



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A

TÍTULO PROYECTO

Caracterización de métricas de calidad orientadas al proceso de pruebas de software

PROPONENTE(S)

Brian Hernando Rodríguez Rodríguez
1049658676
2273600

DIRECTOR

Mg. Martha Susana Contreras Ortiz

Tunja

Fecha de presentación (Julio 25 de 2024)

CONTENIDO

<u>1.</u>	<u>FICHA TÉCNICA DEL PROYECTO</u>	<u>3</u>
<u>2.</u>	<u>PLANTEAMIENTO DEL PROBLEMA</u>	<u>4</u>
<u>3.</u>	<u>JUSTIFICACIÓN</u>	<u>6</u>
<u>4.</u>	<u>OBJETIVOS</u>	<u>7</u>
<u>5.</u>	<u>ESTADO DEL ARTE</u>	<u>8</u>
<u>6.</u>	<u>METODOLOGÍA.....</u>	<u>17</u>
<u>7.</u>	<u>DESARROLLO METODOLÓGICO</u>	<u>19</u>
7.1.	FORMULACIÓN DE LA PREGUNTA DE INVESTIGACIÓN.....	19
7.2.	BÚSQUEDA DE LITERATURA O FUENTES DE INFORMACIÓN	20
7.3.	ESTRATEGIA DE BÚSQUEDA	29
7.4.	PROCESO DE SELECCIÓN	30
7.5.	EXTRACCIÓN DE DATOS	32
7.6.	EVALUACIÓN DE LA CALIDAD DE LOS ESTUDIOS.....	33
7.7.	SÍNTESIS DE LOS RESULTADOS.....	34
7.8.	INFORME DE LA REVISIÓN: CARACTERIZACIÓN DE MÉTRICAS DE CALIDAD ORIENTADAS AL PROCESO DE PRUEBAS DE SOFTWARE	42
<u>8.</u>	<u>CONCLUSIONES Y ANÁLISIS DE RESULTADOS</u>	<u>51</u>
<u>9.</u>	<u>REFERENCIAS</u>	<u>53</u>
<u>10.</u>	<u>ANEXOS.....</u>	<u>57</u>

1. FICHA TÉCNICA DEL PROYECTO

Título	Caracterización de métricas de calidad orientadas al proceso de pruebas de software
Autor (es)	Brian Hernando Rodríguez Rodríguez
Director del proyecto	Mg. Martha Susana Contreras Ortiz
Palabras claves	Calidad del software, Ingeniería del software, Métricas de calidad de software, Testeo de software
Descripción	<p>Proyecto que consistió en desarrollar un análisis de las métricas de calidad implementadas en el proceso de pruebas en el software, que permitió caracterizar los diferentes grupos de métricas y sus respectivas propiedades. Esto mediante un análisis bibliográfico en fuentes de alto impacto para reunir los datos de estas y su posterior estudio. De igual manera, se consultó a expertos en el desarrollo de software, enfatizando en la aplicación de las métricas de calidad con enfoque a la fase de testing de software que presentan en sus proyectos de trabajo y/o que hayan tenido participación.</p>

2. PLANTEAMIENTO DEL PROBLEMA

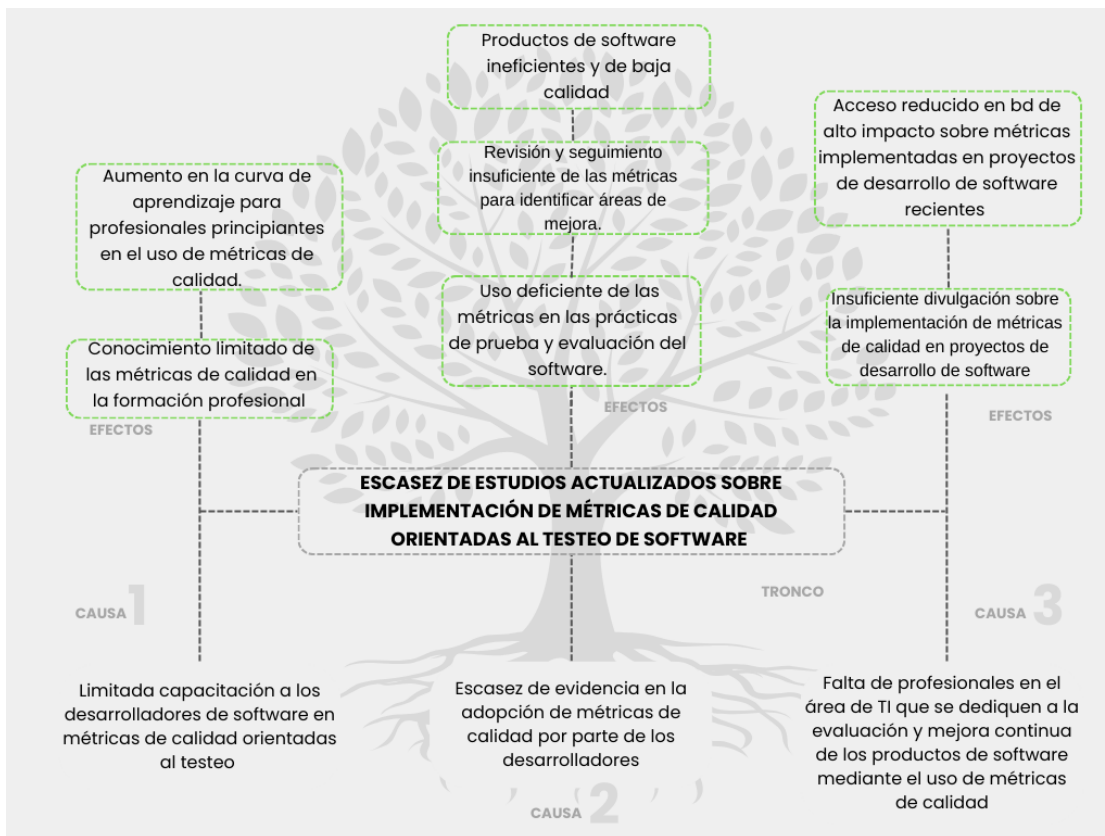
La calidad del software es un pilar fundamental en el desarrollo de los proyectos de una organización. Un software de alta calidad no solo cumple con las funcionalidades y requisitos establecidos (G. L. Saini et al., 2020), sino que significa una experiencia de usuario superior, garantiza la fiabilidad, seguridad y contribuye a la eficiencia y productividad de las operaciones en cada una de las fases del ciclo de vida del desarrollo del software.

Garantizar la calidad de un software no es una tarea sencilla, especialmente en un entorno tecnológico en movimiento y evolución, ya que hay problemas que la afectan y que pueden comprometer la excelencia de los productos. En este contexto, las métricas de calidad adquieren un gran protagonismo, pues se convierten en herramientas esenciales para evaluar el resultado de los productos y así garantizar su correcta funcionalidad y sencillo mantenimiento (Deshpande et al., 2023). En consecuencia, un producto de software con mala calidad o un mal funcionamiento es consecuencia de una incorrecta o nula implementación de las métricas de calidad en cualquier fase del ciclo de vida de un proyecto de desarrollo de software (Garousi et al., 2018). Esto ocurre porque no se tiene suficiente información o incluso desconocimiento total de la implementación de las métricas en los entornos de desarrollo, testing entre otros.

El proceso de testing de un proyecto implica escoger e integrar las métricas adecuadas en los entornos de prueba en cada una de las fases del ciclo de vida del mismo (Sivakrishna et al., 2023a). Para lograrlo, el equipo o persona encargada de esta tarea debe estar capacitado para usar las métricas de calidad enfocadas a revisar la efectividad del proceso de testeado de software (Chahar & Bhatia, 2023). El uso de éstas puede ayudar a superar las dificultades que los productos y servicios de desarrollo de software (Eisty et al., 2018), creando así un valor agregado distintivo para el responsable del proyecto en un mercado en constante competencia. Esto conllevará a una mejora en la eficiencia y a su vez en la productividad, también desembocará en la reducción de costos, así como la satisfacción de los clientes ya que, se puede alcanzar el cumplimiento de los requisitos regulatorios y así mejorar la gestión general del proyecto. Por otro lado, la selección incorrecta de métricas de calidad puede tener impactos negativos, tanto en la etapa de testeado como en cualquier otra etapa dentro del proyecto como la insatisfacción de los clientes, pérdida de

participación en el mercado, aumento de costos y disminución de la eficiencia en la producción (Guamán, 2024). A pesar de que es una tarea vital para la evaluación de los productos de software, hay escasez de estudios en donde se proyecte la aplicación de las métricas de calidad, lo que resulta en el desconocimiento de éstas y por consiguiente su poco uso (Figura 1). No solamente las métricas de calidad presentan esta situación, también existe déficit de documentación de los estándares de calidad de software y por consiguiente su poco uso en proyectos de desarrollo. Tal es el caso de los estándares ISO 25010 e ISO 5055 que están presentes y vigentes en la norma ISO, pero aun así no hay mucha literatura sobre ellos y por consiguiente de su implementación. Por eso surge la necesidad de caracterizar las métricas de calidad que estén enfocadas al proceso de testing, para enriquecer y aportar aún más al conocimiento del profesional en ingeniería de sistemas y así lograr aumentar el número de productos de desarrollo de software con una excelente calidad, funcionamiento, mantenibilidad y por supuesto, éxito en el mercado.

Figura 1. Árbol de Problemas



Fuente: Autor

3. JUSTIFICACIÓN

La inclusión de métricas de calidad en la etapa de pruebas en el desarrollo de software es un paso importante que contribuye a mejorar la calidad del producto final (Cibir & Ayyildiz, 2022). Proporciona información valiosa para una evaluación objetiva de la calidad del software y la identificación de áreas que se pueden mejorar.

Al ejecutar una medición sistemática de la calidad se consigue optimizar el proyecto de desarrollo, aprovechar mejor los recursos y el tiempo, y reducir significativamente los costos a corto y largo plazo. Estos criterios de calidad son herramientas útiles para cumplir con los requisitos y estándares de los clientes, prevenir errores y fallas, y garantizar la fiabilidad y seguridad del producto final (Fuentes et al., 2021). Durante el proceso de pruebas de software, es posible identificar debilidades e ineficiencias, así como determinar áreas de mejora para aumentar la eficiencia y el rendimiento (Wang et al., 2019).

De acuerdo con las situaciones y problemática abordada previamente, se entiende que hay una oportunidad de mejora significativa en el aprendizaje, entendimiento y uso de las métricas de calidad en proyectos de desarrollo de software. Por eso la caracterización de métricas de calidad enfocadas al proceso de testeo de software contribuye a adquirir conocimiento de estas y propone su actualización e integración en los futuros proyectos de desarrollo de software dentro de la comunidad tomasina.

Cabe resaltar que el testing de software reporta una relevancia constante durante los últimos años en el desarrollo de proyectos importantes (Ramadhina et al., 2021). Asimismo, el área de QA (Quality Assurance) viene teniendo un mayor crecimiento últimamente, ya que estos dos campos cada vez son más demandados en los mismos, lo que representa un incremento en la calidad de los productos de software.

Logrando implementar las métricas de calidad en la fase de testing de los proyectos de desarrollo de software, los conocimientos adquiridos por los estudiantes sobre sus características y uso aportarán a la mejora y calidad significativa de los productos de software tanto en los proyectos de los personales como los institucionales, aportando al prestigio y eficacia de la universidad y de los mismos profesionales.

4. OBJETIVOS

4.1. Objetivo General

Caracterizar las métricas de calidad orientadas al testeado de software, para analizar su implementación actual en proyectos de desarrollo de software, a través de un proceso de revisión sistemático de literatura en fuentes de alto impacto.

4.2. Objetivos específicos

A continuación, se presentan los objetivos específicos.

Tabla 1. Objetivos específicos

Nro.	Objetivo específico
1	Describir métricas de calidad vigentes orientadas al testeado de software, a través de un proceso de revisión de literatura de los últimos 5 años, identificando sus correspondientes características.
2	Realizar un análisis comparativo de los hallazgos de la revisión de literatura, con entrevistas a expertos en desarrollo de software, para complementar la descripción de las métricas.
3	Construir una guía de caracterización de métricas de calidad para estudiantes de Ingeniería de Sistemas y afines, que permitan evaluar la calidad de las técnicas de testeado implementadas en sus proyectos de desarrollo de software.

Fuente: Autor

5. ESTADO DEL ARTE

A través de este trabajo se examina la evolución de la calidad del software a lo largo de los últimos cinco años y utiliza los conocimientos actuales para identificar tendencias, logros y áreas de mejora. Este análisis es clave para comprender el estado actual del testing y su impacto en el desarrollo de software.

Esta revisión de literatura permitió identificar los enfoques más recientes en el contexto del testeo de software, centrándose en los criterios de calidad aplicados. También se lograron analizar los métodos y modelos utilizados en diferentes estudios para reconocer las estrategias más eficaces y las posibles limitaciones para el aseguramiento de la calidad del software.

5.1 ANÁLISIS DE MÉTRICAS

(Mladenova, 2020) El artículo de conferencia aborda una variedad de modelos y métricas de calidad, como las métricas de proyecto, proceso y producto. Ejemplos de métricas conocidas tales como tamaño, complejidad, rendimiento, mantenimiento, seguridad y pruebas de software, analiza las métricas de calidad del software y su importancia para evaluar la calidad del desarrollo de software, mediante la revisión de distintos modelos y métricas de calidad, proporcionar ejemplos de métricas y destacar la importancia de implementarlas para contribuir a la mejora de la calidad del software.

Los autores se centran en las métricas de calidad del desarrollo de software como también las enfocadas a la fase del testing, métricas de proyecto, métricas de proceso, y métricas de producto son usadas para diferentes propósitos en el desarrollo y testing de software a lo largo del documento.

