Impacto de las prácticas de DevOps en la calidad de producto de software: un mapeo sistemático

*Impact of DevOps practices on software product quality: a systematic mapping*

Resumen — En la actualidad, varias organizaciones están adoptando DevOps. Se ha sugerido cada vez más que este enfoque tiene un impacto positivo en la calidad del software producido. Este trabajo en progreso tiene como objetivo identificar y analizar las prácticas de DevOps recomendadas en la literatura que contribuyen a mejorar aspectos específicos de la calidad del producto como se representa en el modelo de calidad de la norma ISO/IEC 25010. Para lograr esto, se realizó un mapeo sistemático de la literatura, de manera que se pudiera obtener una visión clara del estado actual del tema, así como encontrar evidencia empírica de las prácticas de DevOps que tienen una influencia positiva en las características de calidad de productos de software. Los resultados preliminares de este trabajo revelan en primer lugar, un aumento en la cantidad de publicaciones sobre el tema en los últimos años. También se encontró que hay una baja adopción de estándares de calidad en este contexto. Por otro lado, se destaca que hay un alto porcentaje de investigaciones que abordan la característica de seguridad, mientras que otras características, como la compatibilidad, reciben menos atención. También se pudieron obtener prácticas de DevOps, que, de acuerdo con la evidencia empírica proporcionada por los estudios primarios, tienen una influencia positiva en la calidad de producto de software.

Palabras Clave - DevOps; características de calidad de producto de software; prácticas; actividades

Abstract — Today, several organizations are embracing DevOps. It has been increasingly suggested that this approach has a positive impact on the quality of the software produced. This work-in-progress aims to identify and analyze DevOps practices recommended in the literature that contribute to improving specific aspects of product quality as represented in the ISO/IEC 25010 quality model. To achieve this, it is carried out a systematic mapping of the literature, so that a clear vision of the current state of the subject could be obtained, as well as finding empirical evidence of DevOps practices that have a positive influence on the quality characteristics of software products. Preliminary results reveal, firstly, an increase in the number of publications on the subject in recent years. It was also found that there is a low adoption of quality standards in this context. On the other hand, it stands out that there is a high percentage of research that addresses the security characteristic, while other characteristics, such as compatibility, receive less attention. It was also possible to obtain DevOps practices, which according to the empirical evidence provided by the primary studies, have a positive influence on the quality of the software product.

*Keywords - DevOps; quality characteristics of software product; practices; activities.*

1. Introducción

El término DevOps, se menciona por primera vez en 2008 por Patrick Debois [1]. DevOps integra los dos mundos de desarrollo y operaciones. Es un cambio organizacional en el que, en lugar de grupos distribuidos en silos que realizan funciones por separado, los equipos multifuncionales trabajan en entregas de funciones operativas continuas [2]. Los beneficios percibidos con el uso de DevOps en el desarrollo de software son: reducción del tiempo de entrega, reducción del tiempo de liberación, mejoras en la resolución de problemas, mejoras en el control de la versión del software, tiempo de comercialización más rápido, reducción de la falta de comunicación entre el equipo de desarrollo, el equipo de operaciones y las partes interesadas, mejoras en la implementación frecuente, mejoras en la productividad, mejoras en el ciclo de comentarios del cliente, costo de desarrollo más bajo, mejoras en la calidad del producto de software, acelera la retroalimentación recibida de clientes y usuarios, eficaz para lograr lanzamientos de proyectos con plazos determinados, entrega puntual y de alto valor, acelera la velocidad del equipo y mayor valor comercial [3].

A pesar de la popularidad y de reconocerse los beneficios de DevOps, se ha prestado poca atención a su impacto en la calidad de software [4] y aunque algunos artículos señalan que tiene una influencia positiva en la calidad del producto, se reporta poca información respecto a las actividades o prácticas recomendadas que tienen una influencia directa sobre ella [5]. Muchos estudios vinculados a DevOps y a su relación con la calidad de software recomiendan la necesidad de abordar la perspectiva de calidad de producto [4], ampliar la investigación en muchas áreas de DevOps como medición, desarrollo de métricas de diferentes etapas para evaluar su desempeño, cultura, prácticas para garantizar la garantía de calidad y la necesidad de abordar factores de calidad como usabilidad, eficiencia, mantenibilidad de software, portabilidad y seguridad necesitan ser más abordados [5].

Un mayor acercamiento al estudio de la calidad de software a través del uso de DevOps sería, identificar prácticas de DevOps recomendadas en la literatura relacionadas con las características de calidad de producto teniendo en cuenta el modelo de calidad del producto definido por la ISO/IEC 25010 [6], por lo que se necesita más evidencia para poder reportar la existencia de un impacto de DevOps en la calidad del software y una influencia en esta calidad cuando se llevan a la práctica [1].

Considerando lo expuesto anteriormente, el objetivo de este mapeo sistemático de literatura es conocer el estado actual del tema y además identificar prácticas de DevOps informadas en la literatura que influyan específicamente en las características de calidad del producto que se encuentran representados en el modelo de calidad de la norma ISO/IEC 25010 [6].

La estructura del documento es como sigue. En la sección II, se presenta brevemente las principales revisiones de literatura que abordan la calidad del software en entornos DevOps. La sección III describe la metodología del mapeo sistemático mientras que, en la sección IV se presentan los resultados preliminares de la revisión de literatura. La discusión de los resultados se presenta en la sección V, y las amenazas a la validez en la sección VI. Finalmente, las conclusiones se presentan en la sección VII.

1. Trabajos relacionados

En esta sección se presentan conceptos claves para la comprensión del tema, así como una visión general de algunos trabajos que abordan la relación de DevOps con la calidad de software.

