

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**IMPLEMENTACIÓN DE MÉTRICAS PARA LA EVALUACIÓN DEL
PROCESO DE CONTROL DE CALIDAD EN PROYECTOS DE
DESARROLLO DE SOFTWARE PARA LA EMPRESA LOGICIEL**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL GRADO DE
MAGÍSTER EN SOFTWARE MENCIÓN CALIDAD**

ING. CAIZA CHACÓN GABRIELA ALEXANDRA

`gabriela.caiza@epn.edu.ec`

DIRECTOR: PhD. EDISON FERNANDO LOZA AGUIRRE

`edison.loza@epn.edu.ec`

QUITO, FEBRERO 2019

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Gabriela Alexandra Caiza Chacón bajo mi supervisión.

PhD. Edison Loza Aguirre
DIRECTOR DE PROYECTO

DECLARACIÓN

Yo, Gabriela Alexandra Caiza Chacón, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo los derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Ing. Gabriela Alexandra Caiza Chacón

DEDICATORIA

Dedico este trabajo a Dios, por ser mi guía en cada paso que doy y por darme la sabiduría y fortaleza necesaria para cumplir este sueño.

En especial dedico este trabajo a mi madre, por su abnegación, sabiduría y compañía en cada momento de mi vida, su luz de amor me ha guiado por el camino del bien. Ella es mi mayor inspiración para seguir adelante y lograr mis metas. Por brindarme siempre su apoyo incondicional en este extenso y arduo camino.

También dedico este trabajo a mi esposo David, quién durante este último año de mis estudios ha pasado a ser parte de mi vida y me ha demostrado su amor y su apoyo constante para culminar con éxito esta meta.

Por acompañarme en este sueño que empezó unos años atrás, por estar conmigo cuando he llorado y he reído; pero sobre todo por disfrutar conmigo el dulce sabor que deja la satisfacción de una meta cumplida.

Gabriela Caiza

AGRADECIMIENTO

A Dios, por darme la vida y sobre todo por permitirme culminar con éxito tan anhelado sueño. Gracias, por derramar sobre mí tus bendiciones y levantarme siempre cuando me sentí derrotada. Gracias, por guiarme en el camino correcto por el cual debo ir y hoy gracias a ti he cumplido uno de mis grandes sueños.

A mis padres, por su apoyo constante, sobre todo por el sacrificio que hicieron por mí, sin ellos no hubiera podido lograr esta meta. Mil gracias por sus consejos, esfuerzo y dedicación.

A mi madre, por el amor, la comprensión y el apoyo que me ha dado durante toda mi vida y más aún durante mi carrera universitaria, gracias por creer en mí y por estar siempre a mi lado.

A mi hermana, por estar siempre pendiente de mí brindándome su cariño y apoyo incondicional. Gracias por estar conmigo en los momentos más importantes de mi vida.

A mi sobrina Jocelyne, que es una de mis mayores inspiraciones para lograr esta meta, que siempre tiene una sonrisa o una travesura para alegrar mi vida.

A mi esposo David, por mostrarme lo maravillosa que es la vida cuando se cuenta con una persona tan especial que celebra mis triunfos y me tiende una mano para levantarme en mis tropiezos.

Al Ing. Edison Loza por su ayuda y guía en el desarrollo de este proyecto, gracias por su paciencia y enseñanza.

Gracias a todas las personas que hicieron esto posible, a la empresa LOGICIEL y a todas las personas que me abrieron las puertas para poder realizar este trabajo.

Gabriela Caiza

CONTENIDO

1 INTRODUCCIÓN	1
1.1 Antecedentes de la empresa LOGICIEL	1
1.1.1 Descripción de la empresa [1]	1
1.1.2 Misión [1].....	1
1.1.3 Valores [1]	2
1.1.4 Enfoque [1].....	2
1.1.5 Cadena de valor [1]	2
1.1.6 Proceso de pruebas [2].....	3
1.1.6.1 Política de pruebas del software	10
1.1.6.2 Funciones del grupo de pruebas del software	10
1.2 Planteamiento del problema	11
1.3 Objetivos	12
1.3.1 Objetivo general	12
1.3.2 Objetivos específicos.....	12
1.4 Marco teórico.....	12
1.4.1 Conceptos	13
1.4.1.1 Calidad de software	13
1.4.1.2 Medida	13
1.4.1.3 Métrica	13
1.4.1.4 Medición	14
1.4.1.5 Indicador.....	15
1.4.1.6 Error	15
1.4.1.7 Defecto.....	15
1.4.1.8 Fallo	15
1.4.2 Fundamentos de pruebas	15
1.4.2.1 Proceso de pruebas.....	15
1.4.2.2 Objetivos de las pruebas	18
1.4.2.3 Niveles de pruebas.....	18
1.4.2.4 Tipos de pruebas.....	19
1.4.3 Gestión de pruebas	20
1.4.3.1 Seguimiento y estimación del estado de las pruebas	20
1.4.3.2 Métricas de calidad para el proceso de pruebas.....	21
1.4.3.3 Productividad del equipo de pruebas.....	22
1.4.4 Modelos de calidad.....	23
1.4.4.1 Enfoques de los modelos de calidad de software	24
1.4.4.2 Evolución de los modelos de calidad a nivel de producto	24
1.4.5 Modelos de calidad a nivel de producto.....	25

1.4.5.1	Modelo de Calidad – McCall	25
1.4.5.2	Estándar de Calidad – ISO/IEC 9126	26
1.4.5.2.1	División del Estándar ISO/IEC 9126	26
1.4.5.2.2	Relación de la ISO/IEC 9126 con la ISO/IEC 14598	28
1.4.5.3	Modelo de Calidad – ISO/IEC 25000	30
1.4.5.3.1	División de la Norma ISO/IEC 25000 [24]	30
1.4.5.3.2	Ciclo de vida de la calidad del Producto de Software [24]	32
1.4.5.3.3	Métricas para la calidad interna, externa y en uso - ISO/IEC 25022 e ISO/IEC 25023 [25] [26]	33
1.4.6	Metodologías de medición	38
1.4.6.1	Metodología GQM	38
2	METODOLOGÍA	42
2.1	Introducción	42
2.2	Visión general de la metodología	42
2.3	Aplicación de la metodología	45
2.3.1	Diagnosticar	45
2.3.1.1	Definición del Problema	45
2.3.1.2	Recolección de datos	46
2.3.1.3	Análisis de datos	47
2.3.1.4	Resultados principales	48
2.3.2	Planificar acciones	55
2.3.2.1	Matrices de impacto	55
2.3.2.2	Marco de referencia	62
2.3.2.3	Comparación del Marco de Referencia con el Diagnóstico Realizado a la Empresa Logiciel	64
2.3.3	Implementar acciones	66
2.3.3.1	Proyectos a ser evaluados	66
2.3.3.2	Definición de métricas existentes a ser usadas	67
2.3.3.3	Implementación de métricas a ser usadas	74
2.3.3.4	Procedimientos para la recolección de datos	92
2.3.3.5	Procedimientos para la evaluación del producto de software y del proceso de pruebas	93
2.3.4	Evaluación	96
2.3.5	Especificación el aprendizaje	101
3	RESULTADOS Y DISCUSIÓN	110
3.1	Aplicación de las métricas implementas	110
3.1.1	Caso de estudio 1	110
3.1.1.1	Antecedentes del Proyecto:	110
3.1.1.2	Recopilación de datos	110
3.1.1.3	Presentación de resultados	111

3.1.1.4	Lecciones aprendidas.....	122
3.1.2	Caso de estudio 2.....	123
3.1.2.1	Antecedentes del Proyecto:.....	123
3.1.2.2	Recopilación de datos	123
3.1.2.3	Presentación de resultados	124
3.1.2.4	Lecciones aprendidas.....	135
3.1.3	Caso de estudio 3.....	136
3.1.3.1	Antecedentes del Proyecto:.....	136
3.1.3.2	Recopilación de datos	136
3.1.3.3	Presentación de resultados	137
3.1.3.4	Lecciones aprendidas.....	148
3.2	Evaluación por parte de los usuarios finales.....	149
3.3	Mejoras para el proceso de pruebas.....	153
4	CONCLUSIONES Y RECOMENDACIONES	155
4.1	Conclusiones.....	155
4.2	Recomendaciones.....	157
5	REFERENCIAS BIBLIOGRÁFICAS	158
6	ANEXOS.....	161
	ANEXO I: ENTREVISTAS REALIZADAS	161
	ANEXO II: CODIFICACIÓN DE ENTREVISTAS	161
	ANEXO III: PLANTILLAS PARA LA RECOLECCIÓN DE DATOS	161
	ANEXO IV: MATRICES DE EVALUACIÓN.....	161
	ANEXO V: EVALUACIÓN CASO DE ESTUDIO 1	161
	ANEXO VI: EVALUACIÓN CASO DE ESTUDIO 2	161
	ANEXO VII: EVALUACIÓN CASO DE ESTUDIO 3	161

ÍNDICE DE TABLAS

CAPITULO 1

Tabla 1.1 Métricas sobre el avance del proceso en la ejecución de las pruebas exploratorias [15]	22
Tabla 1.2 Métricas para la Calidad Interna y Externa, adaptado de [27]	34
Tabla 1.3 Métricas de Calidad en Uso, adaptado de [26].....	37
Tabla 1.4 Formato para definir métricas usando la metodología GQM [28].....	41

CAPITULO 2

Tabla 2.1 Guía para entrevistas, elaborado por: Caiza Gabriela	46
Tabla 2.2 Guía de codificación, elaborado por: Caiza Gabriela.....	47
Tabla 2.3 Codificación de las entrevistas, elaborado por: Caiza Gabriela	48
Tabla 2.4 Conclusiones de las entrevistas realizadas, elaborado por: Caiza Gabriela	49
Tabla 2.5 Matriz de impacto del proceso de pruebas vs las necesidades relacionadas a métricas, elaborado por: Caiza Gabriela	56
Tabla 2.6 Matriz de impacto de las necesidades vs las soluciones, elaborado por: Caiza Gabriela	58
Tabla 2.7 Matriz de impacto del proceso de pruebas vs las soluciones relacionadas a métricas, elaborado por: Caiza Gabriela	60
Tabla 2.8 Métricas seleccionadas para evaluar el producto, elaborado por: Caiza Gabriela	64
Tabla 2.9 Métricas seleccionadas para evaluar el proceso de pruebas, elaborado por: Caiza Gabriela	65
Tabla 2.10 Porcentaje de importancia de las métricas sobre el 100% para evaluar el producto de software, elaborado por: Gabriela Caiza	94
Tabla 2.11 Porcentaje de importancia de las métricas sobre el 100% para evaluar el proceso de pruebas, elaborado por: Gabriela Caiza	95
Tabla 2.12 Niveles de puntuación final para la calidad del producto de software y del proceso de pruebas [35]	96
Tabla 2.13 Preguntas enfocadas en las características principales del modelo TAM.....	99
Tabla 2.14 Guía para la entrevista de evaluación, elaborado por: Caiza Gabriela	100
Tabla 2.15 Entrevista de evaluación, elaborado por: Caiza Gabriela	101
Tabla 2.16 Matriz para evaluar la calidad del producto de software, elaborado por: Caiza Gabriela	102

Tabla 2.17 Matriz para evaluar el proceso de pruebas, elaborado por: Caiza Gabriela..105

CAPITULO 3

Tabla 3.1 Matriz de evaluación de la calidad del producto desarrollado en el proyecto del caso de estudio 1, elaborado por: Caiza Gabriela.....112

Tabla 3.2 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 1, elaborado por: Caiza Gabriela114

Tabla 3.3 Calificación de calidad del producto y del proceso de pruebas desarrollado en el proyecto del caso de estudio 1122

Tabla 3.4 Matriz de evaluación de la calidad del producto de software desarrollado en el proyecto del caso de estudio 2, elaborado por: Caiza Gabriela125

Tabla 3.5 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 2, elaborado por: Caiza Gabriela127

Tabla 3.6 Calificación de calidad del producto de software y del proceso de pruebas desarrollado en el proyecto del caso de estudio 2134

Tabla 3.7 Matriz de evaluación de la calidad del producto desarrollado en el proyecto del caso de estudio 3, elaborado por: Caiza Gabriela.....138

Tabla 3.8 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 3, elaborado por: Caiza Gabriela140

Tabla 3.9 Calificación de calidad del producto y del proceso de pruebas desarrollado en el proyecto del caso de estudio 3148

Tabla 3.10 Resultados obtenidos de las encuestas aplicadas relacionadas a las preguntas enfocadas en la Utilidad percibida, elaborado por: Caiza Gabriela149

Tabla 3.11 Resultados obtenidos de las encuestas aplicadas relacionadas a las preguntas enfocadas en la Facilidad de uso percibida, elaborado por: Caiza Gabriela151

ÍNDICE DE FIGURAS

CAPITULO 1

Figura 1.1 Cadena de Valor de LOGICIEL [1].....	3
Figura 1.2 Macro proceso de Gestión de la Calidad de Software [2].....	4
Figura 1.3 Modelo de desarrollo que usa la empresa Logiciel [3].....	5
Figura 1.4 Fases del proceso de pruebas de la empresa LOGICIEL [2]	7
Figura 1.5 Proceso de Pruebas [13].....	16
Figura 1.6 Modelo V general [13].....	16
Figura 1.7 Modelo Iterativo [14]	17
Figura 1.8 Niveles de Pruebas [13].....	18
Figura 1.9 Modelo de calificación de los Tester [16].....	23
Figura 1.10 Modelos de calidad a nivel de producto [6]	24
Figura 1.11 Factores de calidad de McCall [4]	26
Figura 1.12 División de la Norma ISO/IEC 9126 [6]	27
Figura 1.13: Relación normas ISO/IEC 9126 e ISO/IEC 14598 [19] [23].....	28
Figura 1.14: Relación entre ISO/IEC 9126, ISO/IEC 14598 e ISO/IEC 25000 [19] [23] [24]	29
Figura 1.15 División de la Norma ISO/IEC 25000 [24]	30
Figura 1.16 Ciclo de Vida de la Calidad del Producto de Software [24].....	32
Figura 1.17 Relación entre la ISO/IEC 2502n e ISO/IEC 9126 [25] [26].....	33
Figura 1.18 Relación entre los tipos de métricas de calidad [25].....	34
Figura 1.19 Niveles de GQM [29].....	39
Figura 1.20 Proceso GQM [28]	40

CAPITULO 2

Figura 2.1 Modelo canónico de la metodología Investigación-Acción [31].....	43
Figura 2.2 Marco de Referencia para Evaluar el Proceso de Pruebas [24], [13], [33].....	63
Figura 2.3 Estados de los requerimientos registrados en la herramienta Test Manager [34]	75
Figura 2.4 Estados de los casos de prueba registrados en la herramienta Test Manager [34]	76
Figura 2.5 Estados de los errores registrados en la herramienta Test Manager [34].....	77
Figura 2.6 Modelo de Aceptación Tecnológica (TAM) [36].....	97
Figura 2.7 Escalas de valoración Likert [39].....	98

Figura 2.8 Rangos de contestación y valoración para la entrevista de evaluación99

CAPITULO 3

Figura 3.1 Resultados obtenidos de la evaluación del producto de software desarrollado en el proyecto del caso de estudio 1118

Figura 3.2 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 1120

Figura 3.3 Resultados obtenidos de la evaluación del producto desarrollado en el proyecto del caso de estudio 2131

Figura 3.4 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 2133

Figura 3.5 Resultados obtenidos de la evaluación del producto desarrollado en el proyecto del caso de estudio 3144

Figura 3.6 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 3146

Figura 3.7 Resultados consolidados de la encuesta aplicada correspondiente a las preguntas enfocadas a la Utilidad percibida150

Figura 3.8 Resultados obtenidos de la encuesta aplicada correspondiente a las preguntas enfocadas a la Facilidad de uso percibida152

RESUMEN

En el presente proyecto de titulación se presenta el desarrollo de métricas para evaluar el proceso de control de calidad en proyectos de desarrollo de software para la empresa LOGICIEL. El objetivo del trabajo subyace en ayudar a los directivos de la empresa a la toma de decisiones que permitan mejorar la calidad del proceso de pruebas y del producto final.

Previo al desarrollo de las métricas, se realizó un reconocimiento del proceso de control de calidad implementado en la empresa LOGICIEL con el fin de conocer a detalle el proceso de pruebas definido. Adicionalmente, se realizó una revisión literaria de los fundamentos relacionados al proceso de pruebas, modelos de calidad y metodologías de medición.

En este trabajo se escogió aplicar la metodología Investigación-Acción, la cual nos permitió alcanzar nuestro objetivo. Para ello, primero se realizó el diagnóstico del proceso de pruebas y en base a los resultados obtenidos se planificaron acciones para implementar las métricas. De acuerdo con la revisión literaria previamente realizada, se pudo observar que ya existían algunas métricas para evaluar el proceso de pruebas; sin embargo, no existían métricas específicas que se adapten a los objetivos que persigue la empresa LOGICIEL. Por ello se vio la necesidad de proponer algunas métricas que permitan evaluar el proceso de pruebas de acuerdo con los objetivos planteados por la empresa.

Para la implementación de las métricas se tomó como base la norma ISO/IEC 25023 y los fundamentos de ISTQB (Junta Internacional de Calificaciones para Pruebas de Software). De acuerdo a los resultados obtenidos de la evaluación realizada usando las métricas implementadas, se formularon mejoras para el proceso de pruebas y se pudo constatar que la calidad del proceso de pruebas influye en la calidad del producto final.

Palabras clave: control de calidad, proceso de pruebas, métricas, modelos de calidad, metodologías de medición.

ABSTRACT

In the present project we present the development of metrics to evaluate the process of quality control in projects of software development for the company LOGICIEL. The objective of the work lies in helping the executives of the company to make decisions that improve the quality of the testing process and the final product.

Prior to the development of the metrics, an analysis was made about the quality control process implemented in LOGICIEL. The objective was to know in detail the defined testing process. Additionally, a literature review was made about the fundamentals related to the testing process, quality models and measurement methodologies.

We followed a Research-Action approach, which allowed us to achieve the objective set out in this work. To do this, a diagnosis of the testing process was performed and, based on the results obtained, actions were planned to implement the metrics. According to the literature review previously carried out, it was observed that there were some metrics to evaluate the testing process; however, there were no specific metrics adapted to the objectives pursued by LOGICIEL. Therefore, it was necessary to implement some metrics that allow evaluating the testing process according to the objectives set by the company.

For the implementation of the metrics, the ISO/IEC 25023 standard and the foundations of ISTQB (International Board of Qualifications for Software Testing) were taken as a basis. According to the results obtained from the evaluation carried out using the metrics implemented, improvements were made to the testing process and it was found that the quality of the testing process influences the quality of the final product.

Keywords: quality control, testing process, metrics, quality models, measurement methodologies

PRESENTACIÓN

En este proyecto se detalla la implementación de métricas para la evaluación del proceso de control de calidad en proyectos de desarrollo de software para la empresa LOGICIEL.

El trabajo comienza presentando los antecedentes de la empresa LOGICIEL y la descripción del proceso de control de calidad definido en la empresa. A continuación, se muestra el planteamiento del problema, los objetivos generales y específicos del presente trabajo.

Luego, se presenta el marco teórico en el cual se incluye ciertos conceptos básicos sobre pruebas (fundamentos, gestión, modelos de calidad y metodologías de medición para implementar métricas). A continuación, se detalla la metodología Investigación – Acción y se presenta el desarrollo de cada una de las fases de la metodología aplicada durante el presente proyecto para la implementación de las métricas.

Una vez implementadas las métricas se las aplica a tres proyectos de desarrollo de software de la empresa LOGICIEL y se muestran los resultados obtenidos de cada uno de los casos de estudio evaluados. Posteriormente, se presenta la evaluación realizada por parte de los usuarios de la empresa LOGICIEL y se plantean las mejoras para el proceso de pruebas. Finalmente se presentan las conclusiones y recomendaciones obtenidas a lo largo del proyecto de titulación.

1 INTRODUCCIÓN

1.1 Antecedentes de la empresa LOGICIEL

1.1.1 Descripción de la empresa [1]

LOGICIEL Compañía Limitada, es una organización de responsabilidad limitada, orientada en la entrega de servicios en tecnología de la información para cubrir las necesidades de TI en el segmento financiero-bancario. La empresa ha visto la necesidad de mejorar continuamente sus procesos de desarrollo y gestión de sus productos de software, con el objetivo de generar sistemas de alta calidad y aumentar su competitividad en el medio.

En el 2010, LOGICIEL obtuvo la certificación de calidad de procesos técnicos y del negocio IT MARK. Esta certificación se basa en las áreas de proceso del modelo CMMI (Integración de Modelos de Madurez de Capacidades), avalada por el Instituto Europeo de Software ESI, y se centra en tres perspectivas de la empresa: la gestión general de la empresa, la seguridad de la información, y los procesos de desarrollo de software. Ésto le permitió a la empresa formalizar sus procesos.

1.1.2 Misión [1]

La misión de LOGICIEL está definida como sigue:

“Brindar soluciones integradas en Tecnología de la Información que apoyen a las empresas de producción y servicio en la consecución de sus metas críticas entregando productos y servicios con valor agregado que superen las expectativas y necesidades de nuestros clientes.

Buscamos además que el conocimiento, la calidad de vida y los valores compartidos sean el eje para el desarrollo tanto personal de nuestros empleados y usuarios, así como de nuestras relaciones “.

1.1.3 Valores [1]

La empresa ha definido cinco valores para sus operaciones:

- La integridad de nuestro personal y por lo tanto de nuestra empresa, basa en valores humanos como la honestidad, solidaridad, equidad y responsabilidad.
- Cultura de servicio a los clientes (calidad, efectividad, compromiso, comunicación, pro-actividad).
- La rentabilidad basada en la efectividad, el conocimiento y experiencia de nuestros recursos humanos.
- Reconocimiento y motivación al recurso productivo, efectivo, innovador.
- Buscar el éxito personal, de la empresa y de nuestros clientes.

1.1.4 Enfoque [1]

Para garantizar que los productos y servicios entregados a clientes sean de calidad, LOGICIEL utiliza un enfoque basado en los siguientes elementos:

- Integración de sistemas.
- Modelo de desarrollo de sistemas de información.
- Gestión de calidad del software:
- Plan de pruebas de software.
- Definición y uso de estándares.
- Participación de usuarios.

1.1.5 Cadena de valor [1]

En la Figura 1.1, se presentan los macros procesos de la empresa LOGICIEL, estructurados en su cadena de valor. Se incluyen los procesos principales y los procesos secundarios que han sido agrupados en dos categorías procesos estratégicos – administrativos y procesos de soporte.

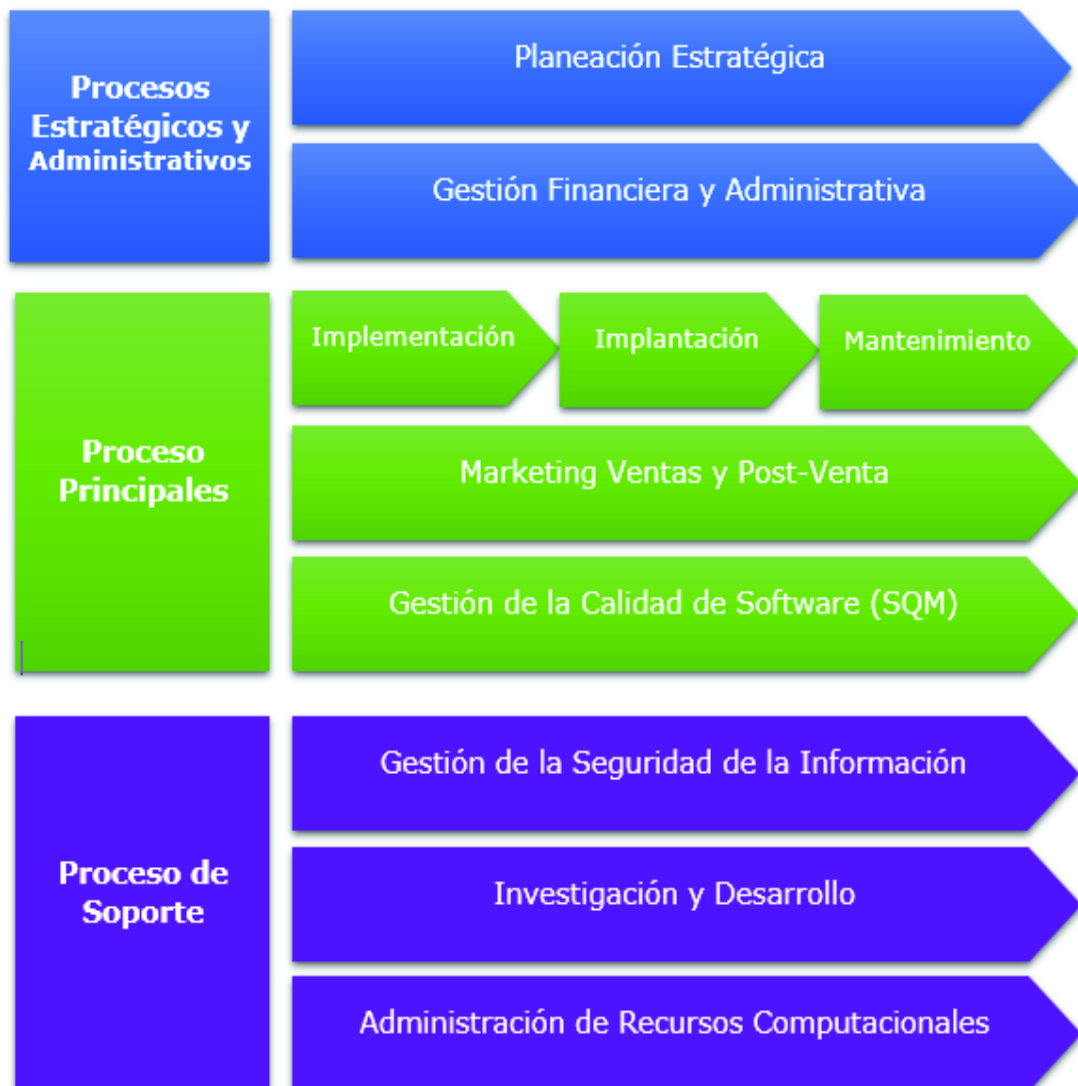


Figura 1.1 Cadena de Valor de LOGICIEL [1]

1.1.6 Proceso de pruebas [2]

En la Figura 1.2 se muestra el proceso de pruebas que se encuentra definido dentro del macro proceso de Gestión de la Calidad de Software.

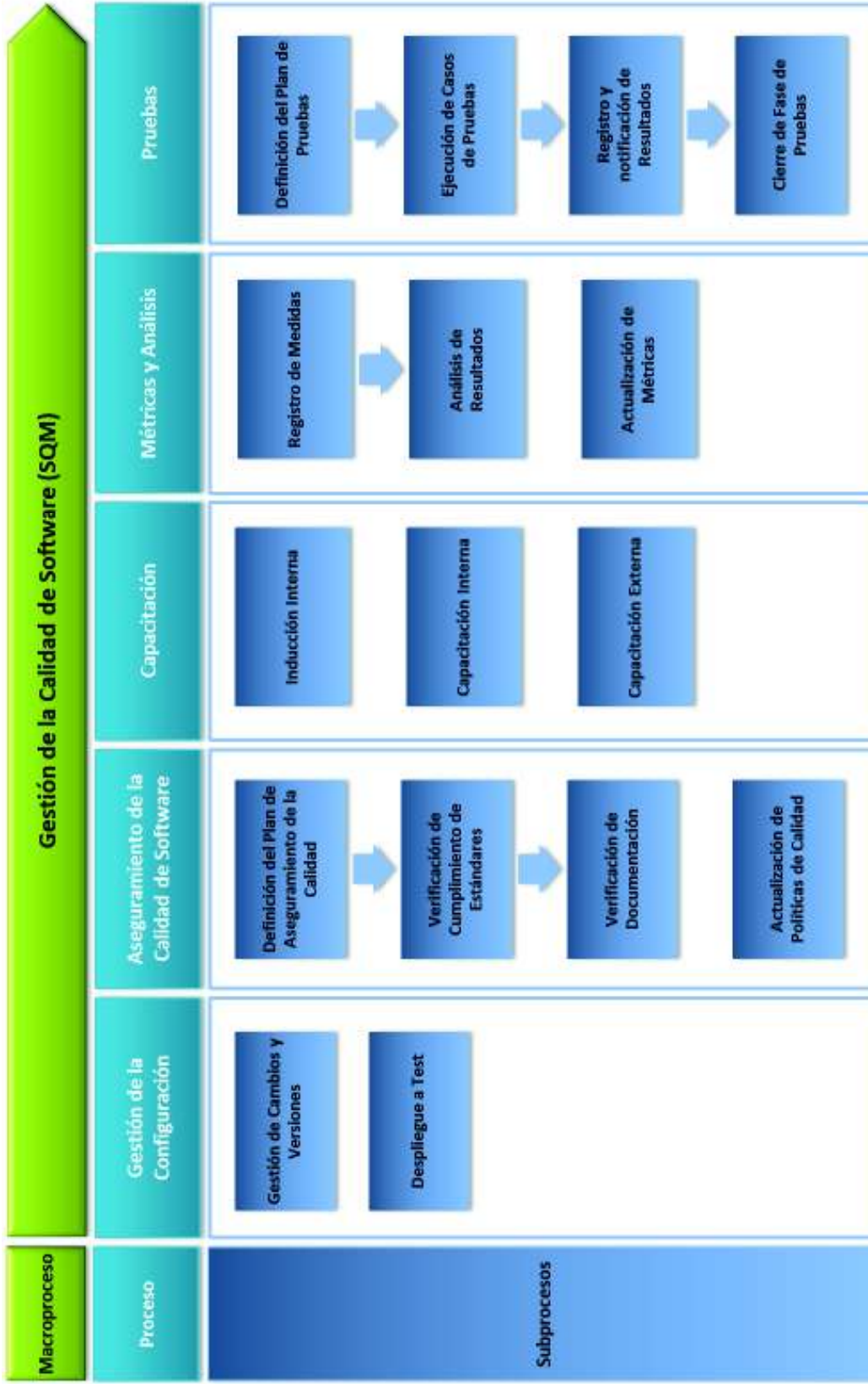


Figura 1.2 Macro proceso de Gestión de la Calidad de Software [2]

El modelo de desarrollo propuesto por la empresa Logiciel, es una aproximación semi-formal para el desarrollo de aplicaciones distribuidas en n-capas, sustentado en componentes, considerando un proceso iterativo e incremental como se muestra en la Figura 1.3, respetando las fases del “ciclo de vida del desarrollo de sistemas de información” de manera que se mantenga un acercamiento consistente entre sus fases [3].

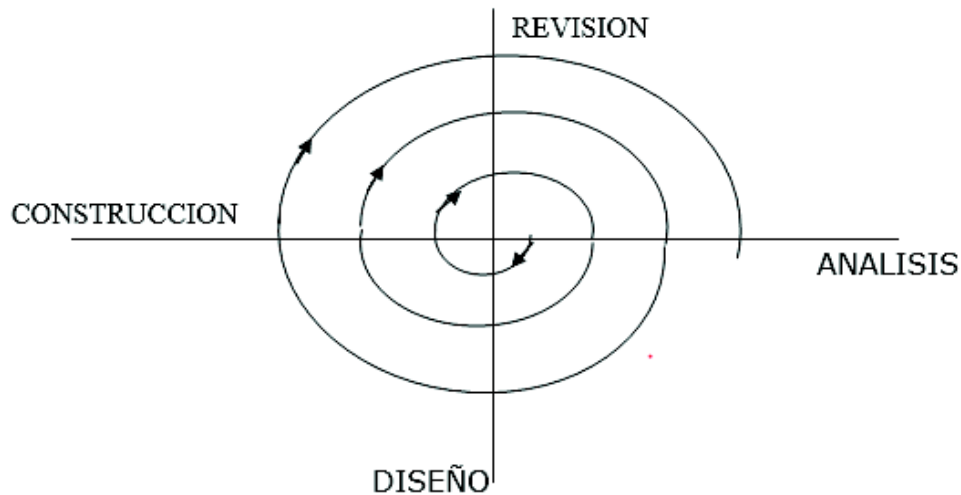


Figura 1.3 Modelo de desarrollo que usa la empresa Logiciel [3]

El modelo contempla las siguientes fases [3]:

- Análisis de Requerimientos
- Análisis del Sistema (Diseño Conceptual)
- Diseño Lógico
- Diseño Físico
- Construcción
- Pruebas

El proceso de desarrollo de software tiene como entrada los requerimientos que debe satisfacer la aplicación, y como salida la funcionalidad y los entregables generados durante el desarrollo del sistema, para su explotación por parte de los usuarios.

Se debe aclarar que el proceso de pruebas sobre el sistema de información deberá ser llevado por un equipo de Gestión de la Calidad del Software, destinado para este fin, mediante un proceso paralelo al desarrollo [3].

Los objetivos de la estrategia de pruebas son:

- Planificar las pruebas necesarias en cada iteración, incluyendo las pruebas de unidad, integración y las pruebas de sistema. Las pruebas de unidad y de integración son necesarias dentro de la iteración, mientras que las pruebas de sistema son necesarias sólo al final de la iteración.
- Diseñar e implementar las pruebas creando los casos de prueba que especifican qué probar, cómo realizar las pruebas y creando, si es posible, componentes de prueba ejecutables para automatizar las pruebas.
- Realizar diferentes pruebas y manejar los resultados de cada prueba sistemáticamente. Los productos de desarrollo de software en los que se detectan defectos son probados de nuevo y posiblemente devueltas a otra etapa, como diseño o implementación, de forma que los defectos puedan ser arreglados.

Para conseguir estos objetivos el flujo de trabajo de la etapa de pruebas consta de las siguientes fases:

- Planificación de las pruebas.
- Diseño de las pruebas.
- Implementación de las pruebas.
- Ejecución de las pruebas.
- Evaluación de las pruebas.

Los productos de desarrollo del software fundamentales que se desarrollan en la etapa de pruebas son:

- Plan de pruebas.
- Casos de prueba.
- Modelo de pruebas, que incluye: clases de prueba, entorno de configuración de pruebas, componentes de prueba y los datos de prueba.
- Informe de evaluación de pruebas.

En la Figura 1.4 se presenta el flujo de interacción de las fases del proceso de pruebas con los respectivos entregables que se deben generar en cada una de ellas:

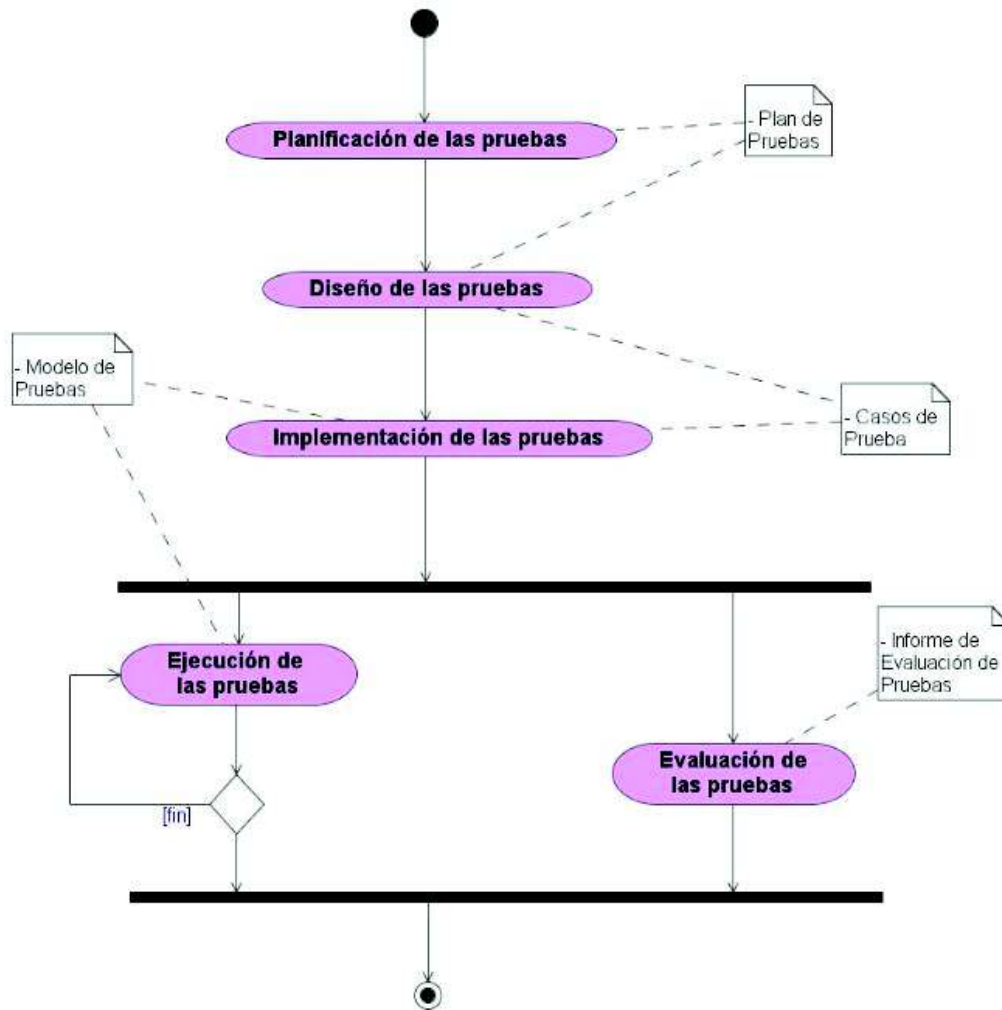


Figura 1.4 Fases del proceso de pruebas de la empresa LOGICIEL [2]

A continuación, se detalla cada una de las fases de proceso de pruebas que tiene implementado la empresa:

a) Planificación de las pruebas

El propósito es identificar y describir las pruebas que se van a implementar y ejecutar, llevando a cabo las siguientes tareas:

- Describiendo una estrategia de prueba.
- Estimando los requisitos para el esfuerzo de la prueba, por ejemplo, los recursos humanos y sistemas necesarios.
- Planificando el esfuerzo de la prueba.

En esta actividad se deben identificar y describir todas las pruebas que se van a realizar para garantizar la calidad del producto desarrollado. Este objetivo se consigue generando un plan de pruebas que contiene todos los requerimientos necesarios para realizar las pruebas y una estrategia de pruebas, es decir: definir el tipo de pruebas a ejecutar, cómo ejecutar dichas pruebas, cuándo ejecutarlas y cómo determinar si el esfuerzo de prueba tiene éxito.

- Se debe desarrollar un plan de pruebas simple que describa todos los tipos de pruebas que se tienen que implementar y ejecutar.
- Esta actividad permite medir y gestionar los esfuerzos necesarios para llevar a cabo la etapa de pruebas.

Productos de desarrollo software que se obtienen

De esta actividad se obtiene:

- Plan de pruebas.

b) Diseño de las pruebas

El propósito de esta actividad es identificar, describir y generar los casos de prueba y el modelo de pruebas correspondiente, especificando cómo realizar los casos de prueba.

Esta actividad permite estructurar y organizar bien las pruebas. El diseñador de pruebas analiza los objetivos de las pruebas que se han planificado y desarrolla los casos de prueba y el modelo de pruebas necesario para llevarlas a cabo.

- Los diseñadores de prueba deberían crear un conjunto de casos de prueba que hicieran posible alcanzar los objetivos establecidos en el plan de prueba con un esfuerzo mínimo. Para poder hacer esto, los diseñadores de pruebas intentan encontrar un conjunto de casos de prueba con un solapamiento mínimo. Cada caso de prueba debe evaluar un camino o escenario interesante.