Como resultado, los autores destacan que:

- Las métricas de software ayudan a las organizaciones a ganar una mejor comprensión en los procesos de desarrollo y pruebas, asumir riesgos y garantizar la calidad de sus productos y servicios de software.

La evaluación de la calidad es un proceso exhaustivo que debería ejecutarse en cada fase del desarrollo de software

5.2 CALIDAD DEL SOFTWARE

(Sivakrishna et al., 2023) El paper de conferencia consistió en determinar el estado actual del conocimiento en la gestión de calidad de productos de software para orientar futuras investigaciones en este campo.

Los autores señalan que las métricas y evaluaciones de los hallazgos de medición sirven como indicadores confiables de la efectividad de las operaciones organizacionales y/o la calidad del producto. Se enfatiza también en la importancia de definir adecuadamente las métricas para lograr el objetivo utilizando métricas calibradas y validadas.

Como resultado se recalcó la importancia de utilizar métricas de calidad para detectar y corregir defectos en el software y mejorar su rendimiento y calidad. Además, se destacó la importancia de la estandarización de métricas de calidad en la industria de software para garantizar que las pruebas de software sean uniformes y comparables.

5.3 CALIDAD EN TESTING

(Obaid Barraood et al., 2023) El paper de conferencia se basó en identificar las características y métricas utilizadas para medir la calidad de los casos de prueba. El objetivo de este estudio está enfocado a la mejora y medición de la calidad en el testing para garantizar la producción de software de alta calidad y su posterior entrega ágil a los clientes. Los autores afirman que algunas métricas de calidad no están claramente relacionadas con características específicas. Sugieren que sea posible que estas métricas requieran una revisión adicional para determinar las características relacionadas que miden.

Se hallaron varios resultados dentro de los que más se destacan los siguientes:

- Identificaron 50 métricas que se utilizan para medir la calidad de los casos de prueba.
- De los documentos que consultaron en la revisión de literatura, solo unos pocos estudios se centraron en la calidad de los casos de prueba, lo que sugiere la necesidad de más investigaciones en este campo.

5.4 CALIDAD WEB

(Deshpande et al., 2023) El artículo de conferencia examina varios tipos de pruebas, sus ventajas y desventajas, y analiza la importancia y retos de las pruebas de software para mantener la calidad de las aplicaciones web. Por otra parte, presentan el llamado modelo SQA-PAK, un modelo que consiste en varios componentes de aseguramiento de la calidad del software y así discutir el rol del testing de software en la garantía de la calidad de las aplicaciones web.

Los autores a lo largo del estudio examinan cómo las métricas se pueden aplicar en el proceso de evaluación de la calidad de los procesos de desarrollo y pruebas del software y afirman que las métricas se utilizan para evaluar componentes o proyectos de software y medir el éxito de una organización.

También llegaron a las siguientes conclusiones:

- Concluyeron que era necesario mejorar una variedad de componentes de los procesos de desarrollo de software, incluida la gestión de requisitos, la planificación de proyectos, el seguimiento y la supervisión de proyectos, la gestión de configuración y los programas de capacitación.

Una de las categorías de métricas de calidad más empleadas en los procesos de desarrollo y pruebas de software son las métricas de rendimiento o funcionamiento

5.5 COMPROBABILIDAD DEL SOFTWARE

(Albattah, 2022) Consiste en un artículo de conferencia que examina una variedad de métricas relacionadas con el desarrollo y el desarrollo ágil de software. Destaca que las métricas como la cohesión de paquetes, el acoplamiento de paquetes y el tamaño de paquetes son cruciales para predecir la facilidad de prueba del software.

Para predecir la facilidad de prueba del software durante la fase de diseño, propone una métrica de cohesión de paquetes. También demuestra su eficacia para evaluar la cohesión del diseño y predecir los costos de prueba. Asimismo, busca Investigar cómo la verificabilidad del software se relaciona con métricas como la cohesión de paquetes, el acoplamiento y el tamaño. El propósito de la investigación es descubrir cómo estas métricas

pueden predecir la verificabilidad del software, así como sugerir nuevas investigaciones sobre métricas alternativas y su eficacia en varios lenguajes de programación.

Se utilizan ampliamente las métricas de software para predecir los atributos de calidad del software. Destaca que la cohesión del software es una métrica fundamental para medir la cohesión del software y se ha demostrado que proporciona predicciones útiles para una variedad de aspectos, como las pruebas que se le realizan a los productos de software, la propensión a fallos y la gravedad de defectos de estos.

Los autores concluyeron lo siguiente:

- La métrica de cohesión de paquetes resulta ser una herramienta útil para predecir los esfuerzos de pruebas y mantenimiento del software.
- La métrica de cohesión se estableció sobre la base de los principios de cohesión de paquetes y demostró su impacto mediante los análisis estadísticos realizados en la predicción de la mantenibilidad del software y los esfuerzos de prueba.

5.6 ESTIMACIÓN DE ESFUERZO

(Cibir & Ayyildiz, 2022) Este paper de revista examina la creación de una nueva forma de estimar el esfuerzo de pruebas de software en proyectos de defensa utilizando las métricas de testing de software usando la regresión lineal, aplicado a 15 proyectos de software de certificación nivel 3 de CMMI (Capability Maturity Model Integration) ya concluidos. En él se abordan los retos de la estimación del esfuerzo de prueba en proyectos de defensa, brindando un enfoque al mejoramiento de la predicción precisa en la estimación del esfuerzo de pruebas de software.

El estudio considera que las métricas de calidad monitorean y evalúan la calidad de los procesos del testing de software. Entre las métricas utilizadas dentro de la investigación se destaca el tiempo de creación de un ambiente de pruebas, como también la revisión del período de los requisitos de prueba y el tiempo de creación de un plan de ejecución de pruebas de software.

Como hallazgos se pudieron identificar los siguientes:

- Se logra identificar que el método propuesto con las métricas implementadas proporciona una “calidad de predicción aceptable”

- Los valores de regresión lineal del “R-cuadrado” (aptitud de la predicción) resultantes estuvieron por arriba del 99%, lo que indica que las predicciones fueron exitosas.

5.7 EVALUACIÓN DEL TESTING DE SOFTWARE

(Barreto & Cavalcanti, 2019) Este paper de conferencia propone desarrollar un conjunto de métricas para evaluar la efectividad en la ejecución de pruebas de software, utilizando la metodología GQM (Goal-Question-Metric) que comienza con la definición explícita de los objetivos de medición. Estos objetivos se refinan en varias preguntas. Cada pregunta se refina luego en métricas que deben proporcionar información para responder a esas preguntas.

El estudio caracteriza las métricas de software como esenciales para distribuir información de ingeniería y gestión, que facilitan evaluar la calidad del producto de software, garantizar el cumplimiento de plazos y costos del proyecto y mejorar la agilidad de los procesos de desarrollo de software.

Los autores concluyen que las métricas propuestas para durante el proceso de validación, pueden tener oportunidades de mejora tales como la definición de objetivos adicionales referentes a la cantidad de pruebas hechas en un período de tiempo determinado, así como la cantidad y gravedad de las fallas.

5.8 IMPACTO EN EL SOFTWARE

(Hooda et al., 2022) Se propone demostrar la eficacia de la técnica «GA-FC», una técnica implementada en el testing, para mejorar la calidad del software en contraste con otras técnicas de testeo, haciendo énfasis en los factores de calidad externos, internos y la calidad de gestión.

En este paper de conferencia los autores realizan un análisis en el impacto de la calidad del software mediante el uso de «GA-FC» implementando métricas de calidad como corrección, integridad, precisión, adaptabilidad, fiabilidad, y usabilidad para medir su impacto en la calidad del software.

Como resultado de la investigación se lograron apreciar los siguientes hallazgos:

- Se halló que la implementación de "GA-FC" resulta en el mejoramiento de la calidad del software del 4% para aplicaciones de tamaño medio y del 20% para aplicaciones pequeñas.
- Afirman que GA-FC es más eficaz que otras técnicas de testing con el fin de mejorar los aspectos de la calidad del software
- Se presenta un 14% de mejora en los factores de calidad externa y un 4% de incremento en el valor total de la calidad después de haber usado "GA-FC"

5.9 MARCO DE INFRAESTRUCTURA EN TESTING

(Locke et al., 2023) Este paper de conferencia propone el marco QEX: un modelo para integrar fuentes de datos de pruebas comunes con el fin de obtener supervisión sobre el desarrollo de pruebas de una manera que proporcione información transparente clara y útil, diseñado para brindar capacidad de observación sobre el desarrollo de pruebas, en especial a los ingenieros de aseguramiento de calidad (QA).

Se logra reconocer un análisis referente a la importancia de seleccionar y utilizar métricas de calidad en el contexto de la supervisión y la simplicidad de las pruebas y la experiencia del desarrollador de control de calidad al marco QEX.

Por otro lado, enfatizan en la necesidad de una variedad de métricas que puedan proporcionar información significativa a los QA y a otras partes interesadas acerca de la ejecución de las pruebas y su impacto en el ciclo de vida del desarrollo de software.

En él se utilizaron varias métricas y estadísticas relacionadas con la ejecución de pruebas como la puntuación activa, puntuación de estabilidad entre otras.

Los autores destacan la potencial importancia del marco QEX enfatizando en cómo el uso de las métricas de calidad impacta directamente en la experiencia del desarrollador QA

5.10 MEDICIÓN DE LA CALIDAD

(Terragni et al., 2020) En este artículo de conferencia, los autores buscan analizar y explorar la correlación entre métricas de diseño orientado a objetos, métricas de calidad de testing en clases Java de proyectos de código abierto y poder mejorar la predicción del esfuerzo de prueba.

Proponen usar las métricas como la cobertura de líneas, la cobertura de ramas y la puntuación de mutaciones para aproximar la calidad de un conjunto de pruebas en términos de su capacidad para revelar fallos, comprendiendo el uso de las métricas de calidad de pruebas para estandarizar y ajustar el esfuerzo en el proceso de pruebas de software.

Se lograron hallar los siguientes resultados:

- El tamaño, la complejidad, el acoplamiento y la cohesión son las características del diseño orientado a objetos que más influyen en la verificabilidad.
- Se descubrió que normalizar el esfuerzo de prueba con respecto a la calidad de prueba mejoraba significativamente la correlación entre el esfuerzo de prueba y las métricas de clase, lo que resulta en un mayor poder de predicción.

5.11 MODELOS DE CALIDAD

(G. L. Saini et al., 2020) Este artículo de revista buscó realizar un estudio comparativo de diferentes modelos de aseguramiento de la calidad del software mediante la revisión de alrededor de 100 artículos de investigación bien indexados.

Mediante el análisis realizado por los investigadores, se pudo hacer una comparación entre los modelos de aseguramiento de la calidad más utilizados.

También se pudieron identificar los diferentes atributos de calidad que cada uno de los modelos utilizados para la comparación podían brindar y satisfacer en el aseguramiento de la calidad del software.

La investigación arrojó que el modelo ISO 9126 es el modelo más eficiente y utilizable ya que puede adaptarse a cualquier tipo de calidad de software para evaluarla de forma cuantitativa.

5.12 TÉCNICAS DE TESTING

(Neelofar et al., 2023) El artículo de revista busca analizar las suficiencias e insuficiencias de distintas técnicas automatizadas de pruebas de software, usando el marco de análisis del espacio de instancias explorando la aptitud de las técnicas, identificando las fortalezas y debilidades de cada una. Resalta la importancia de métricas comunes para evaluar el rendimiento de las técnicas de testing y cómo estas influyen en el análisis del espacio de instancias del desarrollo de software como lo son la cobertura, tamaño del conjunto de pruebas y el puntaje de mutación.

Los autores concluyen que:

- En los distintos cuadrantes las métricas poseen características que afectan la cobertura de las técnicas de pruebas en mayor o menor facilidad, lo que refleja las respectivas fortalezas y debilidades de cada una
- Los benchmarks (referencias o puntos de comparación usados para medir y evaluar el rendimiento, la calidad o las características de un producto) comúnmente utilizados en pruebas de software, pueden no comprender la amplia gama de instancias necesarias para sintetizar las fortalezas y debilidades de las técnicas de testeo, por lo que sugieren ampliar la variedad de benchmarks empleados.

5.13 TENDENCIA DE ANÁLISIS

(Purohit et al., 2023) El artículo de conferencia consistió en un análisis detallado de las tendencias y las métricas de testing del software. Para lograrlo, se realizó una revisión exhaustiva de la literatura existente sobre la facilidad de prueba del software donde se pudieron identificar una cierta cantidad de métricas así como modelos aplicables al testeo de software. Tuvo como objetivo abordar la falta de investigación y normalización de manera adecuada en el campo de las métricas de testing de software. Ejecutan una revisión de 29 estudios seleccionados para determinar las métricas de testeo utilizadas con frecuencia y varios modelos de testing.

En cuanto a métricas, brindan un profundo análisis de las métricas de calidad en el testeo de software y su importancia dentro del desarrollo de software y en la evaluación de la calidad y el rendimiento de los productos de software. Identifican tres categorías de métricas principales: métricas de proceso, métricas de producto y métricas de proyecto.

Como resultado principal, se identificó la necesidad de estandarización y de más investigación para mejorar la fiabilidad de las métricas de testing del software ya que no hay gran tendencia de documentos de investigación registrados con enfoque en este campo.