La calidad de software se define como “el grado en el cual un producto de software satisface las necesidades enunciadas e implícitas cuando se usa en condiciones especificadas” [6]. La calidad del software se puede estudiar desde distintas perspectivas y cada una de ellas enfoca su atención a diferentes aspectos que pueden ser medidos. Las dos perspectivas más utilizadas en la evaluación de la calidad del software son la basada en la vista de proceso y la vista de producto [6] [7].

La calidad de producto de software significa que hay un modelo de calidad subyacente que definen un conjunto de características de calidad del producto [7]. La vista de producto de la calidad del software está basada en un modelo que define un conjunto de características de calidad del producto [8]. En esta perspectiva, la calidad es vista como apropiada para su propósito y depende de las propiedades del producto que satisfacen las necesidades del usuario [8]. La calidad del producto se mide considerando el contexto de la tarea o escenario [8]. Para analizar y estudiar dichos escenarios, los modelos de calidad del software describen un conjunto de características de calidad que son útiles para identificar, medir y evaluar la calidad del producto de software [5].

Respecto de los modelos de calidad del producto de software, el ISO/IEC 25010 [6] proporciona una guía para identificar los requisitos de calidad del producto. Este se compone por ocho características de calidad que debería tener un producto de software: adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad. Cada una de estas características tiene otras subcaracterísticas de calidad y juntas se tienen en cuenta al evaluar un producto de software.

Respecto de otras revisiones de literatura, Mishra y Otaiwi [5], realizaron un mapeo sistemático con el objetivo de analizar las implicaciones de las características de DevOps en la calidad del software. Este estudio tuvo como hallazgos positivos: que la cultura, la colaboración, la automatización, las mediciones y el monitoreo son los habilitadores claves de DevOps para la calidad del software; y que la calidad del software aumentará si aumentan los factores automatización, medición, cultura e intercambio; que la automatización es el factor crítico de éxito para mejorar la calidad del software en el entorno DevOps.

Por otra parte, Domínguez-Acosta y García-Mireles [4], con base en una revisión exploratoria de literatura, identificaron diez prácticas de DevOps con influencia en la calidad de software. Sus conclusiones sugieren la necesidad de abordar la perspectiva de calidad de producto, y cómo las prácticas individuales de DevOps tienen su efecto en esta perspectiva de calidad. Por su parte, Céspedes et al. [9] reporta una revisión sistemática de literatura en donde analiza la influencia de la aplicación de DevOps en la calidad del producto de software considerando el ISO/IEC 25010 [6]. Sus resultados indican un fuerte interés por estudiar algunas características de calidad del producto, específicamente confiabilidad y mantenibilidad. Además, las prácticas asociadas a DevOps, como el producto mínimo viable, la automatización de la implementación, la automatización de pruebas, la computación en la nube y la cooperación en equipo, muestran una relación con la mejora en la calidad del producto de software.

Independientemente de que en estos estudios se analicen aspectos vinculados con la relación entre DevOps, o sus prácticas, y la calidad del software, este trabajo en progreso tiene como objetivo la búsqueda de evidencia empírica que pueda sustentar una influencia positiva de prácticas específicas de DevOps en determinadas características de calidad de producto tal como se describen en el modelo ISO/IEC 25010 [6], de modo que podamos obtener una lista de prácticas con influencia positiva en ciertas características de calidad de producto, basadas en la evidencia empírica recopilada.

1. METODOLOGÍA

Un mapeo sistemático de literatura es un tipo de revisión sistemática de literatura donde se hace una revisión amplia de estudios primarios en un área temática específica que tiene como objetivo identificar qué evidencia está disponible sobre el tema [10]. Para la realización del mapeo sistemático en cuestión, se basó en Kitchenham y Charters [10]. El proceso de búsqueda tuvo en cuenta las siguientes base de datos: ACM Digital Library, IEEE Xplore, Scopus y Web of Science, las cuales fueron seleccionadas debido a su relevancia en el área de la Ingeniería de Software. Se utilizó la cadena general de búsqueda:

*(DevOps) AND (9126 OR 25010 OR usability OR maintainability OR security OR efficiency OR portability OR reliability OR functionality OR compatibility OR "non-functional requirements" OR "nonfunctional requirements" OR nfr OR "quality characteristics" OR "quality objectives" OR "quality factors" OR "quality requirements" OR "quality attributes" OR "software quality" OR "product quality")*.

* 1. Especificación de las preguntas de investigación

La pregunta principal de investigación es: ¿Qué impacto tienen las prácticas de DevOps en la calidad de producto definida según el modelo de calidad de producto de software de la norma ISO/IEC 25010? De esta pregunta general se derivan las siguientes preguntas específicas:

* RQ1. ¿Qué características de calidad del producto se informan en los estudios relacionados con DevOps?
* RQ2. ¿Qué prácticas de DevOps se informan que tienen influencia en la calidad del producto de software?
* RQ3. ¿Qué evidencia se tiene de que las prácticas informadas tengan influencia en la calidad?
* RQ4. ¿Qué recomendaciones se deben tomar para mejorar la calidad del producto al adoptar DevOps?
  1. Criterios de inclusión y exclusión

Criterios de inclusión:

* Estudios relacionados con Ingeniería de Software.
* Estudios que describen una relación entre DevOps y la calidad de productos de software.
* El artículo debe estar escrito en idioma inglés y haber sido publicado a partir del 2008 y hasta el 31 de diciembre de 2022.
* El resumen del artículo debe incluir al menos una característica o subcaracterística de calidad como las describe la norma ISO/IEC 25010 [6]. También puede hacer referencia a un término de calidad de producto como objetivos de calidad o requisitos de calidad.
* Estudios que describan casos empíricos relacionados con DevOps y calidad de producto de software o propuestas que aborden calidad de producto.