Productos de desarrollo software que se obtienen

De esta actividad se obtiene:

- Casos de prueba.
- Modelo de pruebas.
- Plan de pruebas (revisado).

c) Implementación de las pruebas

El propósito de esta actividad es implementar los casos de prueba que se han definido en la actividad anterior de diseño de las pruebas, automatizando todos los que sean posibles. Los componentes de prueba, clases de prueba y datos de prueba se crean usando los casos de prueba como entrada.

Los componentes de prueba usan a menudo grandes cantidades de datos de entrada para ser probados y producen grandes cantidades de datos de salida como resultado de las pruebas. Es útil poder visualizar estos datos de forma clara e intuitiva de manera que puedan especificarse correctamente y los resultados de las pruebas puedan ser interpretados.

Productos de desarrollo software que se obtienen

De esta actividad se obtiene:

- Modelo de pruebas (Clases de prueba, componentes de prueba, entorno de configuración de pruebas).

d) Ejecución de las pruebas

El propósito de esta actividad es llevar a cabo las pruebas de unidad, integración y sistema del producto desarrollado para garantizar su calidad.

- Se realizan las pruebas de integración necesarias para cada uno de los productos de desarrollo software desarrollado en una iteración y se recopilan los resultados de las pruebas
- Se busca realizar las pruebas de sistema necesarias en cada iteración y el recopilar los resultados de las pruebas. La prueba de sistema puede empezar cuando las pruebas de integración indican que el sistema satisface los objetivos de calidad de integración fijados en el plan de prueba de la iteración actual.

Productos de desarrollo software que se obtienen

De esta actividad se obtiene:

- Defecto
- Documento de control de cambios

e) Evaluación de las pruebas

El propósito de la actividad de pruebas es evaluar los resultados (medidas cuantificables) de las pruebas para determinar la calidad del producto desarrollado y la calidad del proceso de pruebas. Los diseñadores de pruebas llevan a cabo esta actividad revisando y evaluando los resultados de las pruebas, para lo cual comparan los resultados obtenidos con los objetivos esbozados en el plan de prueba.

- Los diseñadores de prueba deben preparar métricas que le permitan determinar el nivel de calidad del software y determinar qué cantidad de pruebas es necesario realizar.

Productos de desarrollo software que se obtienen

- Informe de evaluación de pruebas.

1.1.6.1 Política de pruebas del software

Las políticas definidas por LOGICIEL para sus pruebas son las siguientes:

- a) La función de pruebas del software debe estar incluida en todos los proyectos.
- b) El proceso de pruebas unitarias y de integración (estrategia, plan de pruebas, ejecución de las pruebas) es responsabilidad del grupo de desarrollo del sistema de información.
- c) El proceso de pruebas de sistema (estrategia, plan de pruebas, ejecución de las pruebas) es responsabilidad del grupo de pruebas del software.
- d) El grupo de pruebas del software tiene un canal de reporte a la supervisión, es independiente del líder del proyecto y del grupo de desarrollo del sistema de información.
- e) La supervisión revisa periódicamente las actividades y resultados del grupo de pruebas del software.

1.1.6.2 Funciones del grupo de pruebas del software

En LOGICIEL, son funciones del grupo de pruebas:

- a) Brindar soporte en la elaboración de los planes de prueba unitarias y de integración.

- b) Brindar soporte en la ejecución de pruebas unitarias y de integración.
- c) Elaboración de la estrategia y plan de pruebas del sistema de información y de aceptación para cada proyecto.
- d) Ejecución de las pruebas de sistema de información, y cuando se requiera brindar el soporte para la ejecución de las pruebas de aceptación.
- e) Capacitación a los miembros del grupo de desarrollo del sistema de información en el rol, responsabilidades, autoridad y valor del grupo de pruebas del software.

1.2 Planteamiento del problema

LOGICIEL como una empresa de desarrollo de software cuyo segmento principal es el financiero – bancario, ha visto la necesidad de entrar en un esquema de mejoramiento continuo de su proceso de gestión de calidad de software para generar sistemas de calidad y aumentar su competitividad en el mercado.

Actualmente, LOGICIEL cuenta con un proceso de control de calidad basado en CMMI nivel 2 para los proyectos de desarrollo de software. Este proceso de pruebas fue definido en el año 2010 cuando obtuvo la certificación de calidad de procesos de IT Mark. Desde esa fecha hasta la actualidad, el proceso no ha sido ni evaluado ni mejorado; lo que provoca muchos problemas que afectan directamente a la calidad del producto final. Entre los principales problemas podemos mencionar: mala planificación de las pruebas, baja detección de defectos durante las pruebas de integración y de sistema, falta de tiempo para terminar de ejecutar las pruebas, falta de tiempo para la corrección de todos los defectos encontrados durante el proceso de pruebas, entre otros. Esto se pudo evidenciar en la reunión que se mantuvo con la gerente del área de QA, los Tester y analistas de pruebas de la empresa.

Además, el proceso de control de calidad implementado no se adapta a las necesidades actuales que requieren los diferentes proyectos de desarrollo que está llevando a cabo la empresa LOGICIEL. Se trata de un proceso que no es percibido como ágil y requiere de mucha documentación innecesaria. Por lo que el proceso de control de calidad se está volviendo incontrolable dentro de los proyectos de desarrollo.

El presente trabajo de titulación tiene como objetivo principal implementar métricas para evaluar el proceso de control de calidad con el fin de mejorar la calidad del producto final y solventar los problemas mencionados.

1.3 Objetivos

1.3.1 Objetivo general

Implementar métricas para evaluar el proceso de control de calidad en proyectos de desarrollo de software para la empresa LOGICIEL.

1.3.2 Objetivos específicos

- Diagnosticar el proceso de control de calidad de la empresa LOGICIEL.
- Definir métricas para evaluar el proceso de control de calidad.
- Aplicar y evaluar las métricas en tres proyectos de desarrollo de software.
- Formular mejoras para el proceso de control de calidad de la empresa LOGICIEL.

1.4 Marco teórico

En esta sección vamos a abordar los principales conceptos relacionados a calidad, fundamentos de pruebas, gestión de pruebas y modelos de calidad, los cuales se requieren revisarlos con el fin de tenerlos claros durante el desarrollo del presente trabajo. También se detalla el enfoque y la evolución de los modelos de calidad más aceptados para la evaluación de calidad.

Finalmente, se ofrece una pequeña introducción sobre las metodologías de medición y se amplían los conceptos fundamentales de la metodología GQM (Meta-Pregunta-Métrica) la cual se usará en el presente trabajo.

En este trabajo nos centraremos en la norma ISO/IEC 25000:2005 y los fundamentos de ISTQB, los cuales servirán como base para plantear algunas métricas a usar para evaluar el proceso de pruebas de la empresa LOGICIEL, así como para implementar otras métricas que se requieren usaremos la metodología GQM.

1.4.1 Conceptos

Es necesario pasar por una revisión de conceptos básicos requeridos para el desarrollo del presente trabajo, con el fin de tener claro los aspectos fundamentales a ser tratados.

1.4.1.1 Calidad de software

La calidad de software se refiere al grado de desempeño de las principales características con las que debe cumplir un sistema durante su ciclo de vida. Dichas características garantizan que el cliente cuente con un sistema confiable [4] [5].

Para garantizar la calidad de software es importante implementar algún modelo o estándar de calidad que permita evaluar las características del producto de software desarrollado [6].

1.4.1.2 Medida

Las medidas proporcionan una indicación cuantitativa de algún atributo de un producto o proceso. Los requisitos del software son la base de las medidas de calidad; la falta de concordancia con estos requisitos es una falta de calidad [4].

1.4.1.3 Métrica

“Una métrica es una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo determinado” [4].

“Cuando puedes medir lo que estás diciendo y expresarlo en número, sabrás algo acerca de eso; pero cuando no puedes medirlo, cuando no puedes expresarlo en número, tus conocimientos serán escasos y no satisfactorios”, según Lord Kelvin [7].

Los estándares y modelos relacionados con métricas de calidad del software tradicionales se han centrado fundamentalmente en las métricas de procesos, de productos y de recursos [8].

Las métricas deben cumplir las siguientes características [9]:

- Deben ser cuantificables, deben basarse en hechos.
- Deben ser independientes.
- Deben ser explicables, debe documentarse información.
- Deben ser precisas, se requiere establecer un nivel de tolerancia permitido cuando se mide.

1.4.1.4 Medición

La medición es el acto de determinar una medida; la cual ocurre como resultado de la recopilación de uno o más datos. “La medición es un elemento clave en cualquier proceso de ingeniería”, según Pressman [4].

Según Sommerville, “La medición es una actividad que forma parte de un proceso, que consiste en asociar valores numéricos a atributos de productos o procesos de software”. La medición de los procesos y sus características hacen posible la mejora de la calidad del proceso y de sus productos [10].

No obstante, como todo proceso perfectible, no se pueden alcanzar mejoras sin mediciones que nos permitan identificar tanto los procesos alcanzados como los errores a corregirse. Así, existen cuatro razones importantes para medir los procesos de software, productos o recursos [11]:

- Medir para caracterizar
- Medir para evaluar
- Medir para predecir
- Medir para mejorar

Un proceso de medición se caracteriza por cinco actividades [4]:

- **Formulación:** Selección de medidas y métricas apropiadas para el procesos o producto a ser evaluado.
- **Recolección:** Mecanismo a ser usado para la recolección de datos necesarios para aplicar las métricas formuladas
- **Análisis:** Cálculo de las métricas.
- **Interpretación:** Evaluación de las métricas.
- **Retroalimentación:** Recomendaciones derivadas de la interpretación de las métricas.

Incluso cuando un proyecto no está en problemas, la medición no sólo es útil sino necesaria. Por lo tanto, es necesaria la medición, al menos, para evaluar el estado de los proyectos, productos, procesos y recursos. Cada acción de medición debe estar motivada por un objetivo particular o una necesidad [12].

1.4.1.5 Indicador

Un indicador es una métrica o una combinación de métricas que proporcionan conocimiento acerca de un proyecto, proceso o producto de software. Este conocimiento permitirá tomar decisiones claves para mejorar el proyecto, el proceso o el producto [4].

1.4.1.6 Error

“Un error es una acción humana que produce un resultado incorrecto”, según la IEEE 610 [5].

1.4.1.7 Defecto

“Un defecto es un desperfecto en un componente o sistema que puede causar un fallo en el desempeño de las funciones requeridas. Los defectos causan fallos” [13].

1.4.1.8 Fallo

“Un fallo es una manifestación física o funcional de un defecto. Si un defecto es encontrado durante la ejecución de una aplicación puede producir un fallo” [13].

1.4.2 Fundamentos de pruebas

En esta sección se revisarán algunos aspectos sobre los fundamentos de pruebas que se requieren tener claros para poder evaluar el proceso de pruebas.

1.4.2.1 Proceso de pruebas

La ejecución de pruebas es sólo una parte del proceso de pruebas, ya que el proceso incluye varias fases, las cuales se presenta en la Figura 1.5 [13]:

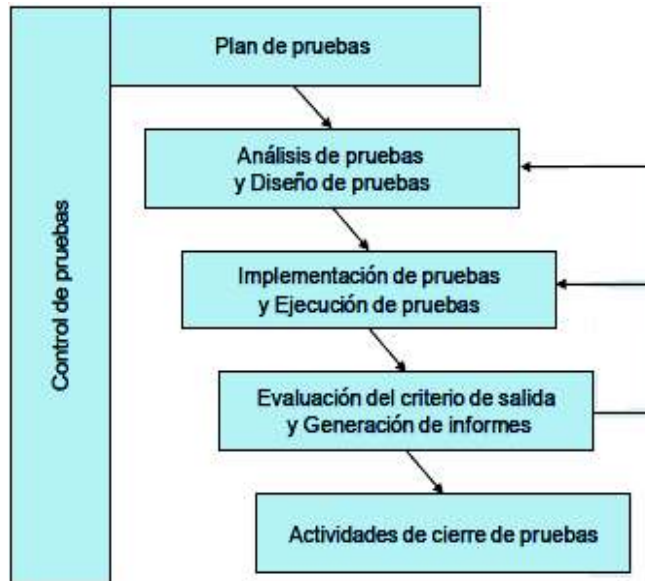


Figura 1.5 Proceso de Pruebas [13]

Las fases del proceso de pruebas se deben ejecutar de forma concurrente con las fases del proceso de desarrollo de software. El modelo V general es uno de los modelos de desarrollo de software más utilizado, en el cual se presenta el proceso de desarrollo y el proceso de pruebas en dos ramas iguales, cada nivel de desarrollo tiene su correspondiente nivel de pruebas como se indica a continuación en la Figura 1.6 [13]:

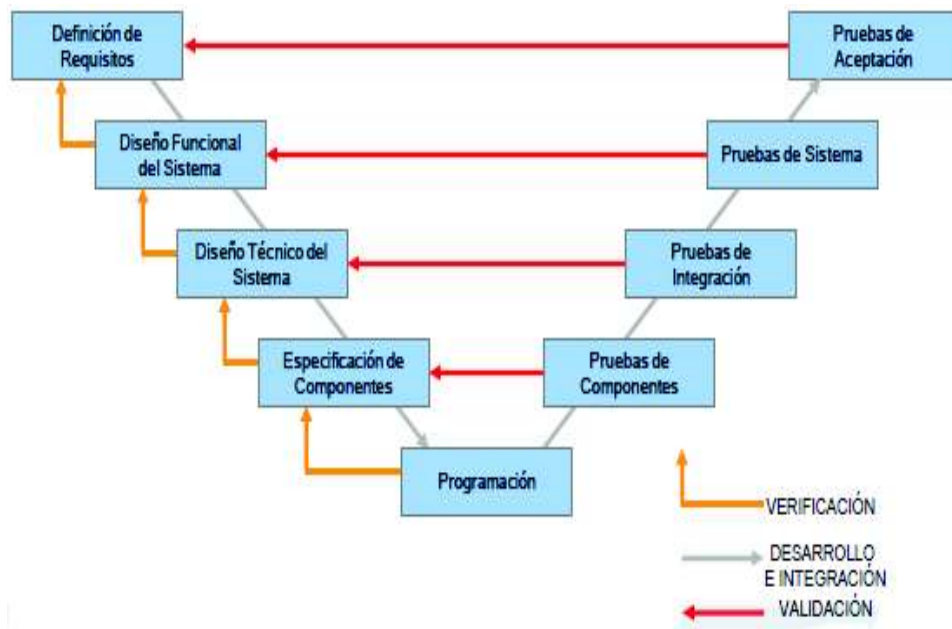


Figura 1.6 Modelo V general [13]

Las actividades que corresponden al proceso de pruebas tienen lugar a través del ciclo de vida completo del desarrollo del software.

En un modelo iterativo se presenta el proceso de desarrollo y el proceso de pruebas en forma de espiral, en el cual se indica que se debe realizar con respecto al proceso de pruebas en cada una de las iteraciones realizadas para desarrollo del producto software, como se indica a continuación en la Figura 1.7 [14]:

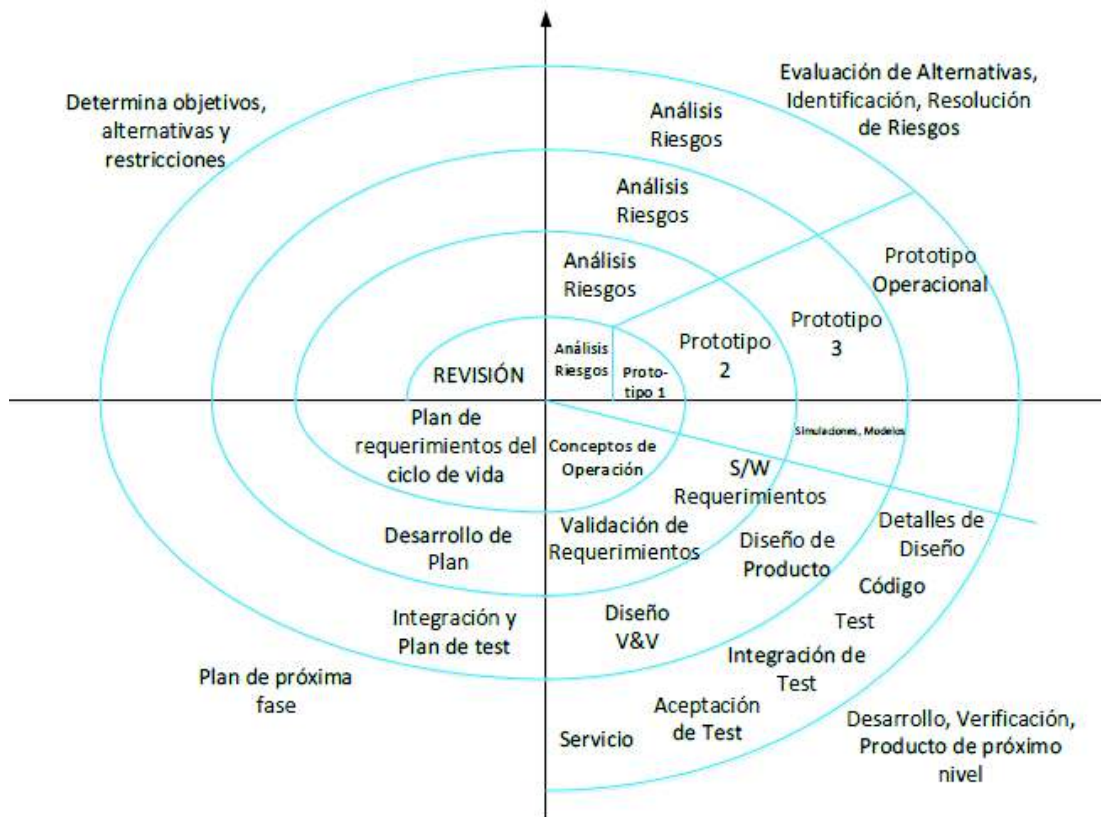


Figura 1.7 Modelo Iterativo [14]

1.4.2.2 Objetivos de las pruebas

El principal objetivo de las pruebas es la detección de defectos, a continuación, se listan algunos objetivos adicionales de las pruebas [13]:

- Generación de confianza respecto al nivel de calidad.
- Aportación de información para la toma de decisiones.
- Prevención de defectos.

1.4.2.3 Niveles de pruebas

Los diferentes niveles de pruebas se pueden ver en la Figura 1.8, que se muestra a continuación:

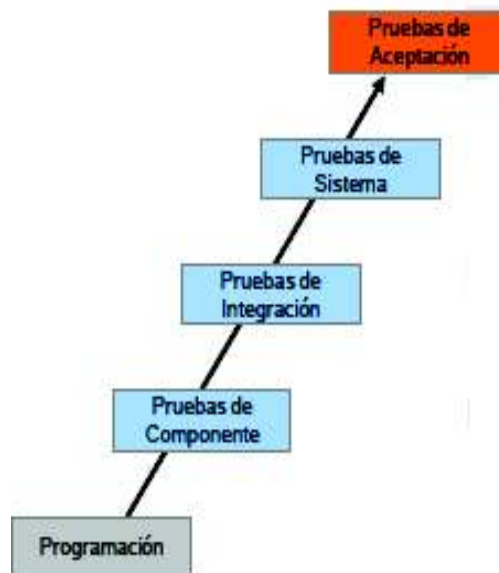


Figura 1.8 Niveles de Pruebas [13]

Definición de los niveles de pruebas [13]:

- a) **Pruebas de componentes:** conocidas también como pruebas unitarias, son pruebas de cada componente tras su construcción. Los componentes pueden ser módulos, clases o unidades. Las pruebas de componentes podrán comprobar características funcionales y no funcionales de un sistema.

- b) Pruebas de integración:** comprueban las funciones externas, la correcta interacción entre los componentes de distintos sistemas. El propósito de las pruebas de integración es detectar defectos en las interfaces. La integración ocurre de forma ascendente, descendente o en forma de Big Bang (Integrar todos los módulos y una vez juntos probar el sistema esperando la explosión de errores).
- c) Pruebas de sistema:** es el proceso de probar un sistema integrado con el objeto de comprobar el cumplimiento de requisitos especificados. Significa probar el comportamiento completo del sistema. Para ejecutar este tipo de pruebas se requiere elaborar casos de pruebas que podrán ser obtenidos a partir de:
- Especificaciones funcionales
 - Casos de uso
 - Procesos de negocio
 - Evaluación de riesgos
- d) Pruebas de aceptación:** es el proceso de verificar la adecuación al uso del sistema por parte de usuarios del negocio. Normalmente el cliente selecciona los casos de prueba para las pruebas de aceptación. Las pruebas de aceptación corresponden a las pruebas de sistema por parte del cliente.

1.4.2.4 Tipos de pruebas

Los diferentes tipos de pruebas se presentan a continuación [13]:

- a) Pruebas funcionales:** su objetivo es probar una función.
- b) Pruebas no funcionales:** su objetivo es probar las características del producto. Algunas pruebas no funcionales típicas son:
- Pruebas de carga
 - Pruebas de rendimiento
 - Pruebas de volumen
 - Pruebas de estrés
 - Pruebas de fiabilidad
 - Pruebas de robustez
 - Pruebas de usabilidad

- c) **Pruebas estructurales:** su objetivo es probar la estructura / arquitectura de un software. La finalidad de las pruebas es medir el grado en el cual la estructura del objeto de prueba ha sido cubierta por los casos de prueba.

- d) **Pruebas de asociadas al cambio:** su objetivo es probar el sistema después de implementarse cambios. Algunas pruebas asociadas al cambio típicas son:
 - Repetición de pruebas
 - Pruebas de regresión

Todos los tipos de pruebas se pueden llevar a cabo en todos los niveles de prueba.

1.4.3 Gestión de pruebas

La gestión de pruebas es la gestión del proceso de pruebas en los proyectos de desarrollo de software. Las actividades de la gestión de pruebas son necesarias a lo largo de todo el proceso de pruebas [13].

1.4.3.1 Seguimiento y estimación del estado de las pruebas

El seguimiento y monitoreo del estado de pruebas debe ser realizado en base a criterios medibles que aporten información necesaria para gestionar el proceso de pruebas, entre las cuales podemos mencionar [13]:

- Métricas en base a errores, por ejemplo: tasa de detección de errores, defectos detectados/corregidos, resultados de repetición de pruebas.
- Métricas en base a casos de prueba, por ejemplo: cobertura de casos de prueba, cobertura de requisitos, casos de prueba exitosos o fallidos, cobertura de código, cobertura de riesgo.
- Métricas en base a costos, por ejemplo: costo de detección de errores, costo de pruebas de regresión, costo de recursos externos.

Los resultados obtenidos de la medición aportan información al proyecto y a la dirección de la empresa respecto al proceso de las pruebas, por ello se deben presentar los informes periódicamente [13].

1.4.3.2 Métricas de calidad para el proceso de pruebas

La medición es una herramienta eficaz en las pruebas de software, es la base para detectar las desviaciones del rendimiento aceptable en los procesos y producto de software. Además, brinda la oportunidad de mejorar, identificar y priorizar las principales preocupaciones, dar seguimiento a la solución y mejorar la calidad del producto. Las mediciones permiten además cuantificar tanto el proceso como el producto [15]. En el proceso de prueba las mediciones pueden ser usadas para [15]:

- Monitorizar el proceso de prueba: Mostrar visibilidad sobre las actividades de pruebas.
- Reportar las pruebas: Métricas recolectadas al finalizar cada etapa de prueba para evaluar la adecuación de los objetivos de cada etapa.
- Controlar las pruebas: Acciones correctivas tomadas como el resultado de la información, las métricas tomadas y reportadas.

Métricas internas

Existe un grupo de métricas internas que, aunque son aplicadas fundamentalmente a los productos que se desarrollan a lo largo del ciclo de vida de desarrollo, las mismas pueden ser adaptadas para reportar el avance de la ejecución de las pruebas en evaluaciones realizadas al final del proceso de pruebas [15].

En la Tabla 1.1 se listan las métricas sobre el avance del proceso de pruebas durante la ejecución de las pruebas exploratorias [15]:

Tabla 1.1 Métricas sobre el avance del proceso en la ejecución de las pruebas exploratorias [15]

ELEMENTO	MÉTRICA	PROPONE MEDIR
Funcionalidad	Cubrimiento de funcionalidades	Cantidad de casos de prueba ÷ Total de funcionalidades a probar.
	Tiempo por funcionalidad	Total de tiempo en ejecutar las pruebas ÷ Total de funcionalidades probadas.
	Densidad de defectos por funcionalidad	Total de No conformidades (NC) encontrados ÷ Total de funcionalidades probadas.
Pruebas exploratorias	Esfuerzo de las pruebas exploratorias (PE)	Contar el tiempo dedicado a las PE en el ciclo de pruebas.
	NC encontradas en las PE	Contar las NC encontradas durante las PE.
	Funcionalidades exploradas	Contar las funcionalidades exploradas durante las PE del ciclo de prueba.
Esfuerzo	Esfuerzo planificado (5)	Sumar el tiempo planificado para todas las actividades del proyecto de prueba.
	Esfuerzo realizado en el ciclo de prueba	Sumar el tiempo invertido en todas las actividades del ciclo de prueba.
	Esfuerzo realizado hasta el momento (3)	Sumar el tiempo realizado desde que el proyecto comenzó hasta el último ciclo realizado.
	Esfuerzo por realizar (4)	Sumar el tiempo que se planifica para cada actividad hasta que el proyecto de prueba termine.
	Esfuerzo total (6)	Sumar (3) y (4).
	Desviación en el esfuerzo planificado	Restar (6) menos (5).
	Esfuerzo por persona	Contar el esfuerzo por persona del equipo de prueba.

1.4.3.3 Productividad del equipo de pruebas

Para obtener la calificación de los “Tester”, se requiere medir el conocimiento que éste tiene de la aplicación que va a probar. Por otra parte, se mide la experiencia de acuerdo a los años de experiencia en pruebas de software [16] . En la Figura 1.9 se muestra un modelo de calificación de los Tester:

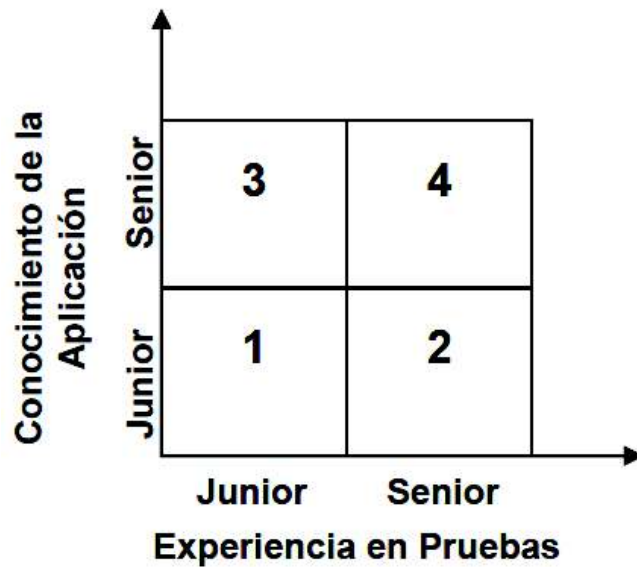


Figura 1.9 Modelo de calificación de los Tester [16]

Aplicando el método indicado en la figura anterior, la calificación de los Tester da una idea más real de las condiciones del equipo de pruebas [16].

1.4.4 Modelos de calidad

Un modelo de calidad es un conjunto de factores de calidad y la relación entre ellos, que proporcionan una base para la especificación de requisitos de calidad y para la evaluación de la calidad de los componentes software [17].

Para garantizar la calidad de software es importante implementar algún modelo o estándar de calidad que permita evaluar las características del producto de software desarrollado. Los modelos de evaluación de calidad apoyan a las organizaciones para tener una mejora continua y ser más competentes, para así poder medir la calidad y brindar productos de alto nivel [6].

A continuación, se presenta el enfoque que tienen los modelos de calidad de software y la evolución de los modelos de calidad a nivel de producto; estos temas son importantes para el desarrollo del presente trabajo.

1.4.4.1 Enfoques de los modelos de calidad de software

Los modelos de calidad de software se clasifican de acuerdo al enfoque de evaluación, de la siguiente manera [6]:

a) Calidad a nivel de producto

La calidad del producto de software permite especificar y evaluar el cumplimiento de criterios del producto, para lo cual se aplican métricas internas, externas y en uso.

b) Calidad a nivel de proceso

La calidad del proceso de software consiste en realizar un examen disciplinado usando un conjunto de criterios para determinar la capacidad del mismo, de manera que el proceso sea realizado dentro de los objetivos de calidad, costo y planificación.

1.4.4.2 Evolución de los modelos de calidad a nivel de producto

Existen diferentes normas, modelos, metodologías, guías y estándares de calidad que permiten evaluar los productos de software. A continuación, se presenta de forma cronológica algunos modelos que se consideran pioneros base del desarrollo de otros recientes [6].

En la figura 1.10 se presenta la línea de tiempo de algunos modelos de evaluación a nivel de producto:

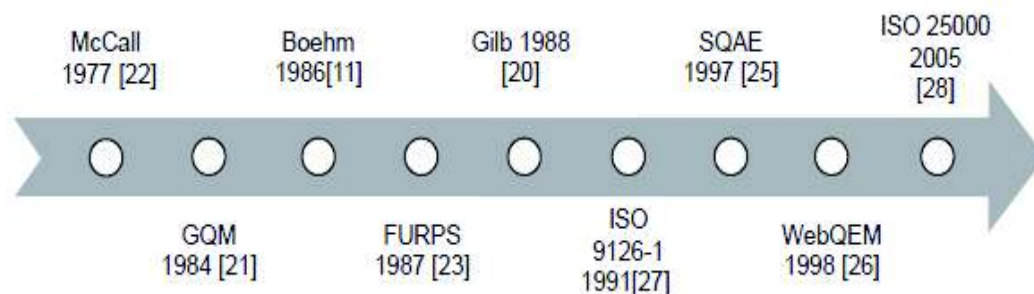


Figura 1.10 Modelos de calidad a nivel de producto [6]

Como podemos ver en la Figura 1.10 el modelo de McCall creado en el año de 1977 es uno de los modelos pioneros en la evaluación de la calidad de software. Por otro lado, el último modelo de evaluación es la norma ISO/IEC 25000 creada en el año 2005, la cual también es llamada como SQuaRE. Algunos modelos de calidad clásicos han servido de base para crear los modelos de calidad más recientes; lo que ha permitido que los nuevos modelos de calidad se consoliden en base a la evolución del software, para así optimizar los procesos organizaciones [6].

En este caso nos enfocaremos en modelos de calidad enfocados al producto, cuyo principal objetivo es especificar y evaluar el cumplimiento de criterios del producto, para lo cual se aplican métricas. Adicionalmente se presentará el listado de todas las métricas que se incluyen en las normas ISO/IEC 25010, 25022 y 25030 que serán usadas en el presente proyecto.

1.4.5 Modelos de calidad a nivel de producto

Los modelos de calidad más aceptados a nivel de producto de software son [18]:

- Modelo de McCall
- ISO/IEC 9126-1
- ISO/IEC 25000

De cada uno de los modelos se presentará una descripción de las características más relevantes, su estructura y objetivos. A continuación, se presenta el detalle de estos modelos de acuerdo al aporte que éstos pueden dar a los objetivos planteados en el presente trabajo.

1.4.5.1 Modelo de Calidad – McCall

El modelo de McCall es uno de los modelos pioneros en la evaluación de la calidad de software. El modelo, identifica tres factores que afectan a la calidad del software como se puede ver en la Figura 1.11. Estos factores se centran en aspectos importantes de un producto de software: sus características operativas, su capacidad de cambio y su adaptabilidad a nuevos entornos [4].



Figura 1.11 Factores de calidad de McCall [4]

El modelo de McCall se basa en 11 criterios de calidad los cuales son: Exactitud, confiabilidad, eficiencia, integridad, usabilidad, mantenibilidad, testeabilidad, flexibilidad, portabilidad, reusabilidad e interoperabilidad [4].

La desventaja que presenta el modelo McCall frente a los modelos ISO/IEC 9126-1 E ISO/IEC 25020 es que no diferencia entre calidad interna, externa y en uso, solamente se enfoca en el desarrollo de factores de calidad basado en tres ejes orientados al producto de software [6].

1.4.5.2 Estándar de Calidad – ISO/IEC 9126

La ISO/IEC 9126 es un estándar internacional para la evaluación de software y se basa en el modelo de McCall. Esta norma permite identificar los atributos clave de calidad para los productos de software [6].

1.4.5.2.1 División del Estándar ISO/IEC 9126

La ISO/IEC 9126 está articulada en varias partes como se muestra en la Figura 1.12 [6].

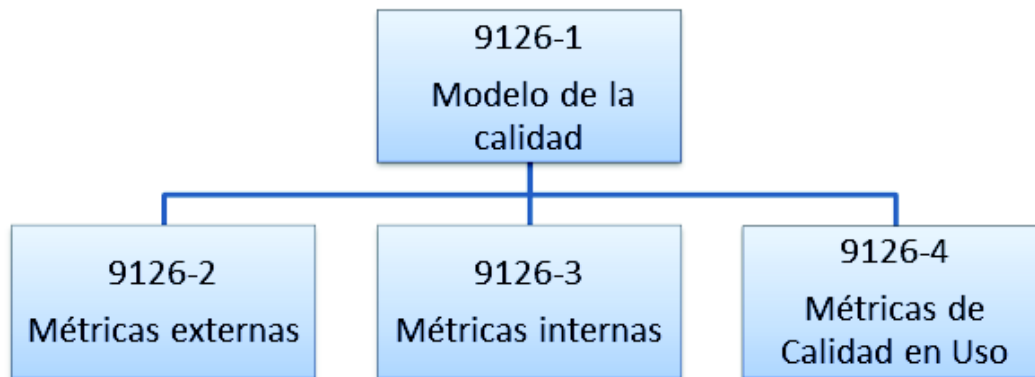


Figura 1.12 División de la Norma ISO/IEC 9126 [6]

En la Figura 1.12 se puede apreciar las tres partes de la norma ISO/IEC 9126 que están relacionadas a métricas de calidad. La información de esta sección del estándar contribuirá a cumplir el objetivo del presente trabajo. A continuación, se brinda una breve descripción de cada una de las partes indicadas en la Figura 1.12:

1. ISO/IEC 9126-1: Modelo de la calidad

La ISO/IEC 9126 - 1 clasifica a la calidad del software en un conjunto estructurado de características y subcaracterísticas. Esta parte se basa en 6 criterios de calidad los cuales son: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad, portabilidad [19].

2. ISO/IEC 9126-2: Métricas externas

La segunda parte de la norma ISO/IEC 9126 presenta el informe técnico de la norma referente a las métricas externas aplicadas al producto en ejecución. Este informe contiene: la terminología, el uso de las métricas durante el ciclo de vida del software, las métricas externas para cada característica y subcaracterísticas de calidad de software, interpretación de las medidas, y las propiedades de las métricas externas [20].

3. ISO/IEC 9126-3: Métricas internas

La parte 3 de la ISO/IEC 9126 presenta las métricas externas para evaluar los atributos de calidad definidos en la norma. Esta norma se aplicará a un producto de software no ejecutable y se aplicará durante la etapa de desarrollo [21].

4. ISO/IEC 9126-4: Métricas de calidad en uso

La parte 4 de la norma ISO/IEC 9126 da a conocer el concepto de calidad en uso, define las características de calidad asociadas a la calidad en uso y presenta las métricas asociadas a las características definidas. Esta parte se basa en 4 criterios de calidad los cuales son: efectividad, productividad, seguridad de acceso y satisfacción [22].

1.4.5.2.2 Relación de la ISO/IEC 9126 con la ISO/IEC 14598

El estándar ISO/IEC 14598 contiene requisitos generales y provee un marco de trabajo para evaluar la calidad de los productos de software, además establece requisitos para métodos de medición [23].

En la Figura 1.13 se muestra la forma en que se relaciona la ISO/IEC 9126 y la ISO/IEC 14598:

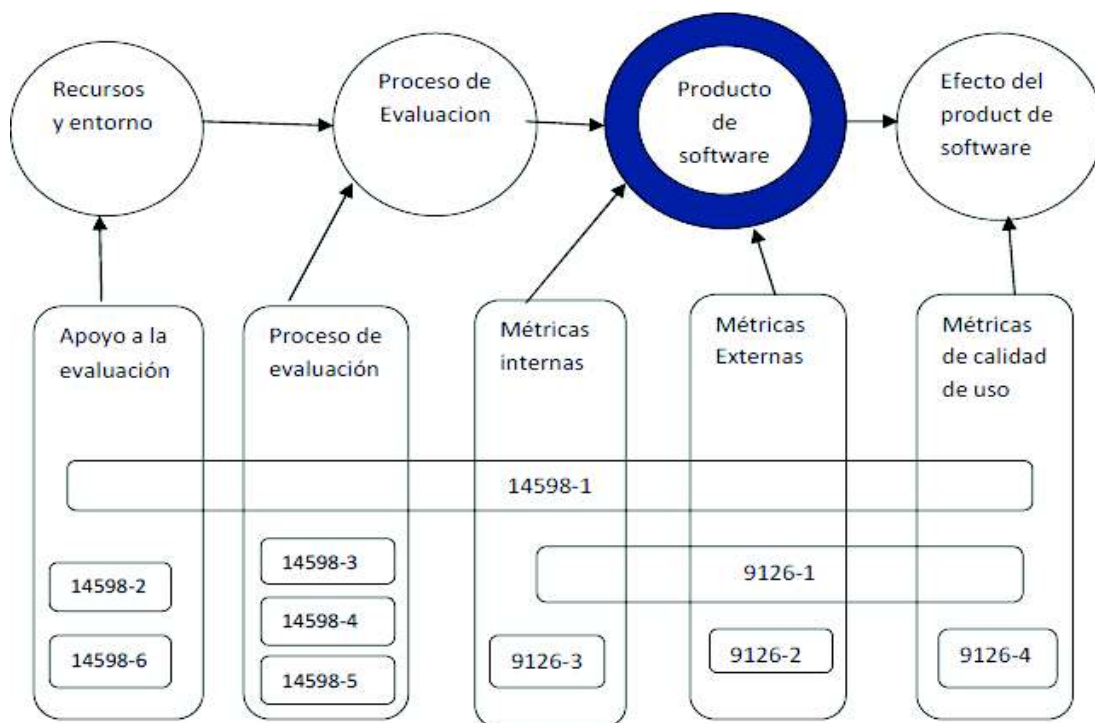


Figura 1.13: Relación normas ISO/IEC 9126 e ISO/IEC 14598 [19] [23]

La relación entre estas dos normas consiste en [19] [23]:

- La norma ISO/IEC 9126 debe ser aplicada en conjunto con la norma ISO/IEC 14598.

- La norma ISO/IEC 14598 define las actividades necesarias para analizar, especificar, diseñar y ejecutar las acciones requeridas para aplicar la norma ISO/IEC 9126-1.
- El modelo de evaluación de la ISO/IEC 9126-1 selecciona como modelo de calidad a la norma ISO/IEC 9126.

Las normas ISO/IEC 14598 y la ISO/IEC 9126 han sido incluidas en una nueva propuesta denominada proyecto SQuaRE (Requisitos y Evaluación de Calidad de Productos Software). Esta nueva propuesta está conformada por una familia de estándares que permiten la creación de un marco de trabajo común para evaluar la calidad del producto de software.

En la Figura 1.14 se puede observar la relación entre las normas ISO/IEC 9126, ISO/IEC 14598 e ISO/IEC 25000.