5.14 USO DE MÉTRICAS

(Fuentes et al., 2021) El artículo de conferencia aborda la aplicación de métricas en el testing de software. Se aplicaron tres métricas a un proyecto de seguridad que atravesó por una importante rotación de equipos durante el desarrollo. También brinda información valiosa sobre el proceso del proyecto y cómo demostrar la utilidad de estas métricas en un entorno de pruebas. El objetivo principal es analizar el uso de métricas de proceso en el desarrollo de software, especialmente en el testing de software, para detectar posibles problemas anticipadamente y demostrar la aplicabilidad de estas métricas en proyectos de desarrollo de software.

Destacan el uso de métricas entre procesos en las pruebas de software y recalcan la eficacia de dichas métricas en las pruebas de software y la gestión de la calidad de los proyectos.

Los autores del estudio presentan los siguientes resultados:

- Presentan casos de éxito en la implementación de las métricas, tales como: Rochester Software Development Lab de IBM entre otras organizaciones del mundo real.
- Concluyen en que el uso de métricas de proceso en el proceso de pruebas de software puede proporcionar información valiosa sobre el progreso del equipo y la detección precoz de problemas.
- Hacen una comparación en uso de métricas de pruebas de software entre equipos de Costa Rica y Estados Unidos que revelaron diferencias significativas en el número de tareas no concluidas y defectos emergentes

Se llega a la conclusión de que la comunicación con el cliente significaba un gran impacto en la reducción de defectos.

6. METODOLOGÍA

La presente monografía tiene un fuerte componente investigativo basado en los estudios desarrollados por expertos en pruebas de software y métricas de calidad. En este sentido, la metodología aplicada al desarrollo de la revisión de literatura corresponde a PRISMA (Preferred Reporting Items for Systematic reviews and Meta-Analyses). Esta metodología genera pautas detalladas para informar sobre una revisión sistemática de literatura de manera clara y completa (Page et al., 2021). PRISMA consta de 3 fases principales: **Identificación**, **Búsqueda** e **Inclusión**. Estas tres fases principales cuentan con tareas específicas como lo son: Formulación de la pregunta de investigación, Búsqueda de literatura, Estrategia de búsqueda, Selección de estudios, Extracción de datos, Evaluación de la calidad de los estudios, Síntesis de los resultados e Informe de la revisión. (Figura 2)

Figura 2. Metodología PRISMA



Adaptado de: (Page et al., 2021)

Tabla 2. Actividades de la metodología PRISMA

Actividad	Características
Formulación de la pregunta de investigación	Se define el propósito de la revisión de literatura mediante la pregunta o preguntas generales que guiarán la investigación
Búsqueda de literatura o fuentes de información	Se desarrolla una estrategia de búsqueda que incluye documentos relevantes de investigación para poder ser estudiados durante el desarrollo del proyecto
Estrategia de búsqueda	Presentar las posibles estrategias implementadas o búsquedas completas para las bases de datos o fuentes consultadas, así como los filtros aplicados, palabras clave entre otras
Proceso de selección	Efectuar criterios tanto de inclusión como de exclusión predefinidos para así escoger los documentos que van de acuerdo con la pregunta de investigación
Extracción de datos	Desempeñar de manera sistemática la recolección de información destacada de cada uno de los documentos agrupados
Evaluación de la calidad de los resultados	Evaluar la calidad metodológica de los documentos incluidos y considerar el riesgo de sesgo en los resultados
Síntesis de los resultados	Es la fase de analizar, interpretar sintetizar los datos de los estudios incluidos (encuestas) y se destacan las posibles limitaciones del estudio.
Informe de la revisión	Se presentan los resultados hallados durante el proceso de revisión de literatura

Fuente: Autor

Al implementar estas actividades de manera rigurosa a la monografía, se pudo hacer efectiva la garantía de la calidad y la integridad de la revisión sistemática de la literatura consultada y su posterior metaanálisis, lo que contribuyó a facilitar la interpretación y utilización de los resultados hallados durante el desarrollo de este trabajo (Tabla 2).

Por otro lado, se realizó un proceso de encuestas a expertos en el área de desarrollo de software con el fin de complementar los conocimientos y resultados obtenidos a lo largo del desarrollo del trabajo.

7. DESARROLLO METODOLÓGICO

7.1. Formulación de la pregunta de investigación

Definiendo el alcance y propósito del desarrollo de la monografía se llega como conclusión a la definición de la pregunta general que guió toda la investigación realizada

Al establecer claramente los objetivos de la monografía, se establecieron las bases para una investigación bien estructurada y enfocada a los mismos. Este proceso permitió establecer un marco de referencia específico y determinar qué aspectos del tema se afrontarían en profundidad. Además, al definir el propósito de la investigación, se consiguió asegurar que los resultados y hallazgos obtenidos fuesen relevantes y aplicables al proceso del testing de software.

Consiguiendo definir el alcance y propósito, se pudo llegar a la definición de la pregunta general que guió toda la investigación y a su vez orientar la temática requerida en el ejercicio de recopilación y análisis de documentos consultados. Esta pregunta es el eje central alrededor del cual giró el trabajo de investigación dentro de la monografía y definió la dirección a seguir:

¿Cuáles son las principales métricas de calidad utilizadas en el testeado de software, y cómo se aplican actualmente en los proyectos de desarrollo para mejorar la eficiencia y efectividad del proceso de pruebas?

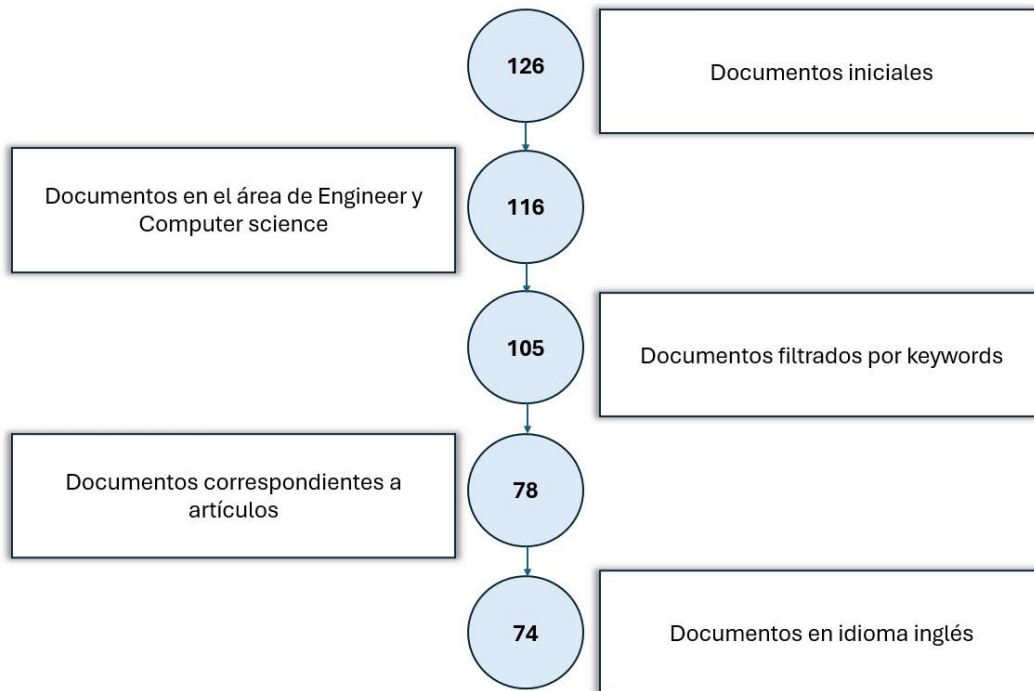
7.2. Búsqueda de literatura o fuentes de información

La calidad es una de las características fundamentales del software. El proceso de testing impacta de manera significativa en los productos de software, por lo que es importante estar al día con las más recientes prácticas, estrategias y métodos empleados (Sheketa et al., 2020). Realizar una revisión bibliográfica en fuentes de alto impacto garantiza el acceso a información precisa y certificada en esta área.

El enfoque bibliográfico de la monografía comprende la revisión e investigación sobre las métricas de calidad orientadas al proceso de pruebas en proyectos de desarrollo de software. De acuerdo con lo anterior, se realiza la búsqueda de dicha bibliografía en la base de datos de Scopus y en la base de datos de IEEE Xplore. Para lograr encontrar los estudios realizados se aplicó la siguiente cadena de búsqueda: “Software Testing” AND “Quality Metrics” y a su vez se establecieron los siguientes criterios de inclusión para así asegurar el enfoque de la temática de la investigación:

- Documentos publicados en las bases de datos desde el año 2019 hasta el año 2024 (contando el año 2024 para posibles nuevos estudios) con el fin de garantizar la actualidad en la temática.
- Documentos enfocados y limitados a las áreas de ciencias de la computación e ingeniería
- Documentos limitados a artículos de revistas, artículos de conferencias y revisiones de literatura.
- Documentos limitados con palabras clave enfocadas al Software Testing y la calidad del Software.
- Documentos publicados preferiblemente en idioma inglés

Figura 3. Recopilación de literatura en la base de datos de Scopus

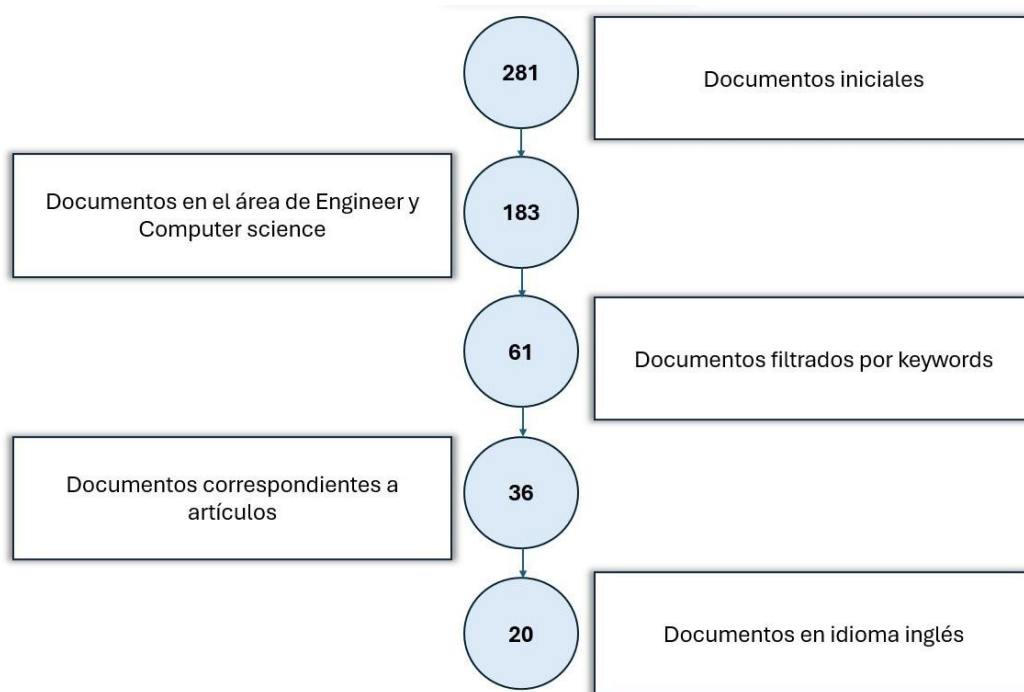


Fuente: Autor

En la base de datos de Scopus se compiló un total de 126 documentos como resultado del uso de las cadenas de búsqueda ingresadas en la base de datos, se redujo el número al descartar los documentos que no fuesen correspondientes a las áreas de ciencias de la computación ni de ingeniería. A continuación, se filtraron los documentos por las 'keywords' o palabras claves para enfocar la temática de la búsqueda a las áreas de Métricas de calidad, Software Testing, Quality Assurance, Calidad de software entre otras.

Por último, se seleccionaron los documentos que fuesen artículos de conferencias o artículos de revistas, y a su vez los documentos que estuvieran en idioma inglés, obteniendo como resultado 74 documentos susceptibles a ser utilizados en la investigación. Cabe resaltar que no hubo necesidad de filtrar los papers o documentos publicados en el rango de años desde 2019 al presente ya que, dicha acción se realizó en un primer instante al ingresar las cadenas de búsqueda. (Figura 3)

Figura 4. Recopilación de literatura en la base de datos de IEEE Xplore

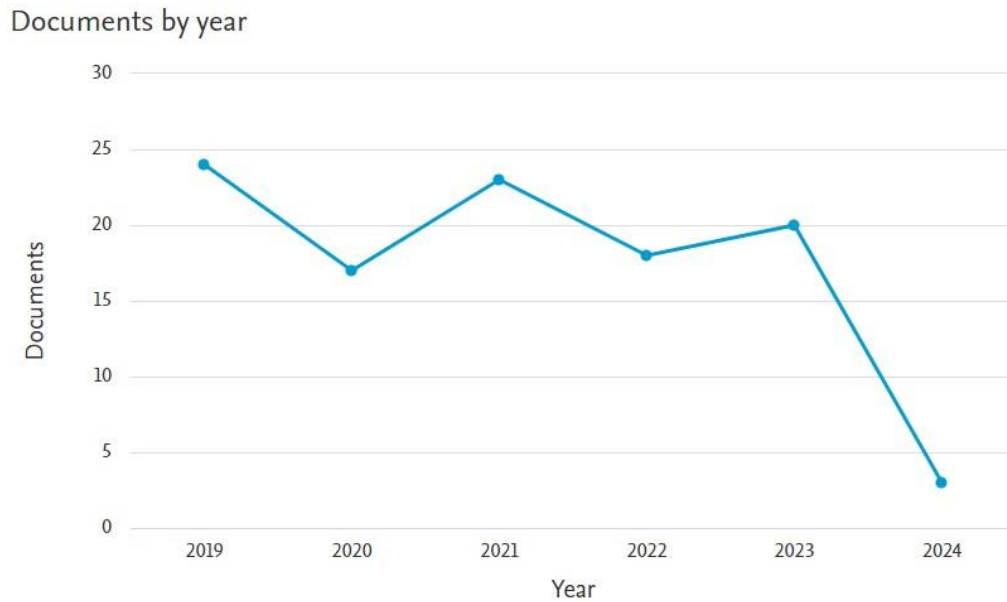


Fuente: Autor

En la base de datos de IEEE Xplore se reunieron 281 documentos usando las cadenas de búsqueda ingresadas en la base de datos (se hizo la especificación del rango de años de publicación de los documentos desde el 2019 al presente), se seleccionaron los documentos correspondientes a áreas de ciencias de la computación y/o ingeniería. Luego se escogieron los documentos resultantes de filtrar por palabras claves similares a las que se usaron en la base de datos de Scopus: Software Testing, Quality Assurance, Calidad de software y demás.