Criterios de exclusión:

* Estudios no relacionados con DevOps.
* Estudios que no se centren en la calidad de producto de software a través del uso de DevOps.
* Estudios en idioma distinto al inglés.
* Estudios que sean de fuentes secundarias.
* Estudios que no describan casos empíricos relacionados con DevOps y calidad de producto de software o propuestas que no aborden calidad de producto.
* Artículos duplicados.
* Artículos extendidos, donde se tomará la última versión publicada.
  1. Proceso de selección de estudios primarios

Para la selección de los estudios primarios se realizó una búsqueda en las bases de datos mencionadas, los resultados obtenidos se redujeron solo a los que fueran del área de Ciencias de la Computación, siendo esta la de interés para este estudio. En el caso de IEEE Xplore y ACM, la búsqueda se realizó de manera individual por título, palabras claves del autor y resumen, el resultado de cada búsqueda se agrupó en una sola lista de la cual se eliminaron los duplicados que pudieron haberse repetido al realizar la búsqueda por separado. Los resultados se tenían en listas individuales por cada base de datos por lo que se debieron integrar para conformar una única lista, una vez realizada esta integración se eliminaron los duplicados de modo que solo resultara una lista con resultados únicos.

Los criterios de selección fueron aplicados por el primer autor al revisar todos los resultados únicos que se obtuvieron del proceso anterior. Una primera revisión se realizó en el título, donde se descartaron estudios que no correspondían con el tema. Una segunda revisión se realizó en el resumen, para definir si verdaderamente el estudio estaba relacionado con el tema que se investiga, esto nos permitió obtener una lista de posibles candidatos. En caso de los trabajos en los que se tuvo dudas, se realizó una lectura de la introducción para identificar cualquier característica de calidad o un enfoque utilizado para tratarla, por lo que se decidió incluir términos genéricos de calidad del producto.

La tercera revisión se realizó para verificar si estos estudios resultantes cumplían con los criterios de inclusión que se definieron, con el fin de obtener una lista final de estudios primarios. El segundo autor verificó el proceso de selección para determinar la consistencia de los resultados.

1. RESULTADOS

Utilizando las cadenas de búsqueda para las cuatro bases de datos se obtuvieron un total de 1235 resultados. De la integración de los registros devueltos por las cuatro bases de datos y la eliminación de duplicados, se obtuvo una lista de 758 resultados únicos, la que fue revisada como se describe en el apartado *C.* de la sección anterior.

Después de la primera y segunda revisión se obtuvo una lista preliminar de 72 estudios, a los cuales se les aplicó rigurosamente los criterios de inclusión y exclusión, de lo que resultó una única lista de 19 estudios, los que finalmente fueron los seleccionados para realizar el proceso de extracción de datos para responder las preguntas de investigación de nuestro estudio. Los artículos que componen la lista de estudios primarios son: S01 [11], S02 [12], S03 [13], S04 [14], S05 [15], S06 [16], S07 [17], S08 [18], S09 [19], S10 [20], S11 [21], S12 [22], S13 [23], S14 [24], S15 [25], S16 [26], S17 [27], S18 [28] y S19 [29].

* 1. Frecuencia de publicaciones por año

Como lo muestra la Figura 1, la frecuencia de publicaciones va en aumento. Esta tendencia puede reflejar un creciente interés y reconocimiento de la importancia del tema en la comunidad científica y puede estar impulsado por la necesidad de abordar problemas urgentes como la calidad en medio de la agilidad del desarrollo de software, el desarrollo de mejores prácticas que impulsen resultados satisfactorios o encontrar soluciones innovadoras.

Figura 1. Frecuencia de publicaciones por año

* 1. Tipos de publicaciones

Un 71% de publicaciones en conferencias como lo indica la Figura 2, puede significar que la comunidad científica está más interesada en compartir y discutir resultados tempranos o trabajos en desarrollo en lugar de investigaciones más completas o finales ya que las conferencias suelen ser espacios para presentar trabajos en progreso, resultados preliminares o investigaciones en curso.

* 1. Tipos de artículos

Los artículos del estudio se clasificaron de acuerdo con lo propuesto por Wieringa [30]. Como lo indica la Figura 3, el 53% de los estudios se clasifican como investigación de validación, estos estudios están centrados en comprobar la validez o la aplicabilidad de conocimientos previos. El 21% de los estudios clasificados como propuestas de solución indica que una proporción significativa de los estudios se enfoca en abordar problemas o desafíos específicos, y presentan soluciones o enfoques nuevos para resolverlos.

Figura 2. Procedencia de los artículos primarios

Por último, el 26% de los estudios categorizados como investigación de evaluación indica que una parte de los estudios se centra en evaluar o medir el impacto, la efectividad y los resultados de las intervenciones en el sector industrial en el área de Ingeniería de Software.