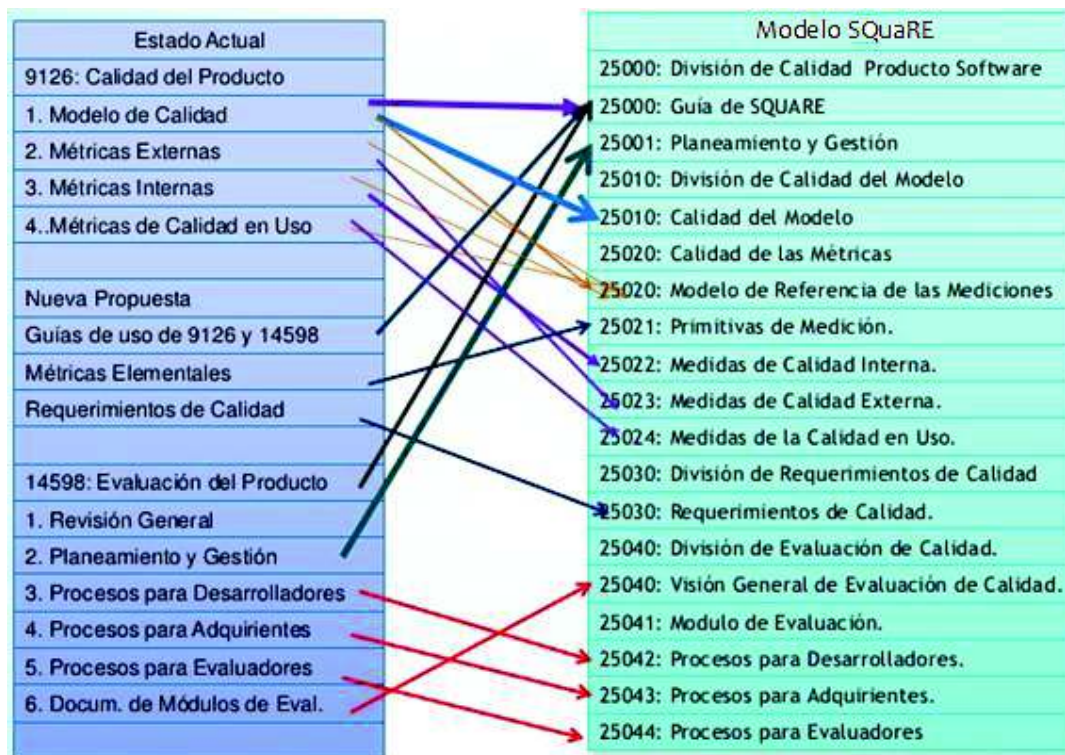


Figura 1.14: Relación entre ISO/IEC 9126, ISO/IEC 14598 e ISO/IEC 25000 [19] [23] [24]

En la Figura 1.14 se puede observar claramente como el modelo SQuaRE engloba las normas ISO/IEC 9126 y la ISO/IEC 14598. Así, este nuevo modelo nos permite evaluar la

calidad del producto de software considerando como base los modelos clásicos para evaluar la calidad.

1.4.5.3 Modelo de Calidad – ISO/IEC 25000

La norma ISO/IEC 25000 provee una guía para evaluar la calidad de los productos software. La norma es conocida como SQuaRE es un marco de trabajo común para evaluar la calidad de los productos software, sustituyendo a la ISO/IEC 9126 e ISO/IEC 14598. La norma 25000 se convierte de esta manera en la piedra angular de esta área en la Ingeniería de Software [24].

1.4.5.3.1 División de la Norma ISO/IEC 25000 [24]

La ISO/IEC 25000 está articulada en varias divisiones como se muestra en la Figura 1.15:

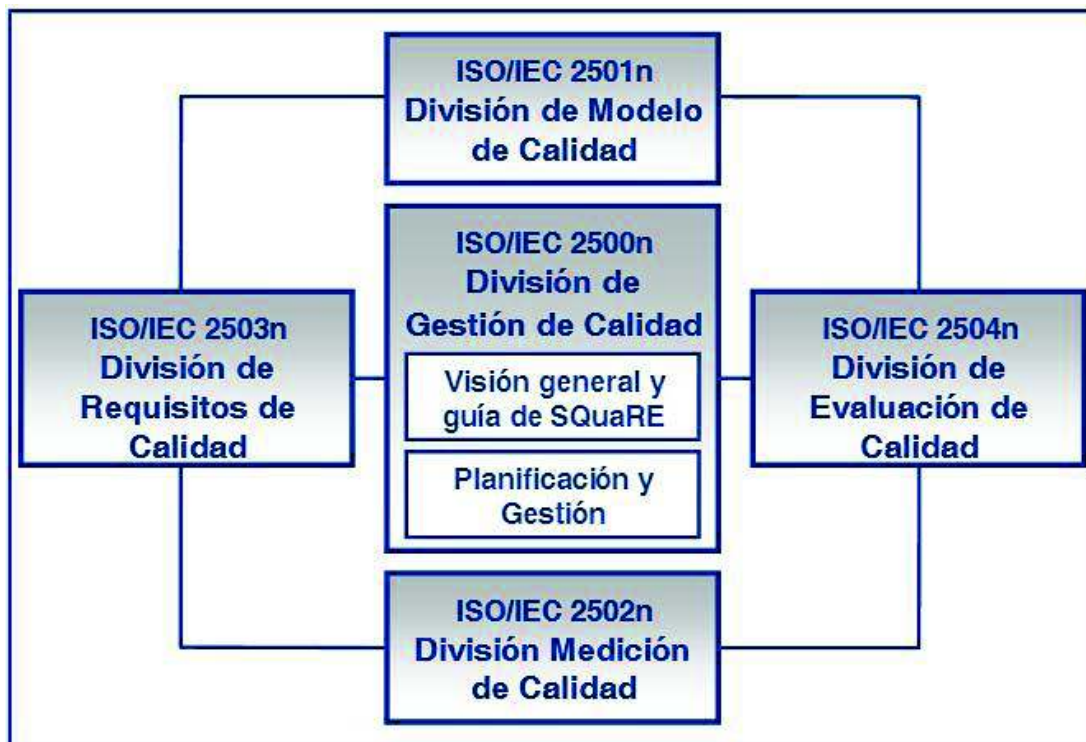


Figura 1.15 División de la Norma ISO/IEC 25000 [24]

En la Figura 1.15 se puede apreciar que una de las partes de la ISO/IEC 25000 está relacionada a la medición de calidad. La información de esta sección del estándar

contribuirá a cumplir el objetivo del presente trabajo; por este motivo se requiere una descripción detallada de la división indicada.

A continuación, se detallan cada una de las divisiones indicadas en la Figura 1.15:

1. ISO/IEC 2500n: Gestión de calidad.

Las normas que forman esta división definen modelos, términos y definiciones comunes, utilizadas por los demás estándares, las cuales son:

- ISO/IEC 25000: Guía de SQuaRE.
- ISO/IEC 25001: Planificación y Gestión.

2. ISO/IEC 2501n: Modelo de calidad.

Las normas que forman esta división presentan características para la calidad interna, externa y en uso del producto de software, las cuales son:

- ISO/IEC 25010: Modelo de calidad de software y sistema.
- ISO/IEC 25012: Modelo de calidad de datos.

3. ISO/IEC 2502n: Medición de la calidad.

Las normas que forma esta división, incluyen un modelo de referencia de la medición de la calidad del producto, métricas de calidad y guías para su aplicación, las cuales son:

- ISO/IEC 25020: Modelo de referencia de medición y guía.
- ISO/IEC 25021: Elementos de medida de la calidad.
- ISO/IEC 25022: Medidas de la Calidad en Uso.
- ISO/IEC 25023: Medidas de Calidad del Producto de Software.

4. ISO/IEC 2503n: Requisitos de calidad.

La norma que forma parte de esta división ayuda en la especificación de requisitos, la cual es:

- ISO/IEC 25030: Calidad de requerimientos.

5. ISO/IEC 2504n: Evaluación de calidad.

Las normas que forma parte de esta división, proporcionan requisitos, recomendaciones y guías para la evaluación del producto de software, las cuales son:

- ISO/IEC 25040: Modelo de referencia de evaluación y guía.

- ISO/IEC 25041: Guía de evaluación para desarrolladores, compradores y evaluadores independientes.
- ISO/IEC 25042: Módulos de evaluación.
- ISO/IEC 25045: Módulos de evaluación de la capacidad de recuperación.

1.4.5.3.2 Ciclo de vida de la calidad del Producto de Software [24]

El ciclo de vida de la calidad de un producto de software es similar al proceso de desarrollo, tal como se muestra en a Figura 1.16:

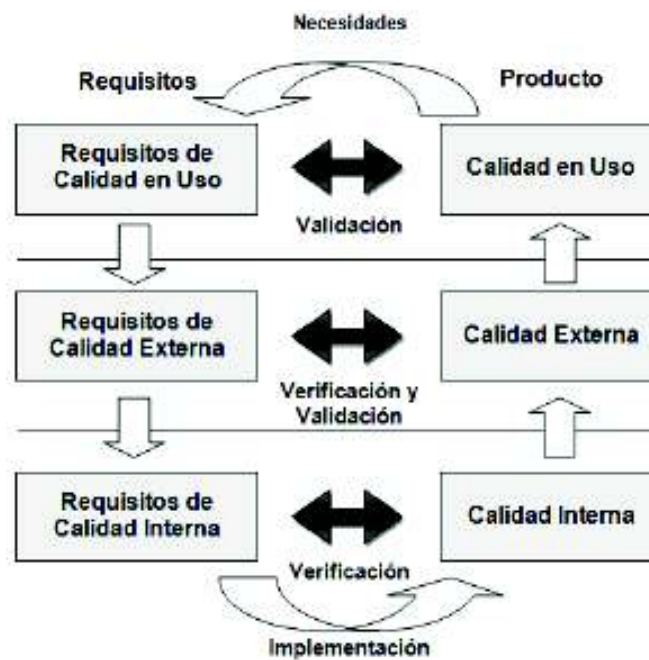


Figura 1.16 Ciclo de Vida de la Calidad del Producto de Software [24]

El ciclo de vida SQuaRE maneja la calidad del producto en tres fases, las cuales son:

- **Calidad Interna:** cuando el producto de software se encuentra en desarrollo.
- **Calidad Externa:** cuando el producto de software se encuentra en funcionamiento.
- **Calidad en Uso:** cuando el producto de software se encuentra en uso.

Para el presente trabajo nos centraremos en la calidad interna y externa de los productos de software.

1.4.5.3.3 Métricas para la calidad interna, externa y en uso - ISO/IEC 25022 e ISO/IEC 25023 [25] [26]

La norma ISO/IEC 25023 e ISO/IC 25022, proporcionan un conjunto de métricas de calidad, tanto para la calidad interna, externa y en uso. De acuerdo al contexto, estas métricas pueden modificarse e incluso es posible utilizar métricas que no estén definidas en la norma, siempre y cuando se especifique como la métrica se relaciona con la norma ISO/IEC 25010.

En la Figura 1.17 se puede observar la relación entre la ISO/IEC 2502n y la ISO/IEC 9126.

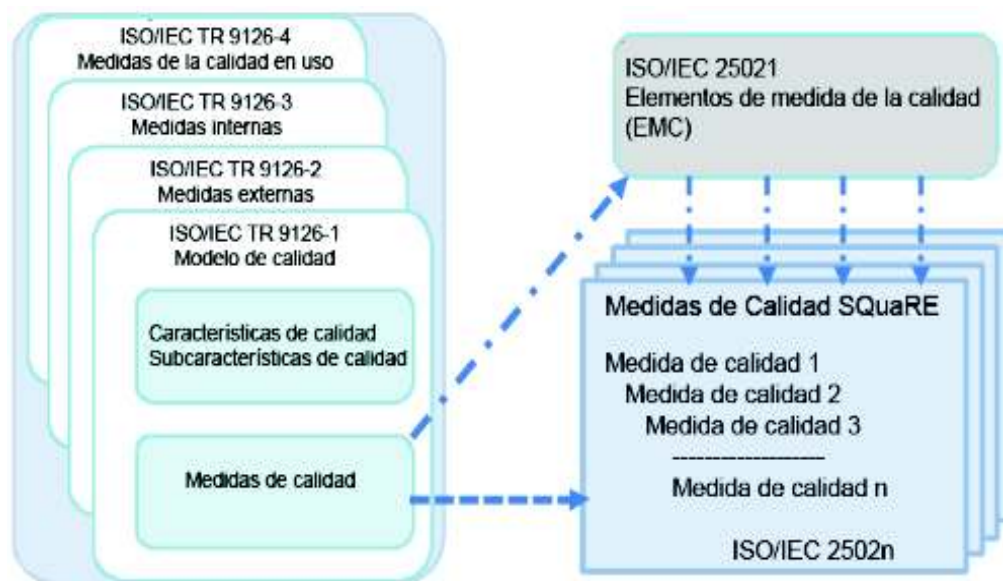


Figura 1.17 Relación entre la ISO/IEC 2502n e ISO/IEC 9126 [25] [26]

En la Figura 1.17 podemos notar que, para determinar las medidas de calidad más apropiadas a ser usadas, primero se requiere determinar las características y subcaracterísticas de calidad del producto de software a ser evaluado. Una vez identificadas las características del producto de software a ser evaluado se requiere determinar qué tipo de métricas de calidad se requiere usar.

En la Figura 1.18 se muestra la relación que existe entre las fases de calidad con sus respectivas métricas y la influencia que existen entre ellas.

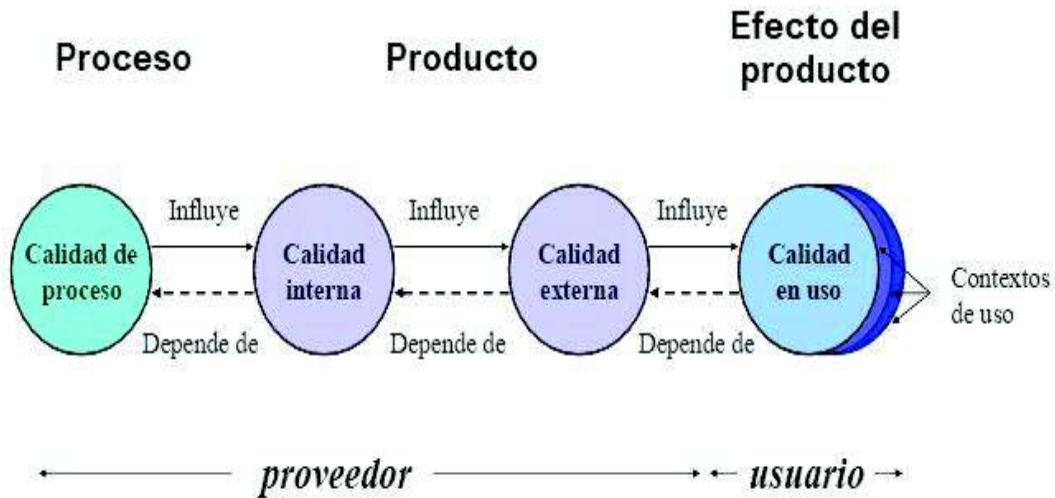


Figura 1.18 Relación entre los tipos de métricas de calidad [25]

Las métricas de calidad interna pueden ser aplicadas durante la etapa de desarrollo del producto de software, con el fin de poder identificar los problemas de calidad e iniciar las acciones correctivas lo más antes posible.

Las métricas de calidad externa pueden ser usadas solo durante las etapas de pruebas y en algunas etapas de operación. Las mediciones se deben llevar a cabo cuando el sistema de software este en ejecución.

Las métricas de calidad en uso tienen como objetivo medir si el sistema software satisface las necesidades específicas del usuario.

Las métricas para la calidad interna y externa se describen en la Tabla 1.2:

Tabla 1.2 Métricas para la Calidad Interna y Externa, adaptado de [27]

MÉTRICAS PARA LA CALIDAD INTERNA / EXTERNA		
CARACTERÍSTICAS	SUBCARACTERÍSTICAS	MÉTRICA
Adecuación funcional	Complejidad funcional	Complejidad de la implementación funcional.
	Exactitud funcional	Exactitud.
		Precisión computacional.
Fiabilidad	Madurez	Eliminación de errores.
		Cobertura de pruebas.
		Tiempo medio entre fallos.

	Disponibilidad	Tiempo de servicio.
		Tiempo medio de inactividad.
	Tolerancia a fallos	Prevención de fallas.
		Redundancia (componentes).
	Anulación de operación incorrecta.	
	Recuperabilidad	Tiempo medio de recuperación.
Eficiencia en el desempeño	Comportamiento temporal	Tiempo de respuesta.
		Tiempo de espera.
		Rendimiento.
	Utilización de recursos	Línea de código
		Utilización de CPU
		Utilización de la memoria.
		Utilización de los dispositivos de E/S.
	Capacidad	Número de peticiones online
		Número de accesos simultáneos.
Sistema de transmisión de ancho de banda.		
Facilidad de uso	Capacidad de reconocer su adecuación	Integridad de descripción.
		Capacidad de demostración.
	Capacidad de ser entendido	Funciones evidentes.
		Efectividad de la documentación del usuario o ayuda del sistema.
	Operatividad	Recuperabilidad de error operacional.
		Claridad de mensajes.
		Consistencia operacional.
		Posibilidad de personalización
	Protección contra errores del usuario	Verificación de entradas válidas.
		Prevención del usuario incorrecto.
Estética de la interfaz de usuario	Personalización de la apariencia de la interfaz del usuario.	
Accesibilidad técnica	Accesibilidad física.	
Seguridad	Confidencialidad	Capacidad de control de acceso.
		Encriptación de datos.
	Integridad	Prevención de corrupción de datos.
	No repudio	Utilización de firma digital.
	Responsabilidad	Capacidad de auditoría de acceso.
Autenticidad	Métodos de autenticación.	
Compatibilidad	Co-Existencia	Co-existencia disponible

	Interoperabilidad	Conectividad con sistemas externos.
		Capacidad de intercambiar de datos.
Mantenibilidad	Modularidad	Capacidad de condensación.
		Acoplamiento de clases.
	Reusabilidad	Ejecución de reusabilidad.
	Capacidad de ser analizado	Capacidad de pistas de auditoría.
		Diagnóstico de funciones suficientes.
	Capacidad de ser modificado	Complejidad ciclomática.
		Profundidad de herencia.
		Grado de localización de corrección de impacto.
		Complejidad de modificación.
	Capacidad de ser probado	Índice de éxito de modificación.
		Complejidad funcional de funciones de pruebas.
		Capacidad de prueba autónoma.
		Capacidad de reinicio de pruebas.
Portabilidad	Adaptabilidad	Adaptabilidad en entorno hardware.
		Adaptabilidad en entorno de software.
		Adaptabilidad en entorno organización.
	Capacidad de ser instalado	Eficiencia en el tiempo de instalación.
		Facilidad de instalación.
	Capacidad de ser reemplazado	Consistencia en la función de soporte al usuario.
		Inclusividad funcional.
		Uso continuo de datos.

Las métricas que evalúan la calidad en uso se describen en la Tabla 1.3:

Tabla 1.3 Métricas de Calidad en Uso, adaptado de [26]

MÉTRICAS DE CALIDAD EN USO		
CARACTERÍSTICAS	SUBCARATERÍSTICAS	MÉTRICA
Efectividad	Efectividad	Compleitud de la tarea.
		Efectividad de la tarea.
		Frecuencia de error.
Eficiencia	Eficiencia	Tiempo de la tarea.
		Tiempo relativo de la tarea.
		Eficiencia de la tarea.
		Eficiencia relativa de la tarea.
		Porcentaje productivo.
		Número relativo de las acciones del usuario.
Satisfacción	Utilidad	Nivel de satisfacción.
		Uso discrecional de las funciones.
		Porcentaje de quejas de los clientes
Libertad de riesgo	Libertad de riesgo económico	Retorno de la inversión (ROI)
		Tiempo para lograr el retorno de la inversión.
		Rendimiento relativo de negocios.
		Balanced Score Card.
		Tiempo de entrega.
		Ganancias para cada cliente.
		Errores con consecuencias económicas.
	Corrupción del software.	
	Libertad de riesgo de salud y seguridad	Frecuencia de problemas en la salud y seguridad del usuario.
		Impacto en la salud y seguridad del usuario.
Seguridad de las personas afectadas por el uso del sistema.		
Libertad del riesgo ambiental	Impacto ambiental.	
Cobertura de contexto	Compleitud de contexto	Compleitud de contexto
	Flexibilidad	Función flexible del diseño.

1.4.6 Metodologías de medición

Para mantener un control de cualquier proceso, proyecto o producto se requiere mantener un control exhaustivo de la situación en la que se encuentra. Para hacer más fácil la implementación de sistemas de medición, existen diferentes metodologías como por ejemplo GQM (Global Question Metric), GDSM (Goal - Driven Software Measurement) o PSM (Practical Software Measurement) entre otras.

Para este trabajo elegiremos la metodología GQM porque dadas las necesidades de información que requiere la empresa LOGICIEL y la naturaleza del presente trabajo, esta metodología se considera la más apropiada y práctica; ya que permite diseñar métricas teniendo en cuenta los objetivos perseguidos por la organización, por lo tanto, esta metodología nos ayudará en la implementación de métricas para evaluar el proceso de pruebas.

1.4.6.1 Metodología GQM

La metodología conocida por sus siglas GQM, cuyas siglas Meta-Pregunta-Métrica hacen referencia a los fundamentos de dicha metodología. La metodología fue definida por Basili y Weiss en 1984 y extendida posteriormente por Rombach en 1990 como resultado de años de experiencia práctica e investigación en la materia [28].

Esta metodología GQM ayuda a [29]:

- Alinear las métricas con las metas de la organización.
- Mejorar el proceso del software.
- Mejorar la calidad del producto.
- Definir mediciones del proceso y del producto.

La metodología GQM se puede aplicar a todo el ciclo de vida del producto, procesos, recursos y se puede alinear fácilmente con el ambiente organizacional. GQM permite identificar métricas útiles y significativas para la organización desde el punto de vista de los objetivos del negocio, además permite tomar decisiones adecuadas y elegir métricas que se relacionen a las metas más importantes de los problemas más urgentes [29].

- 4) **Preparar la recolección de datos:** Desarrollar mecanismos para la recolección de datos.
- 5) **Recolectar, Validar y Analizar los datos para la toma de decisiones:** Recoger, validar y analizar los datos en tiempo real, para proporcionar la realimentación de proyectos en una acción correctiva.
- 6) **Analizar los datos para el logro de los objetivos y el aprendizaje:** Analizar los datos, una vez alcanzada una meta, para determinar el grado de conformidad y hacer las recomendaciones para mejoras futuras.

Las principales características de los pasos que componen la metodología GQM se las puede ver en la Figura 1.20:

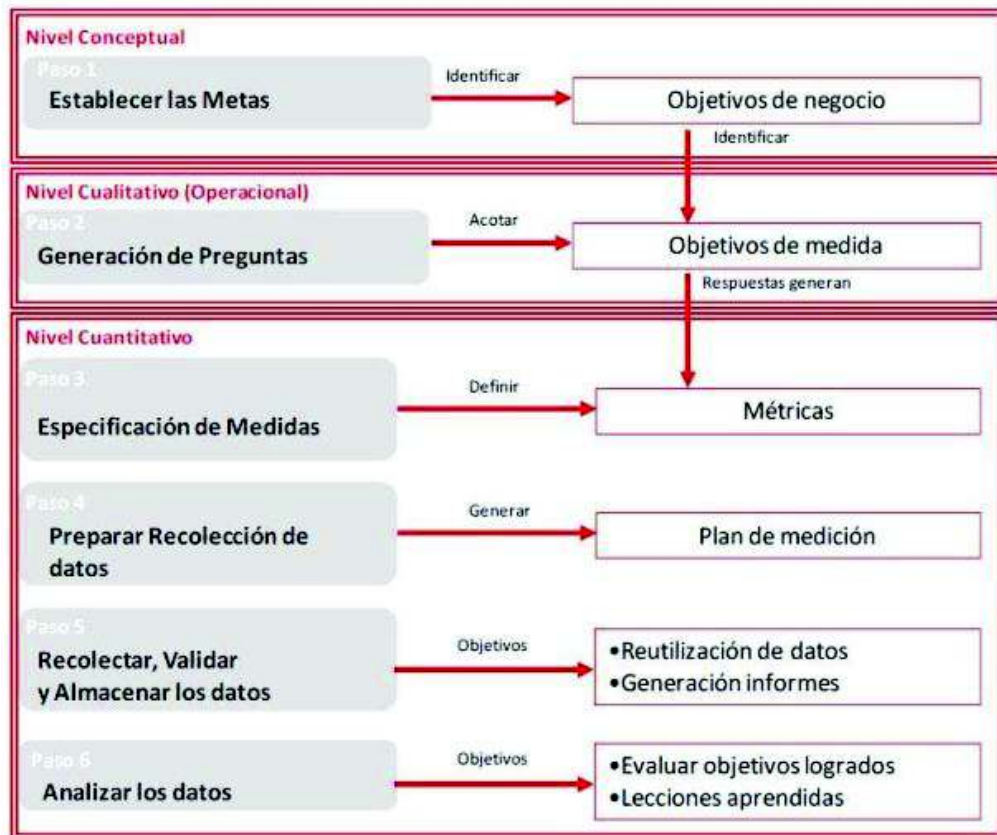


Figura 1.20 Proceso GQM [28]

Para implementar métricas usando la metodología GQM se usan esquemas en formato de tabla como se indica en la Tabla 1.4:

Tabla 1.4 Formato para definir métricas usando la metodología GQM [28]

Objetivo GQM	
Analizar	
Con el propósito de	
Con respecto a	
Desde el punto de vista de	
En el contexto de	
PREGUNTAS	
Pregunta 1	
Pregunta 2	
MÉTRICAS	
Métrica 1	
Métrica 2	

2 METODOLOGÍA

2.1 Introducción

Para lograr los objetivos de este proyecto se utilizará la metodología Investigación-acción, la cual será aplicada durante la evaluación del proceso de pruebas en proyectos de desarrollo de software para la empresa LOGICIEL.

Dicha metodología fue seleccionada porque presenta una estrategia de cambio práctico, ya que permite la participación de forma activa de todos los involucrados en el proceso a ser evaluado y mejorado; lo que permitirá encontrar una solución contextualizada.

En este capítulo se presente una visión general de la metodología y el detalle de cómo se ha ido aplicando cada una de las fases de la metodología durante el desarrollo del presente proyecto.

2.2 Visión general de la metodología

La investigación-acción es un proceso cognitivo que depende de la interacción social entre los observadores y su entorno, por lo tanto, esta metodología implica una participación colaborativa. La metodología de investigación-acción, es reconocida como una estrategia de cambio práctico, debido a que participan de forma activa todos los involucrados en el proceso a ser analizado y mejorado, lo que permite encontrar una solución contextualizada [30].

La metodología persigue cuatro objetivos, los cuales son [30]:

1. Proponer un enfoque en el cual el conocimiento científico es la base para la conceptualización y el desarrollo de un artefacto.
2. Encontrar nuevos conocimientos teóricos mediante el uso del artefacto en una organización en particular.
3. Permitir a los practicantes resolver problemas.
4. Brindar orientación para integrar los conceptos con los principios de intervención de la Investigación- acción.

En la Figura 2.1 se muestra el modelo canónico de la metodología Investigación-acción.

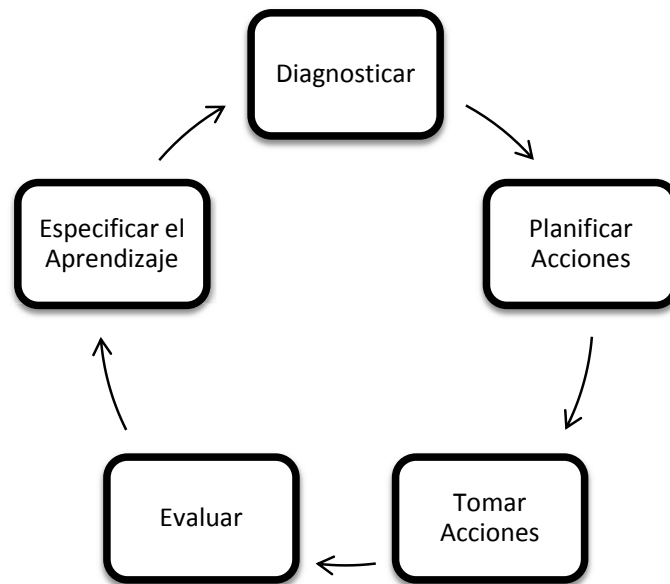


Figura 2.1 Modelo canónico de la metodología Investigación-Acción [31]

La metodología investigación–acción que se presenta en la Figura 2.1, plantea las siguientes etapas: diagnosticar, planificar acciones, tomar acciones, evaluar y especificar el aprendizaje. Esta metodología sigue un proceso iterativo y reflexivo [31]. A continuación, se presenta una descripción de cada una de las fases de la metodología [32]:

a) Diagnosticar

El diagnóstico corresponde a la identificación de los problemas principales que son las causas subyacentes del deseo de la organización para el cambio. Implica un análisis conjunto de la situación social por el investigador y los sujetos de la investigación. Las teorías se formulan en relación con la naturaleza del campo de investigación. Este diagnóstico desarrolla ciertos supuestos teóricos (es decir, una hipótesis de trabajo) sobre la naturaleza de la organización y su dominio del problema.

b) Planificar Acciones

Esta actividad especifica acciones organizacionales que se deberían resolver o mejorar los problemas identificados. El desarrollo de las acciones planificadas se guía por un marco teórico. Este marco se escoge en función de algún estado futuro

deseado para la organización, tanto como de los cambios que lograrían tal estado. Así, el plan establece el objetivo para el cambio y el enfoque para el cambio.

c) Tomar Acciones

Implementación de las acciones planificadas. Los investigadores y los profesionales colaboran en la intervención activa en la organización del cliente. Se pueden adoptar varias formas de estrategia de intervención, la intervención puede ser directiva, en la cual la investigación "dirige" el cambio; o indirecta.

d) Evaluar

Una vez completadas las acciones, los investigadores y profesionales colaboradores evalúan los resultados. La evaluación incluye determinar si los efectos teóricos de la acción se realizaron y si estos efectos resolvieron los problemas. Cuando el cambio fue exitoso, la evaluación debe cuestionarse críticamente para determinar cuál fue la única causa del éxito. Cuando el cambio no tuvo éxito, se debe establecer un marco para la próxima iteración del ciclo de investigación-acción (que incluya el ajuste de las hipótesis).

e) Especificar el Aprendizaje

Consisten en la identificación de nuevos conocimientos para la comunidad científica como resultado del éxito o fracaso de las acciones tomadas. Si bien la actividad de especificar el aprendizaje se realiza formalmente al final, generalmente es un proceso continuo.

El ciclo de investigación-acción puede continuar, tanto si la acción fue exitosa como si no, para desarrollar más conocimiento sobre la organización y la validez de los marcos teóricos relevantes. Como resultado de los estudios, la organización aprende más sobre su naturaleza y entorno, y la constelación de elementos teóricos de la comunidad científica continúa beneficiándose y evolucionando.

La metodología presenta cuatro características principales las cuales son [31]:

- Apunta a una mayor comprensión de la situación social inmediata.
- Asiste simultáneamente en resolución de problemas prácticos y amplía el conocimiento científico.
- Se lleva a cabo en colaboración y mejora de las competencias de los respectivos involucrados.

- Es principalmente aplicable para la comprensión de los procesos de cambio en los sistemas sociales.

2.3 Aplicación de la metodología

El proyecto comenzará con el diagnóstico del proceso de pruebas que tiene actualmente la empresa LOGICIEL con el fin de identificar los problemas que presenta el mismo. A continuación, se definirán objetivos, procedimientos, propiedades, datos de entrada y salida requeridos para la creación de métricas. Posteriormente se implementarán las métricas de acuerdo a los parámetros definidos en la etapa de planificación.

Luego, se procederá a recopilar los datos de tres proyectos de desarrollo de software de la empresa LOGICIEL. Sobre los datos recopilados se aplicarán las métricas implementadas y se evaluarán los resultados obtenidos. Finalmente, se identificarán los nuevos conocimientos obtenidos durante el desarrollo del proyecto.

2.3.1 Diagnosticar

2.3.1.1 Definición del Problema

Actualmente la empresa LOGICIEL cuenta con un proceso de control de calidad basado en CMMI nivel 2 para los proyectos de desarrollo de software. El proceso de pruebas fue definido en el año 2010 cuando obtuvo la certificación de calidad de procesos de IT Mark. Desde esa fecha hasta la actualidad, el proceso no ha sido ni evaluado ni mejorado; lo que provoca muchos problemas que afectan directamente a la calidad del producto final. Entre los principales problemas podemos mencionar: mala planificación de las pruebas, baja detección de defectos durante las pruebas de integración y de sistema, falta de tiempo para terminar de ejecutar las pruebas, falta de tiempo para la corrección de todos los defectos encontrados durante el proceso de pruebas, entre otros. Esto se pudo evidenciar en la reunión que se mantuvo con la gerente del área de QA, los Tester y analistas de pruebas de la empresa.

Para diagnosticar cual es el problema fundamental dentro del proceso de pruebas de la empresa LOGICIEL, se han aplicado entrevistas al personal de la organización de diferentes áreas y cargos; con el fin de obtener información relevante para poder evaluar el proceso.

2.3.1.2 Recolección de datos

Antes de realizar las entrevistas se realizó un análisis previo de los temas que se quieren tratar en las entrevistas y los roles que intervendrán en este proceso. En base a este análisis, se elaboró una Guía para las entrevistas la cual se presenta en la Tabla 2.1:

Tabla 2.1 Guía para entrevistas, elaborado por: Caiza Gabriela

Nº	PREGUNTAS	Gerente General	Gerente de Desarrollo	Gerente de Implantación	Gerente de QA	Lider de QA	Tester
IMPORTANCIA DEL PROCESO DE PRUEBAS							
1	¿Cuáles son los objetivos definidos para el proceso de control de calidad en la empresa?	X	X	X	X	X	X
2	¿Qué necesidades entorno al control de calidad tiene la empresa?	X	X	X	X	X	
3	¿Por qué cree usted que es importante el proceso de pruebas en la empresa?				X	X	X
MEJORAS PARA EL PROCESO DE PRUEBAS							
4	¿Considera usted que el proceso actual de pruebas está ayudando a mejorar la calidad de los productos desarrollados? SI/NO porque.				X	X	X
5	¿Cómo creen usted que podemos mejorar el proceso de pruebas?	X	X	X	X	X	X
ASPECTOS A EVALUAR DEL PROCESO DE PRUEBAS							
6	¿Qué información es importante para usted sobre el proceso de pruebas?	X	X	X	X	X	
7	¿Qué quisiera evaluar del proceso de pruebas?				X	X	
8	¿Qué porcentaje de pruebas considera usted que deberían haberse ejecutado para dar como finalizada la fase de pruebas cuando no hay el tiempo necesario para terminar de ejecutar todas las pruebas planificadas? Porque				X	X	
9	¿Cómo determinaría el número de casos de pruebas requeridos para un proyecto?				X	X	X
10	¿Cómo determinaría el tiempo requerido para ejecutar todos los casos de pruebas planificados para un proyecto?				X	X	X
11	¿Cuántos casos de prueba ejecuta diariamente?				X	X	X
12	¿Cuántos defectos detecta a diario durante las pruebas que realiza?				X	X	X
CONOCIMIENTO DEL PROCESO DE PRUEBAS							
13	¿Conoce usted cuales son las actividades o fases del proceso de pruebas?		X	X	X	X	X
14	¿Conoce usted que niveles de pruebas se aplican en los proyectos de desarrollo e implantación?		X	X	X	X	X
15	¿Conoce usted que tipos de pruebas se aplican en los proyectos de desarrollo e implantación?		X	X	X	X	X
16	¿Considera usted que es necesario realizar casos de prueba antes de ejecutar las pruebas? Si/No porque				X	X	X
17	¿Cómo priorizaría las pruebas en un proyecto?				X	X	X
18	¿Cree usted que es importante conocer el objetivo de las pruebas antes de ejecutarlas?				X	X	X
19	¿Usted lleva un reporte de los defectos encontrados y sus correcciones? Si/No Porque				X	X	X
20	¿Qué le resulta a usted complicado al momento de ejecutar las pruebas?				X	X	X

En el ANEXO I se puede consultar las entrevistas con toda la información recopilada.

2.3.1.3 Análisis de datos

A continuación, con el fin de poder interpretar de forma adecuada la información recopilada en las entrevistas, se definió una Guía de codificación la cual se presenta en la Tabla 2.2:

Tabla 2.2 Guía de codificación, elaborado por: Caiza Gabriela

CÓDIGO	TEMA	SUBTEMAS	PREGUNTAS - ENTREVISTA	
PRI - 01	PROCESO DE PRUEBAS	IMPORTANCIA DEL PROCESO DE PRUEBAS	¿Cuáles son los objetivos definidos para el proceso de control de calidad en la empresa?	
PRI - 02			¿Qué necesidades entorno al control de calidad tiene la empresa?	
PRI - 03			¿Por qué cree usted que es importante el proceso de pruebas en la empresa?	
PRM - 01		MEJORAS PARA EL PROCESO DE PRUEBAS	¿Considera usted que el proceso actual de pruebas está ayudando a mejorar la calidad de los productos desarrollados? Si/NO porque	
PRM - 02			¿Cómo creería usted que podemos mejorar el proceso de pruebas?	
PRA - 01		ASPECTOS A EVALUAR DEL PROCESO DE PRUEBAS	¿Qué información es importante para usted sobre el proceso de pruebas?	
PRA - 02			¿Qué quisiera evaluar del proceso de pruebas?	
PRA - 03			¿Qué porcentaje de pruebas considera usted que deberían haberse ejecutado para dar como finalizada la fase de pruebas cuando no hay el tiempo necesario para terminar de ejecutar todas las pruebas planificadas? Porque	
			¿Cómo determinaría el número de casos de pruebas requeridos para un proyecto?	
PRA - 04			¿Cómo determinaría el tiempo requerido para ejecutar todos los casos de pruebas plantificados para un proyecto?	
PRA - 05			¿Cuántos casos de prueba ejecuta diariamente?	
PRA - 06			¿Cuántos defectos detecta a diario durante las pruebas que realiza?	
PRC - 01			CONOCIMIENTO DEL PROCESO DE PRUEBAS	¿Conoce usted cuales son las actividades o fases del proceso de pruebas?
PRC - 02				¿Conoce usted que niveles de pruebas se aplican en los proyectos de desarrollo e implantación?
PRC - 03				¿Conoce usted que tipos de pruebas se aplican en los proyectos de desarrollo e implantación?
PRC - 04				¿Considera usted que es necesario realizar casos de prueba antes de ejecutar las pruebas? Si/No porque
PRC - 05		¿Cómo priorizaría las pruebas en un proyecto?		
PRC - 06		¿Cree usted que es importante conocer el objetivo de las pruebas antes de ejecutarlas?		
PRC - 07		¿Usted lleva un reporte de los defectos encontrados y sus correcciones? Si/No Porque		
PRC - 08		¿Qué le resulta a usted complicado al momento de ejecutar las pruebas?		

Al aplicar la guía de codificación sobre la información recopilada en las entrevistas obtuvimos los verbatines más importantes para cada uno de los temas base de las entrevistas.

En la Tabla 2.3 se presenta el resumen de la codificación realizada sobre las entrevistas.