Finalmente, se filtran los documentos que correspondan a artículos de conferencias y/o artículos de revistas, y documentos en inglés, obteniendo un número de 20 documentos que pueden ser utilizados en la investigación. (Figura 4)

Figura 5. Tendencia de publicación de documentos en Scopus

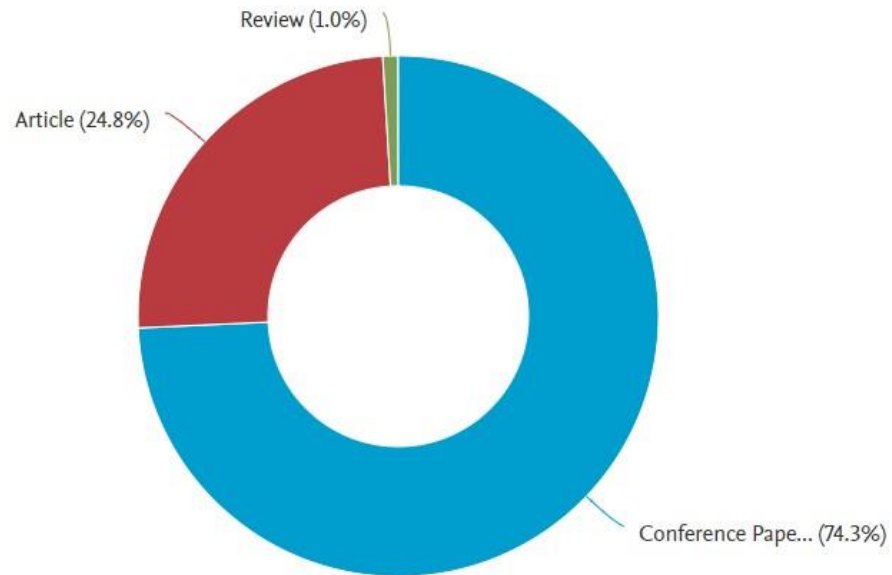


Fuente: Scopus

Conforme a la Figura 5, se logra apreciar la tendencia de publicación de los documentos desde el año 2019 como la cantidad de estos en el período de 2019 a 2024. Asimismo, se evidencia que se ha mantenido el interés por la temática, alcanzando un máximo de 24 estudios en 2019. La reducción del 2024 se puede deber a que las búsquedas fueron realizadas en el primer trimestre del año en curso.

Figura 6. Tipo de documentos en Scopus

Documents by type



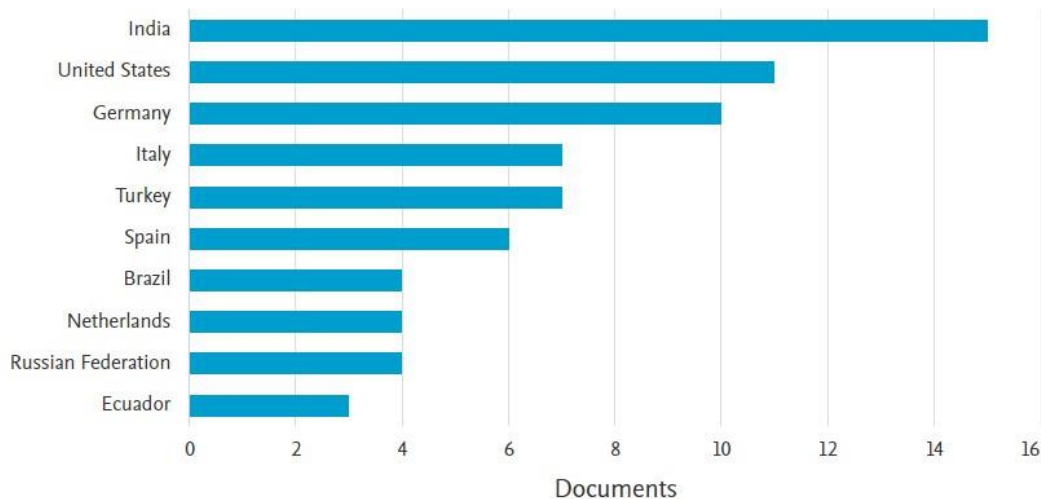
Fuente: Scopus

En la Figura 6 se logra observar el tipo de documentos encontrados en la revisión de literatura siendo los artículos de conferencias el grupo más grande de los documentos con un porcentaje de 74.3% seguido por los artículos de revistas con el 24.8%.

Figura 7. Recopilación de literatura en la base de datos de Scopus

Documents by country or territory

Compare the document counts for up to 15 countries/territories.



Fuente: Scopus

De acuerdo con la Figura 7, se consigue evidenciar los países con mayor cantidad de publicación a lo largo de los últimos 5 años, notando que India es el principal emisor de documentos, mientras que por el lado del continente americano Estados Unidos es el publicador predominante y con respecto a Suramérica, Los dos países líderes en publicaciones son Brasil y Ecuador.

Para el análisis de documentos se utilizó la herramienta VOSviewer, que es un software especializado en la creación, análisis e identificación de patrones de correlación entre palabras, documentos y autores. Dichos patrones se basan en metadatos bibliográficos, que se utilizan ampliamente en los campos de la bibliometría cuantitativa y la infometría.

La creación de las gráficas utiliza la selección de keywords o palabras claves que comprenden enfocar las gráficas hacia el tema específico de la investigación de los documentos. Asimismo, facilitan el proceso de selección al excluir los papers que no cumplan con el área de estudio de la investigación.

Para la elaboración del mapa de correlación en VOSviewer se realizó la acción de descartar las palabras clave que no cumplan con el asunto de la investigación:

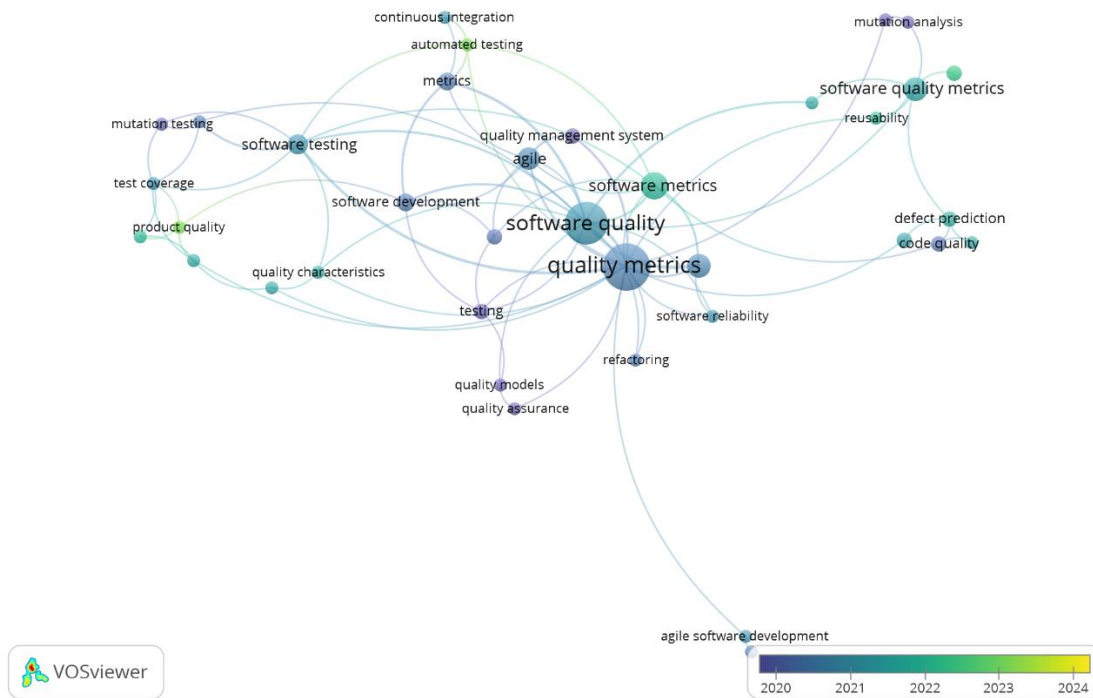
Figura 8. Selección de keywords en VOSviewer

Selected	Keyword	Occurrences	Total link strength
<input checked="" type="checkbox"/>	software quality	22	39
<input checked="" type="checkbox"/>	quality metrics	26	34
<input checked="" type="checkbox"/>	software engineering	7	20
<input type="checkbox"/>	github	3	14
<input type="checkbox"/>	sonarqube	3	14
<input type="checkbox"/>	commits	2	12
<input type="checkbox"/>	sentiments analysis	2	12
<input checked="" type="checkbox"/>	software metrics	9	12
<input checked="" type="checkbox"/>	software testing	5	12
<input type="checkbox"/>	toxic comment classification	2	12
<input checked="" type="checkbox"/>	agile	6	11
<input checked="" type="checkbox"/>	metrics	4	10
<input checked="" type="checkbox"/>	software quality metrics	7	10
<input checked="" type="checkbox"/>	software development	4	8
<input checked="" type="checkbox"/>	automated testing	2	6
<input checked="" type="checkbox"/>	quality management system	3	6
<input type="checkbox"/>	systematic literature review	3	6
<input checked="" type="checkbox"/>	test coverage	2	6
<input type="checkbox"/>	waterfall	2	6
<input checked="" type="checkbox"/>	quality characteristics	2	5

Fuente: VOSviewer

La construcción preliminar del mapa de correlación implicó la selección de las palabras clave de forma manual con la condición de “límite de ocurrencias de palabras clave” con un valor de 2 que corresponde a la cantidad de veces que una palabra clave emerge en la elaboración de la gráfica. Estas son, las que el programa tomará en consideración para graficar el mapa (Figura 8).

Figura 9. Mapa de correlación preliminar

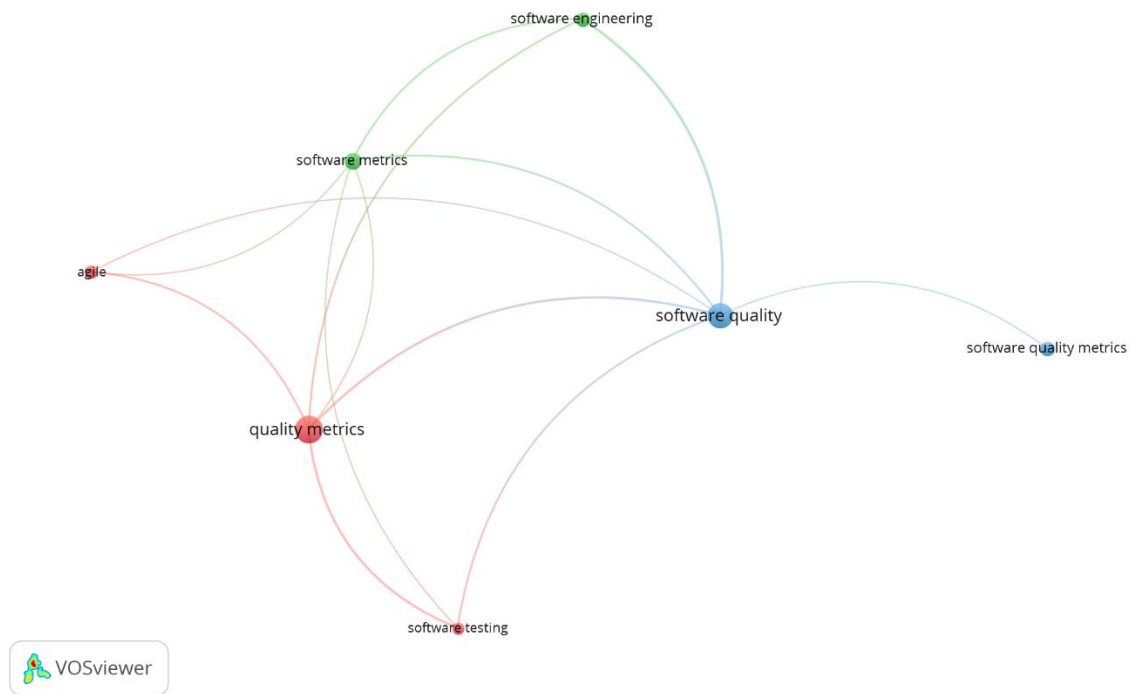


Fuente: VOSviewer

En la figura 9 se logra identificar los distintos clusters de clasificación, a su vez se aprecia la relación de conceptos entre clusters como el software testing, las métricas de calidad, el desarrollo de software, la calidad de los productos entre otros.

También se puede evidenciar la periodicidad de la cantidad de documentos publicados en cada año de forma descendente, siendo el color más oscuro equivalente al año donde más hubo publicaciones de acuerdo con la temática del proyecto y el color más claro correspondiente al año en donde hubo menos publicaciones de documentos en las áreas de interés de la investigación.