Figura 3. Tipos de artículo

* 1. Características de calidad mencionadas

Figura 4. Características de calidad mencionadas

Según los estudios seleccionados, la Figura 4 muestra las características de calidad de producto identificadas. Funcionalidad o adecuación funcional, fiabilidad, usabilidad, eficiencia de desempeño, mantenibilidad y portabilidad se menciona en los estudios S02 [12], S08 [18]. También se menciona fiabilidad en el estudio S12 [22] y la eficiencia de desempeño en el estudio S11 [21]. La seguridad se identificó como una de las características de calidad de producto más estudiadas con un total de 9 trabajos que se centran únicamente en ella: S03 [13], S04 [14], S05 [15], S06 [16], S08 [18], S10 [20], S13 [23], S15 [25] y S16 [26]. Lo anterior puede estar indicando una tendencia a que se aborden la seguridad, fiabilidad y eficiencia de desempeño, lo que puede indicar una creciente preocupación por garantizar entornos seguros, protección de los datos y la privacidad de los usuarios, además una búsqueda por establecer sistemas o procesos confiables y consistentes y un mayor interés en optimizar recursos y maximizar el rendimiento. En el sector industrial, donde se realizaron la mayoría de los estudios que abordan la seguridad, las fallas de seguridad pueden tener consecuencias graves, como en el caso de dominios críticos como el automotriz, abordado en el estudio S13 [23].

* 1. Prácticas de DevOps relevantes para mejorar la calidad

A continuación, se presentan las prácticas de DevOps que se identificaron en los estudios y que basadas en la evidencia empírica proporcionada tienen una influencia positiva en la calidad de producto. Se presentan tres tablas, la Tabla I representa las prácticas de DevOps que tienen influencia en las ocho características de calidad del modelo ISO/IEC 25010 [6], pero dado que el estudio que las recopila S08 [18] no hace uso del modelo, pero sí utiliza las mismas características, se asume que hacían referencia al modelo ISO/IEC 25010 [6]. Sin embargo, no se tiene información de cómo fueron medidas.

Tabla i. Prácticas de devops con influencia en todas las características de calidad de producto

|  |  |
| --- | --- |
| Prácticas de DevOps con influencia en todas las características de calidad de producto | Estudios |
| * Configuración del entorno automatizado * Gestión del entorno de ejecución * Revisión * Gestión de pruebas * Gestión de configuración de activos de prueba * Administración de incidentes * Gestión de defectos, seguimiento de defectos, seguimiento de errores * Monitoreo de prueba * Relación de datos como base para informes de verificación y validación * Informe de verificación y validación | S08 [18] |

La Tabla II presenta prácticas de DevOps con influencia en cuatro características de calidad de producto específicas: fiabilidad, mantenibilidad, portabilidad y seguridad. Estas prácticas se pudieron recopilar de los estudios S02 [12], S08 [18] en los que los autores coinciden en que analizar el código y la automatización continua mejoran estas características.

La Tabla III presenta las prácticas con influencia positiva en la característica de producto eficiencia de desempeño, en la que los estudios S02 [12], S11 [21] han coincidido. Además, se presentan una lista de prácticas con influencia en la seguridad, siendo esta la característica más estudiada como lo indica la Figura 4 y de la que se pudo recopilar la mayor cantidad de prácticas.

Tabla ii. Prácticas de devops con influencia en algunas características de calidad de producto

|  |  |  |
| --- | --- | --- |
| Prácticas de DevOps | Características de calidad | Estudios |
| * Analizar el código * Automatización continua | Fiabilidad, Mantenibilidad, Portabilidad, Seguridad | S02 [12], S08 [18] |

Tabla iii. Prácticas de devops con influencia en la característica eficiencia de desempeño

|  |  |  |
| --- | --- | --- |
| Prácticas de DevOps | Características de calidad | Estudios |
| * Utilización de sistemas de control de versiones * Integración continua * Despliegue continuo * Aprovisionamiento continuo * Realización de evaluaciones de desempeño a través de modelos de desempeño | Eficiencia de desempeño | S02[12], S11[21] |

Las prácticas que a continuación se relacionan tienen influencia únicamente en la característica de seguridad, según la evidencia empírica que pudo ser recolectada en los estudios S03 [13], S04 [14], S05 [15], S06 [16], S08 [18], S10 [20], S13 [23], S16 [26]:

• Monitoreo automatizado

• Implementación automática de software

• Pruebas automáticas

• Entrega de software en pequeños incrementos

• Colaboración y comunicación entre diferentes departamentos de la organización del software

• Proporcionar capacitación en seguridad para los miembros del equipo de desarrollo

• Uso de actividades de seguridad no automatizadas

• Buena documentación y registro, automatización de los procesos

• Cumplimiento de la separación de roles.

• Pruebas de seguridad basadas en código

• Integración Continua de la seguridad

• Bucles de retroalimentación entre equipos de prácticas de desarrollo, seguridad y operaciones

• Participación de las partes interesadas

• Modelado de amenazas

• Escaneo de vulnerabilidades

• Pruebas de penetración

• Evaluación continua de vulnerabilidades a través de análisis de código estático y pruebas dinámicas

• Automatización de pruebas de seguridad

• Evaluación de peligros y riesgos

• Argumentación de seguridad basada en modelos

• Diseño seguro

• Implementación segura

• Pruebas seguras de verificación y validación

• Pautas de seguridad

• Gestión de actualizaciones de seguridad

• Gestión de seguridad

* 1. Modelos de calidad usados

Los estándares asociados a la calidad están siendo poco utilizados. En los estudios analizados solo se mencionan el ISO/IEC 62443, en el S16 [26], como norma que tiene por objetivo proteger los sistemas contra amenazas cibernéticas, asegurar la continuidad operativa, salvaguardar la integridad de los datos y proteger los activos críticos. Otro estándar utilizado es el ISO/IEC 29110 mencionado en el S14 [24], el cual busca cubrir la necesidad de establecer un marco de trabajo práctico y escalable que permita a las microempresas mejorar sus prácticas de ingeniería de software y desarrollar productos de calidad de manera eficiente. La norma se basa en los principios generales de las buenas prácticas de ingeniería de software, pero se adapta para ser aplicada en entornos con recursos limitados.