Tabla 2.3 Codificación de las entrevistas, elaborado por: Caiza Gabriela

TEMA	SUBTEMA	NÚMERO DE VERBATINES
Proceso de Pruebas	Importancia del Proceso de Pruebas	8
	Mejoras para el proceso de pruebas	7
	Aspectos a Evaluar del proceso de pruebas	15
	Conocimiento del proceso de pruebas	8

En el ANEXO II se encuentra la codificación completa de las entrevistas realizadas a los empleados de la empresa LOGICIEL.

2.3.1.4 Resultados principales

En base a los resultados obtenidos de la codificación de las entrevistas se procedió a realizar una síntesis de los mismos tomando como base el tema del presente trabajo. Los resultados se detallan en la Tabla 2.4:

Tabla 2.4 Conclusiones de las entrevistas realizadas, elaborado por: Caiza Gabriela

TEMA	TIPO DE MÉTRICAS	VERBATIN MÁS REPRESENTATIVO	CONCLUSIÓN	SINTESIS	NECESIDAD
PROCESO DE PRUEBAS	Métricas relacionadas al producto	<p>“Entregar los productos software a los clientes con el menor número de errores. Realizando trabajo en conjunto con desarrollo y pruebas”</p> <p>“A nivel del proyecto sé que el objetivo de las pruebas es detectar el mayor número de defectos a fin de que poder entregar un producto de calidad”.</p> <p>“Evitar el mayor número de fallos en las aplicaciones mediante un plan de pruebas en conjunto con la planificación del proyecto “</p> <p>“Los líderes de proyectos siempre requieren que durante las pruebas internas se trate de detectar el mayor número de defectos para que en las pruebas de aceptación no se detecten defectos muy graves que sean difíciles de corregir”.</p>	<p>El objetivo del proceso de pruebas es detectar el mayor número de defectos durante el proceso de pruebas para contribuir a entregar un producto de calidad al cliente</p>	<p>Un objetivo debe ser medible para poder cumplirlo por lo tanto se requiere tener una estadística del número de defectos detectados durante el proceso de pruebas de cada uno de los proyectos desarrollados que ayuden a la toma de decisiones.</p>	<p>Se requiere saber el número de defectos detectados y corregidos durante el proceso de pruebas.</p>
	Métricas relacionadas al producto	<p>“El proceso de pruebas debe evaluar la calidad interna, la calidad externa y la calidad en uso de los productos desarrollados”.</p> <p>“Se requiere establecer estándares para evaluar la interfaz de usuarios, la navegabilidad y la facilidad de uso de los productos desarrollados”.</p> <p>“El proceso de pruebas debe facilitar el mantenimiento de las aplicaciones”.</p>	<p>La empresa requiere que el proceso de pruebas ayude a evaluar la calidad interna y la calidad externa de los productos desarrollados.</p>	<p>Para evaluar los productos software desarrollado se requiere definir las características de calidad a evaluar y en base a ello seleccionar las métricas más adecuadas. Las características de calidad</p>	<p>Se requiere saber el número de requerimientos implementados</p> <p>Se requiere saber si todos los requerimientos implementados han sido probados.</p>

		<p>“Garantizar el producto/servicio entregado al cliente, de manera que se cumpla sus expectativas”.</p> <p>“Certificar el producto que se entrega al cliente. El producto es confiable y de calidad”.</p> <p>“Es importante para asegurar que el software que se provee ayude a los usuarios finales con el menor impacto de errores”.</p>		<p>importantes para la empresa son: adecuación funcional, fiabilidad, facilidad de uso y mantenibilidad,</p>	<p>Se requiere evaluar la facilidad de uso de los productos desarrollados relacionados a los siguientes aspectos: -Facilidad de ser probado -Operatividad -Protección contra errores de usuario</p> <p>Se requiere evaluar la mantenibilidad de los productos desarrollados relacionada a los siguientes aspectos: -Capacidad de ser modificado -Capacidad de ser probado</p>
<p>Métricas relacionadas a los Tester asignados al proceso de pruebas</p>	<p>“El proceso de pruebas puede mejorar teniendo un equipo con mayor experiencia y conocimiento de las reglas de negocio”.</p> <p>“Actualmente se han presentado muchas quejas por parte de los diferentes líderes de proyecto indicando que no se están realizando las pruebas correctamente y que se están detectando muchos defectos durante las pruebas de aceptación. Pienso que esto se debe a que los Tester asignados a cada proyecto no se les involucra desde las etapas tempranas al proyecto, y solo se les toma en cuenta al momento de ejecutar las pruebas; en ese momento ellos no conocen claramente todo el Core del negocio.</p>	<p>Para mejorar el proceso de pruebas se requiere evaluar el conocimiento del equipo de pruebas tanto en el proceso de pruebas como en las reglas del negocio del proyecto en el cual va a participar.</p>	<p>Se requiere evaluar el grado de conocimiento que tiene un Tester con respecto a la aplicación que va a probar y la experiencia que tiene en pruebas, con el fin de poder realizar comparaciones y tener información que permita tomar decisiones.</p>	<p>Se requiere saber el grado de experiencia que tiene el Tester en pruebas y el grado de conocimiento que tiene respecto a la aplicación que va a probar.</p>	

		<p>Pienso que hay varios aspectos que se deben mejorar en las pruebas como, por ejemplo:</p> <ul style="list-style-type: none"> -Incluir a los Tester desde etapas tempranas en el proyecto -Capacitar a los Tester para que tengan claro todo el proceso de pruebas". "La rotación del personal, priorización de determinados proyectos y manejo de cambios". "Hay proyectos que tienen 7 personas o más mientras que otros tienen 3 o más, es necesaria una mejor distribución". "La lógica del negocio a veces es complicada e involucra muchas variables, lo que da pie a cometer errores en el diseño y ejecución de los casos de prueba". 			
<p>Métricas relacionadas al cronograma del proceso de pruebas</p>	<p>"En la mayoría de productos se prueba sin seguir todo el proceso por falta de tiempo, o participación de QA desde etapas iniciales del proyecto".</p> <p>"Existen en ciertos proyectos un tiempo muy corto para poder realizar pruebas".</p> <p>"Integrando al equipo de QA a tiempo en los proyectos.</p> <p>"Capacitación".</p> <p>"Proactividad de cada recurso".</p> <p>"Porcentaje de desviación entre lo planificado y ejecutado para que en futuras planificaciones se estime más acertadamente".</p>	<p>Para mejorar el proceso se requiere tener una estadística sobre el tiempo asignado para el proceso de pruebas en los diferentes proyectos con el fin de poder tener una base para mejorar los tiempos asignados al mismo.</p>	<p>Se requiere saber el tiempo planificado para el proceso de pruebas versus el tiempo real invertido para el proceso en los diferentes proyectos de desarrollo.</p>	<p>Se requiere saber si el tiempo planificado para el proceso de pruebas es el adecuado o se requiere ampliarlo.</p>	

	<p>Métricas relacionadas a los defectos detectados durante el proceso de pruebas</p>	<p>“Requiero saber del proceso de pruebas: -Estadísticas por proyectos de los diferentes tipos de errores y como se corrigieron -Estadísticas por proyecto sobre el número de errores detectados con el objetivo de identificar que está pasando con los desarrolladores del proyecto con el mayor número de errores; y poder hacer algo para mejorar desde el origen de la introducción de defectos”. “Quisiera saber lo siguiente: -Porcentaje de defectos detectados”. “Cuando me asignan el 100% a ejecutar solo pruebas, el mayor número de defectos que he detectado más o menos son unos 4 defectos diarios”. “Si llevo un reporte de defectos para garantizar que hayan sido corregidos y verificados”. “Sí, es importante tener un registro de las correcciones aplicadas, para proveer información de la corrección en caso que el defecto se vuelve a presentar”.</p>	<p>Para la empresa es importante tener información sobre los defectos detectados durante el proceso de pruebas con el fin de poder tomar decisiones oportunas.</p>	<p>Para evaluar los defectos detectados durante el proceso de pruebas se requiere priorizarlos y llevar un control de su registro y corrección de los mismos.</p>	<p>Se requiere saber si los defectos detectados durante el proceso de prueba han sido priorizados.</p>
					<p>Se requiere saber si todos los defectos priorizados con gravedad crítica y alta han sido corregidos</p>
	<p>Métricas relacionadas a los casos de prueba</p>	<p>“Quisiera saber lo siguiente: -Número de casos de pruebas diseñados por proyecto. -Porcentaje de casos de pruebas ejecutados</p>	<p>La empresa requiere tener información confiable relacionada a los casos de prueba diseñados y</p>	<p>Para obtener información adecuada de los casos de prueba y poder llevar un control de los mismos</p>	<p>Se requiere saber si los casos de prueba diseñados cubren todos los requerimientos a ser implementados.</p>

		<p>-Cuántas veces se ha probado un caso de pruebas probados y exitosos.</p> <p>-casos de pruebas probados y fallidos”.</p> <p>“Cuando me asignan el 100% a ejecutar solo pruebas, he ejecutado más o menos unos 3 casos de prueba grandes”.</p> <p>“Es importante crear casos de prueba sobre todo cuando los Tester no tienen experiencia”.</p> <p>“Pienso que los casos de prueba son el punto de partida para poder ejecutar correctamente una prueba”.</p> <p>“Al realizar un caso de pruebas se especifica y limita la característica o funcionalidad a probar”.</p> <p>“Los casos de prueba los podría priorizar tomando en cuenta las reglas de negocio, identificar los puntos de que son críticos para el proyecto”.</p> <p>“Priorizaría las pruebas de acuerdo a las reglas de negocio del proyecto, y a los riesgos que puedan generar lo que se desea probar”.</p>	<p>ejecutados durante el proceso de pruebas, con el fin de poder tomar decisiones correctas y a tiempo que permitan mejorar el proceso de pruebas</p>	<p>se requiere: priorizarlos y mantener un registro de todos los posibles estados que pueden tener los casos de prueba durante el proceso.</p>	<p>Se requiere saber si todos los casos de prueba han sido priorizados.</p>
					<p>Se requiere saber el número de casos de prueba ejecutados</p>
					<p>Se requiere saber el número de casos de prueba ejecutados exitosamente</p>
					<p>Se requiere saber el número de casos de prueba que han sido fallidos.</p>
					<p>Se requiere saber el número de veces que se han ejecutado todos los casos de prueba durante el proceso de pruebas.</p>
<p>Métricas relacionadas a los casos de prueba</p>		<p>“El 90% considerando que las de menor prioridad constituyen el 10% faltante para dar por finalizadas las pruebas planificadas”.</p> <p>“En la empresa se ha manejado el 80/20, es decir se espera que al menos se haya ejecutado el 80% de las pruebas para poder dar como</p>	<p>Para dar por certificado el producto de software desarrollado y dar por finalizado el proceso de pruebas la empresa requiere tener una información confiable que le permita tomar</p>	<p>Para dar por finalizado el proceso de pruebas cuando existe algún motivo que impide poder ejecutar el 100% de los casos de prueba diseñados, se requiere haber ejecutado al</p>	<p>Se requiere saber si se han ejecutados todos los casos de prueba prioritarios para dar por finalizado el proceso de pruebas</p>

		<p>aceptadas y que pase a la siguiente fase el proyecto. Aunque actualmente no es la mejor opción debido a que los casos de prueba no se priorizan y puede ser que en el 20% de los casos de prueba que faltan de ejecutar, no se hayan ejecutado los casos de prueba más importantes”.</p> <p>“Considero que al menos se debería cumplir con el 100% de los casos de prueba que son críticos”.</p> <p>“Se debería priorizar las pruebas y ejecutar las que tengan mayor impacto en el producto o que afecten el funcionamiento del mismo”.</p> <p>“El 80% porque se asegura que el software proporcionado pueda ayudar al usuario final en sus actividades y no detenga su trabajo diario”.</p>	<p>una decisión correcta cuando por algún motivo ya se por falta de tiempo, retrasos en el desarrollo, entre otros no se pueda terminar de ejecutar todos los casos de prueba diseñados.</p>	<p>menos los casos de prueba prioritarios para el proyecto.</p>	
<p>Métricas relacionadas a los requerimientos a ser validados durante el proceso de pruebas</p>	<p>“Al menos un caso de prueba por cada requerimiento”.</p> <p>“Por lo general, se crea un caso de prueba por requerimiento. Si un requerimiento es extenso, se puede dividir en dos o más depende del alcance de la aplicación”.</p> <p>“Por cada requerimiento debería haber al menos un caso de prueba. Si un requerimiento es complejo debería tener más de un caso de prueba”.</p>	<p>La empresa requiere saber el número de casos de prueba que se deben diseñar para probar toda la funcionalidad a ser implementado en los diferentes proyectos de desarrollo</p>	<p>Para obtener información se requiere evaluar si los casos de prueba diseñados y ejecutados están ayudando a validar de forma adecuada cada uno de los requerimientos implementados.</p>	<p>Se requiere saber el número de requerimientos que han sido probados en base a los casos de prueba ejecutados</p> <p>Se requiere saber el número de requerimientos que han sido probados exitosamente en base a los casos de prueba ejecutados</p>	

2.3.2 Planificar acciones

De acuerdo a los resultados obtenidos del análisis de datos en la sección de diagnóstico, se llegó a la conclusión que se requieren implementar métricas enfocadas tanto al producto de software como al proceso de pruebas.

En base a las necesidades detectas y a las soluciones identificadas se procederá a implementar matrices de impacto para determinar cómo influyen las necesidades y las soluciones propuestas en cada una de las fases del proceso de pruebas.

En el capítulo 1, en la sección del marco teórico se presentó una síntesis sobre las métricas que existen actualmente para evaluar el producto de software la ISO 25000:2005 y para evaluar el proceso de pruebas lo expuesto por ISTQB (Junta Internacional de Calificaciones para Pruebas de Software), TMMI (Integración del Modelo de Madurez de Pruebas) y otras fuentes investigadas, en base a ello se elaborará un marco de referencia en el cual se detalla lo que se debería evaluar del proceso de pruebas.

A continuación, se compara el marco de referencia implementado con los resultados obtenidos del diagnóstico realizado a la empresa Logiciel, con el fin de identificar las métricas que se requieren implementar de acuerdo a las necesidades de la empresa.

Para implementar las métricas nuevas, usaremos la metodología GQM para estos casos.

2.3.2.1 Matrices de impacto

En el presente trabajo se utilizará las siguientes matrices para determinar el impacto:

- Matriz de impacto del proceso de pruebas versus las necesidades relacionadas a métricas.
- Matriz de impacto de las necesidades identificadas versus las soluciones propuestas.
- Matriz de impacto del proceso de pruebas vs las soluciones relacionadas a métricas.

Tabla 2.5 Matriz de impacto del proceso de pruebas vs las necesidades relacionadas a métricas, elaborado por: Caiza Gabriela

NECESIDADES		FASES DEL PROCESO DE PRUEBAS					
ID		Planificación	Diseño	Implementación	Ejecución	Evaluación	
R1	Se requiere saber el número de requerimientos implementados	X					
R2	Se requiere saber el número de defectos detectados y corregidos durante el proceso de pruebas.				X	X	
R3	Se requiere saber si todos los requerimientos implementados han sido probados.					X	
R4	Se requiere evaluar la facilidad de uso de los productos desarrollados relacionados a los siguientes aspectos: -Facilidad de ser probado. -Operatividad. -Protección contra errores de usuario.					X	
R5	Se requiere evaluar la mantenibilidad de los productos desarrollados relacionada a los siguientes aspectos: -Capacidad de ser modificado. -Capacidad de ser probado.					X	
R6	Se requiere saber si todos los casos de prueba han sido priorizados.			X	X		
R7	Se requiere saber el número de casos de prueba ejecutados.				X	X	
R8	Se requiere saber el número de casos de prueba ejecutados exitosamente.				X	X	

R9	Se requiere saber el número de casos de prueba que han sido fallidos.									X	X
R10	Se requiere saber si se han ejecutados todos los casos de prueba prioritarios para dar por finalizado el proceso de pruebas									X	X
R11	Se requiere saber el número de veces que se han ejecutado todos los casos de prueba durante el proceso de pruebas.										X
R12	Se requiere saber si los casos de prueba diseñados cubren todos los requerimientos a ser implementados.						X			X	
R13	Se requiere saber el número de requerimientos que han sido probados en base a los casos de prueba ejecutados.									X	X
R14	Se requiere saber el número de requerimientos que han sido probados exitosamente en base a los casos de prueba ejecutados									X	X
R15	Se requiere saber el grado de experiencia que tiene el tester en pruebas y el grado de conocimiento que tiene respecto a la aplicación que va a probar.							X			
R16	Se requiere saber si los defectos detectados durante el proceso de prueba han sido priorizados.									X	X
R17	Se requiere saber si todos los defectos priorizados con gravedad crítica y alta han sido corregidos.									X	
R18	Se requiere saber si todos los defectos reportados han sido corregidos.									X	X
R19	Se requiere saber si el tiempo planificado para el proceso de pruebas es el adecuado o se requiere ampliarlo.									X	X

Tabla 2.6 Matriz de impacto de las necesidades vs las soluciones, elaborado por: Caiza Gabriela

		NECESIDADES RELACIONADAS A MÉTRICAS																			
SOLUCIÓN - MÉTRICAS		R 1	R 2	R 3	R 4	R 5	R 6	R 7	R 8	R 9	R 10	R 11	R 12	R 13	R 14	R 15	R 16	R 17	R 18	R 19	
M1	Complejidad de la implementación funcional probada	X																			
M2	Eliminación de defectos		X																		
M3	Cobertura de pruebas			X																	
M4	Capacidad de demostración				X																
M5	Claridad de mensajes				X																
M6	Verificación de entradas válidas.				X																
M7	Complejidad de modificación					X															
M8	Capacidad de prueba autónoma.					X															
M9	Porcentaje de casos de prueba priorizados						X														
M10	Porcentaje de casos de prueba ejecutados							X													
M11	Porcentaje de casos de prueba exitosos								X												
M12	Porcentaje de casos de prueba fallidos									X											

Tabla 2.7 Matriz de impacto del proceso de pruebas vs las soluciones relacionadas a métricas, elaborado por: Caiza Gabriela

		FASES DEL PROCESO DE PRUEBAS					
ID	SOLUCIÓN - MÉTRICAS	Existe la métrica requerida	Planificación	Diseño	Implementación	Ejecución	Evaluación
M1	Complejidad de la implementación funcional probada	Si	X				
M2	Eliminación de defectos	Si			X		X
M3	Cobertura de pruebas	Si					X
M4	Capacidad de demostración	Si		X			
M5	Claridad de mensajes	Si					X
M6	Verificación de entradas válidas.	Si					X
M7	Complejidad de modificación	Si					X
M8	Capacidad de prueba autónoma.	Si				X	X
M9	Porcentaje de casos de prueba priorizados	No			X		
M10	Porcentaje de casos de prueba ejecutados	No				X	
M11	Porcentaje de casos de prueba exitosos	No				X	
M12	Porcentaje de casos de prueba fallidos	No				X	

M13	Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas	No					X	X
M14	Promedio de ejecuciones de casos de prueba	No					X	X
M15	Porcentaje de requerimientos cubiertos con los casos de prueba implementados	No					X	X
M16	Porcentaje de requerimientos probados	No					X	X
M17	Porcentaje de requerimientos probados exitosamente	No					X	X
M18	Grado de conocimiento y experiencia en pruebas de los Tester	No			X			
M19	Porcentaje de defectos priorizados	No					X	X
M20	Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos	No					X	
M21	Porcentaje de defectos reportados que aún no han sido corregidos	No					X	
M22	Desviación en el esfuerzo planificado	Si					X	X

2.3.2.2 Marco de referencia

La calidad del proceso de pruebas influye en la calidad del producto de software desarrollado por este motivo se requiere implementar un marco referencial que incluya todos los aspectos que se deben evaluar de forma conjunta para evaluar por completo y de forma objetiva el proceso de pruebas.

En el capítulo 1, en la sección del marco teórico se presentó una síntesis sobre las métricas que existen actualmente para evaluar el producto de software la ISO 25000:2005 y para evaluar el proceso de pruebas lo expuesto por ISTQB (Junta Internacional de Calificaciones para Pruebas de Software), TMMI (Integración del Modelo de Madurez de Pruebas) y otras fuentes investigadas, en base a ello se elaboró un marco de referencia en el cual se detalla que se debería evaluar en cada una de las fases de proceso de pruebas.

Se vio la necesidad de implementar un marco de referencia, el cual servirá de apoyo en el presente trabajo, con el fin de identificar de forma clara que se debe evaluar y monitorear en cada una de las fases del proceso de pruebas.

Este marco de referencia nos permitirá comparar y proponer las métricas más adecuadas que se acoplen a las necesidades de la empresa, con el fin de evaluar cada una de las fases del proceso de pruebas, de tal forma que se pueda proponer mejoras para cada uno de los puntos críticos del proceso de pruebas.

Sin embargo, es importante recordar que, para aplicar las métricas requeridas para la evaluación del proceso de pruebas, se requiere tener datos de entrada necesarios que permitan aplicar las métricas, caso contrario si se decide seleccionar una métrica y no se tienen los datos de entrada requeridos, se obtendrá un resultado inválido que no aportará ningún valor durante la evaluación del proceso de pruebas.

A continuación, en la Figura 2.2 se presenta el marco de referencia implementado para evaluar el proceso de pruebas:

2.3.2.3 Comparación del Marco de Referencia con el Diagnóstico Realizado a la Empresa Logiciel

De acuerdo a los resultados obtenidos del diagnóstico realizado a la empresa y a las matrices de impacto implementadas, se comparó el marco de referencia con lo que la empresa requiere evaluar del proceso de pruebas de acuerdo a sus necesidades, de dicha comparación se pudo notar que la empresa requiere evaluar todas las fases del proceso de pruebas.

Sin embargo, no se podrá evaluar el proceso de pruebas de acuerdo al marco referencial implementado, ya que la empresa no cuenta actualmente con todos los datos necesarios para poder aplicar las métricas indicadas en el marco de referencia, por este motivo se ha decidido implementar solamente las métricas que puedan aportar resultados con valores válidos, que permitan tomar acciones para mejorar el proceso de pruebas actual que tiene la empresa.

Por lo tanto las métricas que se han seleccionado para evaluar el proceso de pruebas de la empresa Logiciel, han sido seleccionadas en base a las necesidades e información que tiene la empresa sobre el proceso de pruebas.

El marco de referencia implementado nos permitió comparar y proponer las métricas más adecuadas que se acoplen a las necesidades de la empresa.

A continuación en la Tabla 2.8 y Tabla 2.9, se listan las métricas seleccionadas a ser implementadas para evaluar el producto y el proceso de pruebas una vez terminada de realizar la respectiva comparación:

Tabla 2.8 Métricas seleccionadas para evaluar el producto, elaborado por: Caiza Gabriela

MÉTRICAS PARA EVALUAR EL PRODUCTO
Complejidad de la implementación funcional probada
Eliminación de defectos
Cobertura de pruebas
Capacidad de demostración

Claridad de mensajes
Verificación de entradas válidas
Complejidad de modificación
Capacidad de pruebas autónomas

Tabla 2.9 Métricas seleccionadas para evaluar el proceso de pruebas, elaborado por: Caiza Gabriela

MÉTRICAS PARA EVALUAR EL PROCESO DE PRUEBAS
Porcentaje de casos de prueba priorizados
Porcentaje de casos de prueba ejecutados
Porcentaje de casos de prueba exitosos
Porcentaje de casos de prueba fallidos
Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas
Promedio de ejecuciones de casos de prueba
Porcentaje de requerimientos cubiertos con los casos de pruebas diseñados
Porcentaje de requerimientos probados
Porcentaje de requerimientos probados exitosamente
Grado de conocimiento y experiencia en pruebas de los Tester
Porcentaje de defectos priorizados
Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos
Porcentaje de defectos reportados que aún no han sido corregidos
Desviación en el esfuerzo planificado.

2.3.3 Implementar acciones

Una vez terminada la planificación de las acciones a realizar, en esta sección se procederá a su implementación.

Primero, se presenta un pequeño detalle de los tres proyectos que serán evaluados usando las métricas propuestas en este trabajo. A continuación, se detalla las métricas existentes que serán usadas y se procede a proponer las nuevas métricas requeridas. Finalmente, se indican los procedimientos a usar para la recolección de datos requeridos para aplicar las métricas.

2.3.3.1 Proyectos a ser evaluados

A continuación, se presenta un breve detalle de los tres proyectos en los cuales se aplicarán las métricas propuestas en este trabajo para evaluar el proceso de pruebas en la empresa LOGICIEL.

a) Proyecto: CASO DE ESTUDIO 1

El proyecto tiene como finalidad implementar mejoras a la automatización del proceso de Nómina Militar del caso de estudio 1. El objetivo del proyecto es el agilizar el proceso disminuyendo errores operativos ocasionados por el diseño del proceso actual, lo cual genera un alto impacto en el servicio a los clientes.

Se ha seleccionado este proyecto para ser evaluado porque tuvo varios inconvenientes debido a que el proceso que tenía el banco se encontraba mal diseñado. Para el presente proyecto se planificaron las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

b) Proyecto: CASO DE ESTUDIO 2

El proyecto consiste en implementar mejoras al proceso de PES (BID LAPE MUJERES) que al momento de nuestra evaluación contaba con un flujo ya implementado. Sin embargo, el flujo requería ser mejorado debido a que no satisfacía las necesidades de los clientes.

Se ha seleccionado este proyecto porque se requirió el realizar mejoras urgentes debido a que en el proyecto inicial no se hizo un levantamiento correcto de los requerimientos como requería el banco.

Para el presente proyecto se ha planificado las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

c) Proyecto: CASO DE ESTUDIO 3

Al momento, el área de Banca Empresas del caso de estudio 3 no cuenta con un flujo automático para la generación de líneas de créditos de los clientes de Banca Empresas. Los Oficiales Comerciales de Banca Empresas realizan actualmente el ingreso y proceso de cupos de créditos mediante una hoja Excel denominada MAC (Memorando de Aprobación de Crédito). Se requiere automatizar este proceso con la finalidad de disminuir tiempos de respuesta, mejorar el servicio y poder agilizar el proceso de gestión del crédito de estos clientes. Esta automatización debe facilitar el registro que actualmente es manual en archivos en Excel de forma que se genere información consolidada de uso general para los diferentes procesos de crédito.

Se ha seleccionado este proyecto para ser evaluado porque tiene una complejidad muy alta en el desarrollo debido a que se requiere automatizar una gran cantidad de reglas de negocio complejas.

Para el presente proyecto se ha planificado las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

2.3.3.2 Definición de métricas existentes a ser usadas

Las métricas existentes que usaremos en este trabajo las hemos extraído de la ISO/IEC 25000 del 2005, adicionalmente usaremos las métricas indicadas por ISTQB entre otros, de acuerdo a lo presentado en el marco teórico. El detalle de cada una de las métricas se presenta a continuación.

Elemento	Adecuación funcional – Completitud funcional
Id.	Métrica 1 – M1
Nombre de la métrica	Completitud de la implementación funcional
Tipo	Cuantitativo
Descripción	¿Cuán completa es la implementación de acuerdo a la especificación de requerimientos? Contar el número los requerimientos indicados en la especificación de requerimientos y el número de requerimientos que faltan o están incorrectos.
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de requerimientos que no fueron implementados. B=Número de requerimientos establecidos en la especificación de requisitos.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ El más cercano a 0 es lo mejor.
Parte del proceso de pruebas	Fase de planificación
Recursos utilizados	Especificación de requerimientos, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de planificación
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Fiabilidad – Madurez
Id.	Métrica 2 – M2
Nombre de la métrica	Eliminación de defectos
Tipo	Cuantitativo
Descripción	¿Cuántos errores detectados han sido corregidos? Contar el número de defectos corregidos durante el proceso de pruebas y el número de defectos detectados en las pruebas.
Mecanismo de Medición	$X = \frac{A}{B}$

	<p>A=Número de defectos corregidos durante el proceso de pruebas. B=Número de defectos detectados durante las pruebas.</p> <p style="text-align: center;">Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor.
Parte del proceso de pruebas	Fase de ejecución y Fase de evaluación.
Recursos utilizados	Documento de Casos de pruebas, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución. Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Fiabilidad – Madurez
Id.	Métrica 3 – M3
Nombre de la métrica	Cobertura de pruebas
Tipo	Cuantitativo
Descripción	¿Cuántos casos de prueba requeridos han sido ejecutados durante la etapa de pruebas? Contar el número de casos de prueba realizados en un escenario de operación durante la prueba y el número de casos de prueba a ser realizados para cubrir los requerimientos.
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de casos de pruebas ejecutados. B=Número de casos de prueba a ser ejecutados para cubrir los requerimientos.</p> <p style="text-align: center;">Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ Cuanto más se acerque a 1 es lo mejor
Parte del proceso de pruebas	Fase de evaluación

Recursos utilizados	Especificación de requerimientos, Documento de casos de pruebas, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Facilidad de uso – Capacidad de reconocer su adecuación
Id.	Métrica 4 – M4
Nombre de la métrica	Capacidad de demostración
Tipo	Cuantitativo
Descripción	¿Qué cantidad de requerimientos tienen la capacidad de demostración? Contar el número de requerimientos implementadas con capacidad de demostración y contar el número total de requerimientos que requieren capacidad de demostración
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de requerimientos implementados con capacidad de demostración B=Número total de requerimientos que requieren capacidad de demostración.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ El más cercano a 1 es el mejor.
Parte del proceso de pruebas	Fase de diseño.
Recursos utilizados	Especificación de requerimientos, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de diseño.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Facilidad de uso – Operatividad
Id.	Métrica 5 – M5

Nombre de la métrica	Claridad de mensaje
Tipo	Cuantitativo
Descripción	¿Qué cantidad de mensajes son auto explicativo? Contar el número de mensajes implementados con explicaciones claras y el número total de mensajes implementados
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de mensajes implementados con explicaciones claras. B=Número total de mensajes implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ El más cercano a 1 es el mejor.
Parte del proceso de pruebas	Fase de evaluación.
Recursos utilizados	Especificación de requerimientos, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Facilidad de uso – Protección contra errores de usuario
Id.	Métrica 6 – M6
Nombre de la métrica	Verificación de entradas válidas
Tipo	Cuantitativo
Descripción	¿Qué cantidad de ítems de entrada son validados? Contar el número de ítems de entrada validados y el número de ítems que necesitan ser validados
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de campos de entrada que son validados. B=Número de campos que necesitan ser validados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>

Evaluación	$0 \leq X \leq 1$ El más cercano a 1 es el mejor.
Parte del proceso de pruebas	Fase de evaluación
Recursos utilizados	Especificación de requerimientos, Desarrollador, Tester.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Mantenibilidad – Capacidad de ser modificado
Id.	Métrica 7 – M7
Nombre de la métrica	Complejidad de modificación
Tipo	Cuantitativo
Descripción	¿Con qué facilidad el desarrollador puede modificar el software para resolver problemas? Tomar el tiempo de trabajo que le toma al desarrollador modificar y contar el número de modificaciones.
Mecanismo de Medición	$X = \frac{A}{T}$ <p>A=Número de defectos detectados que requieren modificaciones en el software. T=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados.</p> <p>Dónde: $T > 0$</p>
Evaluación	$0 \leq X \leq 1$ El más lejano a 0/t es el mejor.
Parte del proceso de pruebas	Fase de evaluación
Recursos utilizados	Desarrollador, reportes de prueba.
Periodicidad de medición	Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Mantenibilidad – Capacidad de ser probado
Id.	Métrica 8
Nombre de la métrica	Capacidad de prueba autónoma
Tipo	Cuantitativo
Descripción	¿Qué tan independiente es el software al ser probado? Contar el número de pruebas que están dependiendo de otros sistemas y contra el número total de pruebas dependientes con otros sistemas.
Mecanismo de Medición	$X = \frac{A}{B}$ <p>A=Número de pruebas que están dependiendo de otros sistemas. B=Número total de pruebas dependientes con otros sistemas.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	$0 \leq X \leq 1$ El más cercano a 0 es el mejor.
Parte del proceso de pruebas	Fase de ejecución y Fase de evaluación.
Recursos utilizados	Desarrollador
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución. Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada de la ISO/IEC 25023 [25].

Elemento	Esfuerzo
Id.	Métrica 22 – M22
Nombre de la métrica	Desviación en el esfuerzo planificado.
Tipo	Cuantitativo
Descripción	Diferencia entre el esfuerzo planificado y el esfuerzo invertido para el proceso de pruebas.
Mecanismo de Medición	$X = A_t - B_t$ $A_t = \sum \text{Tiempo planificado en las actividades de pruebas}$

	$Bt = \sum \text{Tiempo invertido en las actividades de pruebas}$
Evaluación	Valor Positivo =Por debajo de lo planificado en el cronograma Valor Neutro =De acuerdo con el cronograma Valor Negativo =Por encima de lo planificado en el cronograma
Parte del proceso de pruebas	Fase de ejecución, Fase de evaluación.
Recursos utilizados	Cronograma del proyecto, Líder de QA, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución. Se requiere calcular una sola vez en la fase de evaluación.
Fuente	Tomada del artículo “Aplicando métricas de calidad a proyectos y procesos durante las pruebas exploratorias” [15].

2.3.3.3 Implementación de métricas a ser usadas

Para implementar las métricas requeridas para este trabajo usaremos la metodología GQM. El detalle de dicha metodología se encuentra en el marco teórico.

Para evaluar el proceso de pruebas se van a considerar los siguientes elementos:

- Estados de los requerimientos
- Estados de los casos de prueba
- Prioridad de los casos de prueba
- Estados de los defectos
- Gravedad de los defectos

Los estados de cada uno de los elementos indicados en el listado anterior los vamos a extraer de la herramienta TEST MANAGER, la cual es usada por la empresa LOGICIEL para la gestión de las pruebas.

En la Figura 2.3 se muestran los estados que puede tener un requerimiento:

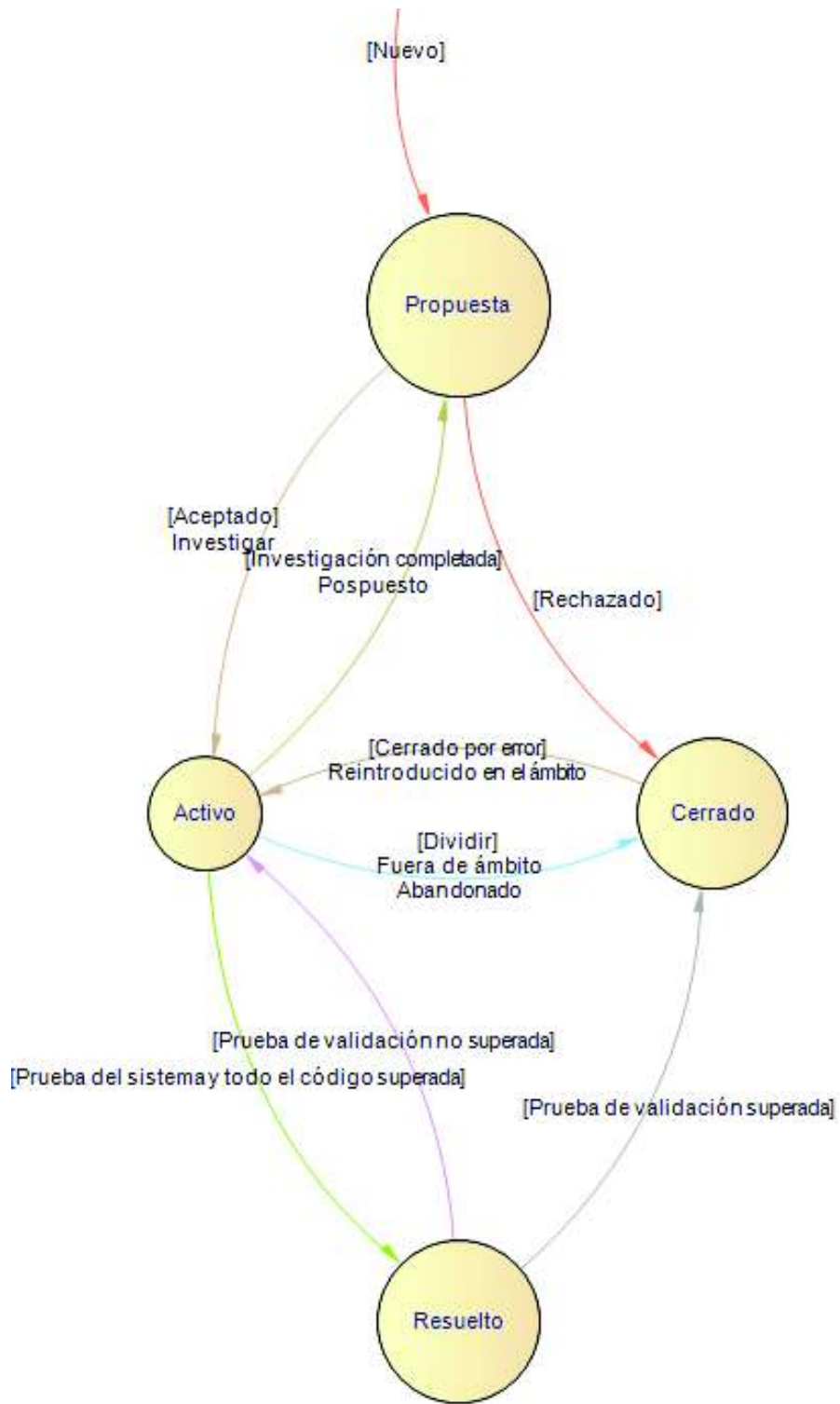


Figura 2.3 Estados de los requerimientos registrados en la herramienta Test Manager [34]

La definición de cada uno de los estados que se muestran en la Figura 2.3 es la siguiente:

- **Nuevo:** Estado inicial del requerimiento cuando se lo crea por primera vez.
- **Activo:** Estado del requerimiento una vez que se haya terminado de registrarlo.
- **Resuelto:** Estado del requerimiento una vez que se han ejecutado todos los casos de prueba de forma exitosa.
- **Cerrado:** Estado del requerimiento una vez que se hayan superado las pruebas de sistema o aceptación.

En la Figura 2.4 se pueden ver los estados que puede tener un caso de prueba:

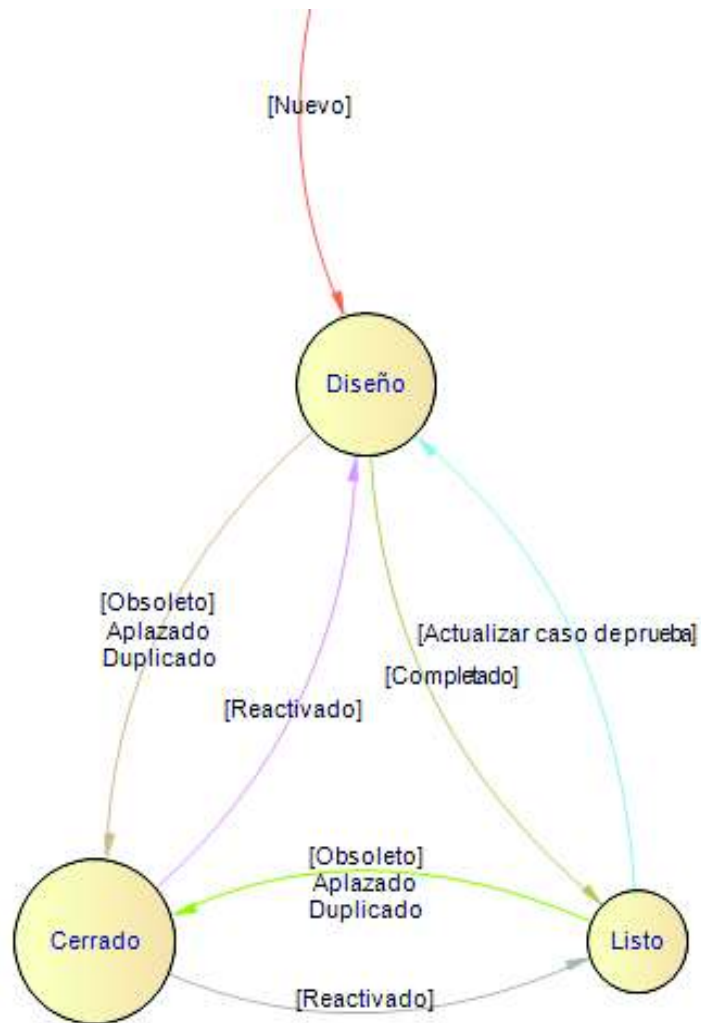


Figura 2.4 Estados de los casos de prueba registrados en la herramienta Test Manager [34]

La definición de cada uno de los estados que se muestran en la Figura 2.4 es la siguiente:

- **Nuevo:** Estado inicial del caso de prueba cuando se lo crea por primera vez.
- **Diseño:** Estado del caso de prueba durante su registro y actualización.
- **Listo:** Estado el caso de prueba cuando está completo y ha sido revisado.
- **Cerrado:** Estado del caso de prueba cuando está obsoleto.