Figura 10. Mapa de correlación



Fuente: VOSviewer

La Figura 10 corresponde a la gráfica del mapa de correlación resultante a la selección automática de las keywords donde el “límite de ocurrencias de palabras clave” corresponde a 5.

En el mapa se identifican 3 clusters principales: Software Testing del cual surgen los enlaces de quality metrics y “agile” que corresponde al agile software development. Por su parte, el cluster de software engineering desprende el enlace con Software Metrics y el cluster de Software Quality se enlaza con Software Quality Metrics.

De acuerdo con lo anterior, se logran identificar las principales áreas comunes de coincidencia en los autores de los documentos recolectados, logrando así alcanzar el enfoque de la investigación y a su vez la actualidad del área de interés del proyecto de investigación.

7.3. Estrategia de búsqueda

Para lograr recopilar los estudios o documentos anteriores, se ejecutó una cadena de búsqueda específica para cada una de las bases de datos que se iba a consultar. De esta forma se obtuvieron resultados relevantes a las métricas de calidad y su enfoque al proceso de testing del software.

En la tabla 3 se logra apreciar la estructura de cada cadena de búsqueda ingresada a cada una de las bases de datos consultadas

Tabla 3. Cadenas de búsqueda utilizadas

Fuente	Instrucción
Scopus	("Software Testing" OR "Software Development") AND ("Quality Metrics") AND PUBYEAR > 2019
IEEE Xplore	("Software testing) AND ("Quality metrics") AND Model 2019 - 2024

Fuente: Autor

A pesar de aplicar todos los filtros necesarios para hacer más específico el tema y propósito de búsqueda, se identificaron documentos que coincidían con alguna de las keywords o incluso alguna palabra relevante al tema general de búsqueda, pero que no tenían relación alguna con el proceso del testing de software ni la implementación de las métricas de calidad orientadas a este proceso.

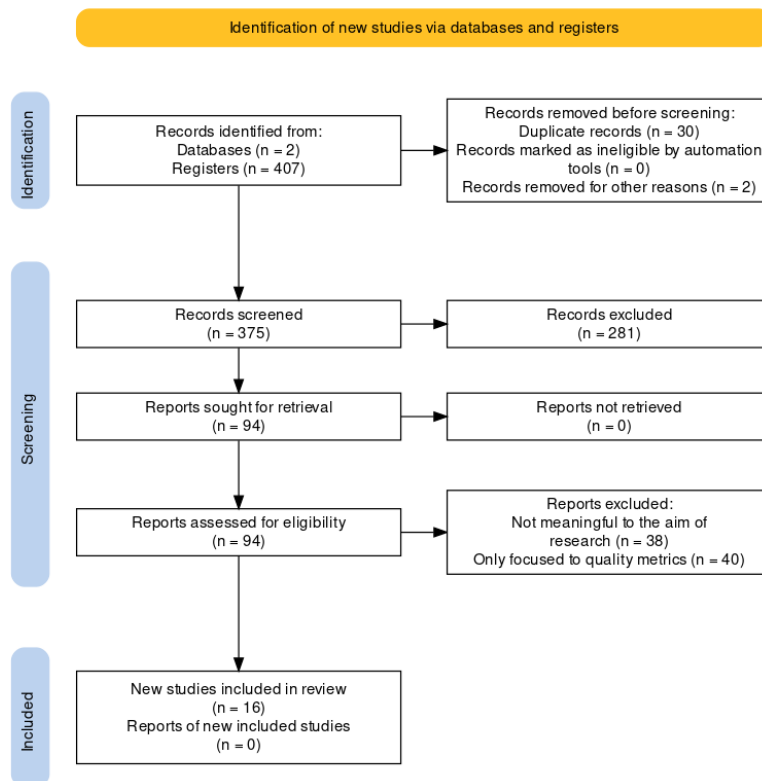
Buscando solucionar esta situación, fue necesario hacer una revisión de documentos preliminar seleccionando los estudios referentes a la ingeniería de software/sistemas y/o ciencias de la computación. Esto con el fin de poder filtrar y seleccionar posteriormente los documentos relevantes y propensos de estudiar en el proyecto de investigación.

7.4. Proceso de selección

El proceso de revisión sistemática de literatura comprendió varios factores o criterios de inclusión y exclusión. Estos criterios se pueden interpretar como filtros que permiten identificar los documentos que pueden ser utilizados como material relevante de investigación en la monografía, ya que contienen información valiosa para poder completar el proceso investigativo y la integración, interpretación y agrupación de datos significativos referente al testing de software y las métricas de calidad. También permiten evidenciar los documentos que, a pesar de poder tener alguna coincidencia dentro de los temas generales, no abordan con suficiencia los temas requeridos y que como resultado no contienen datos que puedan ser analizados y posteriormente anexados al trabajo de investigación por lo que no son material de utilidad para el desarrollo de la investigación

La figura 11 muestra el desarrollo de la revisión de literatura mediante un diagrama de flujo basado en la declaración de PRISMA

Figura 11. Revisión sistemática de literatura



Fuente: PRISMA 2020 flow diagram

De acuerdo con el diagrama de flujo, se logran identificar 3 fases principales en el proceso de revisión:

Identification: Es la fase en donde se identifican los documentos recopilados de las bases de datos seleccionadas. Durante esta fase, se hace la exclusión de los documentos que estén duplicados principalmente y se descartan los que no sean del tema principal de la revisión

Screening: En esta fase se seleccionan los documentos que cumplan con los criterios de inclusión, como lo son las palabras clave y los documentos que aborden las métricas de calidad en testing.

Included: Finalmente, en esta fase se identifican los documentos resultantes que cumplen satisfactoriamente los criterios de la investigación, por lo que se pueden usar en el desarrollo de la investigación.

La tabla 4 presenta los criterios de inclusión y exclusión utilizados en la revisión de literatura

Tabla 4. Criterios de inclusión y exclusión

Criterio	Inclusión	Exclusión
Año de publicación	2019 en adelante	Anterior a 2019
Área de enfoque	Ciencias de la computación e ingeniería	Diferente a las ciencias de la computación e ingeniería
Temática General	Desarrollo de Software	Diferente al proceso de desarrollo de software
Tipo de documento	Artículo de revista o conferencia	No corresponde a artículos de revista o conferencia
Keywords	Enfocadas al desarrollo y testing del software	No enfocadas al desarrollo o testing de software
Coincidencia de documento	No aplica	Si aplica
Idioma	Idioma inglés	Diferente al idioma inglés
Métricas de calidad	Abarca los modelos y/o métricas de calidad orientadas al testing de software	No abarca los modelos y/o métricas de calidad orientadas al testing de software

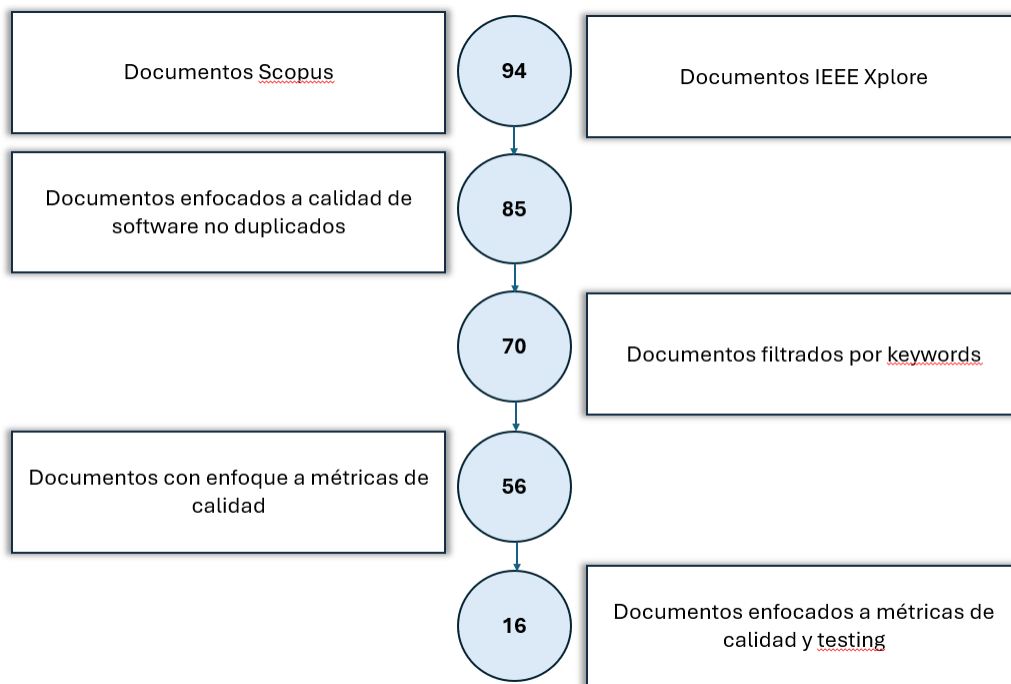
Fuente: Autor

7.5. Extracción de datos

La unión de los documentos encontrados en la base de datos de Scopus y de la base de datos de IEEE resultó en un total de 94 documentos, se redujo el número de documentos al filtrarlos por documentos que estuviesen repetidos. Después, se filtraron los documentos por las 'keywords' para las áreas de Métricas de calidad, Software Testing, Quality Assurance, Calidad de software entre otras.

Luego, se seleccionaron los documentos que estuvieran enfocados hacia las métricas de calidad de software, y finalmente, los documentos que estuvieran enfocados en las métricas de calidad al testeo de software, lo que dio un resultado final de 16 documentos que serían utilizados en el desarrollo del proyecto (Figura 12).

Figura 12. Unión de literatura hallada en la base de datos de IEEE Xplore y Scopus



Fuente: Autor

Los 16 documentos seleccionados cumplían con conformidad los aspectos requeridos para poder realizar con éxito el proceso de investigación. Estos estudios contenían los detalles más relevantes en la calidad del software mediante las métricas orientadas al testing y a su vez los modelos de calidad en tendencia actual (Tabla 5).

Tabla 5. Documentos Incluidos

Autor(es)	Tipo de publicación	Criterio de Inclusión	Año
(Tran et al., 2023)	Conferencia	Métricas de Calidad y modelos de calidad	2023
(Barreto & Cavalcanti, 2019)	Conferencia	Métricas de Calidad	2019
(Purohit et al., 2023)	Conferencia	Métricas de Calidad	2023
(Cibir & Ayyildiz, 2022)	Revista	Métricas de Calidad	2022
(Fuentes et al., 2021)	Conferencia	Métricas de Calidad	2021
(Sheketa et al., 2020)	Conferencia	Métricas de Calidad	2020
(Deshpande et al., 2023)	Conferencia	Métricas de Calidad	2023
(Hooda et al., 2022)	Conferencia	Métricas de Calidad	2022
(Neelofar et al., 2023)	Revista	Métricas de Calidad	2023
(Terragni et al., 2020)	Conferencia	Métricas de Calidad	2020
(G. L. Saini et al., 2020)	Revista	Métricas de Calidad	2020
(Locke et al., 2023)	Conferencia	Métricas de Calidad	2023
(Albattah, 2022)	Conferencia	Métricas de Calidad	2022
(Mladenova, 2020)	Conferencia	Métricas de Calidad	2020
(Sivakrishna et al., 2023b)	Conferencia	Métricas de Calidad	2023
(Obaid Barraood et al., 2023)	Conferencia	Métricas de Calidad	2023

Fuente: Autor

7.6. Evaluación de la calidad de los estudios

Los documentos que no fueron seleccionados para su posible estudio dentro de la investigación no cumplían con los criterios de inclusión requeridos en la búsqueda de la garantía de la calidad de la información necesaria para el enfoque general de la monografía, Al ejecutar el proceso de inclusión y exclusión de los estudios o documentos, se pudo identificar que una gran parte de estos no fueron tomados en cuenta, gracias a que no cumplían con los criterios de inclusión planteados anteriormente o no trataban a

profundidad el uso de las métricas de calidad. Por otro lado, se encontraron dos documentos que cumplían requisitos parciales como el enfoque y que abordan las métricas de calidad, pero no estaban dentro del rango de años de publicación establecido para poder ser tenidos en cuenta al momento de integrarlos a la investigación, por lo cual se optó por descartarlos. Esto busca garantizar la vigencia y calidad de la información que sería utilizada durante el desarrollo del proyecto. Por otra parte, los estudios arrojaron resultados muy importantes acerca de la implementación de las métricas de calidad, resultados sintetizados en la sección: 7.8 Informe de la revisión: Caracterización de métricas de calidad orientadas al proceso de pruebas de software.

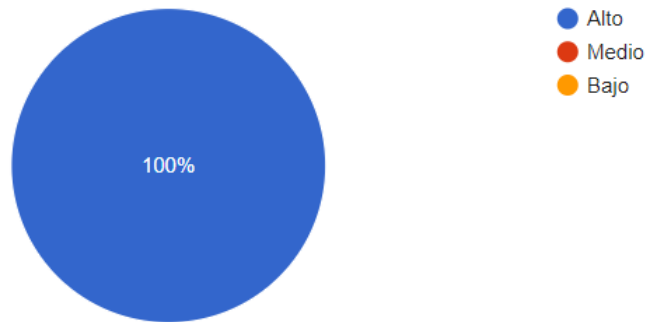
7.7. Síntesis de los resultados

Con el fin de complementar la actividad y proceso de caracterización de las métricas de calidad en la fase de pruebas del software, se realizó una encuesta a 9 profesionales ingenieros de sistemas y afines vinculados a la Universidad Santo Tomás con experiencia de entre 1 a 27 años aproximadamente en el área de desarrollo de software, sobre su aplicación en los proyectos personales o profesionales que implicaran desarrollo de software. Se les consultó sobre el uso de las métricas de calidad y su opinión del área de especialización QA (Quality Assurance), en la cual se hace énfasis acerca de la evaluación de la calidad del software.