Estándares como el ISO/IEC 25010 [6] que se centra en la calidad del producto y proporciona un marco de referencia para evaluar características específicas relacionadas con el software, no se utilizan en los estudios analizados a pesar de que sí se mencionan las características de calidad que contiene el modelo, pero sin llegar a utilizarlo.

* 1. Evidencias encontradas

El estudio S04 [14] se realizó en el sector industrial con encuestados que eran representantes de 9 organizaciones internacionales que han adoptado DevOps y también tuvieron en cuenta 66 artefactos de internet para identificar actividades relacionadas con DevOps hacia la mejora de la seguridad. Indican que las pruebas automatizadas y la revisión de código, se referencian en más de diez artefactos de Internet y la automatización de pruebas lo practican ocho de los nueve profesionales encuestados, y de revisión de código lo practican 7. Con estas prácticas el estudio indica que se mejora la seguridad.

El estudio S5 [25], también se llevó a cabo en el sector industrial, específicamente realizado en IBM. Seleccionaron cinco proyectos que desarrollan e implementan aplicaciones internas de inteligencia de negocios de IBM. Entrevistaron a nueve personas, partes interesadas de los proyectos. Cinco de los nueve participantes sugirieron que la adopción de un proceso DevOps con implementación y pruebas automatizadas haría que el sistema fuera más seguro. Los cinco participantes mencionaron que la implementación automatizada daría como resultado implementaciones más rápidas, lo que a su vez reduciría la duración de la implementación de objetos de seguridad y correcciones de seguridad. Dos participantes sugirieron que las pruebas automatizadas para el control de acceso lo harían más sólido. Un participante mencionó que las pruebas manuales se limitan a las pruebas positivas y que esto podría mejorarse en presencia de la automatización. Otro participante mencionó que la prueba de las definiciones de vista de usuario debe automatizarse porque generalmente se olvida durante las implementaciones manuales.

El estudio S10 [20], se llevó a cabo en el sector industrial. Entrevistaron a representantes de cinco organizaciones de Medio Oriente y África del Norte que están adoptando SecOps en sus actividades diarias de gestión de servicios de tecnologías de la información. Los estudios de caso se realizaron en forma de entrevistas con los equipos DevSecOps de las empresas. Dado que su estudio tenía como objetivo explorar qué o cómo las prácticas de DevSecOps influyen en el trabajo de los profesionales en el proceso de gestión de servicios de tecnología de la información, utilizaron entrevistas semiestructuradas. También se utilizaron otras técnicas de recolección de datos, como la extracción de datos de los informes de desempeño y la observación directa. Las prácticas: integración continua de la seguridad y bucles de retroalimentación entre equipos de desarrollo, seguridad y operaciones, son implementadas por casi todos los participantes para la mejora de la seguridad.

El estudio S11 [21] realiza una encuesta con una variedad de participantes del sector industrial, cuya cantidad no se indica, solo se conoce que procesaron la información de 26 respuestas completas. La finalidad de la encuesta fue conocer el estado actual de las prácticas para abordar los problemas de rendimiento en las aplicaciones industriales DevOps. La mayoría de los participantes aplican la integración continua (54 %), mientras que el despliegue continuo (12 %) y el aprovisionamiento continuo (4 %) se aplican con menos frecuencia. La integración continua se aplica a menudo (38 %) en combinación con procesos ágiles, como Scrum. La mayoría de los participantes (54 %) utilizan datos en tiempo real para la mejora de procesos. Todos los profesionales encuestados utilizan sistemas de control de versiones (VCS). La gran mayoría usa Git (77 %) y/o SVN (38 %) como VCS. Jenkins es la solución "extremo a extremo" más popular para CI. La mayoría del 77 % de los profesionales usan Jenkins para compilaciones continuas y el 65 % de los profesionales usan Jenkins para implementar su software. Con estas prácticas el estudio indica que se ve mejorado el rendimiento.

El estudio S16 [26], es un caso de estudio también en el sector industrial, utilizaron un enfoque sistemático para integrar actividades de seguridad basadas en estándares en el conjunto de herramientas y procesos automatizados de DevOps y destacaron su potencial de automatización. Realizaron una evaluación de su enfoque en una gran empresa industrial considerando el estándar de seguridad IEC 62443­4­1. Los resultados arrojan que los sujetos destacaron el impacto positivo en la mejora de la seguridad.

* 1. Recomendaciones para mejorar la calidad de producto

El estudio S01 [11] recomienda para las organizaciones que comienzan con DevOps, adoptar todo como código bajo control de versión. Además, sugieren centrarse en la adopción de pruebas automatizadas en entornos, implementaciones continuas automatizadas, supervisión continua y automatizada y gestión de configuración. Estas prácticas tienen el impacto más positivo en la madurez y el desempeño. El estudio S02 [12] recomienda que todos los miembros del equipo deben ser conscientes de las tareas individuales dentro de la cultura DevOps. Cuando todos son conscientes de sus roles y responsabilidades, la mayoría de los desafíos se pueden superar. La automatización, la implementación continua y el cambio de las medidas de la implementación inicial de la organización son importantes para implementar DevOps con éxito.

El estudio S08 [18] menciona que es posible administrar continuamente un cierto nivel de calidad practicando un proceso general de integración continua y entrega continua y que para utilizar de manera efectiva las características de calidad, recomiendan que los gerentes de proyecto clasifiquen la calidad de los objetivos del software por características de calidad y, si es posible, también definan la calidad que se logrará en cada hito por características de calidad, clasificar las pruebas automatizadas utilizando características de calidad no solo permite considerar la cobertura de las pruebas, sino que también aumenta la productividad del desarrollo y que si se considera la utilización de atributos de calidad desde la etapa de planificación del proyecto, se pueden tomar las siguientes medidas para proporcionar evidencia de garantía de calidad para la gestión del proyecto: declarar la utilización de las características de calidad en el acta de constitución del proyecto, asignar las características de calidad a varias actividades, los gastos generales de utilizar las características de calidad deben incluirse en la estimación aproximada antes de que comience el proyecto.