En la Figura 2.5 se pueden ver los estados que puede tener un error:



Figura 2.5 Estados de los errores registrados en la herramienta Test Manager [34]

A continuación, se definen cada uno de los estados que se muestran en la Figura 2.5:

- **Nuevo:** Estado del error cuando se crea.
- **Activo:** Estado del error cuando se ha iniciado su revisión.
- **Resuelto:** Estado del error cuando ha sido corregido.
- **Cerrado:** Estado del error cuando ha sido comprobada su corrección.

Así, mediante el proceso GQM se definen las métricas adicionales como se detalla a continuación.

Objetivo GQM											
Analizar	El proceso de pruebas										
Propósito	Evaluar el proceso de prueba										
Enfoque	Priorización de pruebas										
Punto de Vista	El área de QA										
Entorno	Ambiente de pruebas										
PREGUNTA											
Preguntas	¿Se requiere saber el número de casos de prueba por prioridad? ¿Se requiere saber el número de casos de prueba que han sido priorizados? ¿Se requiere saber el número de casos de prueba que no han sido priorizados?										
MÉTRICA											
Id.	Métrica 9 – M9										
Nombre	Porcentaje de casos de prueba priorizados										
Tipo	Cuantitativo										
Descripción	Número de casos de prueba por prioridad de acuerdo a los siguientes niveles de prioridad que se presentan en la herramienta Test Manager: <table border="1" data-bbox="634 1583 1255 1843"> <thead> <tr> <th>Prioridad de los casos de prueba</th> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Media</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager	Crítica	1	Alta	2	Media	3	Baja	4
Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager										
Crítica	1										
Alta	2										
Media	3										
Baja	4										

	La priorización de todos los casos de prueba va a permitir determinar el orden de ejecución de los casos de prueba.
Mecanismo de Medición	$X = ((A+B+C+D) / E) * 100$ <p>A= \sum [Casos de prueba con prioridad crítica] B= \sum [Casos de prueba con prioridad alta] C= \sum [Casos de prueba con prioridad media] D= \sum [Casos de prueba con prioridad baja] E=Número de casos de prueba implementados</p> <p>Dónde: E>0</p> <p>El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>
Evaluación	Valor deseado 100%.
Parte del proceso de pruebas	Fase de Implementación.
Recursos utilizados	Casos de prueba, Tester
Periodicidad de medición	Se requiere calcular semanalmente durante la fase de implementación.
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Ejecución de pruebas
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Preguntas	¿Se requiere saber el número de casos de prueba que han sido ejecutados hasta un determinado momento?
MÉTRICA	
Id.	Métrica 10 – M10
Nombre	Porcentaje de casos de prueba ejecutados.
Tipo	Cuantitativo

Descripción	Número de casos de prueba que han sido ejecutados durante la fase de ejecución de pruebas.
Mecanismo de Medición	$X=(A/B) *100$ <p>A=Número de casos de prueba ejecutados. B=Número de casos de prueba implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>
Evaluación	Valor deseado 100%
Parte del proceso de pruebas	Fase de Ejecución.
Recursos utilizados	Casos de prueba, Tester
Periodicidad de medición	Se requiere calcular semanalmente durante la fase de ejecución.
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Estados de los casos de prueba
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el número de casos de prueba ejecutados exitosamente?
MÉTRICA	
Id.	Métrica 11 – M11
Nombre	Porcentaje de casos de prueba exitosos.
Tipo	Cuantitativo
Descripción	Del número de casos de pruebas ejecutados contar el número de casos de prueba cuyo resultado ha sido exitoso con el fin de poder validar el porcentaje de pruebas exitosas.
Mecanismo de Medición	$X=(A/B) *100$

	<p>A=Número de casos de prueba exitosos. B=Número de casos de prueba ejecutados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>El número de casos de prueba exitosos corresponden a los casos de prueba con estado "Superado".</p> <table border="1"> <thead> <tr> <th>Resultados de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución del caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Activa</td> <td>Pendiente</td> </tr> <tr> <td>Superada</td> <td>Exitoso</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </tbody> </table>	Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Activa	Pendiente	Superada	Exitoso	Error	Fallido
Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba								
Activa	Pendiente								
Superada	Exitoso								
Error	Fallido								
Evaluación	Valor deseado 100%								
Parte del proceso de pruebas	Fase de ejecución								
Recursos utilizados	Casos de prueba, Tester								
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución.								
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].								

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Estados de los casos de prueba
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el número de casos de prueba fallidos?
MÉTRICA	
Id.	Métrica 12 – M12
Nombre	Porcentaje de casos de prueba fallidos
Tipo	Cuantitativo

Descripción	Del número de casos de pruebas ejecutados contar el número de casos de prueba cuyo resultado ha sido fallido con el fin de poder validar el porcentaje de pruebas fallidos.								
Mecanismo de Medición	$X=(A/B) *100$ <p>A=Número de casos de prueba fallidos B=Número de casos de prueba ejecutados</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>El número de casos de prueba fallidos corresponden a los casos de prueba con estado "Error".</p> <table border="1"> <thead> <tr> <th>Resultados de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución del caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Activa</td> <td>Pendiente</td> </tr> <tr> <td>Superada</td> <td>Exitoso</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </tbody> </table>	Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Activa	Pendiente	Superada	Exitoso	Error	Fallido
Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba								
Activa	Pendiente								
Superada	Exitoso								
Error	Fallido								
Evaluación	Valor deseado 0%								
Parte del proceso de pruebas	Fase de ejecución								
Recursos utilizados	Casos de prueba, Tester								
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución.								
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].								

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Ejecuciones de casos de prueba
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber si se han ejecutados todos los casos de prueba prioritarios para dar por finalizado el proceso de pruebas?
MÉTRICA	

Id.	Métrica 13 – M13																		
Nombre	Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas																		
Tipo	Cuantitativo																		
Descripción	<p>Para dar por finalizado la ejecución de las pruebas se requiere que el 100% de los casos de prueba con prioridad crítica y alta hayan sido ejecutados exitosamente, adicional que al menos se haya ejecutado exitosamente el 50% de los casos de prueba con prioridad media. A continuación, en la tabla se indica el porcentaje que representa cada uno de los casos de prueba ejecutados:</p> <table border="1" data-bbox="625 699 1279 999"> <thead> <tr> <th>Prioridad de casos de prueba</th> <th>Porcentaje requerido a ser ejecutado</th> <th>Porcentaje que representa en la formula</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>100%</td> <td>40%</td> </tr> <tr> <td>Alta</td> <td>100%</td> <td>30%</td> </tr> <tr> <td>Media</td> <td>50% - 100%</td> <td>20%</td> </tr> <tr> <td>Baja</td> <td>0% - 100%</td> <td>10%</td> </tr> <tr> <td></td> <td>TOTAL</td> <td>100%</td> </tr> </tbody> </table>	Prioridad de casos de prueba	Porcentaje requerido a ser ejecutado	Porcentaje que representa en la formula	Crítica	100%	40%	Alta	100%	30%	Media	50% - 100%	20%	Baja	0% - 100%	10%		TOTAL	100%
Prioridad de casos de prueba	Porcentaje requerido a ser ejecutado	Porcentaje que representa en la formula																	
Crítica	100%	40%																	
Alta	100%	30%																	
Media	50% - 100%	20%																	
Baja	0% - 100%	10%																	
	TOTAL	100%																	
Mecanismo de Medición	$X=(A+B+C) *100$ <p>A= \sum [Casos de prueba con prioridad crítica ejecutados exitosamente] B= \sum [Casos de prueba con prioridad alta ejecutados exitosamente] C= \sum [Casos de prueba con prioridad media ejecutados exitosamente] C= \sum [Casos de prueba con prioridad media ejecutados exitosamente]</p>																		
Evaluación	Valor deseado $80 \leq X \leq 100$																		
Parte del proceso de pruebas	Fase de Ejecución, Fase de Evaluación																		
Recursos utilizados	Casos de prueba, Tester																		
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.																		
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].																		

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Ejecuciones de casos de prueba
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el número de veces que se han ejecutado todos los casos de prueba durante el proceso de pruebas?
MÉTRICA	
Id.	Métrica 14 – M14
Nombre	Promedio de ejecuciones de casos de prueba
Tipo	Cuantitativo
Descripción	Número de veces que se han ejecutado cada uno de los casos de prueba diseñados.
Mecanismo de Medición	$X=(A/B)$ <p>A= \sum [Número de ejecuciones de cada caso de prueba] B=Número de casos de prueba implementados</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Evaluación	Valor deseado ≥ 1
Parte del proceso de pruebas	Fase de Ejecución y Fase de Evaluación
Recursos utilizados	Casos de prueba, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada del libro de Fundamentos de Pruebas de ISTQB [13].

Objetivo GQM	
Objeto	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Evaluación de requerimientos probados

Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el número de requerimientos que han sido probados?
MÉTRICA	
Id.	Métrica 15 – M15
Nombre	Porcentaje de requerimientos cubiertos con los casos de prueba diseñados
Tipo	Cuantitativo
Descripción	Determina si los casos de prueba diseñados permitirán probar todos los requerimientos a ser implementados.
Mecanismo de Medición	$X=(A/B) *100$ <p>A= \sum [Requerimientos que tienen casos de prueba diseñados] . B=Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>Se tomará en cuenta los requerimientos que tengan al menos un caso de prueba diseñado.</p>
Evaluación	Valor deseado 100%
Parte del proceso de pruebas	Fase de Ejecución y Fase de Evaluación
Recursos utilizados	Especificación de requerimientos, Casos de prueba, Desarrollador, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [25].

Objetivo GQM	
Objeto	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Evaluación de requerimientos probados
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	

Pregunta 1	¿Se requiere saber si los casos de prueba diseñados cubren todos los requerimientos a ser implementados?
MÉTRICA	
Id.	Métrica 16 – M16
Nombre	Porcentaje de requerimientos probados
Tipo	Cuantitativo
Descripción	Porcentaje de requerimientos que han sido probados durante la fase de pruebas.
Mecanismo de Medición	$X=(A/B) *100$ <p>A= \sum [Requerimientos probados] B=Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>Se dará como requerimiento probado cuando todos los casos de prueba relacionados a un requerimiento hayan sido ejecutados al menos una vez.</p>
Evaluación	Valor deseado 100%
Parte del proceso de pruebas	Fase de Ejecución y Fase de Evaluación
Recursos utilizados	Especificación de requerimientos, Casos de prueba, Desarrollador, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [25].

Objetivo GQM	
Objeto	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Evaluación de requerimientos probados
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el número de requerimientos que han sido probados exitosamente en base a los casos de prueba ejecutados?
MÉTRICA	

Id.	Métrica 17 – M17
Nombre	Porcentaje de requerimientos probados exitosamente
Tipo	Cuantitativo
Descripción	Determinar el número de requerimientos que han sido probados exitosamente tomando en cuenta cada uno de los casos de prueba ejecutados exitosamente.
Mecanismo de Medición	$X=(A/B) *100$ <p>A= \sum [Requerimientos probados exitosamente] B=Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>Se dará como requerimiento probado exitosamente cuando todos los casos de prueba relacionados a un requerimiento sean exitosos.</p>
Evaluación	Valor deseado 100%
Parte del proceso de pruebas	Fase de Ejecución y Fase de Evaluación
Recursos utilizados	Especificación de requerimientos, Casos de prueba, Desarrollador, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [25].

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Conocimiento y experiencia de los Tester
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber el grado de conocimiento que tiene el teste de la aplicación que va a probar? ¿Se requiere saber el grado de conocimiento y experiencia adquirida que tiene el teste sobre el proceso de pruebas?
MÉTRICA	
Id.	Métrica 18 – M18

Nombre	Grado de conocimiento y experiencia en pruebas de los Tester.																							
Tipo	Cuantitativo																							
Descripción	Asignar un valor de acuerdo a la matriz implementada en la cual se toma en cuenta el grado de conocimiento de la aplicación que va a probar y la experiencia en pruebas que tiene el tester.																							
Mecanismo de Medición	<p style="text-align: center;">$X=A/B$</p> <p>A= \sum [Calificación de los tester asignados para ejecutar las pruebas]</p> <p>B=Número de tester asignados para ejecutar las pruebas.</p> <p style="text-align: center;">Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>La calificación a los tester se asignará de acuerdo a la siguiente matriz:</p> <div style="text-align: center;"> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td rowspan="3" style="writing-mode: vertical-rl; transform: rotate(180deg);">Conocimiento de la aplicación</td> <td style="text-align: center;">Alto</td> <td style="text-align: center;">7</td> <td style="text-align: center;">8</td> <td style="text-align: center;">9</td> </tr> <tr> <td style="text-align: center;">Medio</td> <td style="text-align: center;">4</td> <td style="text-align: center;">5</td> <td style="text-align: center;">6</td> </tr> <tr> <td style="text-align: center;">Bajo</td> <td style="text-align: center;">1</td> <td style="text-align: center;">2</td> <td style="text-align: center;">3</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">Junior</td> <td style="text-align: center;">Senior</td> <td style="text-align: center;">Master</td> </tr> <tr> <td></td> <td></td> <td colspan="3" style="text-align: center;">Experiencia en Pruebas</td> </tr> </table> </div>	Conocimiento de la aplicación	Alto	7	8	9	Medio	4	5	6	Bajo	1	2	3			Junior	Senior	Master			Experiencia en Pruebas		
Conocimiento de la aplicación	Alto		7	8	9																			
	Medio		4	5	6																			
	Bajo	1	2	3																				
		Junior	Senior	Master																				
		Experiencia en Pruebas																						
Evaluación	Valor deseado 9																							
Parte del proceso de pruebas	Fase de planificación																							
Recursos utilizados	Tester																							
Periodicidad de medición	Se requiere calcular una sola vez en la fase de planificación.																							
Fuente	Base para determinar la métrica tomada de “Un método para medir la productividad del equipo de pruebas en la estimación del esfuerzo de pruebas de software” [16].																							

Objetivo GQM											
Analizar	El proceso de pruebas										
Propósito	Evaluar el proceso de prueba										
Enfoque	Detección de defectos										
Punto de Vista	El área de QA										
Entorno	Ambiente de pruebas										
PREGUNTA											
Pregunta 1	¿Se requiere saber si los defectos detectados durante el proceso de prueba han sido priorizados?										
MÉTRICA											
Id.	Métrica 19 – M19										
Nombre	Porcentaje de defectos priorizados										
Tipo	Cuantitativo										
Descripción	<p>Número de defectos priorizados de acuerdo a su gravedad, la cual ayudará a los desarrolladores a priorizar los mismos para su posible corrección.</p> <p>Para priorizar los defectos se tomará en cuenta los valores indicados en la siguiente tabla:</p> <table border="1" data-bbox="664 1073 1253 1335"> <thead> <tr> <th>Gravedad de los defectos</th> <th>Gravedad de defectos en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Media</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager	Crítica	1	Alta	2	Media	3	Baja	4
Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager										
Crítica	1										
Alta	2										
Media	3										
Baja	4										
Mecanismo de Medición	$X=(A+B+C+D) *100$ <p>A= \sum [Defectos con gravedad crítica] B= \sum [Defectos con gravedad alta] C= \sum [Defectos con gravedad media] D= \sum [Defectos con gravedad baja]</p>										
Evaluación	Valor deseado 100%										
Parte del proceso de pruebas	Fase de ejecución y Fase de evaluación										
Recursos utilizados	Reporte de pruebas, Tester										

Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [25].

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Detección de defectos
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber si todos los defectos priorizados con gravedad crítica y alta han sido corregidos?
MÉTRICA	
Id.	Métrica 20 – M20
Nombre	Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos
Tipo	Cuantitativo
Descripción	Número de defectos reportados con gravedad crítica y alta que aún no han sido corregidos.
Mecanismo de Medición	$X = (A/B) * 100$ <p>A = \sum [Defectos con gravedad crítica y alta aún no corregidos] B = Número de defectos detectados</p> <p style="text-align: center;">Dónde: $B \geq A \geq 0$; $B > 0$</p>
Valor deseado	Valor deseado 0%
Parte del proceso de pruebas	Fase de ejecución y Fase de evaluación
Recursos utilizados	Reporte de pruebas, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [25].

Objetivo GQM	
Analizar	El proceso de pruebas
Propósito	Evaluar el proceso de prueba
Enfoque	Detección de defectos
Punto de Vista	El área de QA
Entorno	Ambiente de pruebas
PREGUNTA	
Pregunta 1	¿Se requiere saber si todos los defectos reportados han sido corregidos?
MÉTRICA	
Id.	Métrica 21 – M21
Nombre	Porcentaje de defectos reportados que aún no han sido corregidos
Tipo	Cuantitativo
Descripción	Número de defectos reportados que aún no han sido corregidos.
Mecanismo de Medición	$X=(A/B) *100$ <p>A=Número de defectos con estado propuesto y activo B= Número de defectos detectados</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>
Valor deseado	Valor deseado 0%
Parte del proceso de pruebas	Fase de ejecución y Fase de evaluación
Recursos utilizados	Reporte de pruebas, Tester
Periodicidad de medición	Se requiere calcular semanalmente en la fase de ejecución y una sola vez en la fase de evaluación.
Fuente	Base para determinar la métrica tomada de la ISO/IEC 25023 [27].

2.3.3.4 Procedimientos para la recolección de datos

La recopilación de datos requeridos para aplicar las métricas se realizó a través de varias plantillas Excel que fueron elaboradas en base a las métricas para evaluar el producto de software y el proceso de pruebas. A continuación, se presentará el listado de cada una de las plantillas a ser usadas y un breve detalle de la información que se recopilará en cada una de ellas:

- **Plantilla 1 – Datos Cronograma**

En esta plantilla se recopilarán los datos relacionados a las actividades planificadas y recursos asignados para el proceso de pruebas. La plantilla permitirá llevar un seguimiento de las actividades y recursos realmente usados durante todo el proceso de pruebas.

- **Plantilla 2 – Datos Requerimientos**

En esta plantilla se recopilarán los datos relacionados a los requerimientos que serán implementados en un determinado proyecto. La plantilla permitirá llevar un seguimiento de los requerimientos de un proyecto, validando cuales requerimientos serán implementados y cuáles no.

- **Plantilla 3 – Datos Tester**

En esta plantilla se recopilarán los datos relacionadas a los Tester asignados a un determinado proyecto. En la plantilla se registrará a cada uno de los integrantes que forman parte del equipo de pruebas y se asignará una calificación a los Tester de acuerdo a diferentes parámetros.

- **Plantilla 4 – Datos Diseño Casos de Prueba**

En esta plantilla se recopilarán los datos relacionadas a los casos de prueba diseñados para un determinado proyecto. En la plantilla se registrará la información más relevante de cada uno de los casos de prueba diseñados.

- **Plantilla 5 – Datos Ejecución Casos de Prueba**

En esta plantilla se recopilarán los datos relacionados a la ejecución de los casos de prueba en un proyecto determinado. En la plantilla se registrará la información más relevante de cada una de las ejecuciones de los casos de prueba con el resultado obtenido en cada ejecución.

- **Plantilla 6 - Datos Defectos**

En esta plantilla se recopilarán los datos relacionados a los defectos detectados durante el proceso de pruebas en un determinado proyecto. En la plantilla se registrará la información más relevante de cada uno de los defectos y permitirá llevar un seguimiento de estos.

En el ANEXO III se presentan las plantillas implementadas para la recolección de datos.

2.3.3.5 Procedimientos para la evaluación del producto de software y del proceso de pruebas

Para realizar el análisis del producto de software y del proceso de pruebas se utilizarán dos matrices de calidad, las cuales permitirán realizar la evaluación del producto de software y del proceso de pruebas de una manera completa y concisa. A continuación, se presentará el detalle de cada una de las matrices implementadas:

Matriz para evaluar la calidad del producto de software

En la matriz para evaluar el producto de software, se describe lo siguiente:

- a) Característica:** nombre de la característica.
- b) Subcaracterísticas:** nombre de la subcaracterísticas.
- c) Métrica:** nombre de la métrica.
- d) Fórmula:** fórmula de la métrica.
- e) Valor deseado:** umbrales de medida.
- f) Valor obtenido:** valor obtenido una vez aplicada la fórmula.
- g) Ponderación:** valor sobre 10.
- h) Porcentaje de importancia:** porcentaje de importancia de la características sobre el 100% de acuerdo a la Tabla 2.10.

Tabla 2.10 Porcentaje de importancia de las métricas sobre el 100% para evaluar el producto de software, elaborado por: Gabriela Caiza

ID	MÉTRICA	PORCENTAJE DE IMPORTANCIA
M1	Compleitud de la implementación funcional probada	12,5%
M2	Eliminación de defectos	12,5%
M3	Cobertura de pruebas	12,5%
M4	Capacidad de demostración	12,5%
M5	Claridad de mensajes	12,5%
M6	Verificación de entradas válidas.	12,5%
M7	Complejidad de modificación	12,5%
M8	Capacidad de prueba autónoma.	12,5%
		100%

- i) **Valor Final:** es el producto de los valores obtenidos al aplicar las fórmulas por el porcentaje de importancia.
- j) **Resultado de la Calidad del producto:** es la suma de los valores finales.

Matriz para evaluar la calidad del proceso de pruebas

En la matriz para evaluar el proceso de pruebas, se describe lo siguiente:

- a) **Proceso:** nombre del proceso.
- b) **Relación:** relación de los diferentes elementos con el proceso de pruebas.
- c) **Métrica:** nombre de la métrica.
- d) **Fórmula:** fórmula de la métrica
- e) **Valor deseado:** umbrales de medida.
- f) **Valor obtenido:** valor obtenido una vez aplicada la fórmula.
- g) **Ponderación:** valor sobre 10.
- h) **Porcentaje de importancia:** porcentaje de importancia de las métricas sobre el 100% de acuerdo a la Tabla 2.11:

Tabla 2.11 Porcentaje de importancia de las métricas sobre el 100% para evaluar el proceso de pruebas, elaborado por: Gabriela Caiza

ID	MÉTRICA	PORCENTAJE DE IMPORTANCIA
M9	Porcentaje de casos de prueba priorizados	7,15%
M10	Porcentaje de casos de prueba ejecutados	7,15%
M11	Porcentaje de casos de prueba exitosos	7,15%
M12	Porcentaje de casos de prueba fallidos	7,15%
M13	Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas	7,15%
M14	Promedio de ejecuciones de casos de prueba	7,15%
M15	Porcentaje de requerimientos cubiertos con los casos de prueba diseñados	7,15%
M16	Porcentaje de requerimientos probados	7,15%
M17	Porcentaje de requerimientos probados exitosamente	7,15%
M18	Grado de conocimiento y experiencia en pruebas de los Tester	7,15%
M19	Porcentaje de defectos priorizados	7,15%
M20	Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos	7,15%
M21	Porcentaje de defectos reportados que aún no han sido corregidos	7,15%
M22	Desviación en el esfuerzo planificado	7,15%
		100%

- i) **Valor Final:** es el producto de los valores obtenidos al aplicar las fórmulas por el porcentaje de importancia.
- j) **Resultado de la Calidad del proceso:** es la suma de los valores finales.

Las matrices implementadas para evaluar el proceso de pruebas y el producto de software, se presentan en la sección de especificación del aprendizaje en la Tabla 2.16 y Tabla 2.17.

Niveles de puntuación final

Para analizar el resultado final se usará la escala de medición que se presenta en la Tabla 2.12, la cual nos dará el nivel de puntuación final que se le asignará al producto de software y al proceso de pruebas después de su evaluación:

Tabla 2.12 Niveles de puntuación final para la calidad del producto de software y del proceso de pruebas [35]

ESCALA DE MEDICIÓN	NIVELES DE PUNTUACIÓN	GRADO DE SATISFACCIÓN
8,75 – 10	Cumple con los requisitos	Muy Satisfactorio
5 – 8,74	Aceptable	Satisfactorio
2,75 – 4,9	Mínimamente aceptable	Insatisfactorio
0 – 2,74	Inaceptable	

2.3.4 Evaluación

Para evaluar los resultados obtenidos del presente trabajo usaremos una entrevista cuyas preguntas se basarán en el Modelo de Aceptación Tecnológica (TAM) con una escala Likert de 5 niveles, con el fin de evaluar la aceptación de los usuarios.

Modelo de Aceptación Tecnológica

El Modelo de Aceptación Tecnológica (TAM) fue propuesto en el año de 1989 por Davis, Venkatesh y otros, en base a la teoría de la Acción Razonada y a la teoría del Comportamiento Planificado. Estas teorías indican que la percepción de un individuo sobre las consecuencias de una acción o comportamiento puede predecir sus acciones futuras [30].

El modelo TAM lo podemos observar en la siguiente Figura 2.6:

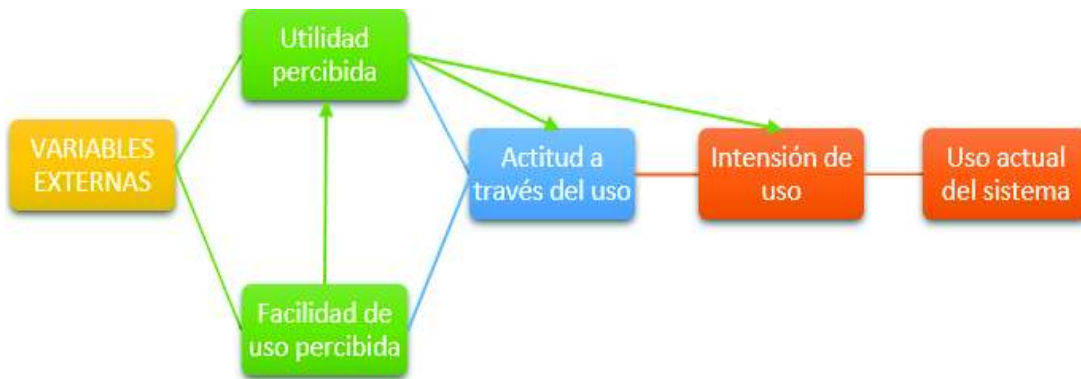


Figura 2.6 Modelo de Aceptación Tecnológica (TAM) [36]

El modelo TAM se usa para predecir el uso de las TICS, basándose en dos características principales [36]:

- La utilidad percibida
- La facilidad de uso percibida

La utilidad percibida: “se refiere al grado en que una persona cree que usando un sistema en particular mejorará su desempeño de trabajo” [36].

La facilidad de uso percibida: “se refiere al grado en que una persona cree que usando un sistema en particular, realizará menos esfuerzo para desempeñar sus tareas” [36].

Los métodos cualitativos son una alternativa para evaluar las soluciones implementadas cuando el número de usuarios es pequeño. La aceptación de los usuarios es un factor clave para la obtención de resultados satisfactorios para los investigadores y profesionales [30]. El modelo TAM ha sido usado en varios mercados para explicar el comportamiento de aceptación de tecnologías y sistemas de información. Además, varias investigaciones lo han utilizado para evaluar artefactos relacionados con tecnología, como es el caso de las métricas propuestas en el presente trabajo.

Escala Likert

La escala Likert fue desarrollada en 1932 por el sociólogo Rensis Likert. Es un tipo de escala ordinal frecuentemente utilizada en cuestionarios de opinión y valoración, y en encuestas para la investigación [37].

Esta escala es una de las más utilizadas para la medición de los conocimientos y actitudes utilizando preguntas cerradas, referentes a un tema específico. La intención de las preguntas planteadas es valorar las opiniones, satisfacciones y el conocimiento frente a un tema determinado [37].

Las escalas Likert son un instrumento psicométrico donde el encuestado debe indicar su acuerdo o desacuerdo sobre una afirmación o ítem, a través de una escala ordenada y unidimensional. Las respuestas pueden presentarse en diferentes niveles de medición, permitiendo escalas de 5, 7 y 9 elementos. Se recomienda que siempre haya un elemento neutral [38]. En la Figura 2.7 se presenta las escalas de valoración de 5 niveles:

1	2	3	4	5
Totalmente en desacuerdo	En desacuerdo	Ni en desacuerdo ni de acuerdo	De acuerdo	Totalmente de acuerdo
-2	-1	0	+1	+2

Figura 2.7 Escalas de valoración Likert [39]

Modalidad de Evaluación

Para evaluar los resultados del presente trabajo se usará una encuesta de población reducida. En base al modelo TAM se seleccionaron las preguntas para la entrevista y se adaptaron para el entorno de la empresa LOGICIEL en donde se llevará a cabo la recolección de los datos.

- **Variable a medir:** Métricas propuestas para evaluar el producto de software y el proceso de pruebas.

La encuesta contiene 10 preguntas con una balanza de Linkert de 5 niveles. A cada respuesta se le asignará un valor con respecto a la escala Linkert. Los rangos de contestaciones y su valoración se presentan en la Figura 2.8:

1	2	3	4	5
Totalmente en desacuerdo	En desacuerdo	Neutral Ni en desacuerdo ni en acuerdo	De acuerdo	Totalmente de acuerdo

Figura 2.8 Rangos de contestación y valoración para la entrevista de evaluación

En la Tabla 2.13 se muestran las preguntas a ser utilizadas en las entrevistas de evaluación enfocadas en las dos características principales del modelo TAM:

Tabla 2.13 Preguntas enfocadas en las características principales del modelo TAM.

Elaborado por: Caiza Gabriela

Nº	ENFOQUE	PREGUNTAS
		EVALUAR LAS MÉTRICAS PROPUESTAS
1	Utilidad percibida	La información obtenida al aplicar las métricas es la información esperada.
2		La información obtenida al aplicar las métricas permite tomar acciones para mejorar el proceso de prueba.
3		Las métricas propuestas están enfocadas a los temas que realmente se necesitaba evaluar del proceso de pruebas.
4		Siento que voy a lograr mejorar el proceso de pruebas con la aplicación de las métricas propuestas.
5		Siento que usando las métricas propuestas es más fácil evaluar el proceso de pruebas que sin ellas.
6		Siento que las métricas implementadas serán un aporte para realizar mejor mi trabajo.
7	Facilidad de uso percibida	Encuentro que las métricas implementadas son fáciles de usar.
8		Las métricas propuestas son claras y entendibles.
9		Creo que la propuesta implementada me permitirá mejorar la productividad del equipo de pruebas.
10		Me siento capaz de interpretar la información obtenida al aplicar las métricas implementadas.
11		Encuentro que es fácil conseguir el resultado una vez aplicadas las métricas implementadas.
12		Si pudiera elegir, seguiría usando las métricas propuestas.

En la Tabla 2.14 se presenta el plan de encuesta que contiene las preguntas a ser evaluadas y los roles a los cuales se aplicará las entrevistas:

Tabla 2.14 Guía para la entrevista de evaluación, elaborado por: Caiza Gabriela

Nº	PREGUNTAS	Gerente General	Gerente de Desarrollo	Gerente de Implantación	Gerente de QA	Lider de QA	Tester
EVALUAR LAS MÉTRICAS PROPUESTAS							
1	La información obtenida al aplicar las métricas es la información esperada.	X	X	X	X	X	X
2	La información obtenida al aplicar las métricas permite tomar acciones para mejorar el proceso de prueba.	X	X	X	X	X	
3	Las métricas propuestas están enfocadas a los temas que realmente se necesitaba evaluar del proceso de pruebas.	X	X	X	X	X	
4	Siento que la propuesta planteada será un aporte para mejorar el proceso de pruebas.	X	X	X	X	X	X
5	Siento que usando las métricas propuestas es más fácil evaluar el proceso de pruebas que sin ellas.				X	X	X
6	Siento que las métricas implementadas serán un aporte para realizar mejor mi trabajo.				X	X	X
7	Encuentro que las métricas implementadas son fáciles de usar.				X	X	X
8	Las métricas propuestas son claras y entendibles.	X	X	X	X	X	X
9	Creo que la propuesta implementada me permitirá mejorar la productividad del equipo de pruebas.				X	X	
10	Me siento capaz de interpretar la información obtenida al aplicar las métricas implementadas.				X	X	
11	Encuentro que es fácil conseguir el resultado una vez aplicadas las métricas implementadas.				X	X	
12	Si pudiera elegir, seguiría usando las métricas propuestas.				X	X	

A continuación, se presenta el formato final de la encuesta usada, la cual contiene cada pregunta y sus respectivas alternativas de respuesta como se puede ver en la Tabla 2.15:

Tabla 2.15 Entrevista de evaluación, elaborado por: Caiza Gabriela

Nº	PREGUNTAS	Totalmente en Desacuerdo	En Desacuerdo	Ni En Desacuerdo Ni Deacuerdo	De Acuerdo	Totalmente Deacuerdo
1	La información obtenida al aplicar las métricas es la información esperada.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	La información obtenida al aplicar las métricas permitirá tomar acciones para mejorar el proceso de prueba.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	Las métricas propuestas están enfocadas a los temas que realmente se necesitaba evaluar del proceso de pruebas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Siento que la propuesta planteada será un aporte para mejorar el proceso de pruebas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	Siento que usando las métricas propuestas es más fácil evaluar el proceso de pruebas que sin ellas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	Siento que las métricas implementadas serán un aporte para realizar mejor mi trabajo.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	Encuentro que las métricas implementadas son fáciles de usar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8	Las métricas propuestas son claras y entendibles.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9	Creo que la propuesta implementada permitirá mejorar la productividad del equipo de pruebas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10	Me siento capaz de interpretar la información obtenida al aplicar las métricas implementadas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11	Encuentro que es fácil conseguir el resultado una vez aplicadas las métricas implementadas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
12	Si pudiera elegir, seguiría usando las métricas propuestas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2.3.5 Especificación el aprendizaje

A partir de los resultados obtenidos se establece un primer catálogo de métricas para evaluar la calidad del producto de software y el proceso de pruebas. A continuación, se presentan las matrices implementadas para realizar la evaluación en la Tabla 2.16 y Tabla 2.17:

Tabla 2.16 Matriz para evaluar la calidad del producto de software, elaborado por: Caiza Gabriela

MATRIZ PARA EVALUAR LA CALIDAD DEL PRODUCTO

ID	CARACTERÍSTICAS	SUBCARACTERÍSTICA	MÉTRICA	FÓRMULA	VALOR DESEADO	VALOR OBTENIDO
1	Adecuación funcional	Complejidad funcional	Complejidad de la implementación funcional probada	A=Número de requerimientos que no fueron implementados	0	A = 0
				B=Número de requerimientos establecidos en la especificación de requisitos		B = 0
				Dónde: $B \geq A \geq 0$; $B > 0$		X = 0,00
2	Fiabilidad	Madurez	Eliminación de defectos	A=Número de defectos corregidos durante el proceso de pruebas	1	A = 0
				B=Número de defectos detectados durante las pruebas		B = 0
				<p>NOTA: El número de defectos corregidos corresponden a los defectos con estado Resuelto y Cerrado</p> Dónde: $B \geq A \geq 0$; $B > 0$		X = 0
3	Fiabilidad	Madurez	Cobertura de pruebas	A=Número de casos de pruebas ejecutados	1	A = 0
				B=Número de casos de prueba a ser ejecutados para cubrir los requerimientos		B = 0
				Dónde: $B \geq A \geq 0$; $B > 0$		X = 0,00
4	Fiabilidad	Capacidad de reconocer su adecuación	Capacidad de demostración	A=Número de requerimientos implementados con capacidad de demostración	1	A = 0
				B=Número total de requerimientos que requieren capacidad de demostración		B = 0
				Dónde: $B \geq A \geq 0$; $B > 0$		X = 0

5	Facilidad de uso	Operatividad	Claridad de mensajes	<p>A=Número de mensajes implementados con explicaciones claras. B=Número total de mensajes implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0
6		Protección contra errores de usuario	Verificación de entradas válidas.	<p>A=Número de campos de entrada que son validados. B=Número de campos que necesitan ser validados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0
7		Capacidad de ser modificado	Complejidad de modificación	<p>A=Número de defectos detectados que requieren modificaciones en el software. T=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados.</p> <p>Dónde: $T \geq A \geq 0$; $T > 0$</p>	<p>Deseado: $\geq 1/60\text{min}$ Peor caso: $\geq 0/60\text{min}$</p>	A =	0
						T =	0
						X =	0,00
8		Capacidad de ser probado	Capacidad de prueba autónoma.	<p>A=Número de pruebas que están dependiendo de otros sistemas. B=Número total de pruebas dependientes con otros sistemas.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0,00

5	Facilidad de uso	Operatividad	Claridad de mensajes	<p>A=Número de mensajes implementados con explicaciones claras. B=Número total de mensajes implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0
6		Protección contra errores de usuario	Verificación de entradas válidas.	<p>A=Número de campos de entrada que son validados. B=Número de campos que necesitan ser validados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0
7		Capacidad de ser modificado	Complejidad de modificación	<p>A=Número de defectos detectados que requieren modificaciones en el software. T=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados.</p> <p>Dónde: $T \geq A \geq 0$; $T > 0$</p>	<p>Deseado: $\geq 1/60\text{min}$ Peor caso: $\geq 0/60\text{min}$</p>	A =	0
						T =	0
						X =	0,00
8		Capacidad de ser probado	Capacidad de prueba autónoma.	<p>A=Número de pruebas que están dependiendo de otros sistemas. B=Número total de pruebas dependientes con otros sistemas.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	1	A =	0
						B =	0
						X =	0,00

Tabla 2.17 Matriz para evaluar el proceso de pruebas, elaborado por: Caiza Gabriela