Dicha encuesta se realizó mediante un formulario en la plataforma Google forms, el cual se envió por correo electrónico y por medio del enlace de invitación para que pudiera ser diligenciada (Ver anexo 1)

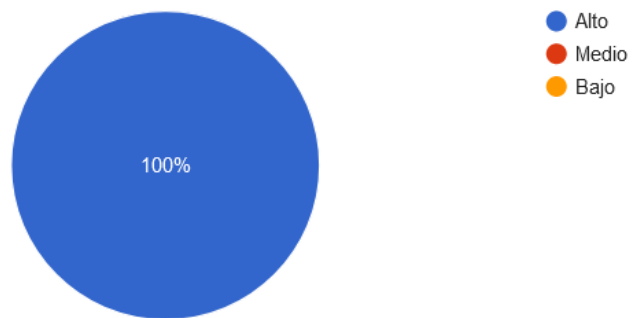
Los profesionales calificaron la legibilidad (Figura 13), comprensibilidad (Figura 14), simplicidad (Figura 15), usabilidad (Figura 16) y fiabilidad (Figura 17) como atributos muy importantes en la calidad del software.

Figura 13. ¿Qué nivel de importancia considera que tiene la Legibilidad como característica del software?



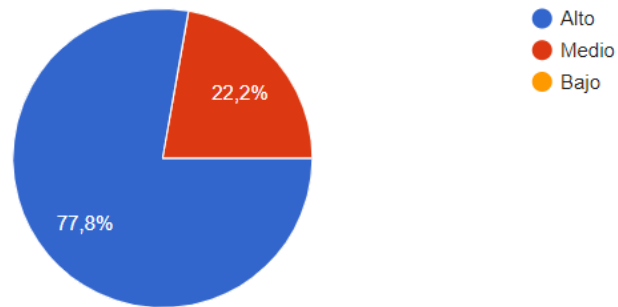
Fuente: Google Forms

Figura 14. ¿Qué nivel de importancia considera que tiene la Comprensibilidad como característica del software?



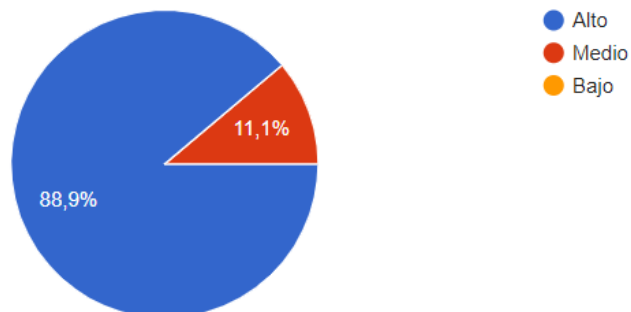
Fuente: Google Forms

Figura 15. ¿Qué nivel de importancia considera que tiene la Simplicidad como característica del software?



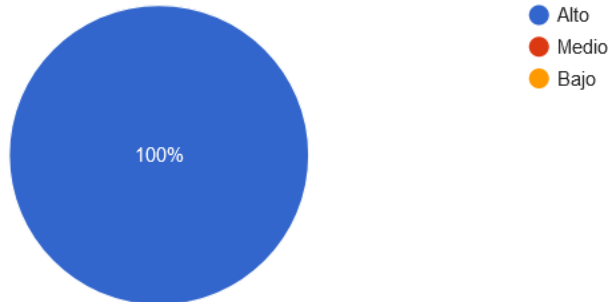
Fuente: Google Forms

Figura 16. ¿Qué nivel de importancia considera que tiene la Usabilidad como característica del software?



Fuente: Google Forms

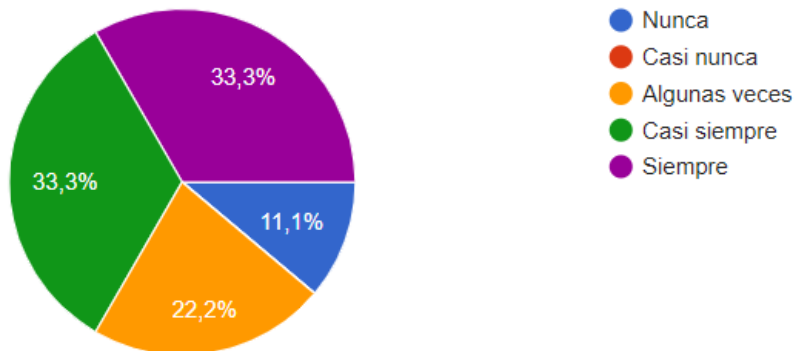
Figura 17. ¿Qué nivel de importancia considera que tiene la Fiabilidad como característica del software?



Fuente: Google Forms

Por otro lado, se les preguntó con qué frecuencia incorporan las métricas de calidad a la fase de pruebas en sus proyectos de desarrollo, identificando una práctica constante de implementación de métricas de calidad en el panorama actual (Figura 18).

Figura 18. ¿Con qué frecuencia utiliza las métricas de calidad en la etapa de testing de sus proyectos?



Fuente: Google Forms

Después se les consultó por las métricas de testing que han utilizado durante su experiencia en los proyectos de desarrollo de software, reconociendo métricas evidenciadas previamente en la fase de revisión de literatura como el registro de algunas nuevas métricas

de calidad. La tabla 6 presenta las métricas identificadas y a cuál atributo de calidad del software corresponden.

Tabla 6. Métricas de testing identificadas a través de la consulta a los profesionales

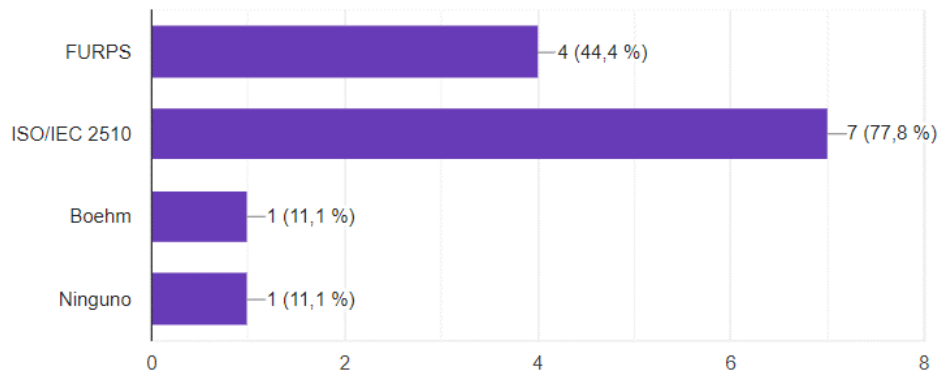
Métrica	Atributo
Jest	Usabilidad, Fiabilidad, Mantenibilidad, Testabilidad, Soporte, Funcionalidad, Modificabilidad, Rendimiento
Cobertura de código	Funcionalidad, Modificabilidad
Número de defectos encontrados y resueltos	Soporte, Eficiencia
Tasa de fallas por prueba	Eficiencia, Funcionalidad
Tiempo promedio de ejecución de pruebas	Rendimiento, Madurez del proceso
Tasa de éxito de pruebas	Usabilidad, Eficiencia
Porcentaje de los requisitos del software	Interoperabilidad, Comprensibilidad, Usabilidad, Legibilidad
Validaciones de pruebas	Usabilidad, Funcionalidad
Tiempo de ejecución de funciones	Rendimiento, Funcionalidad, Eficiencia
Pruebas de automatización	Interoperabilidad, Rendimiento, Comprensibilidad, Simplicidad
Tasa de ejecución de casos de prueba y de uso	Rendimiento, Usabilidad, Testabilidad
Tiempo de respuesta de pruebas	Usabilidad, Simplicidad, Funcionalidad, Rendimiento
Rendimiento antes y después de cambios	Usabilidad, Modificabilidad, Funcionalidad, Fiabilidad, Flexibilidad
Densidad de defectos, causa del defecto, gravedad del defecto, cumplimiento de tiempos.	Rendimiento, Soporte, Mantenibilidad, Madurez del proceso

Fuente: Autor

Las métricas registradas cumplían con características de software propuestas por los modelos de calidad detectados en la revisión de literatura y que fueron examinados con los profesionales. Se les consultó por el uso de los modelos de calidad que integran a sus proyectos de desarrollo para garantizar la calidad de sus productos de software,

determinando así que el modelo ISO 25010 es el modelo aplicado regularmente (Figura 19).

Figura 19. ¿Cuál de los siguientes modelos de calidad del software aplica o ha aplicado a sus proyectos de desarrollo?



Fuente: Google Forms

Por otra parte, se indagó sobre por qué consideran importante la implementación de métricas de calidad en el proceso de desarrollo de software, obteniendo las siguientes respuestas:

- Mejorar la calidad de los ciclos de vida de los proyectos de desarrollo
- Proporcionar una medida objetiva del rendimiento del software
- Evaluar el cumplimiento de los requisitos y la satisfacción del cliente
- Detectar tempranamente, defectos en el proceso de desarrollo y corregirlos
- Reconocer datos cuantificables sobre el rendimiento del software, lo que permite identificar áreas de mejora y evaluar el progreso a lo largo del tiempo
- Garantizar buenas prácticas para el desarrollo, el mantenimiento y la escalabilidad del software
- Mejorar considerablemente los procesos, tiempos de desarrollo, trabajo en equipo, toma de decisiones, automatización y el control de cambios
- Ayudar a mejorar la calidad del producto de software y también su ágil mantenimiento

Aunque el área de Q.A (Quality Assurance) es una de las más importantes para garantizar la calidad del software, no cuenta con muchos profesionales ejerciéndola. Para recibir la percepción de los profesionales encuestados, se les preguntó: Una de las áreas más importante para garantizar la calidad del software es la de Q.A (Quality Assurance) ¿Por qué cree que pocos profesionales optan por especializarse en dicha área? Algunas de sus respuestas fueron:

- La percepción de que QA es una tarea menos glamorosa o emocionante en comparación con el desarrollo de software
- Algunos pueden tener la impresión errónea de que el trabajo en QA es simplemente encontrar errores, en lugar de reconocer su importancia en garantizar la calidad del producto final
- Temas relacionados al salario comparado con las otras divisiones del desarrollo de software
- La percepción de que implica altos costos y demanda mucho tiempo, así como también porque los roles en QA pueden ser subestimados en comparación con otros roles en el desarrollo de software
- Falta mayor conciencia de estos procesos y muchas veces no se recalca en los procesos académicos. Además, porque constantemente emergen nuevos campos del saber en el área tecnológica que generan gran interés por la novedad y el impacto en el mercado
- Falta de reconocimiento, o una percepción errónea, como las oportunidades limitadas
- El desconocimiento y apatía hacia los procesos de calidad de software

De acuerdo con esto, se reconoce la falta de conocimiento en cuanto al área de QA se refiere, resultando en un poco interés por parte de los profesionales para especializarse en dicha área por temas de interés y en gran parte de salario. Esto deriva en poca documentación verificada y confiable acerca de la práctica de la garantía del software y los posibles vacíos en los conocimientos de los futuros profesionales en el área de la ingeniería de software. Sin embargo, la calidad del software comprende el total compromiso de los equipos de desarrollo para poder obtener productos que cumplan las necesidades de los

clientes y al tener una calidad alta, marcar un factor diferenciador en cuanto al éxito en el mercado se refiere.

7.8. Informe de la revisión: Caracterización de métricas de calidad orientadas al proceso de pruebas de software

La calidad es quizás la característica más importante que puede brindar un producto de software. Se refiere a la capacidad del producto de software para satisfacer los requisitos establecidos y las necesidades del usuario final. La calidad es un valor claramente definido que puede medirse, controlarse, gestionarse y mejorarse (Sheketa et al., 2020) . Garantizar la calidad del software es una tarea ardua, dado que comprende la necesidad de ejecutar diversas actividades, como lo son las pruebas de software (testing) durante el ciclo de vida del desarrollo del software.

El testing o pruebas de software consiste en el proceso de examinar una aplicación o sistema de software para encontrar cualquier fallo, problema o error que pueda afectar al funcionamiento o rendimiento de un programa, con el propósito de que cumpla con los requisitos de calidad, seguridad y usabilidad (Hooda et al., 2022), Iniciar este proceso oportunamente, ayuda a los desarrolladores a encontrar problemas en una fase temprana, con el propósito que puedan resolverlos antes de que el programa sea desplegado o puesto en funcionamiento por los usuarios o clientes (Deshpande et al., 2023).

El testing software se puede clasificar en dos grupos principales: la facilidad de testing y la facilidad para revelar fallos. La facilidad de testing se evalúa mediante métricas que miden el esfuerzo de prueba, como el tamaño de las pruebas realizadas. Por otro lado, la facilidad para revelar fallos se mide a través de métricas de calidad del testing, tales como la puntuación de mutación y la cobertura de pruebas (Terragni et al., 2020).

Para poder ejercer el proceso de pruebas a un producto de desarrollo de software, se hace uso de los modelos de calidad. El modelo de calidad es la única herramienta estándar para medir la calidad del producto de software (Hooda et al., 2022), ya que establecen las características esenciales que debe tener un producto de software de alta calidad.

Para poder ejecutar con éxito el proceso de testing a un proyecto de desarrollo de software, se hace uso de las métricas de calidad. Las métricas de calidad de software son los valores

del software que se pueden medir o calcular en función de los resultados de las mediciones, gracias a que controlan y gestionan la calidad de los procesos de desarrollo.

7.8.1 ¿Qué son las métricas de calidad?