El S14 [24] también proporciona recomendaciones para lograr mejoras en el producto, estas son: mapeo, diseño y separación de plataformas y partes de software para personalización mediante la utilización de mecanismos de línea de productos de software, aplicación de conceptos y modelos de diseño de software y que requieren pruebas como parte obligatoria en conjunto con el proceso de diseño de acuerdo con los requisitos del software y las listas de entrada de productos deben incluir el registro de trazabilidad y el registro de riesgos como parte de los requisitos de cierres del proyecto realizados al comienzo del proyecto y actualizados continuamente a lo largo del proyecto.

El estudio S11 [21] que aborda el rendimiento, recomienda asegurarse de que las herramientas se integren sin problemas en todos los procesos automatizados de DevOps, de manera que tanto el equipo de desarrollo y de operaciones puedan colaborar en la compilación e implementación de código en un entorno de producción. Dado que en su estudio observan que Jenkins CI es la herramienta de CI más popular recomiendan que los investigadores proporcionen complementos que permitan una fácil integración de sus herramientas de evaluación de rendimiento en Jenkins.

Algunos de los estudios que abordaron la seguridad también presentaron recomendaciones a tener en cuenta para lograr una mejora de la seguridad en DevOps. El S03 [13] recomienda que la seguridad se puede potenciar con un conjunto de herramientas adecuadas, que debe existir un estándar sobre cómo configurar pruebas de seguridad automatizadas, que podría ser beneficioso para las empresas obtener una descripción general de las características de seguridad que ofrecen los proveedores de servicios en la nube, pero también cuáles son las limitaciones y que lo más importante de la seguridad es que los desarrolladores estén capacitados en ella.

El S04 [14] recomienda como actividades útiles para potenciar la seguridad, el monitoreo automatizado, las pruebas automatizadas y la implementación automatizada de software. Además, sugiere que actividades como la colaboración supervisada con el equipo de seguridad podría ayudar a que la implementación automatizada sea beneficiosa para la seguridad del sistema como una actividad de DevOps.

Otras recomendaciones para mejorar la seguridad se mencionan en el estudio S05 [15]. Se sugiere que el desarrollador no debe diseñar los casos de prueba ya que los casos de prueba funcionan como guardianes de los otros entornos, que las partes interesadas deben iniciar y aprobar una solicitud de cambio antes de la implementación en producción y que debe haber derechos de acceso separados para cada uno de los tres entornos: acceso de desarrollo, acceso de TIC y acceso de producción. Un desarrollador no deberá tener derecho de acceso a los entornos de prueba y producción. El estudio S15 [25] recomienda que el evaluador se incorpore al proceso de desarrollo ágil, de modo que pueda revisar el estado del proyecto en términos de seguridad funcional con frecuencia. Por lo tanto, proponen agregar explícitamente el rol del "Evaluador de seguridad independiente".

1. Discusión

En esta sección se discutirán los resultados preliminares obtenidos de este trabajo en progreso, teniendo en cuenta la extracción de datos que se realizó de los estudios seleccionados para responder las preguntas de investigación que se trazaron.

RQ1. ¿Qué características de calidad del producto se informan en los estudios relacionados con DevOps?

A pesar de que se informan las ocho características de calidad del modelo ISO/IEC 25010 [6], los estudios no mencionan la utilización de este modelo, tampoco definen las características ni se tiene información en todos los casos de cómo fueron medidas. Solo el estudio S11 [21], que aborda el rendimiento, lo define y según la definición que utilizan se podría traducir a eficiencia de desempeño.

Un hallazgo encontrado en nuestro estudio pudiera indicar de una posible tendencia a abordar la fiabilidad, esto coincide con los resultados de Cespedes et al. [9] que también reflejan en su estudio la fiabilidad como una característica muy mencionada. Por otra parte, la característica de seguridad la encontramos en nuestro trabajo en mayor cantidad de estudios que en comparación del trabajo de Céspedes et al. [9]. Estas diferencias podrían estar asociadas al periodo de tiempo en que los estudios primarios fueron seleccionados, así como a criterios específicos de selección que diferencian nuestro mapeo sistemático de la revisión llevada a cabo por Céspedes et al. [9].

RQ2. ¿Qué prácticas de DevOps se informan que tienen influencia en la calidad del producto de software?

En la sección anterior, en el apartado E, Prácticas de DevOps relevantes para mejorar la calidad, se presenta un compendio de las prácticas encontradas en los estudios primarios seleccionados, basadas en la evidencia empírica proporcionada, que indican una relación positiva de estas prácticas con características de calidad de producto como se describen en la norma ISO/IEC 25010 [6]. Comparamos las prácticas que encontramos con la de un par de estudios que también se centran en DevOps y su influencia en la calidad de software. El objetivo de hacer esto era comprobar si las prácticas de nuestros hallazgos también figuraban entre las listadas por otros autores.

La primera comparación fue con el trabajo de Ankur et al. [31] los cuales realizaron una encuesta internacional en el sector industrial con respuestas de 1467 participantes tomadas de un conjunto que contiene una gran cantidad de datos sobre 24 enfoques de desarrollo de software, incluido DevOps. Estos autores listaron las prácticas según la frecuencia con que se implementaban, tomamos en cuenta para la comparación las prácticas que se hacían a menudo y siempre, encontrando como comunes en relación con nuestros hallazgos: integración continua, pruebas automatizadas y revisión.