MATRIZ PARA EVALUAR LA CALIDAD DEL PROCESO DE PRUEBAS

ID	PROCESO	RELACIÓN	METRICA	FORMULA	VALOR DESEADO	VALOR OBTENIDO										
9			Porcentaje de casos de prueba priorizados	$X = \frac{(A+B+C+D)/E}{100}$ <p>A= \sum[Casos de prueba con prioridad crítica] B= \sum[Casos de prueba con prioridad alta] C= \sum[Casos de prueba con prioridad media] D= \sum[Casos de prueba con prioridad baja] E=Número de casos de prueba implementados</p> <p>Dónde: B>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo". Para priorizar los casos de prueba se tomará en cuenta los</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Critica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Media</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Prioridad de los casos de prueba en la herramienta Test Manager	Prioridad de los casos de prueba en la herramienta Test Manager	Critica	1	Alta	2	Media	3	Baja	4	100%	A = 0 B = 0 C = 0 D = 0 E = 0 X = 0,00
Prioridad de los casos de prueba en la herramienta Test Manager	Prioridad de los casos de prueba en la herramienta Test Manager															
Critica	1															
Alta	2															
Media	3															
Baja	4															
10			Porcentaje de casos de prueba ejecutados	$X = \frac{A}{B} * 100$ <p>A=Número de casos de prueba ejecutados B= Número de casos de prueba implementados</p> <p>Dónde: B≥A≥0; B>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>	100%	A = 0 B = 0 X = 0,00										
11	PROCESO DE PRUEBAS	Métricas relacionadas a los casos de prueba	Porcentaje de casos de prueba exitosos	$X = \frac{A}{B} * 100$ <p>A=Número de casos de prueba exitosos B= Número de casos de prueba ejecutados</p> <p>Dónde: B≥A≥0; B>0</p> <p>NOTA: El número de casos de prueba exitosos corresponden a los casos de prueba con estado "Superado".</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Resultados de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución del caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Activa</td> <td>Pendiente</td> </tr> <tr> <td>Superada</td> <td>Exitoso</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </tbody> </table>	Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Activa	Pendiente	Superada	Exitoso	Error	Fallido	100%	A = 0 B = 0 X = 0,00		
Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba															
Activa	Pendiente															
Superada	Exitoso															
Error	Fallido															

12	<p>Porcentaje de casos de prueba fallidos</p>	<p>$X = (A/B) * 100$ A=Número de casos de prueba fallidos B= Número de casos de prueba ejecutados</p> <p>Dónde: $B > A \geq 0$; $B > 0$</p> <p>NOTA: El número de casos de prueba fallidos corresponden a los casos de prueba con estado "Error"</p> <table border="1" data-bbox="438 766 584 1165"> <tr> <td>Resultados de ejecución de los casos de prueba en la herramienta Test Manager</td> <td>Resultado al que corresponde la ejecución del caso de prueba</td> </tr> <tr> <td>Activa</td> <td>Pendiente</td> </tr> <tr> <td>Superada</td> <td>Exitoso</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </table>	Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Activa	Pendiente	Superada	Exitoso	Error	Fallido	<p>0%</p>	<p>A = 0 B = 0 X = 0,00</p>
Resultados de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba											
Activa	Pendiente											
Superada	Exitoso											
Error	Fallido											
13	<p>Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas</p>	<p>$X = (A+B+C) * 100$</p> <p>A= \sum [Casos de prueba con prioridad crítica ejecutados exitosamente] B= \sum [Casos de prueba con prioridad alta ejecutados exitosamente] C= \sum [Casos de prueba con prioridad media ejecutados exitosamente] D= \sum [Casos de prueba con prioridad baja ejecutados exitosamente]</p> <p>NOTA: Dentro del 80% como valor deseado se encuentran el 100% de los casos de prueba con prioridad crítica y alta ejecutados exitosamente. Adicionalmente el 50% de los casos de prueba ejecutados exitosamente.</p>	<p>$80 \leq X \leq 100$</p>	<p>A = 40,00 B = 30,00 C = 20,00 D = 10,00 X = 100,00</p>								
14	<p>Promedio de ejecuciones de casos de prueba</p>	<p>$X = (A/B)$ A= \sum [Número de ejecuciones de cada caso de prueba] B= Número de casos de prueba ejecutados</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	<p>$>= 1$</p>	<p>A = 0 B = 0 X = 0,00</p>								
15	<p>Porcentaje de requerimientos cubiertos con los casos de prueba diseñados</p>	<p>$X = (A/B) * 100$ A= \sum [Requerimientos que tienen casos de prueba diseñados] B= Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>NOTA: Se tomará en cuenta los requerimientos que tengan al menos un caso de prueba diseñado.</p>	<p>100%</p>	<p>A = 0 B = 0 X = 0,00</p>								

16	Métricas relacionadas a los requerimientos	Porcentaje de requerimientos probados	<p>$X = \frac{A}{B} * 100$</p> <p>A = \sum [Requerimientos probados] B = Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>NOTA: Se dará como requerimiento probado cuando todos los casos de prueba relacionados a un requerimiento hayan sido ejecutados al menos una vez.</p>	100%	A = B = X =	26 0 10,00
17	Porcentaje de requerimientos probados exitosamente	Porcentaje de requerimientos probados exitosamente	<p>$X = \frac{A}{B} * 100$</p> <p>A = \sum [Requerimientos probados exitosamente] B = Número de requerimientos implementados.</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>NOTA: Se dará como requerimiento probado exitosamente cuando todos los casos de prueba relacionados a un requerimiento sean exitosos</p>	100%	A = B = X =	26 0 0,00
18	Métricas relacionadas a los tester	Grado de conocimiento y experiencia en pruebas de los tester	<p>A = \sum [Calificación de los tester asignados para ejecutar las pruebas] B = Número de tester asignados para ejecutar las pruebas</p> <p>Dónde: $B \geq A \geq 0$; $B > 0$</p> <p>NOTA: La calificación a los tester se asignará de acuerdo a la siguiente matriz:</p>	9	A = B = X =	0 0 0,00

7	8	9
4	5	6
1	2	3
Bajo Medio Alto	Junior Senior Master	Experiencia en Pruebas

19	Métricas relacionadas a los defectos	Porcentaje de defectos priorizados	$X = (A+B+C+D) * 100$ $A = \sum [\text{Defectos con gravedad crítica}]$ $B = \sum [\text{Defectos con gravedad alta}]$ $C = \sum [\text{Defectos con gravedad media}]$ $D = \sum [\text{Defectos con gravedad baja}]$ $E = \text{Número de defectos detectados}$ <p>NOTA: Para priorizar los defectos se toma en cuenta los valores indicados en la siguiente tabla:</p> <table border="1" data-bbox="548 804 708 1129"> <thead> <tr> <th>Gravedad de los defectos</th> <th>Gravedad de defectos en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Media</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager	Crítica	1	Alta	2	Media	3	Baja	4	100%	<p>A =</p> <p>B =</p> <p>C =</p> <p>D =</p> <p>E =</p> <p>X =</p>	<p>0</p> <p>0</p> <p>0</p> <p>0</p> <p>0</p> <p>0</p>
Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager															
Crítica	1															
Alta	2															
Media	3															
Baja	4															
20		Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos	$X = (A/B) * 100$ $A = \sum [\text{Defectos con gravedad crítica y alta aún no corregidos}]$ $B = \text{Número de defectos detectados}$ <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	0%	<p>A =</p> <p>B =</p> <p>X =</p>	<p>0</p> <p>0</p> <p>0,00</p>										
21		Porcentaje de defectos reportados que aún no han sido corregidos	$X = (A/B) * 100$ $A = \text{Número de defectos con estado propuesto y activo}$ $B = \text{Número de defectos detectados}$ <p>Dónde: $B \geq A \geq 0$; $B > 0$</p>	0%	<p>A =</p> <p>B =</p> <p>X =</p>	<p>0</p> <p>0</p> <p>0,00</p>										
22	Métricas relacionadas al cronograma del proceso de pruebas	Desviación en el esfuerzo planificado	$X = A_t - B_t$ $A_t = \sum [\text{Tiempo planificado en las actividades de pruebas}]$ $B_t = \sum [\text{Tiempo invertido en las actividades de pruebas}]$	<p>Positivo=Por debajo de lo planificado en el cronograma</p> <p>Neutro=De acuerdo con el cronograma</p> <p>Negativo=Por encima de lo planificado en el cronograma</p>	<p>A =</p> <p>B =</p> <p>X =</p>	<p>0,00</p> <p>0,00</p> <p>0</p>										

En el ANEXO IV se presentan las matrices completas en formato Excel con las respectivas formulas implementadas para evaluar la calidad del producto de software y la calidad del proceso de pruebas.

3 RESULTADOS Y DISCUSIÓN

En el desarrollo de este capítulo, se detallan los tres casos de estudio sobre los cuales se aplicarán las métricas implementadas para evaluar la calidad del producto de software y la calidad del proceso de pruebas.

3.1 Aplicación de las métricas implementadas

3.1.1 Caso de estudio 1

3.1.1.1 Antecedentes del Proyecto:

El proyecto tiene como finalidad implementar mejoras a la automatización del proceso de Nómina Militar del caso de estudio 1. El objetivo del proyecto es el agilizar el proceso disminuyendo errores operativos ocasionados por el diseño del proceso actual, lo cual genera un alto impacto en el servicio a los clientes.

Se ha seleccionado este proyecto para ser evaluado porque tuvo varios inconvenientes debido a que el proceso que tenía el banco se encontraba mal diseñado. Para el presente proyecto se planificaron las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

3.1.1.2 Recopilación de datos

Para proceder a realizar la evaluación del producto de software y del proceso de pruebas se aplicó lo detallado en el capítulo 2, en la sección de implementar acciones.

Los datos que se necesitan para realizar el análisis del proyecto se tomaron en cuenta la información que se encuentra registrada en la herramienta Test Manager que usa la empresa LOGICIEL para la gestión de pruebas. Posteriormente, se aplicaron las métricas definidas e implementadas en la sección 2.3.3.2 y 2.3.3.3.

En el ANEXO V se encuentran las plantillas Excel diseñadas con todos los datos recopilados del proyecto del caso de estudio 1.

Una vez terminada la recopilación de los datos, se aplicó las métricas para evaluar el producto de software y el proceso de pruebas; las matrices implementadas se encuentran en la Tabla 2.16 y Tabla 2.17.

3.1.1.3 Presentación de resultados

Los resultados obtenidos de la evaluación de calidad de este proyecto aplicando las métricas implementadas se presenta en las siguientes tablas:

En la tabla 3.1 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación del producto de software desarrollado en el proyecto de caso de estudio 1.

En la tabla 3.2 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación del proceso de pruebas durante el desarrollo del proyecto de caso de estudio 1.

En el ANEXO V se encuentran la documentación completa de la evaluación realizada al proyecto del caso de estudio 1.

Tabla 3.1 Matriz de evaluación de la calidad del producto desarrollado en el proyecto del caso de estudio 1, elaborado por: Caiza Gabriela

MATRIZ PARA EVALUAR LA CALIDAD DEL PRODUCTO

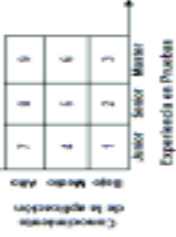
ID	CARACTERÍSTICAS	SUBCARACTERÍSTICA	MÉTRICA	FÓRMULA	VALOR DESEADO	VALOR OBTENIDO	PONDERACIÓN (1/10)	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL PRODUCTO (1/10)
1	Adecuación funcional	Complejidad funcional	Complejidad de la implementación funcional probada	$X=A/B$ A=Número de requerimientos que no fueron implementados B=Número de requerimientos establecidos en la especificación de requisitos Donde: B>A>0; B>0	0	A = 0	10,00	12,5%	1,25	
						B = 18				
						X = 0,00				
2	Fiabilidad	Modulos	Eliminación de defectos	$X=A/B$ A=Número de defectos corregidos durante el proceso de pruebas B=Número de defectos detectados durante las pruebas Donde: B>A>0; B>0 NOTA: El número de defectos corregidos corresponden a los defectos con estado Resuelto y Cerrado	1	A = 56	9,00	12,5%	1,10	
						B = 62				
						X = 0,90				
3			Cobertura de pruebas	$X=A/B$ A=Número de casos de pruebas ejecutados B=Número de casos de prueba a ser ejecutados para cubrir los requerimientos Donde: B>A>0; B>0	1	A = 126	8,51	12,5%	1,06	
						B = 146				
						X = 0,86				
4		Capacidad de reconocer su ejecución	Capacidad de demostración	$X=A/B$ A=Número de requerimientos implementados con capacidad de demostración B=Número total de requerimientos que requieren capacidad de demostración Donde: B>A>0; B>0	1	A = 18	10,00	12,5%	1,25	7,28
						B = 18				
						X = 1,00				

5	Facilidad de uso	Operatividad	Claridad de mensajes	$X=A/B$ A=Número de mensajes implementados con explicaciones claras. B=Número total de mensajes implementados. Donde: $B>A>0$; $B>0$	1	A =	2	2,86	12,5%	0,36
						B =	7			
						X =	0,29			
6		Protección contra errores de usuario	Verificación de entradas válidas.	$X=A/B$ A=Número de campos de entrada que son validados. B=Número de campos que necesitan ser validados. Donde: $B>A>0$; $B>0$	1	A =	62	9,25	12,5%	1,16
						B =	67			
						X =	0,93			
7		Capacidad de ser modificado	Complejidad de modificación	$X=A/B$ A=Número de defectos detectados que requieren modificaciones en el software. B=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados. Donde: $T>A>0$; $T>0$	Desarrollado ≥ 150 min Por caso ≥ 0 (0) min	A =	62	0,00	12,5%	0,00
						T =	1770			
						X =	0,00			
8		Capacidad de ser probado	Capacidad de prueba autónoma.	$X=A/B$ A=Número de pruebas que están dependiendo de otros sistemas. B=Número total de pruebas dependientes con otros sistemas. Donde: $B>A>0$; $B>0$	1	A =	6	8,57	12,5%	1,07
						B =	7			
						X =	0,86			

Tabla 3.2 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 1, elaborado por: Caiza Gabriela

ID	PROCESO	RELACION	METRICA	FORMULA	VALOR DESEADO	VALOR OBTENIDO	PONDERACION (/10)	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL PROCESO (/10)									
9			Porcentaje de casos de prueba prelanzados	<p>$X = \frac{(A+B+C+D+E)}{E} * 100$</p> <p>A= \sum Casos de prueba con prioridad crítica] B= \sum Casos de prueba con prioridad alta] C= \sum Casos de prueba con prioridad media] D= \sum Casos de prueba con prioridad baja] E= Número de casos de prueba implementados</p> <p>Dónde: E>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo". Para prelanzar los casos de prueba se tomará en cuenta los valores indicados en la siguiente tabla:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Prioridad de los casos de prueba</th> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Critica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Medio</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager	Critica	1	Alta	2	Medio	3	Baja	4	100%	A = 0	10,00	7,15%	0,72
					Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager													
					Critica	1													
					Alta	2													
					Medio	3													
Baja	4																		
B = 104																			
C = 4																			
D = 40																			
E = 148																			
X = 100,00																			
10			Porcentaje de casos de prueba ejecutados	<p>$X = \frac{A}{B} * 100$</p> <p>A= Número de casos de prueba ejecutados B= Número de casos de prueba implementados</p> <p>Dónde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>	100%	A = 125	8,51	7,15%	0,61										
						B = 148													
						X = 85,14													
11		Mención relacionada a los casos de prueba	Porcentaje de casos de prueba exitosos	<p>$X = \frac{A}{B} * 100$</p> <p>A= Número de casos de prueba exitosos B= Número de casos de prueba ejecutados</p> <p>Dónde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba exitosos corresponden a los casos de prueba con estado "Superado".</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Resultado de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución del caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Exitoso</td> <td>Exitoso</td> </tr> <tr> <td>Fracaso</td> <td>Fracaso</td> </tr> </tbody> </table>	Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Exitoso	Exitoso	Fracaso	Fracaso	100%	A = 111	8,81	7,15%	0,53				
					Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba													
					Exitoso	Exitoso													
Fracaso	Fracaso																		
B = 126																			
X = 88,10																			

12	<p>Porcentaje de casos de prueba fallidos</p>	<p>$X=(A/B)*100$</p> <p>A= Número de casos de prueba fallidos B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba fallidos corresponden a los casos de prueba con estado "Error"</p> <table border="1" data-bbox="511 1134 690 1417"> <thead> <tr> <th>Resultado de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado de ejecución correspondiente a ejecución de los casos de prueba</th> </tr> </thead> <tbody> <tr> <td>Manejado</td> <td>Exitoso</td> </tr> <tr> <td>Advertido</td> <td>Pre-fallido</td> </tr> <tr> <td>Esperado</td> <td>Error</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </tbody> </table>	Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado de ejecución correspondiente a ejecución de los casos de prueba	Manejado	Exitoso	Advertido	Pre-fallido	Esperado	Error	Error	Fallido	<p>0%</p>	<p>A = 15 B = 126 X = 11,90</p>	<p>8,81</p>	<p>7,16%</p>	<p>0,63</p>
Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado de ejecución correspondiente a ejecución de los casos de prueba																
Manejado	Exitoso																
Advertido	Pre-fallido																
Esperado	Error																
Error	Fallido																
13	<p>Porcentaje de casos de prueba ejecutados para ser por finalizados las pruebas</p>	<p>$X=(A+B+C)*100$</p> <p>A= \sum [Casos de prueba con prioridad crítica ejecutados exitosamente] B= \sum [Casos de prueba con prioridad alta ejecutados exitosamente] C= \sum [Casos de prueba con prioridad media ejecutados exitosamente] D= \sum [Casos de prueba con prioridad baja ejecutados exitosamente]</p> <p>NOTA: Dentro del 80% como valor deseado se encuentran el 100% de los casos de prueba con prioridad crítica y alta ejecutados exitosamente. Adicionalmente el 50% de los casos de prueba ejecutados exitosamente</p>	<p>80%<=(x<=100)</p>	<p>A = 0,00 B = 21,92 C = 10,00 D = 0,25 X = 40,17</p>	<p>4,02</p>	<p>7,16%</p>	<p>0,29</p>										
14	<p>Promedio de ejecuciones de casos de prueba</p>	<p>$X=(A/B)$</p> <p>A= \sum [Número de ejecuciones de cada caso de prueba] B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p>	<p>>=1</p>	<p>A = 126 B = 126 X = 1,00</p>	<p>10,00</p>	<p>7,16%</p>	<p>0,72</p>										

15	PROCESO DE PRUEBAS	Porcentaje de requerimientos cubiertos con los casos de prueba diseñados	<p>$X = \frac{A}{B} * 100$</p> <p>A= \sum[Requerimientos que tienen casos de prueba diseñados] B=Número de requerimientos implementados. Donde: B>A>0; B>0</p> <p>NOTA: Se tomará en cuenta los requerimientos que tengan al menos un caso de prueba diseñado.</p>	100%	<table border="1"> <tr><td>A =</td><td>16</td></tr> <tr><td>B =</td><td>16</td></tr> <tr><td>X =</td><td>100,00</td></tr> </table>	A =	16	B =	16	X =	100,00	10,00	7,15%	0,72
A =	16													
B =	16													
X =	100,00													
16	Métrica relacionada a los requerimientos	Porcentaje de requerimientos probados	<p>$X = \frac{A}{B} * 100$</p> <p>A= \sum [Requerimientos probados] B=Número de requerimientos implementados. Donde: B>A>0; B>0</p> <p>NOTA: Se dará como requerimiento probado cuando todos los casos de prueba relacionados a un requerimiento hayan sido ejecutados al menos una vez.</p>	100%	<table border="1"> <tr><td>A =</td><td>10</td></tr> <tr><td>B =</td><td>18</td></tr> <tr><td>X =</td><td>55,56</td></tr> </table>	A =	10	B =	18	X =	55,56	5,56	7,15%	0,40
A =	10													
B =	18													
X =	55,56													
17		Porcentaje de requerimientos probados exitosamente	<p>$X = \frac{A}{B} * 100$</p> <p>A= \sum [Requerimientos probados exitosamente] B=Número de requerimientos implementados. Donde: B>A>0; B>0</p> <p>NOTA: Se dará como requerimiento probado exitosamente cuando todos los casos de prueba relacionados a un requerimiento sean exitosos</p>	100%	<table border="1"> <tr><td>A =</td><td>11</td></tr> <tr><td>B =</td><td>18</td></tr> <tr><td>X =</td><td>61,11</td></tr> </table>	A =	11	B =	18	X =	61,11	5,11	7,15%	0,44
A =	11													
B =	18													
X =	61,11													
18	Métrica relacionada a los tester	Grado de conocimiento y experiencia en pruebas de los tester	<p>$X = \frac{A}{B}$</p> <p>A= \sum[Calificación de los tester asignados para ejecutar las pruebas] B=Número de tester asignados para ejecutar las pruebas</p> <p>Donde: B>A>0; B>0</p> <p>NOTA: La calificación a los tester se asignará de acuerdo a la siguiente matriz:</p> 	9	<table border="1"> <tr><td>A =</td><td>14</td></tr> <tr><td>B =</td><td>3</td></tr> <tr><td>X =</td><td>4,67</td></tr> </table>	A =	14	B =	3	X =	4,67	5,19	7,15%	0,37
A =	14													
B =	3													
X =	4,67													

19	Métricas relacionadas a los defectos	Porcentaje de defectos priorizados	$X = \frac{A+B+C+D}{E} * 100$ $A = \sum [\text{Defectos con gravedad crítica}]$ $B = \sum [\text{Defectos con gravedad alta}]$ $C = \sum [\text{Defectos con gravedad media}]$ $D = \sum [\text{Defectos con gravedad baja}]$ $E = \text{Número de defectos detectados}$ <p style="text-align: center;">Donde: E=(A+B) B>0</p> <p>NOTA: Para priorizar los defectos se toma en cuenta los valores indicados en la siguiente tabla</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Gravedad de los defectos</th> <th>Gravedad de defectos en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Critica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Medio</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager	Critica	1	Alta	2	Medio	3	Baja	4	100%	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>A =</td> <td>4</td> </tr> <tr> <td>B =</td> <td>10</td> </tr> <tr> <td>C =</td> <td>40</td> </tr> <tr> <td>D =</td> <td>8</td> </tr> <tr> <td>E =</td> <td>62</td> </tr> <tr> <td>X =</td> <td>100</td> </tr> </table>	A =	4	B =	10	C =	40	D =	8	E =	62	X =	100	7,15%	0,715
Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager																												
Critica	1																												
Alta	2																												
Medio	3																												
Baja	4																												
A =	4																												
B =	10																												
C =	40																												
D =	8																												
E =	62																												
X =	100																												
20		Porcentaje de defectos con gravedad crítica y alta que aun no han sido corregidos	$X = \frac{A+B}{E} * 100$ $A = \sum [\text{Defectos con gravedad crítica y alta aun no corregidos}]$ $B = \text{Número de defectos detectados}$ <p style="text-align: center;">Donde: E=(A+B) B>0</p>	0%	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>A =</td> <td>2</td> </tr> <tr> <td>B =</td> <td>14</td> </tr> <tr> <td>X =</td> <td>14,29</td> </tr> </table>	A =	2	B =	14	X =	14,29	7,15%	0,61																
A =	2																												
B =	14																												
X =	14,29																												
21		Porcentaje de defectos reportados que aún no han sido corregidos	$X = \frac{A}{B} * 100$ $A = \text{Número de defectos con estado propuesto y activo}$ $B = \text{Número de defectos detectados}$ <p style="text-align: center;">Donde: E=(A+B) B>0</p>	0%	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>A =</td> <td>0</td> </tr> <tr> <td>B =</td> <td>62</td> </tr> <tr> <td>X =</td> <td>0,68</td> </tr> </table>	A =	0	B =	62	X =	0,68	7,15%	0,66																
A =	0																												
B =	62																												
X =	0,68																												
22	Métricas relacionadas al cronograma del proceso de pruebas	Desviación en el esfuerzo planificado	$X = \frac{A}{B}$ $A = \sum [\text{Tiempo planificado en las actividades de pruebas}]$ $B = \sum [\text{Tiempo invertido en las actividades de pruebas}]$	Positivo=Por debajo de lo planificado en el cronograma Neutro=De acuerdo con el cronograma Negativo=Por encima de lo planificado en el cronograma	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>A =</td> <td>13,00</td> </tr> <tr> <td>B =</td> <td>18,00</td> </tr> <tr> <td>X =</td> <td>-5</td> </tr> </table>	A =	13,00	B =	18,00	X =	-5	7,15%	0																
A =	13,00																												
B =	18,00																												
X =	-5																												

En la Figura 3.1 se presenta el resultado final del análisis de calidad aplicado al producto de software desarrollado en el proyecto del caso de estudio 1.

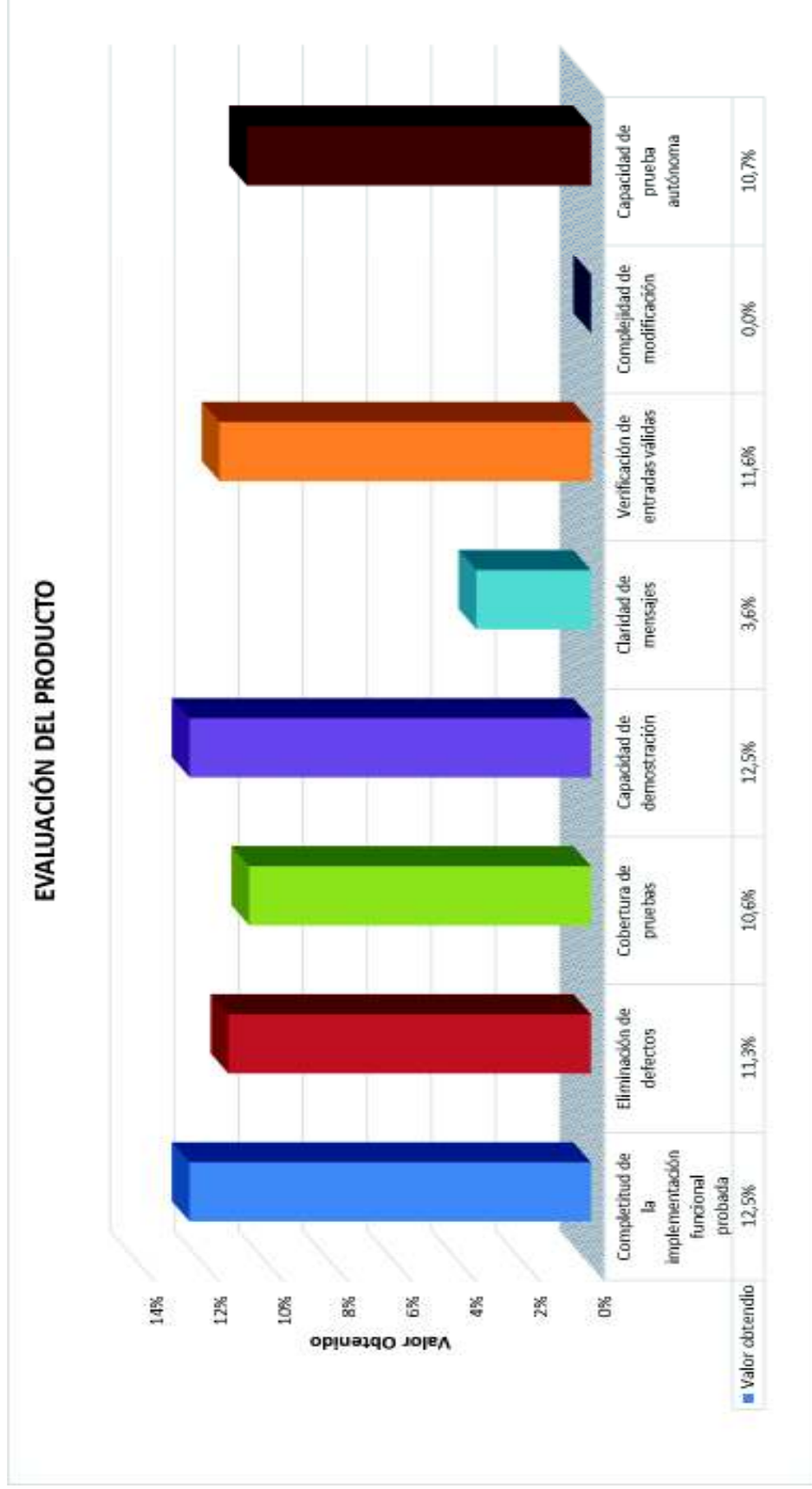


Figura 3.1 Resultados obtenidos de la evaluación del producto de software desarrollado en el proyecto del caso de estudio 1

Una vez realizada la evaluación de calidad del producto de software, en la Figura 3.1 se puede notar que el producto tiene un 0% de 12,5 % en la complejidad de modificación. Esto se debe a que se ha tomado demasiado tiempo en corregir los errores reportados durante las pruebas. Esto indica que el mantenimiento del software implementado va a ser complicado y esto no es bueno; ya que todo software estará sometido a constantes cambios.

Además, se puede ver que se tiene un 3,6% de 12,5% en la claridad de mensajes, lo que nos indica que no se han validado todos los mensajes de alerta requeridos por el cliente.

Adicionalmente, se tiene un 10,6% de 12,5% en la cobertura de pruebas, lo que nos indica que no se han probado al 100% los requerimientos implementados porque no se han terminado de ejecutar todos los casos de prueba diseñados.

Por otro lado, se tiene un 10,7 % de 12,5% en la capacidad de prueba autónoma, lo que nos indica que no se han terminado de ejecutar las pruebas que dependen de otros sistemas externos. Se tiene 11,6% de 12,5% en la verificación de entradas válidas, lo que nos indica que no se han terminado de verificar todas las entradas requeridas por los usuarios.

También, se puede ver que se tiene un 11,3% de 12,5% en la eliminación de defectos, esto nos indica que el producto de software tiene aún defectos que no han sido corregidos.

Todas las características se han evaluado sobre 12,5% con el fin de completar un 100% entre la suma de cada una de las características evaluadas. De las ocho características de software evaluadas solo 2 de ellas han obtenido el porcentaje ideal del 12,5%.

En la Figura 3.2 se presenta el resultado final del análisis de calidad aplicado al proceso de pruebas durante el desarrollo del proyecto del caso de estudio 1.

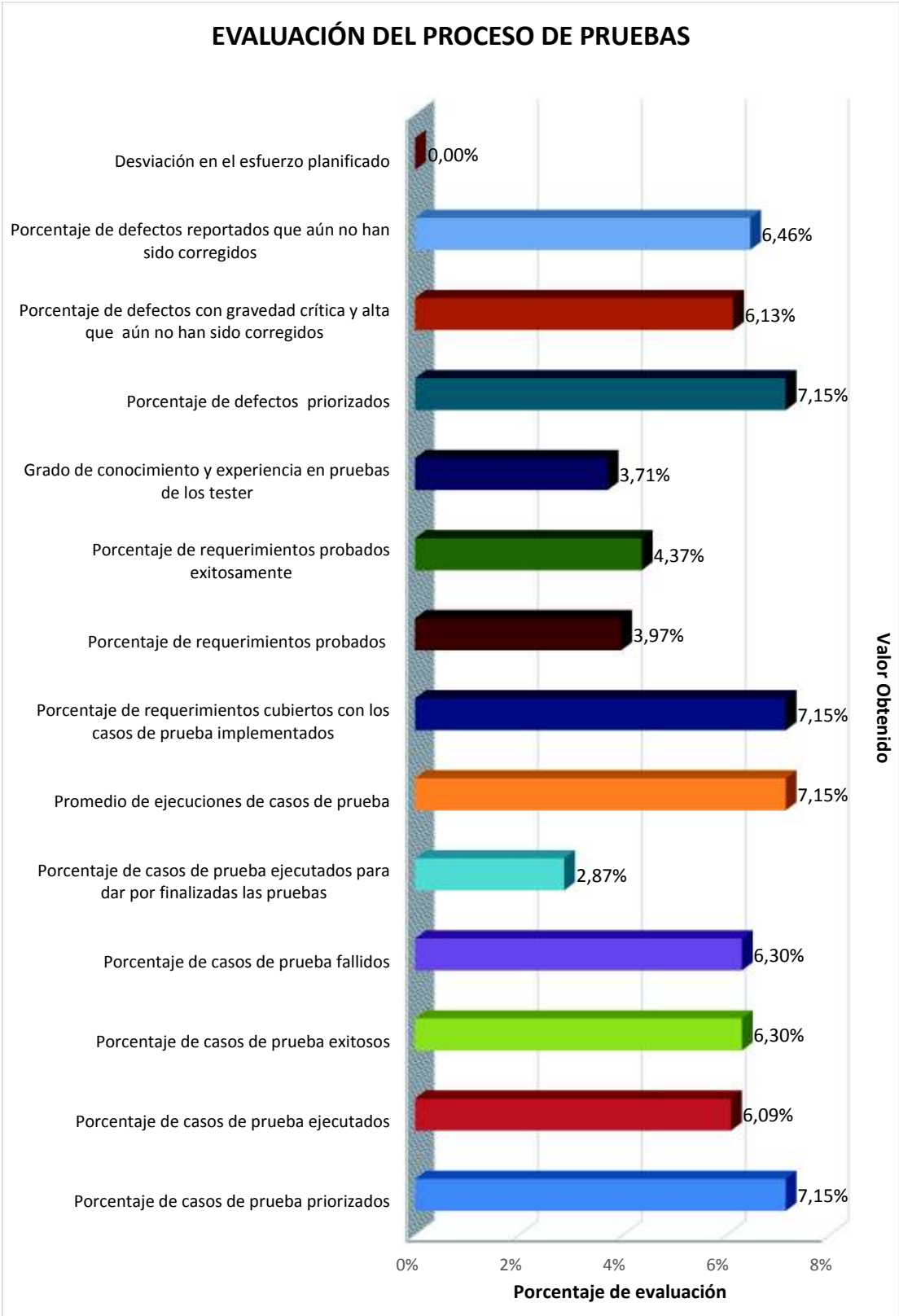


Figura 3.2 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 1

Todas las métricas han sido evaluadas sobre 7,15% con el fin de completar un 100% entre la suma de cada una de los elementos evaluados.

Una vez realizada la evaluación de calidad del proceso de pruebas, en la Figura 3.2 se puede ver que se obtuvo 0% de 7,15% en la desviación en el esfuerzo planificado, esto nos indica que el tiempo planificado es menor al tiempo real empleado para ejecutar el proceso de pruebas por completo, lo cual representa un gran problema para el proceso.

Además, se puede ver que se tiene un 2,87% de 7,15% en el porcentaje de pruebas ejecutadas para dar por finalizadas las pruebas, lo que nos indica que existen dos problemas: por un lado, no se han priorizado correctamente los casos de prueba diseñados, y por otro lado, no se ha ejecutado los casos de prueba realmente importantes para validar los requerimientos implementados.

Adicionalmente, se puede ver que se tiene un 3,71% de 7,15% en el grado de conocimiento y experiencia de los Tester, lo que nos indica que el equipo de pruebas asignado al proyecto no tenía el conocimiento suficiente de la aplicación que iba a validar y le faltaba experiencia en pruebas. Esto provoca que no se pueda ejecutar el proceso de pruebas de forma correcta y completa.

También, en el porcentaje de requerimientos probados exitosamente se alcanzó un 3,97% de 7,15%, lo que indica que no se ha terminado de verificar exitosamente todos los requerimientos implementados en el proyecto.

De los 14 elementos evaluados en el proceso de pruebas solo 4 de ellos han obtenido el porcentaje ideal del 7,5%; esto nos indica que es importante que trabajemos en el resto de elementos que no obtuvieron el porcentaje ideal con el fin de mejorar el proceso de pruebas de acuerdo a los puntos claves que requiere la empresa.

En la tabla 3.3 se puede ver la calificación de calidad sobre 10 que ha obtenido el producto de software y el proceso de pruebas del proyecto del caso de estudio 1.

Tabla 3.3 Calificación de calidad del producto y del proceso de pruebas desarrollado en el proyecto del caso de estudio 1

	CALIFICACIÓN DE CALIDAD	NIVELES DE PUNTUACIÓN	GRADO DE SATISFACCIÓN
Producto de Software	7,28 / 10	Aceptable	Satisfactorio
Proceso de Pruebas	7,48 / 10	Aceptable	Satisfactorio

Se usó la Tabla 2.12 para identificar los niveles de puntuación y los grados de satisfacción.

3.1.1.4 Lecciones aprendidas

- A pesar de usar la herramienta Test Manager para la gestión de pruebas, no se puede controlar el proceso de pruebas sino se definen estándares para su uso; como se evidenció en el proceso de pruebas durante este proyecto. Los bajos valores obtenidos en el proceso obligaron a establecer varios estándares de uso y conllevaron a una capacitación a toda el área de QA.
- La mala planificación tanto para el diseño de casos de prueba como para la ejecución de las pruebas provoca que no se pueda completar el proceso de pruebas correctamente, lo cual puede afectar a la calidad del producto desarrollado. En el cronograma de este proyecto se planificó 1 día para el diseño de casos de prueba, pero de acuerdo con la evaluación realizada, no se tuvo el suficiente tiempo para completar el proceso de pruebas.
- La mala priorización de los casos de prueba diseñados provoca que no se lleguen a ejecutar todos los casos de prueba realmente importantes que permitan validar los requerimientos implementados.

3.1.2 Caso de estudio 2

3.1.2.1 Antecedentes del Proyecto:

El proyecto consiste en implementar mejoras al proceso del caso de estudio 2, que al momento de nuestra evaluación contaba con un flujo ya implementado. Sin embargo, el flujo requería ser mejorado debido a que no satisfacía las necesidades de los clientes.

Se ha seleccionado este proyecto porque se requirió el realizar mejoras urgentes debido a que en el proyecto inicial no se hizo un levantamiento correcto de los requerimientos como requería el banco.

Para el presente proyecto se ha planificado las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

3.1.2.2 Recopilación de datos

Para proceder a realizar la evaluación del producto de software y del proceso de pruebas, se aplicó lo detallado en el capítulo 2, en la sección de implementar acciones.

Los datos que se necesitan para realizar el análisis del proyecto se tomaron en cuenta la información que se encuentra registrada en la herramienta Test Manager que usa la empresa LOGICIEL para la gestión de pruebas. Posteriormente, se aplicaron las métricas definidas e implementadas en la sección 2.3.3.2 y 2.3.3.3.

En el ANEXO VI se encuentran las plantillas Excel diseñadas con todos los datos recopilados del proyecto del caso de estudio 2.

Una vez terminada la recopilación de los datos, se aplicó las métricas para evaluar el producto de software y el proceso de pruebas. Las matrices implementadas se encuentran en la Tabla 2.16 y Tabla 2.17.

3.1.2.3 Presentación de resultados

Los resultados obtenidos de la evaluación de calidad de este proyecto aplicando las métricas implementadas se presenta en las siguientes tablas:

En la tabla 3.4 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación del producto de software desarrollado en el proyecto del caso de estudio 2.

En la tabla 3.5 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación del proceso de pruebas durante el desarrollo del proyecto del caso de estudio 2.

En el ANEXO VI se encuentran la documentación completa de la evaluación realizada al proyecto del caso de estudio 2.