La métrica de calidad es definida como una función cuyas entradas son datos de software y cuya salida es un único valor numérico que puede interpretarse como el grado en que el software posee un determinado atributo que afecta a su calidad. El uso de las métricas puede reducir el número de fallos y defectos durante la evaluación del software, así como proporcionar una base cuantitativa para tomar decisiones informativas y racionales sobre la calidad y el desarrollo de software (Mladenova, 2020)

Existen 3 grupos principales de métricas de calidad orientadas al testing de software:

- **Métricas de Proceso:**

Permiten observar el proceso de desarrollo. Facilitan obtener información sobre los defectos encontrados durante un largo período de tiempo en la programación del software, en el proyecto que se está desarrollando ahora o en proyectos anteriores mejorando la eficacia del ciclo de vida, y ofrecen resultados que pueden predecir problemas futuros. Estas métricas también se utilizan cuando es necesario evaluar la eficacia del equipo.

- **Métricas de Producto:**

Relacionadas con el producto desarrollado y no con el proceso de desarrollo, por lo que se aplican al final de la fase de desarrollo. Se enfocan en la medición del tamaño del software, la complejidad, el rendimiento, eficacia de las pruebas, entre otras

- **Métricas de Proyecto:**

Recopilan y miden los potenciales riesgos elaborando un plan estable de desarrollo en las primeras etapas del proyecto de desarrollo de software. Se utilizan para controlar y supervisar el estado y costos del proyecto en la experiencia y los datos recopilados de los proyectos anteriores.

7.8.2 ¿Qué modelos de calidad de software se usan actualmente?

Actualmente existen diversos modelos de calidad de software que son utilizados de manera regular. Cada uno tiene características diferentes y puede aplicarse en función de las necesidades (Hooda et al., 2022). Los modelos de calidad son utilizados para evaluar y mejorar la calidad de un producto de software ya que, proveen un marco conceptual y posibles metodologías que se pueden implementar para evaluar los aspectos de la calidad del software que se necesiten.

Como resultado de la revisión de literatura, se pudo identificar que, en una gran cantidad de los documentos hallados, los autores abordan el uso de un grupo de distintos modelos de calidad, en especial al momento de realizar las pruebas de los productos de software, donde dichos modelos contenían atributos de métricas de calidad específicos en los que coincidían al realizar el proceso de evaluación de la calidad del software. (Tabla 7)

Tabla 7. Comparación de características de métricas en modelos de calidad

Atributo	Modelo de Calidad				
	BOEHM	DROMEY	FURPS	ISO/IEC 9126	ISO/IEC 2510
Usabilidad					
Eficiencia					
Flexibilidad					
Testabilidad					
Mantenibilidad					
Fiabilidad					
Portabilidad					
Reusabilidad					
Interoperabilidad					
Comprensibilidad					
Funcionalidad					
Rendimiento					
Soporte					
Seguridad					
Modificabilidad					
Madurez del proceso					
Integridad					
Ingeniería Humana					

Fuente: Autor

Los atributos representados en la tabla 7 representan las características que debe brindar un producto de software de alta calidad y en qué modelos de calidad del software se hacen

presente la mayoría de los atributos, por lo que marca una tendencia de uso de éstos dentro del contexto de las pruebas de software.

Con base en la tabla presentada se puede afirmar que el modelo de calidad más completo es el establecido por el estándar ISO/IEC 25010, que comprende la evaluación de los productos de software. Este estándar cuenta con más cobertura en cuanto a las características de métricas de calidad se refiere, por lo que se percibe una total comprensión de que sea el modelo más implementado actualmente en función del testing de software.

7.8.3 ¿Cuáles son las métricas de calidad enfocadas al testing de software implementadas en la actualidad?

Implementar métricas garantiza la obtención de productos de software de alta calidad, ya que ayudan a mejorar aspectos clave del software como lo son el mantenimiento, la productividad, el desarrollo del producto entre muchas otras, logrando así la satisfacción del cliente final.

Las métricas se pueden emplear para varios propósitos, siempre dependiendo de las necesidades y propósitos tanto del desarrollador como del cliente. En la fase de pruebas del software se emplean unas métricas específicas para corroborar, evaluar, verificar la eficiencia del software (Gromova et al., 2019), como también dar una idea más clara acerca del estado de desarrollo y las prácticas usadas, logrando reconocer posibles áreas de mejora

Mediante el proceso de revisión de los documentos incluidos, se pudo caracterizar un grupo selecto de métricas enfocadas a la fase de testing que cada uno de los autores de estos estudios pudo hallar y/o emplear en el desarrollo de sus investigaciones o proyectos, determinando el estado actual de la práctica del testing de software.

La tabla 8 presenta las métricas documentadas que se hallaron durante la revisión de cada documento incluido, como los atributos de calidad a los que corresponde su implementación dentro de los proyectos de software.

Tabla 8. Métricas de calidad halladas por cada estudio

Referencia	Métricas de Testing Aplicadas	Atributo de Calidad
(Tran et al., 2023)	<ul style="list-style-type: none"> - Nivel del Sistema de software bajo prueba - VOLC: Long/complex/verbose/obscure - Rastreo de errores 	Testabilidad, Rendimiento, Soporte
(Barreto & Cavalcanti, 2019)	<ul style="list-style-type: none"> - Número de fallos - Velocidad - Tiempo estimado x tiempo empleado - Corrección de defectos - Gravedad de los defectos - Número de pruebas automatizadas - Eficacia de la automatización - Ahorro de tiempo de automatización 	Rendimiento, Usabilidad, Eficiencia, Testabilidad, Mantenibilidad, Soporte
(Purohit et al., 2023)	<ul style="list-style-type: none"> - Composición de pruebas - Encapsulación de errores - Jerarquización - Polimorfismo - Herencia - Cohesión - Abstracción - Acoplamiento de prueba - Tamaño de fallos 	Usabilidad, Testabilidad, Reusabilidad, Rendimiento, Integridad, Funcionalidad
(Cibir & Ayyildiz, 2022)	<ul style="list-style-type: none"> - Número de escenarios - Tiempo de creación del plan de pruebas - Prueba/Requisitos Revisión del Periodo - Tiempo de creación del entorno de pruebas - Número de reuniones - Número de escenarios de prueba defectuosos 	Interoperabilidad, Integridad, Testabilidad, Flexibilidad, Rendimiento
(Fuentes et al., 2021)	<ul style="list-style-type: none"> - Pruebas de progreso de Curva - Defectos del backlog en el tiempo - Llegada de defectos en el tiempo 	Madurez del proceso, Usabilidad
(Sheketa et al., 2020)	<ul style="list-style-type: none"> - Número de modificaciones fallidas - Número de criterios de evaluación fallidos - Valor de las características de rendimiento antes de cambios - Características de rendimiento después de los cambios - Cambio de compatibilidad - Valor de compatibilidad después de los cambios 	Usabilidad, Rendimiento, Interoperabilidad, Soporte, Testabilidad
(Deshpande et al., 2023)	<ul style="list-style-type: none"> - Número total de requisitos - Total de requisitos funcionales - Total no funcionales - Número total de requisitos acordados 	Funcionalidad, Usabilidad, Madurez del proceso, Comprensibilidad

Referencia	Métricas de Testing Aplicadas	Atributo de Calidad
(Hooda et al., 2022)	<ul style="list-style-type: none"> - Calidad Interna (IQ) - Calidad Externa (EQ) - Gestión de la Calidad (MQ) 	Corrección, Facilidad de uso, Integridad, Precisión, Robustez, Fiabilidad
(Neelofar et al., 2023)	<ul style="list-style-type: none"> - Cobertura del código - Puntuación de mutación - Diversidad de casos de prueba - Longitud/tamaño del conjunto de pruebas resultante 	Interoperabilidad, Testabilidad, Funcionalidad
(Terragni et al., 2020)	<ul style="list-style-type: none"> - Líneas de código (T-LOC) - Número de pruebas (T-NOT) - Número de afirmaciones (T-NOA) - Número de llamadas a métodos (T-NMC) - Métodos ponderados por clase (T-WMC) - Complejidad media de los métodos (T-AMC) - Cobertura de línea (L) - Cobertura de ramas (B) - Puntuación de mutación (M) 	Testabilidad, Interoperabilidad, Rendimiento, Eficiencia, Modificabilidad, Integridad, Usabilidad
(G. L. Saini et al., 2020)	<ul style="list-style-type: none"> - Recuento de defectos - Líneas de código (LOC) - Conjunto de métricas de producto de Halstead - Métricas CK 	Mantenibilidad, Reusabilidad, Complejidad, Fiabilidad
(Locke et al., 2023)	<ul style="list-style-type: none"> - Tasa de aprobación de casos de AT (Automated Testing) - Tiempo de ejecución de los casos de AT - Escasez de casos de AT - Cobertura del servicio por casos de AT - Número de mantenedores por caso de AT - Casos de prueba/servicios con un porcentaje de fallos muy correlacionado - Tasa de ejecución de casos de prueba - Principal ejecutor de casos de prueba - Número de nuevos casos de AT ejecutados diariamente - Servicios con escaso soporte de AT - Frecuencia de cambio de código de AT - Asignación de trabajo de AT - Número de tickets firmados por AT - Número de errores detectados por AT - Porcentaje de casos de AT redactados por persona/equipo - Número de nuevos casos de prueba creados diariamente 	Usabilidad, Testabilidad, Rendimiento, Eficiencia, Reusabilidad, Soporte, Comprensibilidad, Integridad

Referencia	Métricas de Testing Aplicadas	Atributo de Calidad
(Albattah, 2022)	<ul style="list-style-type: none"> - TLOC Líneas de código de prueba - TNOO Número de operaciones en una clase de prueba - TWMC Métodos ponderados por clase - T-NOA Número de aserciones - T-NMC Número de llamadas a métodos - T-AMC Complejidad media de los métodos - TRFC Tamaño del conjunto de respuestas de una clase de prueba 	<p>Complejidad, Acoplamiento, Cohesión, Observabilidad, Controlabilidad, Herencia, Reutilización Comprensibilidad.</p>
(Mladenova, 2020)	<ul style="list-style-type: none"> - Casos de prueba superados Porcentaje - Casos de prueba fallidos Porcentaje - Porcentaje de defectos corregidos - Porcentaje de defectos aceptados - Porcentaje de defectos rechazados - Tiempo medio de reparación de defectos - Número de pruebas tun por periodo - Eficacia del diseño de las pruebas 	<p>Portabilidad Fiabilidad, Eficiencia Responsabilidad, Usabilidad, Ingeniería humana (comprobabilidad) Comprensión de Modificabilidad (flexibilidad)</p>
(Sivakrishna et al., 2023b)	<ul style="list-style-type: none"> - Funcionalidad - Fiabilidad - Eficacia - Mantenibilidad - Utilidad 	<p>Funcionalidad, Fiabilidad, Eficacia Mantenibilidad, Utilidad</p>
(Obaid Barraood et al., 2023)	<ul style="list-style-type: none"> - Cobertura de declaraciones - Cobertura de código - Cobertura de ramas - Cobertura de mutaciones - Cobertura de condiciones modificadas - Métodos ponderados por clase (WMC) - Profundidad del árbol hereditario (DIT) - Número de hijos (NOC) - Falta de cohesión de un método de prueba (LCTM) - Cantidad de aserciones - Número de caracteres en un resumen TC - Número de palabras clave en el TC - Número de ítems de prueba - Número de constantes de una CT - Número de variables de una CT - Número de TC precursoras - Número de aplicaciones que pueden utilizar un CT - Línea de código (LOC) - Respuesta por clase (RFC) - Criterio de flujo de control - Cobertura del flujo de datos - Acoplamiento entre métodos de prueba (CBTM) 	<p>Legibilidad, Comprensibilidad, Simplicidad/Complejidad, Corrección, Independencia, Reutilización Mantenibilidad, integridad, Repetibilidad, Modificabilidad, Universal, Eficacia, Cobertura, Responsabilidad única, Experiencia del personal,</p>

	<ul style="list-style-type: none"> - Cobertura de pares de eventos - Excepciones únicas no detectadas - Coste en horas - Métricas de CT acumuladas - Número de fallos revelados por la CT - Cobertura de código compuesto - Puntuación de todos los pares - Ratios de eliminación de mutaciones - Cobertura de aserciones - Resumen de CT - Número de palabras - Resultado esperado - Número de palabras condicionales para cada prueba - Número de palabras vagas - Cantidad de uso de sinónimos incoherentes - Número de métodos de prueba - Recuento de ramas - Número de llamadas únicas a métodos - Tiempo de ejecución de las CT - Número de pruebas de aceptación del usuario superadas - Número total de defectos detectados por día (semana o mes) - Número de TC ejecutadas en un periodo de tiempo - Defecto por línea de código 	
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Fuente: Autor

Las métricas de testing son aplicadas a cada proyecto dependiendo las necesidades específicas de cada uno de ellos y para ejecutar pruebas en distintas áreas del proyecto (Ramadhina et al., 2021). De las métricas identificadas se puede concluir que la gran mayoría están enfocadas en brindar las características más esenciales que debe llevar y demostrar un producto de software, como lo son la usabilidad, fiabilidad, comprensibilidad, simplicidad, eficiencia y soporte. Muchas de las métricas halladas coincidían en el enfoque de cumplimiento y garantía de la calidad mediante la satisfacción de estos atributos dados por los modelos estudiados.

Una característica fundamental de las métricas de calidad es su enfoque en el tiempo de ejecución y respuesta de las pruebas. Esto es esencial en el desarrollo de software, un área en constante movimiento y actualización. Un producto que responda a las necesidades y peticiones del usuario en el menor tiempo posible, entre otras características vitales, refleja un buen proceso de desarrollo y un entorno de pruebas eficiente. Esto no solo garantiza la

calidad del producto final, sino también uno de los elementos más importantes que debe brindar el software: la satisfacción del cliente (Mladenova, 2020).