La segunda comparación fue con el trabajo de Céspedes et al. [9], que encontró como prácticas con mayor impacto positivo en la calidad la automatización de la implementación, el despliegue automatizado, pruebas automatizadas, responsabilidad personal y cooperación en equipo, computación en la nube y monitoreo y retroalimentación. Encontramos similitudes con nuestros hallazgos en las pruebas automatizadas, colaboración en equipo y retroalimentación.

El análisis indica que prácticas como integración continua, revisión, colaboración en equipo y pruebas automatizadas tienen un impacto positivo en la mejora de la calidad de productos de software. Si bien es cierto que hay consenso en estas prácticas antes mencionadas, hay otras citadas por varios autores que parecieran referirse a lo mismo. Por ejemplo, nuestros hallazgos recogen la práctica *bucles de retroalimentación entre equipos*, y Céspedes et al. [9] encontró *cooperación en equipo*, lo que pareciera ser la misma práctica, pero de lo que no se tiene certeza que hagan referencia a la misma actividad. Por lo tanto, se sugiere que se describan las prácticas para realizar apropiadamente la síntesis relacionada con hallazgos de los estudios empíricos.

RQ3. ¿Qué evidencia se tiene de que las prácticas informadas tengan influencia en la calidad?

Todos los estudios que abordan una o varias características de calidad de producto y que se tuvieron en cuenta para la extracción de prácticas proporcionan evidencia empírica ya sea por medio de encuestas S02 [12], S04 [14], S06 [16], S11 [21], entrevistas S03 [13], S05 [15], o estudios de casos S13 [23], S15 [25], S16 [26], por lo que las prácticas recogidas están respaldadas con evidencia empírica, como se puede ver en la sección anterior en el apartado *G*. Aunque se consideran insuficientes, las evidencias recolectadas sí muestran un impacto positivo en la calidad de producto de software.

RQ4. ¿Qué recomendaciones se deben tomar para mejorar la calidad del producto al adoptar DevOps?

El análisis de las recomendaciones expuestas en el apartado *H.* de la sección anterior muestra que las pruebas automatizadas, implementaciones continuas y automatizadas, la colaboración y comunicación entre los miembros del equipo son las recomendaciones más sugeridas, y que a opinión de los estudios empíricos seleccionados tienen mayor impacto positivo en la calidad de un producto de software. Estas recomendaciones coinciden, además, con ser prácticas de DevOps de las más mencionadas como potenciales para alcanzar mejoras en la calidad de producto de software [5], [9].

1. amenazas a la validez

La contribución de este estudio, presentan algunas amenazas a su validez. Este trabajo, para la selección de los estudios primarios, se ha centrado en el proceso de búsqueda automática de estudios primarios en base de datos. No se realizaron actividades como bola de nieve o búsqueda manual para garantizar la exhaustividad del proceso. Los resultados presentados, hasta este punto de la investigación, se basa en un número limitado de artículos (19) y las prácticas, evidencias y características de calidad, se basan, principalmente, e encuestas de opinión. Por lo tanto, se requiere evidencia de proyectos reales para entender cómo se mide cada una de las características de calidad y cómo se determina la influencia de las prácticas de DevOps. Por otra parte, la diferencia entre los artículos reportados por Céspedes et al. [9] y nuestro mapeo exige una actividad adicional de verificación de la selección de estudios primarios que la consideraremos en la actualización de este estudio.

Otra amenaza a la validez podría ser que al extraer datos de estudios primarios, es importante considerar el contexto en el que se realizaron los estudios, en la totalidad de los estudios seleccionados, la investigación se realizó en el sector industrial pero no en todos los casos se proporcionó información del tamaño de la muestra, el término *calidad* se abordó de manera genérica en la totalidad de los casos y sin ofrecer una definición, por lo que puede ser una limitante debido a la falta de especificidad, ambigüedad y posible inexactitud en la descripción de los elementos a los que se refieren. Por otra parte, la poca utilización de modelos de calidad en los estudios relacionados con DevOps, pero en los que sí existía una relación entre DevOps y ciertas características de calidad, nos hizo asumir esas características como las define el modelo ISO/IEC 25010 [6], pero no son especificadas ni definidas, lo que amenaza con introducir diferencias en la interpretación, lo que puede llevar a conclusiones erróneas o inexactas.

1. conclusiones

Basado en los resultados preliminares del mapeo sistemático, se lograron responder las preguntas de investigación propuestas para abordar el tema, y aunque la evidencia empírica es aún insuficiente, en las que se pudieron obtener sí se aprecia un impacto positivo de prácticas de DevOps en determinadas características de calidad. Respecto al estado actual del tema se puede evidenciar que está ganando relevancia. La presencia significativa de artículos de validación indica un enfoque que estudia las propiedades de DevOps en distintos contextos. Sin embargo, La falta de utilización de estándares de calidad, señala una brecha en la aplicación de marcos de referencia ampliamente aceptados. Esto destaca la necesidad de promover el uso de estándares, como ISO/IEC 25010[6], que proporcionan un marco sólido para evaluar las características de calidad del producto de software. Por su parte, el mayor número de estudios que abordan la característica de seguridad destaca su importancia en el sector industrial y la necesidad de investigar y mitigar las nuevas amenazas emergentes.

