diseñó de tal manera que fuera muy flexible en cuanto al modelo de calidad, ya que permite almacenar múltiples modelos, los cuales se pueden combinar entre si y crear nuevos diseños, permitiendo así la personalización y tener más opciones a la hora de realizar una evaluación.

La arquitectura de QUIS permite la realización de múltiples evaluaciones de forma independiente, permitiendo con esto agilizar procesos cuando se requiere la evaluación de varios productos de forma consecutiva.

Proyectos de este tipo y magnitud permiten una formación más completa en los estudiantes desarrolladores ya que es un área de conocimiento muy importante en nuestro medio y poco explorada y aplicada en la universidad.

5.2 RECOMENDACIONES

Se recomienda como primera medida realizar una capacitación dirigido a los grupos de investigación para que conozcan la herramienta y su utilización, para que se pueda usar de la mejor manera.

Al ser esta herramienta tan flexible, se recomienda actualizar la base de datos con nuevos modelos de calidad, personalizar estos modelos de acuerdo a los productos que se desarrollen en cada grupo, facilitando así la medición.

Al ser QUIS un primer prototipo se recomienda seguir la investigación y el desarrollo; agregándole nuevos módulos, nuevos diseños de tal manera que cada vez mas vaya facilitando la medición y el manejo de esta.

BIBLIOGRAFIA

ICONTEC: Estándar Internacional ISO/IEC 14598. Evaluación del producto software (1999)

ICONTEC: Estándar Internacional ISO/IEC 9126. Modelo de calidad. En Calidad del producto software. (2000).

Mendoza Luis E. Pérez María. Grimán Anna C. Roja Teresita. Algoritmo para la Evaluación de la Calidad Sistémica del Software. Laboratorio de Investigación en Sistemas de Información (LISI). Departamento de Procesos y Sistemas, Universidad Simón Bolívar Caracas – Venezuela. Disponible en: http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_21.pdf

Necesidad de sistemas formales de métricas para proyectos de software OO. Sevilla-JJOO-1997.pdf disponible en: <http://www.di.uniovi.es/~aquilino/ficheros/articulos/Metricas/Sevilla-JJOO-1997.pdf>.

Piattini, M., García, F., Caballero, I., (2007). Calidad de productos software. En Calidad de Sistemas Informáticos (1ª. ed. PP. 81-93). México: Alfa Omega.

Rodríguez Moisés, Garzas Javier, Piattin Mario, Kemis: Un entorno para la medición de la calidad del producto software. Disponible en: www.kybeleconsulting.com.