Tabla 3.4 Matriz de evaluación de la calidad del producto de software desarrollado en el proyecto del caso de estudio 2, elaborado por: Caiza Gabriela

MATRIZ PARA EVALUAR LA CALIDAD DEL PRODUCTO										
ID	CARACTERÍSTICAS	SUBCARACTERÍSTICA	MÉTRICA	FÓRMULA	VALOR DESEADO	VALOR OBTENIDO	PONDERACIÓN (/10)	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL PRODUCTO (/10)
1	Adecuación funcional	Complejidad funcional	Complejidad de la implementación funcional probada	$X=A/B$ A=Número de requerimientos que no fueron implementados B=Número de requerimientos establecidos en la especificación de requisitos Donde: B>A>0; B>0	0	A = 0	10,00	12,5%	1,25	
						B = 43				
						X = 0,00				
2	Fiabilidad	Nadares	Eliminación de defectos	$X=A/B$ A=Número de defectos corregidos durante el proceso de pruebas B=Número de defectos detectados durante las pruebas Donde: B>A>0; B>0 NOTA: El número de defectos corregidos corresponden a los defectos con estado Resuelto y Cerrado	1	A = 63	9,76	12,5%	1,22	
						B = 65				
						X = 0,90				
3			Cobertura de pruebas	$X=A/B$ A=Número de casos de pruebas ejecutados B=Número de casos de prueba a ser ejecutados para cubrir los requerimientos Donde: B>A>0; B>0	1	A = 168	9,88	12,5%	1,24	
						B = 170				
						X = 0,99				
4		Cantidad de reconocer su adecuación	Capacidad de demostración	$X=A/B$ A=Número de requerimientos implementados con capacidad de demostración B=Número total de requerimientos que requieren capacidad de demostración Donde: B>A>0; B>0	1	A = 43	10,00	12,5%	1,25	8,71
						B = 43				
						X = 1,000				

5	Facilidad de uso	Operatividad	Cantidad de mensajes	<p>$X=A/B$</p> <p>A=Número de mensajes implementados con explicaciones claras.</p> <p>B=Número total de mensajes implementados.</p> <p>Donde: B>A>0; B>0</p>	1	A = 5	10	12.5%	1.25
						B = 5			
						X = 1			
6		Protección contra errores de usuario	Verificación de entradas válidas.	<p>$X=A/B$</p> <p>A=Número de campos de entrada que son válidos.</p> <p>B=Número de campos que necesitan ser válidos.</p> <p>Donde: B>A>0; B>0</p>	1	A = 78	10.00	12.5%	1.25
						B = 78			
						X = 1.00			
7	Mantenibilidad	Capacidad de ser modificado	Complejidad de modificación	<p>$X=A/T$</p> <p>A=Número de defectos detectados que requieren modificaciones en el software.</p> <p>B=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados.</p> <p>Donde: T>A>0; T>0</p>	<p>Desarrollo: >=160min</p> <p>Por caso >=0/8min</p>	A = 86	0.00	12.5%	0.00
						T = 2728			
						X = 0.03			
8		Capacidad de ser probado	Capacidad de prueba autónoma.	<p>$X=A/B$</p> <p>A=Número de pruebas que están dependiendo de otros sistemas.</p> <p>B=Número total de pruebas dependientes con otros sistemas.</p> <p>Donde: B>A>0; B>0</p>	1	A = 4	10.00	12.5%	1.25
						B = 4			
						X = 1.00			

Tabla 3.5 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 2, elaborado por: Caiza Gabriela

ID	PROCESO	RELACIÓN	MÉTRICA	FÓRMULA	VALOR DESEADO	VALOR OBTENIDO	PONDERACIÓN (%)	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL PROCESO (%)										
9			Porcentaje de casos de prueba priorizados	$X = \frac{A+B+C+D+E}{100}$ <p>A= \sum Casos de prueba con prioridad crítica] B= \sum Casos de prueba con prioridad alta] C= \sum Casos de prueba con prioridad media] D= \sum Casos de prueba con prioridad baja] E= Número de casos de prueba implementados Donde: E<4</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo". Para priorizar los casos de prueba se tomará en cuenta los valores indicados en la siguiente tabla</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Prioridad de los casos de prueba</th> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Critica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Medio</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager	Critica	1	Alta	2	Medio	3	Baja	4	100%	A = 1 B = 161 C = 6 D = 2 E = 170 X = 100,00	10,00	7,15%	0,72	
				Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager															
Critica	1																			
Alta	2																			
Medio	3																			
Baja	4																			
10			Porcentaje de casos de prueba ejecutados	$X = \frac{A}{B} * 100$ <p>A= Número de casos de prueba ejecutados B= Número de casos de prueba implementados Donde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>	100%	A = 169 B = 170 X = 99,41	9,94	7,15%	0,71											
11		Métricas relacionadas a los casos de prueba	Porcentaje de casos de prueba exitosos	$X = \frac{A}{B} * 100$ <p>A= Número de casos de prueba exitosos B= Número de casos de prueba ejecutados Donde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba exitosos corresponden a los casos de prueba con estado "Superado".</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Resultado de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución de caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Exitosa</td> <td>Exitoso</td> </tr> <tr> <td>Fallida</td> <td>Falso</td> </tr> </tbody> </table>	Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución de caso de prueba	Exitosa	Exitoso	Fallida	Falso	100%	A = 145 B = 169 X = 85,80	8,58	7,15%	0,61					
				Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución de caso de prueba															
Exitosa	Exitoso																			
Fallida	Falso																			

12	<p>Porcentaje de casos de prueba fallidos</p>	<p>$X=(A/B)*100$</p> <p>A=Número de casos de prueba fallidos B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba fallidos corresponden a los casos de prueba con estado "Error"</p> <table border="1" data-bbox="511 1138 690 1419"> <thead> <tr> <th>Finalidad de ejecución de los casos de prueba en la herramienta</th> <th>Resultado de ejecución de los casos de prueba en la herramienta</th> </tr> </thead> <tbody> <tr> <td>Manejo</td> <td>Exitoso</td> </tr> <tr> <td>Salvo</td> <td>Parcialmente</td> </tr> <tr> <td>Excepción</td> <td>Error</td> </tr> <tr> <td>Total</td> <td>100%</td> </tr> </tbody> </table>	Finalidad de ejecución de los casos de prueba en la herramienta	Resultado de ejecución de los casos de prueba en la herramienta	Manejo	Exitoso	Salvo	Parcialmente	Excepción	Error	Total	100%	<p>0%</p>	<p>A = 2</p> <p>B = 100</p> <p>X = 1,18</p>	<p>9,88</p> <p>7,15%</p>	<p>0,71</p>
Finalidad de ejecución de los casos de prueba en la herramienta	Resultado de ejecución de los casos de prueba en la herramienta															
Manejo	Exitoso															
Salvo	Parcialmente															
Excepción	Error															
Total	100%															
13	<p>Porcentaje de casos de prueba ejecutados para del por finalizados las pruebas</p>	<p>$X=(A+B+C)*100$</p> <p>A= \sum Casos de prueba con prioridad crítica ejecutados exitosamente; B= \sum Casos de prueba con prioridad alta ejecutados exitosamente; C= \sum Casos de prueba con prioridad media ejecutados exitosamente; D= \sum Casos de prueba con prioridad baja ejecutados exitosamente;</p> <p>NOTA: Dentro del 80% como valor deseado se encuentran el 100% de los casos de prueba con prioridad crítica y alta ejecutados exitosamente. Adicionalmente al 50% de los casos de prueba ejecutados exitosamente.</p>	<p>80% (X=100)</p>	<p>A = 40,00</p> <p>B = 25,34</p> <p>C = 20,00</p> <p>D = 10,00</p> <p>X = 65,34</p>	<p>9,53</p> <p>7,15%</p>	<p>0,88</p>										
14	<p>Promedio de ejecuciones de casos de prueba</p>	<p>$X=(A/B)$</p> <p>A= \sum Número de ejecuciones de cada caso de prueba] B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p>	<p>X=1</p>	<p>A = 287</p> <p>B = 100</p> <p>X = 1,71</p>	<p>10,00</p> <p>7,15%</p>	<p>0,72</p>										

15	PROCESO DE PRUEBAS		Porcentaje de requerimientos cubiertos con los casos de prueba diseñados	$X = \frac{A \cdot B}{A+B} \cdot 100$ <p>A= \sum [Requerimientos que tienen casos de prueba diseñados] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se tomará en cuenta los requerimientos que tengan al menos un caso de prueba diseñado.</p>	100%	A = 43 B = 43 X = 100,00	10,00	7,15%	0,72																		
16	Métrica relacionada a los requerimientos		Porcentaje de requerimientos probados	$X = \frac{A \cdot B}{A+B} \cdot 100$ <p>A= \sum [Requerimientos probados] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se dará como requerimiento probado cuando todos los casos de prueba relacionados a un requerimiento hayan sido ejecutados al menos una vez.</p>	100%	A = 42 B = 43 X = 97,67	9,77	7,15%	0,70																		
17			Porcentaje de requerimientos probados exitosamente	$X = \frac{A \cdot B}{A+B} \cdot 100$ <p>A= \sum [Requerimientos probados exitosamente] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se dará como requerimiento probado exitosamente cuando todos los casos de prueba relacionados a un requerimiento sean exitosos</p>	100%	A = 29 B = 43 X = 67,44	6,74	7,15%	0,48																		
18	Métrica relacionada a los testar		Grado de conocimiento y experiencia en pruebas de los testar	$X = A \cdot B$ <p>A= \sum [Calificación de los testar asignados para ejecutar las pruebas] B= Número de testar asignados para ejecutar las pruebas Dónde: B>A=0; B>0</p> <p>NOTA: La calificación a los testar se asignará de acuerdo a la siguiente matriz:</p> <table border="1" data-bbox="1161 1165 1307 1396"> <tr> <td rowspan="3">Grado de conocimiento y experiencia</td> <td>7</td> <td>8</td> <td>9</td> </tr> <tr> <td>4</td> <td>5</td> <td>6</td> </tr> <tr> <td>1</td> <td>2</td> <td>3</td> </tr> <tr> <td></td> <td colspan="3">Juniór Senior Maestr</td> </tr> <tr> <td></td> <td colspan="3">Experiencia en Pruebas</td> </tr> </table>	Grado de conocimiento y experiencia	7	8	9	4	5	6	1	2	3		Juniór Senior Maestr				Experiencia en Pruebas			0	A = 20 B = 3 X = 6,67	7,41	7,15%	0,53
Grado de conocimiento y experiencia	7	8	9																								
	4	5	6																								
	1	2	3																								
	Juniór Senior Maestr																										
	Experiencia en Pruebas																										

19	Métricas relacionadas a los defectos	Porcentaje de defectos priorizados	$X = (A+B+C) / 100$ $A = \sum [\text{Defectos con gravedad crítica}]$ $B = \sum [\text{Defectos con gravedad alta}]$ $C = \sum [\text{Defectos con gravedad media}]$ $D = \sum [\text{Defectos con gravedad baja}]$ $E = \text{Número de defectos detectados}$ <p>Dónde: $B \leq A \leq D$, $B > 0$</p> <p>NOTA: Para priorizar los defectos se toma en cuenta los valores indicados en la siguiente tabla.</p> <table border="1" data-bbox="581 1157 740 1388"> <thead> <tr> <th>Gravedad de los defectos</th> <th>Gravedad de defectos en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Media</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager	Crítica	1	Alta	2	Media	3	Baja	4	100%	<table border="1" data-bbox="305 722 740 863"> <tr><td>A =</td><td>1</td></tr> <tr><td>B =</td><td>9</td></tr> <tr><td>C =</td><td>71</td></tr> <tr><td>D =</td><td>4</td></tr> <tr><td>E =</td><td>85</td></tr> <tr><td>X =</td><td>100</td></tr> </table>	A =	1	B =	9	C =	71	D =	4	E =	85	X =	100	10	7,15%	0,715
Gravedad de los defectos	Gravedad de defectos en la herramienta Test Manager																													
Crítica	1																													
Alta	2																													
Media	3																													
Baja	4																													
A =	1																													
B =	9																													
C =	71																													
D =	4																													
E =	85																													
X =	100																													
20		Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos	$X = (A/B) * 100$ $A = \sum [\text{Defectos con gravedad crítica y alta aún no corregidos}]$ $B = \text{Número de defectos detectados}$ <p>Dónde: $B \leq A \leq 0$, $B > 0$</p>	0%	<table border="1" data-bbox="773 722 932 863"> <tr><td>A =</td><td>0</td></tr> <tr><td>B =</td><td>10</td></tr> <tr><td>X =</td><td>0,00</td></tr> </table>	A =	0	B =	10	X =	0,00	10,00	7,15%	0,715																
A =	0																													
B =	10																													
X =	0,00																													
21		Porcentaje de defectos reportados que aún no han sido corregidos	$X = (A/B) * 100$ $A = \text{Número de defectos con estado propuesto y activo}$ $B = \text{Número de defectos detectados}$ <p>Dónde: $B > 0$</p>	0%	<table border="1" data-bbox="963 722 1122 863"> <tr><td>A =</td><td>2</td></tr> <tr><td>B =</td><td>85</td></tr> <tr><td>X =</td><td>2,35</td></tr> </table>	A =	2	B =	85	X =	2,35	9,75	7,15%	0,70																
A =	2																													
B =	85																													
X =	2,35																													
22	Métricas relacionadas al cronograma del proceso de pruebas	Desviación en el esfuerzo planificado	$X = (A/B)$ $A = \sum [\text{Tiempo planificado en las actividades de pruebas}]$ $B = \sum [\text{Tiempo invertido en las actividades de pruebas}]$	<p>Positivo- Por debajo de lo planificado en el cronograma</p> <p>Neutro- De acuerdo con el cronograma</p> <p>Negativo- Por encima de lo planificado en el cronograma</p>	<table border="1" data-bbox="1153 722 1328 863"> <tr><td>A =</td><td>50,00</td></tr> <tr><td>B =</td><td>55,00</td></tr> <tr><td>X =</td><td>5</td></tr> </table>	A =	50,00	B =	55,00	X =	5	0	7,15%	0																
A =	50,00																													
B =	55,00																													
X =	5																													

En la Figura 3.3 se presenta el resultado final del análisis de calidad aplicado al producto de software desarrollado en el proyecto del caso de estudio 2.

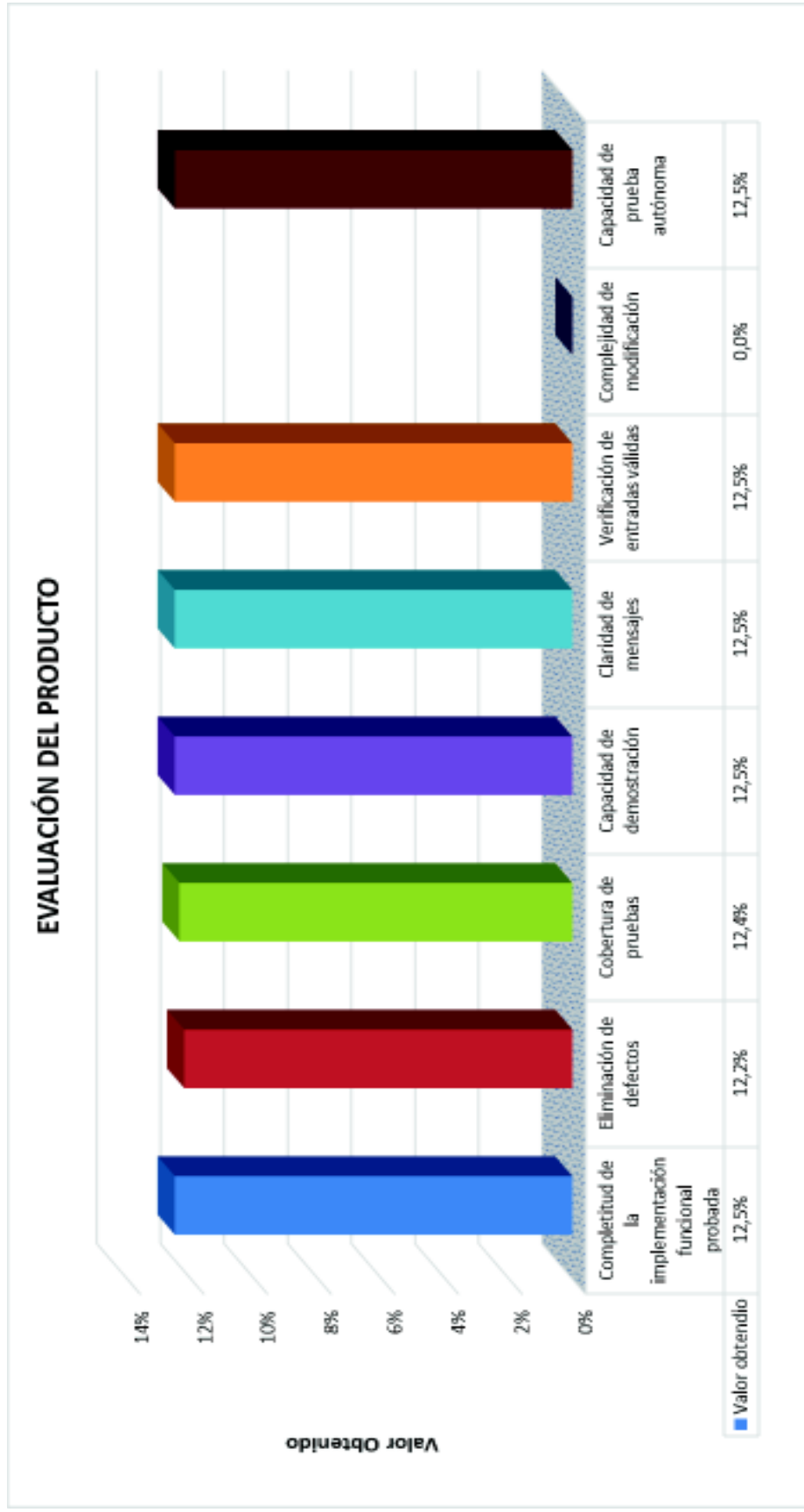


Figura 3.3 Resultados obtenidos de la evaluación del producto desarrollado en el proyecto del caso de estudio 2

Una vez realizada la evaluación de calidad del producto de software, en la Figura 3.3 se puede notar que el software tiene un 0% de 12,5% en la complejidad de modificación, esto se debe a que se ha tomado demasiado tiempo en corregir los errores reportados durante el proceso de pruebas. Esto indica que el mantenimiento del software implementado va a ser complicado y esto no es bueno; ya que todo software estará sometido a constantes cambios.

Además, se puede ver que se tiene un 12,2% de 12,5 en la eliminación de defectos, esto nos indica que el producto de software tiene aún defectos que no han sido corregidos.

También se puede ver que se tiene un 12,4% de 12,5 en la cobertura de pruebas, lo que nos indica que aún se dejó algunos casos de prueba diseñados sin ejecutar.

Todas las características se han evaluado sobre un 12,5% con el fin de completar un 100% entre la suma de cada una de las características evaluadas. De las ocho características de software evaluadas 5 de ellas han obtenido el porcentaje ideal.

En la Figura 3.4 se presenta el resultado final del análisis de calidad aplicado al proceso de pruebas durante el desarrollo del proyecto del caso de estudio 2.

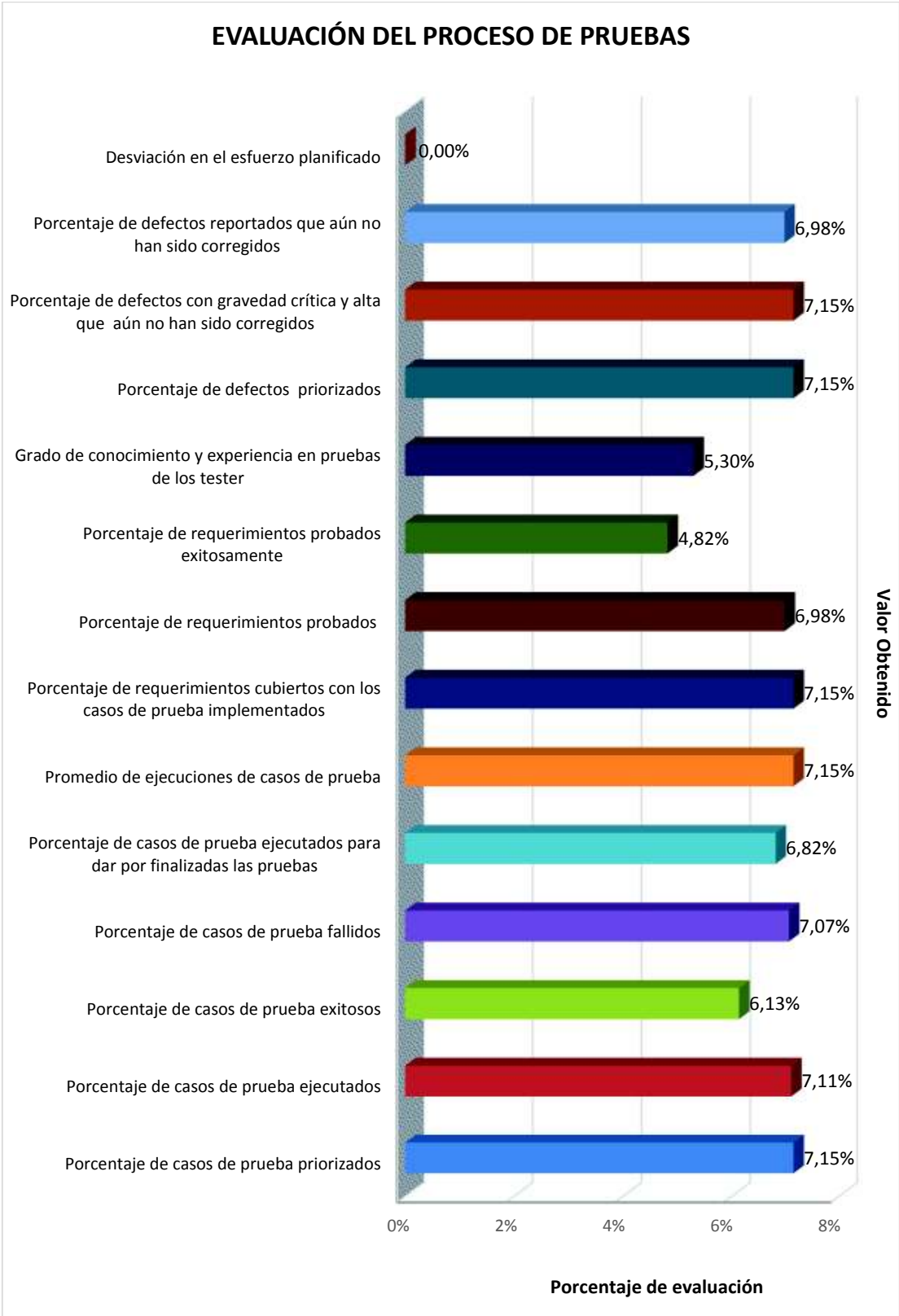


Figura 3.4 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 2

Todas las métricas han sido evaluadas sobre 7,15% con el fin de completar un 100% entre la suma de cada una de los elementos evaluados.

Una vez realizada la evaluación de calidad del proceso de pruebas, en la Figura 3.4 se puede ver que se obtuvo el valor de 0% de 7,15% en la desviación en el esfuerzo planificado, esto nos indica que el tiempo planificado es menor al tiempo real empleado para ejecutar el proceso de pruebas por completo.

Adicionalmente, se tiene un 4,71% de 7,15% en el porcentaje de requerimientos probados exitosamente, lo que indica que no se ha terminado de verificar exitosamente todos los requerimientos implementados en el proyecto.

También se puede ver que se tiene un 5,30% de 7,15% en el grado de conocimiento y experiencia de los Tester, lo que nos indica que el equipo de pruebas asignado al proyecto no tenía el conocimiento suficiente de la aplicación que iba a validar y le falta experiencia en pruebas. Esto provoca que no se pueda ejecutar el proceso de pruebas de forma correcta y completa.

De los 14 elementos evaluados en el proceso de pruebas, solo 4 de ellos han obtenido el porcentaje ideal del 7,5%. Esto nos indica que es importante que trabajemos en el resto de elementos que no obtuvieron el porcentaje ideal, con el fin de mejorar el proceso de pruebas de acuerdo a los puntos claves que requiere la empresa.

En la tabla 3.6 se puede ver la calificación de calidad sobre 10 que ha obtenido el producto de software y el proceso de pruebas del proyecto del caso de estudio 2.

Tabla 3.6 Calificación de calidad del producto de software y del proceso de pruebas desarrollado en el proyecto del caso de estudio 2

	CALIFICACIÓN DE CALIDAD	NIVELES DE PUNTUACIÓN	GRADO DE SATISFACCIÓN
Producto de Software	8,71 / 10	Aceptable	Satisfactorio
Proceso de Pruebas	8,70 / 10	Aceptable	Satisfactorio

Se usó la Tabla 2.12 para identificar los niveles de puntuación y los grados de satisfacción.

3.1.2.4 Lecciones aprendidas

- A pesar de tener diseñado casos de prueba para validar todos los requerimientos implementados, no solo es necesario ejecutar los casos de prueba sino también validar su estado. En este proyecto no se ha terminado de probar todos los requerimientos de forma exitosa porque aún existen casos de prueba ejecutados con estado fallido.
- En este proyecto se terminó el tiempo planificado para completar el proceso de pruebas y no se logró ejecutar todos los casos de prueba diseñados, ni se terminó de corregir todos los defectos reportados.
- El grado de conocimiento en la aplicación a ser probada y la experiencia en pruebas de los Tester en este proyecto se puede ver que si influye en el proceso de pruebas.

3.1.3 Caso de estudio 3

3.1.3.1 Antecedentes del Proyecto:

Al momento, el área de Banca Empresas del caso de estudio 3 no cuenta con un flujo automático para la generación de líneas de créditos de los clientes de Banca Empresas. Los Oficiales Comerciales de Banca Empresas realizan actualmente el ingreso y proceso de cupos de créditos mediante una hoja Excel denominada MAC (Memorando de Aprobación de Crédito). Se requiere automatizar este proceso con la finalidad de disminuir tiempos de respuesta, mejorar el servicio y poder agilizar el proceso de gestión del crédito de estos clientes. Esta automatización debe facilitar el registro que actualmente es manual en archivos en Excel de forma que se genere información consolidada de uso general para los diferentes procesos de crédito.

Se ha seleccionado este proyecto para ser evaluado porque tiene una complejidad muy alta en el desarrollo debido a que se requiere automatizar una gran cantidad de reglas de negocio complejas.

Para el presente proyecto se ha planificado las siguientes pruebas:

- Nivel de pruebas: pruebas de integración y pruebas de sistema.
- Tipo de pruebas: funcionales.

3.1.3.2 Recopilación de datos

Para proceder a realizar la evaluación del producto de software y del proceso de pruebas se aplicó lo detallado en el capítulo 2, en la sección de implementar acciones.

Los datos que se necesitan para realizar el análisis del proyecto se tomaron en cuenta la información que se encuentra registrada en la herramienta Test Manager que usa la empresa LOGICIEL para la gestión de pruebas. Posteriormente, se aplicaron las métricas definidas e implementadas en la sección 2.3.3.2 y 2.3.3.3.

En el ANEXO VII se encuentran las plantillas Excel diseñadas con todos los datos recopilados del proyecto del caso de estudio 3.

Una vez terminada la recopilación de los datos, se aplica las métricas para evaluar el producto de software y el proceso de pruebas; las matrices implementadas se encuentran en la Tabla 2.16 y Tabla 2.17.

3.1.3.3 Presentación de resultados

Los resultados obtenidos de la evaluación de calidad de este proyecto aplicando las métricas implementadas se presenta en las siguientes tablas:

En la tabla 3.7 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación del producto de software desarrollado en el proyecto del caso de estudio 3.

En la tabla 3.8 se muestra a detalle los valores obtenidos de las métricas que fueron aplicadas para la evaluación de calidad del proceso de pruebas durante el desarrollo del proyecto del caso de estudio 3.

En el ANEXO VII se encuentran la documentación completa de la evaluación realizada al proyecto del caso de estudio 3.

Tabla 3.7 Matriz de evaluación de la calidad del producto desarrollado en el proyecto del caso de estudio 3, elaborado por: Caiza Gabriela

MATRIZ PARA EVALUAR LA CALIDAD DEL PRODUCTO

ID	CARACTERÍSTICAS	SUBCARACTERÍSTICA	MÉTRICA	FÓRMULA	VALOR DESEADO	VALOR OBTENIDO	PONDERACIÓN (PI)	PORCENTAJE DE INFORMACIÓN (PI/PO)	VALOR FINAL	CALIDAD DEL PRODUCTO (PI/PO)
1	Adecuación funcional	Complejidad funcional	Complejidad de la implementación funcional probada	$X=A/B$ A=Número de requerimientos que no fueron implementados B=Número de requerimientos establecidos en la especificación de requisitos Donde: $B>A>0$; $B>0$	0	A = 0	10,00	12,5%	1,25	
						B = 25				
						X = 0,00				
2	Fiabilidad	Madurez	Eliminación de defectos	$X=A/B$ A=Número de defectos corregidos durante el proceso de pruebas B=Número de defectos detectados durante las pruebas Donde: $B>A>0$; $B>0$ NOTA: El número de defectos corregidos corresponden a los defectos con estado Resuelto y Cerrado	1	A = 21	9,55	12,5%	1,19	
						B = 22				
						X = 0,95				
3			Cobertura de pruebas	$X=A/B$ A=Número de casos de pruebas ejecutados B=Número de casos de prueba a ser ejecutados para cubrir los requerimientos Donde: $B>A>0$; $B>0$	1	A = 55	6,02	12,5%	0,75	
						B = 93				
						X = 0,60				
4		Capacidad de reconocer su adecuación	Capacidad de demostración	$X=A/B$ A=Número de requerimientos implementados con capacidad de demostración B=Número total de requerimientos que requieren capacidad de demostración Donde: $B>A>0$; $B>0$	1	A = 25	10	12,5%	1,25	6,26
						B = 25				
						X = 1				

5	Facilidad de uso	Operatividad	Cantidad de mensajes	<p>X=AB</p> <p>A=Número de mensajes implementados con explicaciones claras.</p> <p>B=Número total de mensajes implementados.</p> <p>Dónde: B>A; B>0</p>	1	A = 6	10	12,5%	1,25
						B = 6			
						X = 1			
6		Protección contra errores de usuario	Verificación de entradas válidas.	<p>X=AB</p> <p>A=Número de campos de entrada que son validados.</p> <p>B=Número de campos que necesitan ser validados.</p> <p>Dónde: B>A; B>0</p>	1	A = 3	231	12,5%	0,29
						B = 13			
						X = 0,23			
7	Mantenibilidad	Condiciones de ser modificado	Complejidad de modificación	<p>X=A/T</p> <p>A=Número de defectos detectados que requieren modificaciones en el software.</p> <p>T=Tiempo de trabajo que le toma al desarrollador corregir los defectos detectados.</p> <p>Dónde: T>A; T>0</p>	<p>Decreto: >=180min</p> <p>Por caso: >=0,60min</p>	A = 20	0,00	12,5%	0,00
						T = 209			
						X = 0,10			
8		Capacidad de ser probado	Capacidad de prueba autónoma.	<p>X=AB</p> <p>A=Número de pruebas que están dependiendo de otros sistemas.</p> <p>B=Número total de pruebas dependientes con otros sistemas.</p> <p>Dónde: B>A; B>0</p>	1	A = 2	2,22	12,5%	0,28
						B = 9			
						X = 0,22			

Tabla 3.8 Matriz de evaluación de la calidad del proceso de pruebas del proyecto del caso de estudio 3, elaborado por: Caiza Gabriela

ID	PROCESO	RELACIÓN	MÉTRICA	FÓRMULA	VALOR DESEADO	VALORI OBTENIDO	PONDERACIÓN (/10)	PORCENTAJE DE IMPORTANCIA	VALOR FINAL	CALIDAD DEL PROCESO (/10)										
9			Porcentaje de casos de prueba priorizados	$X = \frac{A+B+C+D+E}{100}$ <p>A= \sum Casos de prueba con prioridad crítica] B= \sum Casos de prueba con prioridad alta] C= \sum Casos de prueba con prioridad media] D= \sum Casos de prueba con prioridad baja] E= Número de casos de prueba implementados Dónde: E>0</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo". Para priorizar los casos de prueba se tomará en cuenta los valores indicados en la siguiente tabla:</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Prioridad de los casos de prueba</th> <th>Prioridad de los casos de prueba en la herramienta Test Manager</th> </tr> </thead> <tbody> <tr> <td>Crítica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Medio</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager	Crítica	1	Alta	2	Medio	3	Baja	4	100%	A = 28 B = 14 C = 25 D = 13 E = 93 X = 86,02	8,60	7,15%	0,62	
Prioridad de los casos de prueba	Prioridad de los casos de prueba en la herramienta Test Manager																			
Crítica	1																			
Alta	2																			
Medio	3																			
Baja	4																			
10			Porcentaje de casos de prueba ejecutados	$X = \frac{A}{B} * 100$ <p>A= Número de casos de prueba ejecutados B= Número de casos de prueba implementados</p> <p>NOTA: El número de casos de prueba implementados corresponden a los casos de prueba con estado "Listo".</p>	100%	A = 55 B = 93 X = 60,22	6,02	7,15%	0,43											
11		Métricos relacionados a los casos de prueba	Porcentaje de casos de prueba exitosos	$X = \frac{A}{B} * 100$ <p>A= Número de casos de prueba exitosos B= Número de casos de prueba ejecutados</p> <p>NOTA: El número de casos de prueba exitosos corresponden a los casos de prueba con estado "Superado"</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Resultado de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponde la ejecución del caso de prueba</th> </tr> </thead> <tbody> <tr> <td>Activo</td> <td>Finalizado</td> </tr> <tr> <td>Superado</td> <td>Exitoso</td> </tr> <tr> <td>Error</td> <td>Fallido</td> </tr> </tbody> </table>	Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba	Activo	Finalizado	Superado	Exitoso	Error	Fallido	100%	A = 37 B = 66 X = 66,07	6,61	7,15%	0,47			
Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponde la ejecución del caso de prueba																			
Activo	Finalizado																			
Superado	Exitoso																			
Error	Fallido																			

12	Porcentaje de casos de prueba fallidos	<p>$y=(A/B)*100$</p> <p>A= Número de casos de prueba fallidos B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p> <p>NOTA: El número de casos de prueba fallidos corresponden a los casos de prueba con estado: "Error"</p> <table border="1" data-bbox="509 1136 688 1419"> <thead> <tr> <th>Resultado de ejecución de los casos de prueba en la herramienta Test Manager</th> <th>Resultado al que corresponden las ejecuciones de los casos de prueba</th> </tr> </thead> <tbody> <tr> <td>Exitosa</td> <td>Éxito</td> </tr> <tr> <td>Advertida</td> <td>Advertida</td> </tr> <tr> <td>Suprimida</td> <td>Fracaso</td> </tr> <tr> <td>Error</td> <td>Fallida</td> </tr> </tbody> </table>	Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponden las ejecuciones de los casos de prueba	Exitosa	Éxito	Advertida	Advertida	Suprimida	Fracaso	Error	Fallida	0%	<p>A = 3</p> <p>B = 56</p> <p>X = 5,36</p>	9,46	7,16%	0,68
Resultado de ejecución de los casos de prueba en la herramienta Test Manager	Resultado al que corresponden las ejecuciones de los casos de prueba																
Exitosa	Éxito																
Advertida	Advertida																
Suprimida	Fracaso																
Error	Fallida																
13	Porcentaje de casos de prueba ejecutados para dar por finalizadas las pruebas	<p>$y=(A+B+C)*100$</p> <p>A= \sum[Casos de prueba con prioridad crítica ejecutados exitosamente] B= \sum[Casos de prueba con prioridad alta ejecutados exitosamente] C= \sum[Casos de prueba con prioridad media ejecutados exitosamente] D= \sum[Casos de prueba con prioridad baja ejecutados exitosamente]</p> <p>NOTA: Dentro del 80% como valor deseado se encuentran el 100% de los casos de prueba con prioridad crítica y alta ejecutados exitosamente. Adicionalmente el 50% de los casos de prueba ejecutados exitosamente.</p>	80% (x=100)	<p>A = 48,57</p> <p>B = 26,71</p> <p>C = 6,40</p> <p>D = 3,00</p> <p>X = 53,76</p>	5,30	7,16%	0,30										
14	Promedio de ejecuciones de casos de prueba	<p>$y=(A/B)$</p> <p>A= \sum[Número de ejecuciones de cada caso de prueba] B= Número de casos de prueba ejecutados</p> <p>Donde: B>A>0; B>0</p>	x>1	<p>A = 304</p> <p>B = 56</p> <p>X = 5,96</p>	10,00	7,16%	0,72										

PROCESO DE PRUEBAS		7,23																	
15	<p>Porcentaje de requerimientos cubiertos con los casos de prueba diseñados</p> <p>Métrica relacionada a los requerimientos</p>	<p>$X = \frac{ A }{ A + B } * 100$</p> <p>A= \sum [Requerimientos que tienen casos de prueba diseñados] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se tomará en cuenta los requerimientos que tengan al menos un caso de prueba diseñado.</p>	<p>A = 26</p> <p>B = 26</p> <p>X = 100,00</p> <p>100%</p> <p>10,00</p> <p>7,15%</p> <p>0,72</p>																
16	<p>Porcentaje de requerimientos probados</p> <p>Métrica relacionada a los requerimientos</p>	<p>$X = \frac{A}{ A + B } * 100$</p> <p>A= \sum [Requerimientos probados] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se dará como requerimiento probado cuando todos los casos de prueba relacionados a un requerimiento hayan sido ejecutados al menos una vez.</p>	<p>A = 18</p> <p>B = 26</p> <p>X = 69,23</p> <p>100%</p> <p>6,92</p> <p>7,15%</p> <p>0,50</p>																
17	<p>Porcentaje de requerimientos probados exitosamente</p> <p>Métrica relacionada a los requerimientos</p>	<p>$X = \frac{A}{ A + B } * 100$</p> <p>A= \sum [Requerimientos probados exitosamente] B= Número de requerimientos implementados. Dónde: B>A=0; B>0</p> <p>NOTA: Se dará como requerimiento probado exitosamente cuando todos los casos de prueba relacionados a un requerimiento sean exitosos</p>	<p>A = 9</p> <p>B = 26</p> <p>X = 34,62</p> <p>100%</p> <p>3,46</p> <p>7,15%</p> <p>0,25</p>																
18	<p>Grado de conocimiento y experiencia en pruebas de los tester</p> <p>Métrica relacionada a los tester</p>	<p>$X = A/B$</p> <p>A= \sum [Calificación de los tester asignados para ejecutar las pruebas] B= Número de tester asignados para ejecutar las pruebas</p> <p>Dónde: B>A=0; B>0</p> <p>NOTA: La calificación a los tester se asignará de acuerdo a la siguiente matriz:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td>Junior</td> <td>Senior</td> <td>Master</td> </tr> <tr> <td>Bajo</td> <td>7</td> <td>8</td> <td>9</td> </tr> <tr> <td>Medio</td> <td>4</td> <td>5</td> <td>6</td> </tr> <tr> <td>Alto</td> <td>1</td> <td>2</td> <td>3</td> </tr> </table> <p>Experiencia en Pruebas</p>		Junior	Senior	Master	Bajo	7	8	9	Medio	4	5	6	Alto	1	2	3	<p>A = 14</p> <p>B = 3</p> <p>X = 4,67</p> <p>0</p> <p>5,19</p> <p>7,15%</p> <p>0,37</p>
	Junior	Senior	Master																
Bajo	7	8	9																
Medio	4	5	6																
Alto	1	2	3																

19	Métricas relacionadas a los defectos	Porcentaje de defectos priorizados	<p>$X = (A+B+C) \cdot 100$</p> <p>A= [Defectos con gravedad crítica] B= [Defectos con gravedad alta] C= [Defectos con gravedad medio] D= [Defectos con gravedad baja] E= Número de defectos detectados</p> <p>Dónde: B>A>0; B>0</p> <p>NOTA: Para priorizar los defectos se toma en cuenta los valores indicados en la siguiente tabla:</p> <table border="1" data-bbox="581 1157 732 1388"> <thead> <tr> <th>Gravedad de los defectos</th> <th>Gravedad de defectos en la herramienta (en Manager)</th> </tr> </thead> <tbody> <tr> <td>Critica</td> <td>1</td> </tr> <tr> <td>Alta</td> <td>2</td> </tr> <tr> <td>Medio</td> <td>3</td> </tr> <tr> <td>Baja</td> <td>4</td> </tr> </tbody> </table>	Gravedad de los defectos	Gravedad de defectos en la herramienta (en Manager)	Critica	1	Alta	2	Medio	3	Baja	4	100%	<table border="1" data-bbox="305 726 732 863"> <tr> <td>A =</td> <td>1</td> </tr> <tr> <td>B =</td> <td>5</td> </tr> <tr> <td>C =</td> <td>16</td> </tr> <tr> <td>D =</td> <td>0</td> </tr> <tr> <td>E =</td> <td>22</td> </tr> <tr> <td>X =</td> <td>100</td> </tr> </table>	A =	1	B =	5	C =	16	D =	0	E =	22	X =	100	10	7,15%	0,715
Gravedad de los defectos	Gravedad de defectos en la herramienta (en Manager)																													
Critica	1																													
Alta	2																													
Medio	3																													
Baja	4																													
A =	1																													
B =	5																													
C =	16																													
D =	0																													
E =	22																													
X =	100																													
20	Métricas relacionadas a los defectos	Porcentaje de defectos con gravedad crítica y alta que aún no han sido corregidos	<p>$X = (A/B) \cdot 100$</p> <p>A= [Defectos con gravedad crítica y alta aún no corregidos] B= Número de defectos detectados</p> <p>Dónde: B>A>0; B>0</p>	0%	<table border="1" data-bbox="773 726 924 863"> <tr> <td>A =</td> <td>0</td> </tr> <tr> <td>B =</td> <td>0</td> </tr> <tr> <td>X =</td> <td>0,00</td> </tr> </table>	A =	0	B =	0	X =	0,00	10,00	7,15%	0,72																
A =	0																													
B =	0																													
X =	0,00																													
21	Métricas relacionadas a los defectos	Porcentaje de defectos reportados que aún no han sido corregidos	<p>$X = (A/B) \cdot 100$</p> <p>A= Número de defectos con estado propuesto y activa B= Número de defectos detectados</p> <p>Dónde: B>A>0; B>0</p>	0%	<table border="1" data-bbox="959 726 1110 863"> <tr> <td>A =</td> <td>1</td> </tr> <tr> <td>B =</td> <td>22</td> </tr> <tr> <td>X =</td> <td>4,55</td> </tr> </table>	A =	1	B =	22	X =	4,55	9,55	7,15%	0,68																
A =	1																													
B =	22																													
X =	4,55																													
22	Métricas relacionadas al cronograma del proceso de pruebas	Desviación en el esfuerzo planificado	<p>$X = A/B$</p> <p>A= [Tiempo planificado en las actividades de pruebas] B= [Tiempo ocurrido en las actividades de pruebas]</p>	<p>Positivo=Por debajo de lo planificado en el cronograma Neutro=De acuerdo con el cronograma Negativo=Por encima de lo planificado en el cronograma</p>	<table border="1" data-bbox="1146 726 1315 863"> <tr> <td>A =</td> <td>45,00</td> </tr> <tr> <td>B =</td> <td>63,00</td> </tr> <tr> <td>X =</td> <td>-16</td> </tr> </table>	A =	45,00	B =	63,00	X =	-16	0	7,15%	0																
A =	45,00																													
B =	63,00																													
X =	-16																													

En la Figura 3.5 se presenta el resultado final del análisis de calidad aplicado al producto de software desarrollado en el proyecto del caso de estudio 3.