8. CONCLUSIONES Y ANÁLISIS DE RESULTADOS

La calidad del software se garantiza a través del uso adecuado de métricas de calidad. En la fase de testing, es crucial comprender estas métricas para desarrollar productos que sean competitivos y cumplan con los estándares de calidad establecidos. No obstante, la elección de métricas depende del contexto de la prueba o evaluación, ya que deben adaptarse a las necesidades específicas del cliente y del software en cuestión. Por ello, las métricas seleccionadas deben no solo reflejar estas características, sino también alinearse con los modelos de calidad reconocidos.

Se concluye que hay un déficit de información verídica, ya que no existe una gran cantidad de documentos actualizados en la práctica e implementación de las métricas de calidad en los procesos de testing de software, lo que representó una dificultad para lograr la caracterización de estas.

Las métricas deben satisfacer las características de los atributos suministrados por los modelos de calidad actuales, independientemente en qué fase del ciclo de vida se ejecuten. En este caso, la caracterización de las métricas enfocadas al testing de software derivó en el hallazgo de más de 100 métricas (tabla 8) utilizadas en proyectos de desarrollo de software a lo largo de los últimos 5 años, las cuales cumplían con satisfacción las características requeridas para garantizar la calidad de los productos, procesos y proyectos de software.

La experiencia en el desarrollo contribuye a la perfección en la práctica y aplicación de las métricas de calidad.

Mediante la realización de encuestas a los profesionales fue posible identificar cuál es su uso e importancia actual dentro de la práctica del desarrollo de software y a su vez, complementar los conocimientos adquiridos durante el proceso de la revisión de literatura. Por otra parte, se identifica la falta de profesionales en el área de QA, lo que representa el poco registro de documentación verificada en el ámbito de la garantía del software y por ende, una potencial deficiencia de conocimiento en la garantía de la calidad de los productos de software en la fase del testing, a comparación de otras fases dentro del ciclo de vida de desarrollo. Este déficit es debido al poco interés que el profesional en ingeniería de sistemas/software posee por la garantía de la calidad, debido mayormente a que Q.A

está por debajo de la capacidad salarial comparada con otras áreas de desarrollo de software.

9. REFERENCIAS

- Albattah, W. (2022). Software Package Testability Prediction Using Object-Oriented Cohesion Metrics. *2022 13th International Conference on Information and Communication Systems, ICICS 2022*, 155–161. <https://doi.org/10.1109/ICICS55353.2022.9811192>
- Aleti, A. (2023). Software Testing of Generative AI Systems: Challenges and Opportunities. *Proceedings - 2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering, ICSE-FoSE 2023*, 4–14. <https://doi.org/10.1109/ICSE-FoSE59343.2023.00009>
- Barreto, P. P., & Cavalcanti, A. P. C. (2019). A proposed metrics to evaluate the effectiveness of software testing [Uma Proposta de Métricas Para Avaliar a Efetividade da Execução de Testes de Software]. *Iberian Conference on Information Systems and Technologies, CISTI, 2019-June*. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85070096514&doi=10.23919%2FCISTI.2019.8760646&partnerID=40&md5=481929580e38ee0e5b9416d2a9051dad>
- Biricz, B., Jónás, V., Vincze, M., Benhamida, A., Paulik, R., & Kozlovsky, M. (2022). User friendly virtual reality software development and testing. *2022 13th IEEE International Conference on Cognitive Infocommunications, CogInfoCom 2022*, 87–92. <https://doi.org/10.1109/CogInfoCom55841.2022.10081846>
- Chahar, V., & Bhatia, P. K. (2023). Unveiling the Art of Software Testing Effort Estimation: An In-Depth Study of Current Techniques and their Analysis. *Proceedings - 2023 International Conference on Advanced Computing and Communication Technologies, ICACCTech 2023*, 51–60. <https://doi.org/10.1109/ICACCTech61146.2023.00018>
- Cibir, E., & Ayyildiz, T. E. (2022). An Empirical Study on Software Test Effort Estimation for Defense Projects. *IEEE Access*, 10, 48082–48087. <https://doi.org/10.1109/ACCESS.2022.3172326>
- Deshpande, M. V., Soitkar, P. A., Tripathi, D. R., & Agarmore, S. B. (2023). Ensuring Web Application Quality: The Role of Software Testing as a Form of Quality Assurance. *3rd IEEE International Conference on ICT in Business Industry and Government, ICTBIG 2023*, 1–6. <https://doi.org/10.1109/ICTBIG59752.2023.10456277>
- Eisty, N. U., Thiruvathukal, G. K., & Carver, J. C. (2018). A survey of software metric use in

- research software development. *Proceedings - IEEE 14th International Conference on EScience, e-Science 2018*, 212–222. <https://doi.org/10.1109/eScience.2018.00036>
- Faisal, M. R. (2021). Software Testing: Data Kart and Integrated Pipeline Approach. *International Journal on Recent and Innovation Trends in Computing and Communication*, 9(8), 1–4. <https://doi.org/10.17762/ijritcc.v9i8.5480>
- Fuentes, B. A., Cersosimo, M., & Jenkins, M. (2021). Application of Process Metrics for Software Testing: A Case Study. *Proceedings - 5th Jornadas Costarricenses de Investigacion En Computacion e Informatica, JoCICI 2021*, 1–6. <https://doi.org/10.1109/JoCICI54528.2021.9794340>
- Garousi, V., Felderer, M., & Hacaloglu, T. (2018). What We Know about Software Test Maturity and Test Process Improvement. *IEEE Software*, 35(1), 84–92. <https://doi.org/10.1109/MS.2017.4541043>
- Gromova, A., Itkin, I., Pavlov, S., & Korovayev, A. (2019). Raising the Quality of Bug Reports by Predicting Software Defect Indicators. *Proceedings - Companion of the 19th IEEE International Conference on Software Quality, Reliability and Security, QRS-C 2019*, 198–204. <https://doi.org/10.1109/QRS-C.2019.00048>
- Hooda, S., Lamba, V., Kaur, S., Sharma, V. K., Kumar, R., & Sood, V. (2022). Impact of Software Quality on “gA-FC” Software Testing Technique. *Proceedings of International Conference on Computational Intelligence and Sustainable Engineering Solution, CISES 2022*, 142–147. <https://doi.org/10.1109/CISES54857.2022.9844275>
- Locke, H. L., Keshia, Y. K. T., Yu, J. C. K., & Chua, H. Y. (2023). QEX: Automated Testing Observability and QA Developer Experience Framework. *Proceedings - 2023 IEEE 16th International Conference on Software Testing, Verification and Validation, ICST 2023*, 454–460. <https://doi.org/10.1109/ICST57152.2023.00049>
- Martinez-Ortiz, A.-L., Lizcano, D., & Ortega, M. (2019). Software Metrics Artifacts Making Web Quality Measurable. *2019 IEEE/ACM 14th International Workshop on Automation of Software Test (AST)*, 1–6. <https://doi.org/10.1109/ast.2019.000-2>
- Mladenova, T. (2020). Software Quality Metrics - Research, Analysis and Recommendation. *2020 International Conference Automatics and Informatics, ICAI 2020 - Proceedings*. <https://doi.org/10.1109/ICAI50593.2020.9311361>
- Moreira Nascimento, A., Vismari, L. F., Cugnasca, P. S., Camargo Junior, J. B., & Rady De Almeida Junior, J. (2019). A cost-sensitive approach to enhance the use of ML classifiers in software testing efforts. *Proceedings - 18th IEEE International Conference*

- on Machine Learning and Applications, ICMLA 2019*, 1806–1813.
<https://doi.org/10.1109/ICMLA.2019.00292>
- Neelofar, N., Smith-Miles, K., Munoz, M. A., & Aleti, A. (2023). Instance Space Analysis of Search-Based Software Testing. *IEEE Transactions on Software Engineering*, 49(4), 2642–2660. <https://doi.org/10.1109/TSE.2022.3228334>
- Normas ISO 5055. (s/f). Prezi.com. Recuperado el 22 de agosto de 2024, de <https://prezi.com/p/fsyfbd57n6ou/normas-iso-5055/>
- Obaid Barraood, S., Mohd, H., Baharom, F., & Almogahed, A. (2023). Systematic Literature Review on Test Case Quality Characteristics and Metrics. *2023 3rd International Conference on Emerging Smart Technologies and Applications, ESmarTA 2023*, 1–8. <https://doi.org/10.1109/eSmarTA59349.2023.10293544>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *The BMJ*, 372. <https://doi.org/10.1136/bmj.n71>
- Purohit, S., Singh, S., Agarwal, M., & Verma, N. (2023). A Qualitative and Comprehensive Analysis of Software Testability Metrics and their Trends. *Proceedings of the 2023 2nd International Conference on Electronics and Renewable Systems, ICEARS 2023*, 1545–1552. <https://doi.org/10.1109/ICEARS56392.2023.10085333>
- Ramadhina, S., Bahawares, R. B., Hermadi, I., Suroso, A. I., Rodoni, A., & Arkeman, Y. (2021). Software Defect Prediction Using Process Metrics Systematic Literature Review: Dataset and Granularity Level. *2021 9th International Conference on Cyber and IT Service Management, CITSM 2021*, 1–7. <https://doi.org/10.1109/CITSM52892.2021.9587932>
- Saini, A., Kumar, S., Kumar, R., Kumar, M., & Kumari, A. (2023). Exploratory Testing of Software Product Lines using Distance Metrics. *Proceedings of International Conference on Computational Intelligence and Sustainable Engineering Solution, CISES 2023*, 542–546. <https://doi.org/10.1109/CISES58720.2023.10183535>
- Saini, G. L., Panwar, D., Kumar, S., & Singh, V. (2020). A systematic literature review and comparative study of different software quality models. In *Journal of Discrete Mathematical Sciences and Cryptography* (Vol. 23, Issue 2, pp. 585–593).

<https://doi.org/10.1080/09720529.2020.1747188>

- Sheketa, V., Zorin, V., Chupakhina, S., Kyrsta, N., Pasyeka, M., & Pasioka, N. (2020). Empirical Method of Evaluating the Numerical Values of Metrics in the Process of Medical Software Quality Determination. *2020 International Conference on Decision Aid Sciences and Application, DASA 2020*, 22–26. <https://doi.org/10.1109/DASA51403.2020.9317218>
- Sivakrishna, V., Kumar, P. S., Jhaveri, J., Akanksha, L. S., Rao, P. V., & Sai, N. R. (2023b). Software Testing Based on Software Product Quality Metrics (SPQM). *2023 9th International Conference on Advanced Computing and Communication Systems, ICACCS 2023*, 2151–2156. <https://doi.org/10.1109/ICACCS57279.2023.10113076>
- Terragni, V., Salza, P., & Pezzè, M. (2020). Measuring software testability modulo test quality. *IEEE International Conference on Program Comprehension*, 241–251. <https://doi.org/10.1145/3387904.3389273>
- Tran, H. K. V., Ali, N. Bin, Börstler, J., & Unterkalmsteiner, M. (2023). *Test-Case Quality -- Understanding Practitioners' Perspectives*. 1–16. https://doi.org/10.1007/978-3-030-35333-9_3
- Wang, X., Sun, W., Hu, L., Zhao, Y., Eric Wong, W., & Chen, Z. (2019). Software-Testing Contests: Observations and Lessons Learned. *Computer*, 52(10), 61–69. <https://doi.org/10.1109/MC.2019.2905533>

10. ANEXOS

Anexo 1. Formato de encuestas realizadas

Métricas de calidad en el testing de software

Durante la fase de pruebas (testing) de un proyecto de software, es importante evaluar la calidad del producto haciendo uso de métricas de calidad

El propósito de este formulario es consultar a los profesionales con experiencia en el desarrollo de software sobre la aplicación de métricas en el testing de los proyectos personales o de trabajo.

Agradecemos el tiempo y colaboración al brindar sus respuestas en el siguiente formulario

1. ¿Cuántos años de experiencia tiene en el desarrollo de software?

2. ¿Qué nivel de importancia considera que tiene la **Legibilidad** como característica del software?

- Alto
- Medio
- Bajo

3. ¿Qué nivel de importancia considera que tiene la **Comprensibilidad** como característica del software?

- Alto
- Medio
- Bajo

4. ¿Qué nivel de importancia considera que tiene la **Simplicidad** como característica del software?

- Alto
- Medio
- Bajo

5. ¿Qué nivel de importancia considera que tiene la **Usabilidad** como característica del software?

- Alto
- Medio
- Bajo

6. ¿Qué nivel de importancia considera que tiene la **Fiabilidad** como característica del software?

- Alto
- Medio
- Bajo

7. ¿Con qué frecuencia utiliza las métricas de calidad en la etapa de testing de sus proyectos?

- Nunca
- Casi nunca
- Algunas veces
- Casi siempre
- Siempre

8. ¿Cuáles métricas de testing ha aplicado en sus proyectos?

9. ¿Cuál de los siguientes modelos de calidad del software aplica o ha aplicado a sus proyectos de desarrollo?

Selecciona todos los que correspondan.

- FURPS
- ISO/IEC 25010
- Boehm
- Ninguno

Otro: _____

10. ¿Por qué considera importante la utilización de métricas de calidad en el testing de software?

11. ¿Cómo ha logrado adquirir el conocimiento y experiencia que posee referente a la calidad del software?

- Artículos científicos
- Libros
- Estudios realizados
- Experiencia personal
- Otro:

12. Una de las áreas más importante para garantizar la calidad del software es la de Q.A (Quality Assurance) ¿Por qué cree que pocos profesionales optan por especializarse en dicha área?