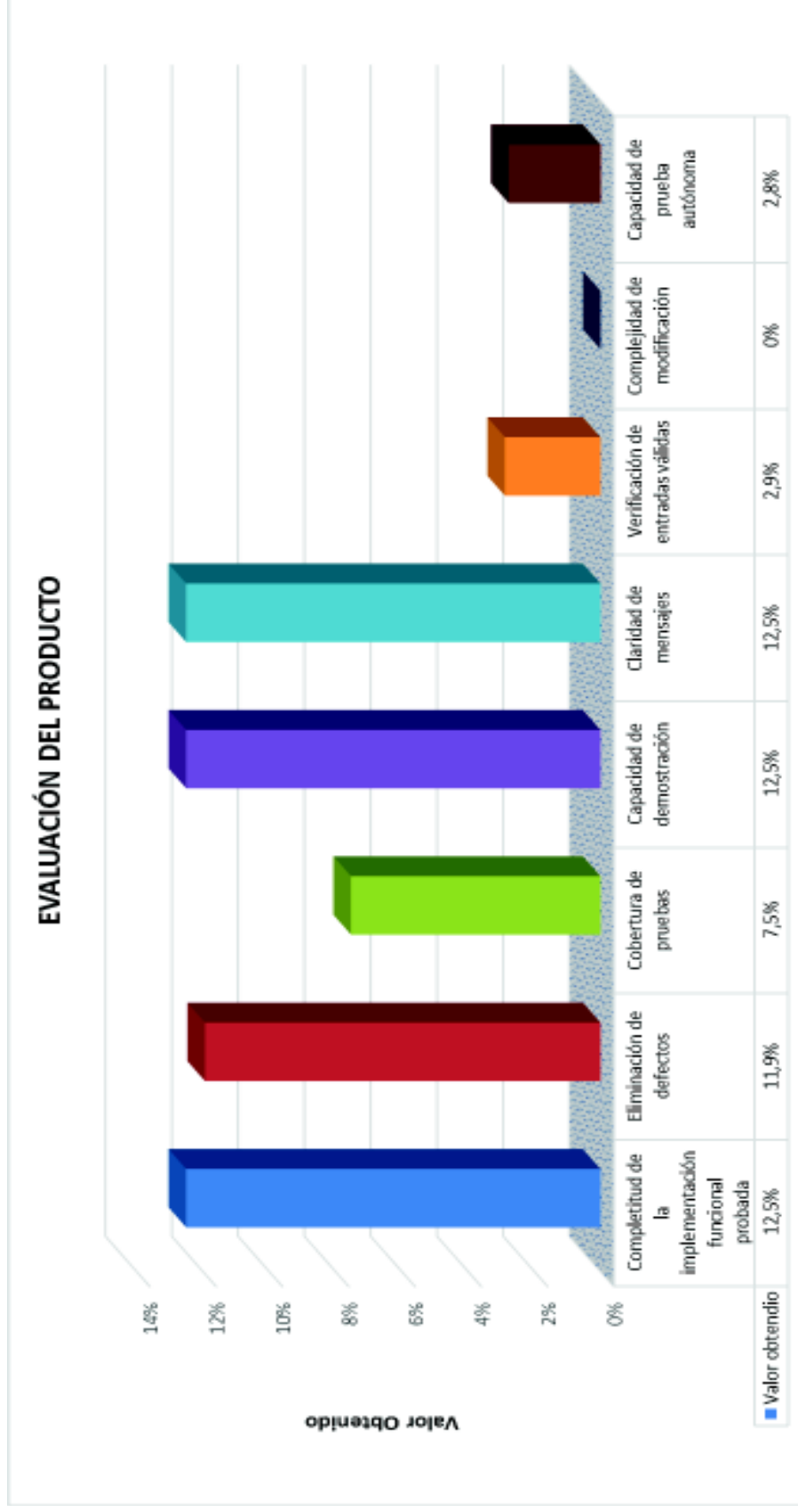


Figura 3.5 Resultados obtenidos de la evaluación del producto desarrollado en el proyecto del caso de estudio 3

Una vez realizada la evaluación de calidad del producto de software, en la Figura 3.5 se puede notar que el producto tiene un 0% de 12,5% en la complejidad de modificación. Esto se debe a que se ha tomado demasiado tiempo en corregir los errores reportados durante el proceso de pruebas, lo que indica que el mantenimiento del software implementado va a ser complicado.

También, se tiene un 2,8% de 12,5% en la capacidad de prueba autónoma, lo que nos indica que no se han terminado de ejecutar las pruebas que dependen de otros sistemas externos. Además, se tiene 2,9% de 12,5% en la verificación de entradas válidas, lo que nos indica que no se han terminado de verificar todas las entradas requeridas por los usuarios.

Adicionalmente, se puede ver que se tiene 7,5% de 12,5% en la cobertura de pruebas, lo que nos indica que no se han probado al 100% todos los requerimientos implementados porque no se han terminado de ejecutar todos los casos de prueba diseñados. También, se puede ver que se tiene un 11,9% de 12,5% en la eliminación de defectos, esto nos indica que el producto de software tiene aún defectos que no han sido corregidos.

Todas las características se han evaluado sobre 12,5% con el fin de completar un 100% entre la suma de cada una de las características evaluadas. De las ocho características de software evaluadas, solo 3 de ellas han obtenido el porcentaje ideal del 12,5%.

En la Figura 3.6 se presenta el resultado final del análisis de calidad aplicado al proceso de pruebas durante el desarrollo del proyecto del caso de estudio 3.

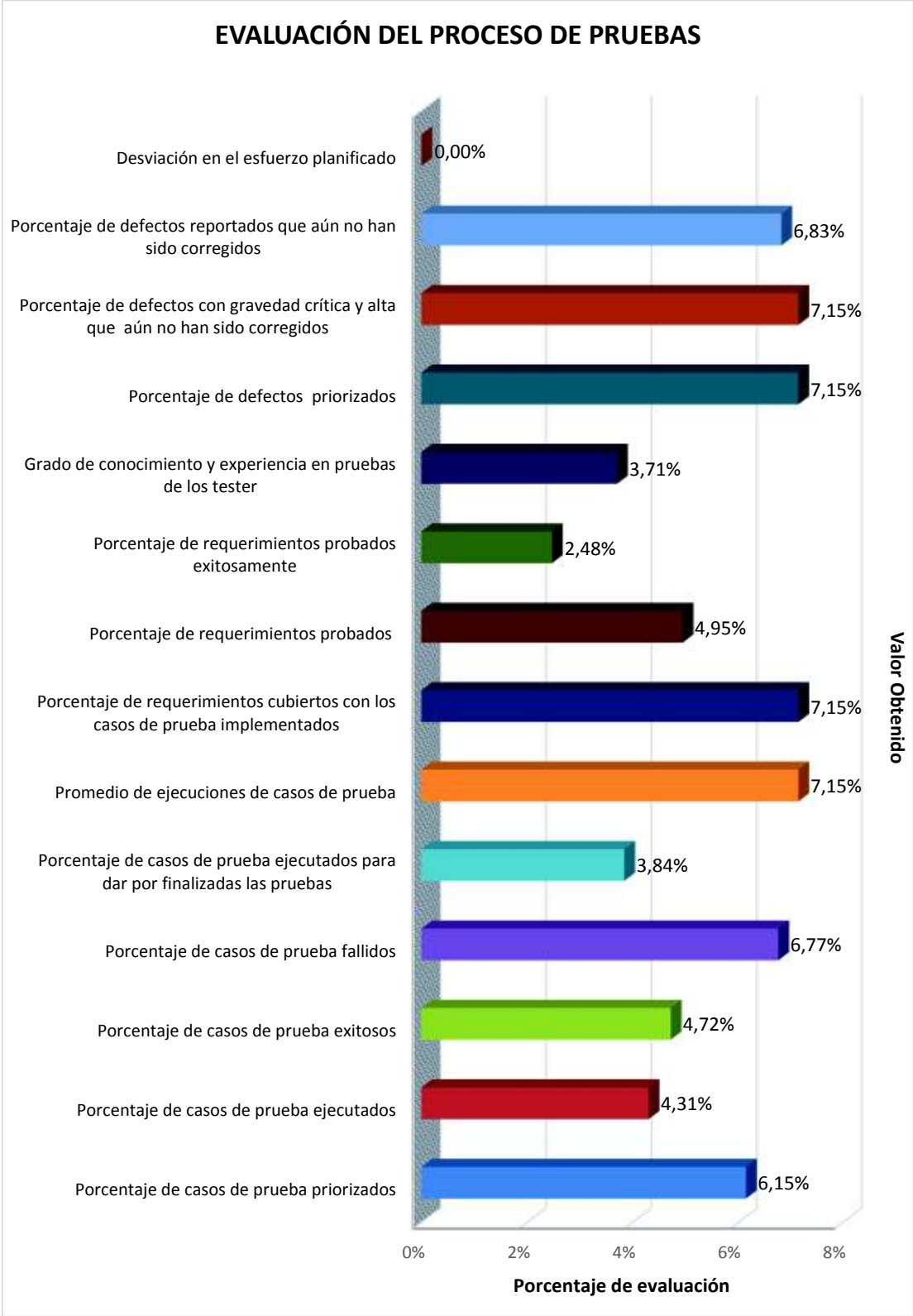


Figura 3.6 Resultados obtenidos de la evaluación del proceso de pruebas en el proyecto del caso de estudio 3

Todas las métricas han sido evaluadas sobre 7,15% con el fin de completar un 100% entre la suma de cada una de los elementos evaluados.

Una vez realizada la evaluación de calidad del proceso de pruebas, en la Figura 3.6 se puede ver que se obtuvo 0% de 7,15% en la desviación en el esfuerzo planificado, esto nos indica que el tiempo planificado es menor al tiempo real empleado para ejecutar el proceso de pruebas por completo, lo cual representa un gran problema para el proceso.

También, se puede ver que se tiene un 2,48% de 7,15% en el porcentaje de requerimientos probados exitosamente, lo que indica que no se ha terminado de verificar exitosamente todos los requerimientos implementados en el proyecto.

Adicionalmente, se puede ver que se tiene un 3,71% de 7,15% en el grado de conocimiento y experiencia de los Tester, lo que nos indica que el equipo de pruebas asignado al proyecto no tenía el conocimiento suficiente de la aplicación que iba a validar y le faltaba experiencia en pruebas. Esto provoca que no se pueda ejecutar el proceso de pruebas de forma correcta y completa.

Además, se puede ver que se tiene un 3,84% de 7,15% en el porcentaje de pruebas ejecutadas para dar por finalizadas las pruebas, lo que nos indica que existen dos problemas: por un lado, no se han priorizado correctamente los casos de prueba diseñados, y por otro lado, no se ha ejecutado los casos de prueba realmente importantes para validar los requerimientos implementados.

De los 14 elementos evaluados en el proceso de pruebas, solo 4 de ellos han obtenido el porcentaje ideal del 7,5%. Esto nos indica que es importante que trabajemos en el resto de elementos que no obtuvieron el porcentaje ideal, con el fin de mejorar el proceso de pruebas de acuerdo a los puntos claves que requiere la empresa.

En la tabla 3.9 se puede ver la calificación de calidad sobre 10 que ha obtenido el producto de software y el proceso de pruebas del proyecto del caso de estudio.

Tabla 3.9 Calificación de calidad del producto y del proceso de pruebas desarrollado en el proyecto del caso de estudio 3

	CALIFICACIÓN DE CALIDAD	NIVELES DE PUNTUACIÓN	GRADO DE SATISFACCIÓN
Producto de Software	6,26 / 10	Aceptable	Satisfactorio
Proceso de Pruebas	7,23 / 10	Aceptable	Satisfactorio

Se usó la Tabla 2.12 para identificar los niveles de puntuación y los grados de satisfacción.

3.1.3.4 Lecciones aprendidas

- Este proyecto obtuvo las calificaciones más bajas tanto en la calidad del producto de software como en la calidad del proceso de pruebas. Esto se debe a que, de los 93 casos de prueba implementados, solo se han ejecutado 56. Este número se debe a la complejidad con la cual se creó cada uno de los casos de prueba y al alto número de iteraciones definidas para cada uno de ellos.
- Las pruebas se centraron en terminar de ejecutar todas las iteraciones de cada uno de los casos de prueba diseñados; lo que fue una mala estrategia debido a que se terminó el tiempo planificado y no se pudo completar el proceso de pruebas de forma satisfactoria.
- En la evaluación de este proyecto se puede ver claramente que una baja calidad en el proceso de pruebas, influye en la baja calidad del producto desarrolla.

3.2 Evaluación por parte de los usuarios finales

Para evaluar las métricas implementadas en el presente trabajo se procedió a aplicar la encuesta de evaluación diseñada en el capítulo 2 sección 2.3.4 en la Tabla 2.15, una vez aplica la encuesta se tabularon las respuestas obtenidas:

Tabla 3.10 Resultados obtenidos de las encuestas aplicadas relacionadas a las preguntas enfocadas en la Utilidad percibida, elaborado por: Caiza Gabriela

		VALOR				
N°	ENFOQUE	1	2	3	4	5
PREGUNTAS		Totalmente en Desacuerdo	En Desacuerdo	Ni En Desacuerdo Ni Acuerdo	De Acuerdo	Totalmente Desacuerdo
EVALUAR LAS MÉTRICAS PROPUESTAS		Sumatoria de respuestas obtenidas en las encuestas				
1	La información obtenida al aplicar las métricas es la información esperada.	0	0	0	8	4
2	La información obtenida al aplicar las métricas permite tomar acciones para mejorar el proceso de prueba.	0	0	1	6	5
3	Las métricas propuestas están enfocadas a los temas que realmente se necesitaba evaluar del proceso de pruebas.	0	0	1	4	7
4	Utilidad percibida Siento que la propuesta planteada será un aporte para mejorar el proceso de pruebas.	0	0	0	5	7
5	Siento que usando las métricas propuestas es más fácil evaluar el proceso de pruebas que sin ellas.	0	0	0	4	8
6	Siento que las métricas implementadas serán un aporte para realizar mejor mi trabajo.	0	0	2	6	4
TOTALES		0 0,00%	0 0,00%	4 5,56%	33 45,83%	35 48,61%
						72 100,00%

En la Tabla 3.10 se tabularon las respuestas obtenidas de las preguntas enfocadas a la utilidad percibida de las cuales se obtuvieron los siguientes resultados:

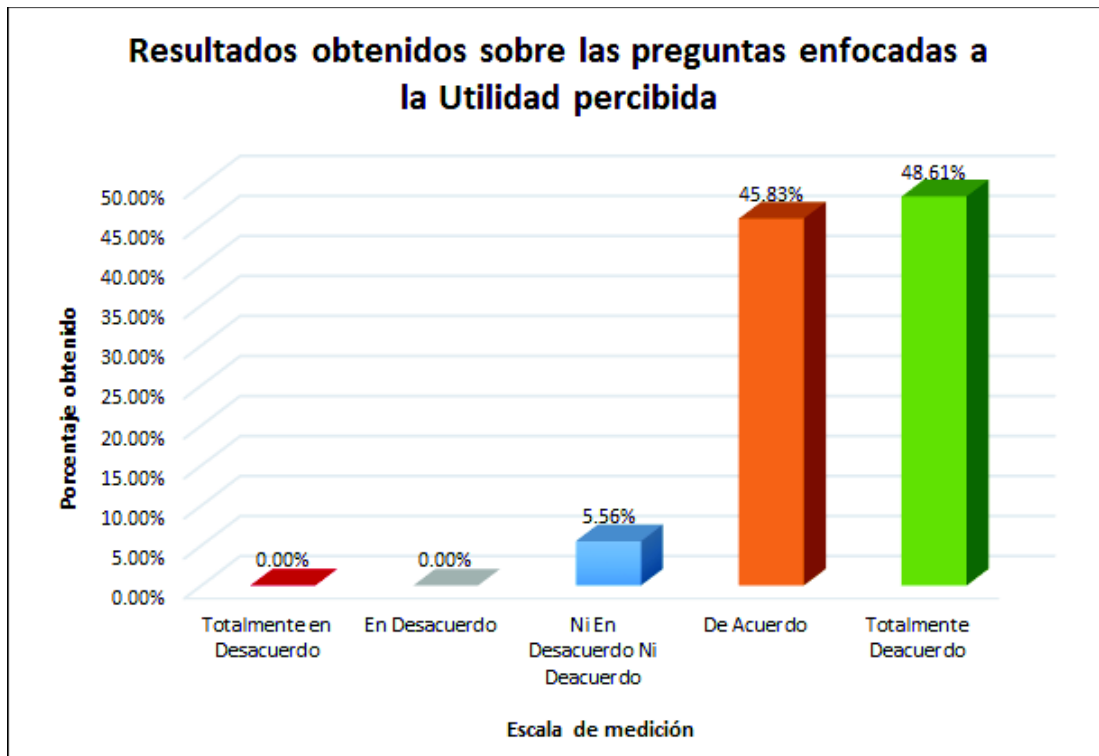


Figura 3.7 Resultados consolidados de la encuesta aplicada correspondiente a las preguntas enfocadas a la Utilidad percibida

En la Figura 3.7 se puede observar los resultados obtenidos de las encuestas aplicadas a 12 personas de la empresa LOGICIEL, las cuales constituyen el 100% del universo encuestado.

Se puede ver que el 94,44% de las respuestas fueron favorables a la idea de que las métricas implementadas serán de utilidad para la empresa.

Tabla 3.11 Resultados obtenidos de las encuestas aplicadas relacionadas a las preguntas enfocadas en la Facilidad de uso percibida, elaborado por: Caiza Gabriela

		VALOR												
N°	ENFOQUE	PREGUNTAS	1		2		3		4		5			
			Totalmente en Desacuerdo	En Desacuerdo	Ni En Desacuerdo Ni Acuerdo	En Desacuerdo	Ni En Desacuerdo Ni Acuerdo	De Acuerdo	Totalmente Desacuerdo					
		EVALUAR LAS MÉTRICAS PROPUESTAS	Sumatoria de respuestas obtenidas en las encuestas											
7	Facilidad de uso percibida	Encuentro que las métricas implementadas son fáciles de usar.	0	0	4	6	2							
8		Las métricas propuestas son claras y entendibles.	0	0	1	8	3							
9		Creo que la propuesta implementada me permitirá mejorar la productividad del equipo de pruebas.	0	0	2	5	5							
10		Me siento capaz de interpretar la información obtenida al aplicar las métricas implementadas.	0	0	1	9	2							
11		Encuentro que es fácil conseguir el resultado una vez aplicadas las métricas implementadas.	0	0	3	7	2							
12		Si pudiera elegir, seguiría usando las métricas propuestas.	0	0	2	6	4							
TOTALES			0	0	13	41	18	0,00%	0,00%	18,06%	56,94%	25,00%	72	100,00%

En la Tabla 3.11 se tabularon las respuestas obtenidas de las preguntas enfocadas a la facilidad de uso percibida de las cuales se obtuvieron los siguientes resultados:

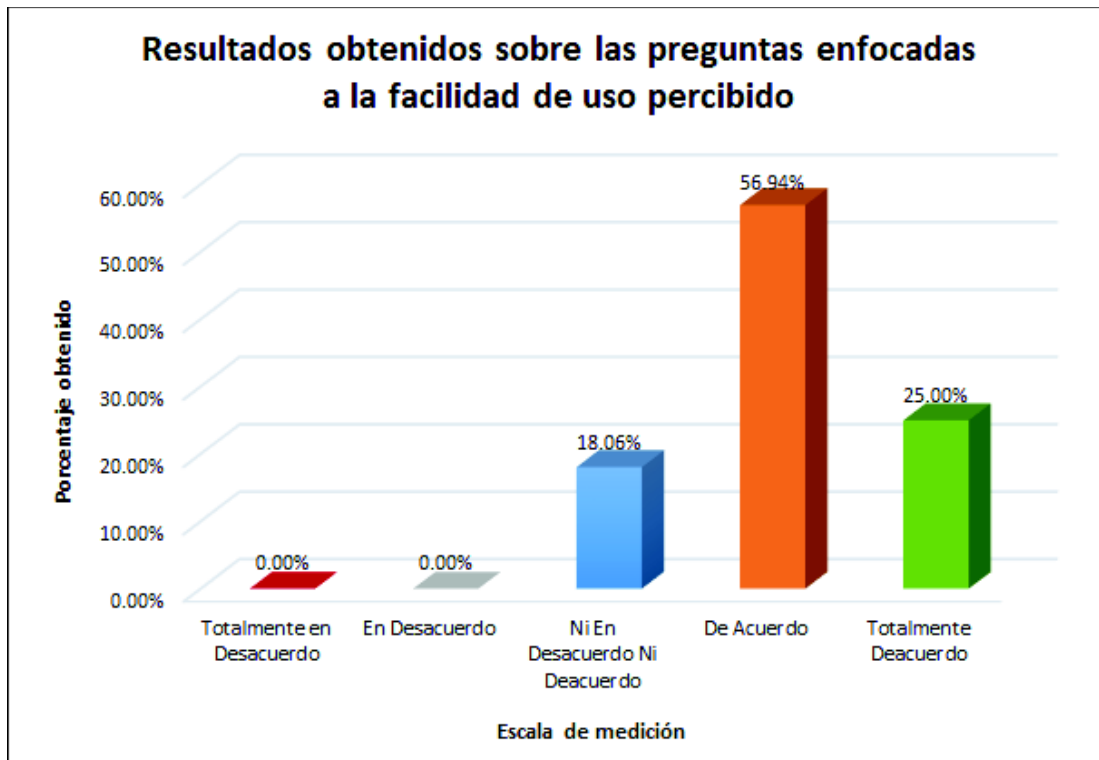


Figura 3.8 Resultados obtenidos de la encuesta aplicada correspondiente a las preguntas enfocadas a la Facilidad de uso percibida

En la Figura 3.8 se puede observar los resultados obtenidos de las encuestas aplicadas a 12 personas de la empresa LOGICIEL, las cuales constituyen el 100% del universo encuestado.

Se puede ver que el 81,94% corresponden a respuestas favorables a la idea de que las métricas implementadas son fáciles de usar para la empresa.

De acuerdo a los resultados obtenidos en la encuesta aplicada a las 12 personas de la empresa LOGICIEL, se puede notar que las métricas implementadas en este trabajo para evaluar el producto de software desarrollo y el proceso de pruebas si será de utilidad para la empresa

3.3 Mejoras para el proceso de pruebas

De acuerdo a los resultados obtenidos de la evaluación realizada a los tres casos de estudio, se proponen las siguientes mejoras para el proceso de pruebas:

- Se recomienda registrar y actualizar la documentación de los proyectos en los respectivos repositorios creados, con el fin de poder llevar un control de los mismos.
- A pesar de que se utilice una herramienta para la gestión de pruebas se requiere definir estándares y lineamientos que permitan dar un mejor uso de la misma.
- Se recomienda realizar una mejor planificación para el proceso de pruebas, teniendo en cuenta que las pruebas no solo constituyen una actividad aislada, sino que implican varias actividades que conforman todo un proceso del cual depende la calidad del producto que va a ser entregado a los clientes.
- Se recomienda incluir a todo el equipo de pruebas en el proyecto desde el inicio, con el fin de que adquieran el respectivo conocimiento del producto de software a ser desarrollado. Esto influirá directamente en el proceso de pruebas.
- Se debe priorizar los casos de prueba siguiendo una escala de medición estandarizada con el fin de que se ejecuten de forma adecuada todos los casos de prueba diseñados para validar los requerimientos implementados correctamente.
- Se debe dar un seguimiento continuo a los casos de prueba, con el fin de tener un control de los mismos e ir verificando semana a semana que pasa con la ejecución de los casos de prueba. Adicionalmente, es importante tener en cuenta que se requieren levantar alertas sobre los casos de prueba con estado fallido, ya que el tener un caso de prueba fallido implica que se tiene que esperar un tiempo no determinado para su corrección y adicionalmente se tiene que volver a ejecutar por completo todo el caso de prueba.
- No es recomendable crear casos de pruebas con demasiadas iteraciones. De ser necesario se recomienda aplicar las diferentes técnicas de diseño que existen para producir un plan de pruebas ejecutable.

- Se recomienda reforzar las capacitaciones a los Tester para mejorar el conocimiento sobre el proceso de pruebas y la herramienta Test Manager que se usa la empresa para la gestión de pruebas.
- La experiencia en pruebas de los Tester es de mucha ayuda durante el proceso de pruebas por lo que se recomienda siempre asignar el personal a los diferentes proyectos de forma equitativa.
- Se deben priorizar los defectos detectados de forma adecuada con el fin de que se permita dar un orden de corrección a los desarrolladores. Esto permitirá agilizar la corrección de los defectos reportados y su respectiva verificación. El tiempo empleado para la corrección de errores reportados durante el proceso de pruebas es un factor importante tanto para la calidad del producto de software desarrollado, como para la calidad del proceso de pruebas.
- Se requiere que durante la planificación del proyecto se tomen en cuenta los tiempos que se requieren para la corrección de defectos y verificación de las correcciones realizadas.

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Para diagnosticar el proceso de pruebas de la empresa LOGICIEL, se aplicaron entrevistas al personal de la organización de diferentes áreas y cargos. Los resultados obtenidos fueron codificados de acuerdo a una guía de codificación implementada y se obtuvieron los verbatines más representativos de cada uno de los temas abordados en las entrevistas y en base a ello se pudo diagnosticar el proceso de pruebas
- Para evaluar el proceso de control de calidad de la empresa LOGICIEL se implementaron métricas, tanto para evaluar el producto de software como el proceso de pruebas de acuerdo a las necesidades e información que dispone la empresa actualmente.
- Las métricas implementadas se aplicaron a tres casos de estudio y se evaluaron los resultados obtenidos. Del análisis realizado se pudo detectar varios problemas que existen actualmente en el proceso de pruebas, los cuales están relacionados principalmente con: la planificación del proyecto, el grado de conocimiento y experiencia de los Tester y el correcto diseño de los casos de prueba.
- De los resultados obtenidos de la aplicación de las métricas implementadas se formularon varias mejoras para el proceso de pruebas de la empresa LOGICIEL, los cuales se detallan en la sección 3.3 del presente trabajo.
- Para evaluar cualquier de proceso de control de calidad usando métricas, se requiere tener un amplio conocimiento y dominio del proceso a ser evaluado con el fin de seleccionar las métricas más adecuadas que permitan realizar una evaluación objetiva del mismo.
- La implementación de métricas para evaluar el proceso de control de calidad no es una tarea fácil, ya que se requiere tener en cuenta varios aspectos importantes entre los cuales podemos mencionar: conocer por completo el proceso a ser evaluado, tener claramente definidos los objetivos de evaluación, conocer

diferentes metodologías para implementar métricas, y conocer todas las métricas existentes que nos ayuden en la implementación.

- Una vez implementadas las métricas que se requieran para evaluar un proceso, es importante realizar un seguimiento de los resultados obtenidos de las mismas con el fin de poder verificar si están cumpliendo los objetivos para los cuales fueron implementadas.
- De acuerdo a los resultados obtenidos en los tres casos de estudio evaluados en el presente trabajo con las métricas implementadas, se pudo constatar que la calidad del proceso de pruebas influye en la calidad del producto de software desarrollado.
- La selección e implementación de las métricas en este presente trabajo, se realizó con el objetivo de ayudar a los directivos a la toma de decisiones y a mejorar la calidad del proceso de pruebas y del producto de software. Además, estas métricas se incluyen en las pruebas de sistema con la idea de brindarle al equipo de desarrollo una evaluación de su producto desarrollado.
- A pesar de que se utilicen herramientas para la gestión de pruebas, se requieren definir estándares y lineamientos que permitan dar un mejor uso de este tipo de herramientas.

4.2 Recomendaciones

- Se recomienda que antes de implementar cualquier tipo de métrica, se realice una investigación previa de los diferentes tipos de metodologías existentes para su implementación, con el fin seleccionar la metodología más adecuada que nos permita alcanzar los objetivos planteados.
- Se recomienda que los datos requeridos para aplicar las métricas sean fáciles de recolectar y fáciles de reportar.
- Se recomienda que las métricas implementadas sean fáciles de entender.
- Se recomienda dar a conocer y concienciar a todo el personal de la empresa, de los beneficios que implica tener un proceso de control de calidad bien definido y controlado.
- Se recomienda almacenar toda la información recopilada a lo largo de los diferentes proyectos sobre el proceso de pruebas, a fin de resguardar toda aquella información útil e indispensable para proyectos similares a futuro.
- Se recomienda realizar una mejor planificación para el proceso de pruebas, teniendo en cuenta que las pruebas no solo constituyen una actividad aislada, sino que implican varias actividades que conforman todo un proceso del cual depende la calidad del producto de software desarrollado.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] LOGICIEL CIA. LTDA., «Planificación Estratégica,» 2012.
- [2] LOGICIEL CIA. LTDA, «Gestión de Calidad de Software - Pruebas del Software,» Quito, 2010.
- [3] LOGICIEL CIA. LTDA, «Modelo de Desarrollo de Sistemas de información,» Quito, 2010.
- [4] R. S. Pressman, Ingeniería de Software, Séptima ed., New York: Mac Graw Hill, 2010.
- [5] The Institute of Electrical and Electronics Engineers, IEEE 610 - Standard Glossary of Software Engineering Terminology, New York, 1990.
- [6] M. Callejas Cuervi, A. C. Alarcón Aldana y A. M. Álvarez Carreño, «Modelo de calidad del software, un estado del arte,» *Entramado*, vol. 13, nº 1, pp. 236-250, 2017.
- [7] F. A. Herrera Jiménez y E. R. Ortiz Jurado, «Guía para la aplicación de métricas para determinar la calidad de un sistema software,» Escuela Politécnica Nacional, Quito, 2010.
- [8] A. M. García Sánchez, «Evaluación de métricas de calidad del software sobre un programa Java,» Universidad Complutense de Madrid, Madrid, 2010.
- [9] I. Darrel y H. Sharp, Planificación y Gestión de Sistemas de Información, McGraw-Hill, 1993.
- [10] I. Sommerville, Ingeniería del Software, Novena ed., Addison-Wesley, 2011.
- [11] A. V. Lopez, A. Sánchez y G. A. Montejano, «Definición de Métricas de Calidad para Productos de Software,» de *XVIII Workshop de Investigadores en Ciencias de la Computación*, Argentina, 2016.
- [12] N. Fenton y J. Bieman, Software Metric A Rigorous and Practical Approach, New York: Taylor y Francis Group, 2015.
- [13] ISTQB - ISEB, Software Testing - Foundation Guide, 2012.
- [14] L. Pauta Ayabaca y S. Moscoso Bernal, «Verificación y Validación de Software,» *Revista Killkana Técnica*, vol. 1, nº 3, pp. 25-32, 2017.
- [15] Y. León Perdomo, A. Enrique Góngora y A. Febles Estrada, «Aplicando métricas de calidad a proyectos y procesos durante las pruebas exploratorias,» *Revista Cubana de Ciencias Informáticas*, vol. 7, nº 2, pp. 193-205, 2013.

- [16] D. M. Torres Ricaurte, « Un método para medir la productividad del equipo de pruebas en la estimación del esfuerzo de pruebas de software,» Universidad Nacional de Colombia, Medellín, 2013.
- [17] M. Estayno, G. Dapozo, L. Cuenca y C. Greiner, «Modelos y Métricas para Evaluar la Calidad de Software,» *Departamento de Informática. Facultad de Ingeniería. Universidad Nacional de Lomas de Zamora.*
- [18] F. Scolone, «Estudio comparativo de los modelos y estándares de calidad del software,» *Universidad Tecnológica Nacional,* 2006.
- [19] ISO/IEC 9126-1, International Standard, Software Engineering - Product Quality - Part 1: Quality Model, Suiza: Final Draft, 2000.
- [20] ISO/IEC 9126 - 2, «International Standard, Software Engineering - Product Quality - Part 2: External Metrics,» Suiza, 2002.
- [21] ISO/IEC 9126 - 3, «International Standard, Software Engineering - Product Quality - Part 3: Internal Metrics,» Suiza, 2002.
- [22] ISO/IEC 9126 - 4, «International Standard, Software Engineering - Product Quality - Part 4: Quality in use Metrics,» Suiza, 2002.
- [23] ISO/IEC 14598-1, «ISO/IEC 14598-1:2000 - Information technology - Software product evaluation - Part 1: General Overview,» *Reference number ISO/IEC 14598-1:1999,* 2000.
- [24] ISO/IEC 25000, ISO/IEC 25000:2005 - Software engineering - Software Product Quality Requirements and Evaluation Square - Guide to Square, Reference number ISO/IEC 25000.
- [25] ISO/IEC 25023, «Use of System and software product quality Measures,» de *Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Measurement of system and software product quality,* 2011, pp. 14-15.
- [26] ISO/IEC 25022, «Quality in use measures,» de *Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Measurement of quality in use,* 2012, pp. 13-20.
- [27] ISO/IEC 25023, «Required Quality Measure,» de *Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Measurement of system and software product quality,* 2011, pp. 16-32.
- [28] N. Lascano, «Un Conjunto de Métricas para Proyectos de Transición de Software Offshore,» 05 2013. [En línea]. Available:

- [http://sedici.unlp.edu.ar/bitstream/handle/10915/32513/Documento_completo__4.4\)%20Un%20Conjunto%20de%20M%C3%A9tricas%20para%20Proyectos%20de%20Transici%C3%B3n%20de%20Software%20Offshore.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/32513/Documento_completo__4.4)%20Un%20Conjunto%20de%20M%C3%A9tricas%20para%20Proyectos%20de%20Transici%C3%B3n%20de%20Software%20Offshore.pdf?sequence=1). [Último acceso: 29 09 2018].
- [29] F. García, «Medición de Software,» 06 2010. [En línea]. Available: <https://alarcos.esi.uclm.es/per/fruiz/curs/santander/fgarcia-medicion-update.pdf>. [Último acceso: 30 09 2018].
- [30] E. F. Loza Aguirre y A. F. Buitrago Hurtado, «Qualitative assessment of user acceptance within Action Design Research and Action Research: two case studies,» *Latin American Journal of Computing LAJC*, vol. 1, nº 1, 2014.
- [31] R. Baskerville y A. Wood-Harper, «Diversity in information systems action research methods,» *European Journal of Information Systems*, vol. 7, pp. 90-106, 1998.
- [32] R. L. Baskerville, «Investigating Information Systems With Action Research,» *Communcations of the Association for Information System*, vol. 2, nº 19, 1999.
- [33] TMMI Foundation, «Test Maturity Model integration,» Erik van Veenendaal, Irlanda, 2015.
- [34] Microsoft , *Herramienta Test Manager*, 2016.
- [35] E. A. Balseva Chisaguano, «Evaluación de Calidad de Producto Software en Empresas de Desarrollo de Software Aplicando la Norma ISO/IEC 25000,» Escuela Politécnica Nacional, Quito, 2014.
- [36] L. A. Yong Varela, «Modelo de aceptación tecnológica (TAM) para determinar los efectos de las dimensiones de cultura,» *Revista Internacional de Ciencias Sociales y Humanidades, SOCIOTAM*, vol. XIV, nº 1, pp. 131-171, 2004.
- [37] S. R. Súa, «Análisis estadístico comparativo de tres escalas de valoración: Likert, fuzzy-Likert y fuzzy de respuesta libre,» Universidad de Oviedo, Oviedo, 2012.
- [38] QuestionPro, «¿Qué es la escala de Likert y como utilizarla?,» 2018. [En línea]. Available: <https://www.questionpro.com/blog/es/que-es-la-escala-de-likert-y-como-utilizarla/>. [Último acceso: 27 09 2018].
- [39] J. Cordero Cajiao, «Escala Liker,» 6 10 2016. [En línea]. Available: <https://www.slideshare.net/Manchas44/escala-de-likert-66818914>. [Último acceso: 27 09 2018].

6 ANEXOS

Los anexos que se enumeran a continuación se encuentran en el disco adjunto.

ANEXO I: ENTREVISTAS REALIZADAS

ANEXO II: CODIFICACIÓN DE ENTREVISTAS

ANEXO III: PLANTILLAS PARA LA RECOLECCIÓN DE DATOS

ANEXO IV: MATRICES DE EVALUACIÓN

ANEXO V: EVALUACIÓN CASO DE ESTUDIO 1

ANEXO VI: EVALUACIÓN CASO DE ESTUDIO 2

ANEXO VII: EVALUACIÓN CASO DE ESTUDIO 3