Referencias Bibliográfica

1. [3] Debois, P. (2008, August). Agile infrastructure and operations: how infra-gile are you?. In Agile 2008 Conference (pp. 202-207). IEEE.
2. Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *Ieee Software*, *33*(3), 94-100.
3. Lazuardi, M., Raharjo, T., Hardian, B., & Simanungkalit, T. (2021). Perceived Benefits of DevOps Implementation in Organization: A Systematic Literature Review. In 2021 10th International Conference on Software and Information Engineering, ICSIE 2021 (pp. 10-16). (ACM International Conference Proceeding Series). Association for Computing Machinery. https://doi.org/10.1145/3512716.3512718
4. Domínguez-Acosta, M. F., & García-Mireles, G. A. (2021, October). Identifying Activities for Enhancing Software Quality in DevOps Settings. In 2021 10th International Conference On Software Process Improvement (CIMPS) (pp. 84-89). IEEE.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
5. Mishra, A., & Otaiwi, Z. (2020). DevOps and software quality: A systematic mapping. Computer Science Review, 38, 100308.
6. ISO, 2010. ISO/IEC FCD 25010: Systems and software engineering – system and software product quality requirements and evaluation (SQauRE) – System and software quality models.
7. García-Mireles, G. A., Moraga, M. Á., García, F., & Piattini, M. (2015). Approaches to promote product quality within software process improvement initiatives: a mapping study. *Journal of Systems and* *Software*, *103*, 150-166.
8. Kitchenham, B., & Pfleeger, S. L. (1996). Software quality: the elusive target [special issues section]. *IEEE software*, *13*(1), 12-21.
9. Céspedes, D., Angeleri, P., Melendez, K., Dávila, A. (2020). Software Product Quality in DevOps Contexts: A Systematic Literature Review. In: Mejia, J., Muñoz, M., Rocha, Á., A. Calvo-Manzano, J. (eds) Trends and Applications in Software Engineering. CIMPS 2019. Advances in Intelligent Systems and Computing, vol 1071. Springer, Cham. <https://doi.org/10.1007/978-3-030-33547-2>.
10. Kitchenham, B. A. & Charters, S. (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering* (EBSE 2007-001). Keele University and Durham University Joint Report.
11. T. Offerman, R. Blinde, C. J. Stettina and J. Visser, "A Study of Adoption and Effects of DevOps Practices," 2022 IEEE 28th International Conference on Engineering, Technology and Innovation (ICE/ITMC) & 31st International Association For Management of Technology (IAMOT) Joint Conference, Nancy, France, 2022, pp. 1-9, doi: 10.1109/ICE/ITMC-IAMOT55089.2022.10033313.
12. P. Perera, M. Bandara and I. Perera, "Evaluating the impact of DevOps practice in Sri Lankan software development organizations," 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), Negombo, Sri Lanka, 2016, pp. 281-287, doi: 10.1109/ICTER.2016.7829932.
13. N. Tomas, J. Li and H. Huang, "An Empirical Study on Culture, Automation, Measurement, and Sharing of DevSecOps," 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), Oxford, UK, 2019, pp. 1-8, doi: 10.1109/CyberSecPODS.2019.8884935.
14. A. A. U. Rahman and L. Williams, "Software Security in DevOps: Synthesizing Practitioners’ Perceptions and Practices," 2016 IEEE/ACM International Workshop on Continuous Software Evolution and Delivery (CSED), Austin, TX, USA, 2016, pp. 70-76.
15. V. Mohan, L. ben Othmane and A. Kres, "BP: Security Concerns and Best Practices for Automation of Software Deployment Processes: An Industrial Case Study," 2018 IEEE Cybersecurity Development (SecDev), Cambridge, MA, USA, 2018, pp. 21-28, doi: 10.1109/SecDev.2018.00011.
16. Roshan Namal Rajapakse, Mansooreh Zahedi, and Muhammad Ali Babar. 2021. An Empirical Analysis of Practitioners' Perspectives on Security Tool Integration into DevOps. In Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '21). Association for Computing Machinery, New York, NY, USA, Article 6, 1–12. <https://doi.org/10.1145/3475716.3475776>
17. Mali Senapathi, Jim Buchan, and Hady Osman. 2018. DevOps Capabilities, Practices, and Challenges: Insights from a Case Study. In Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 (EASE '18). Association for Computing Machinery, New York, NY, USA, 57–67. <https://doi.org/10.1145/3210459.3210465>
18. Daiju Kato, Ayumu Shimizu, and Hiroshi Ishikawa. 2022. Quality Classification for Testing Work in DevOps. In Proceedings of the 14th International Conference on Management of Digital EcoSystems (MEDES '22). Association for Computing Machinery, New York, NY, USA, 156–162. <https://doi.org/10.1145/3508397.3564840>
19. Trigo, A., Varajão, J., & Sousa, L. (2022). DevOps adoption: Insights from a large european telco. Cogent Engineering, 9(1) doi:10.1080/23311916.2022.2083474
20. Zaydi, M., & Nassereddine, B. (2021). DevSecOps Practices for an Agile and Secure IT Service Management. Defense AR Journal, 28(2), 239-239
21. Cor-Paul Bezemer, Simon Eismann, Vincenzo Ferme, Johannes Grohmann, Robert Heinrich, Pooyan Jamshidi, Weiyi Shang, André van Hoorn, Monica Villavicencio, Jürgen Walter, and Felix Willnecker. 2019. How is Performance Addressed in DevOps? In Proceedings of the 2019 ACM/SPEC International Conference on Performance Engineering (ICPE '19). Association for Computing Machinery, New York, NY, USA, 45–50. <https://doi.org/10.1145/3297663.3309672>
22. Bertolino, A., Angelis, G. D., Guerriero, A., Miranda, B., Pietrantuono, R., & Russo, S. (2023). DevOpRET: Continuous reliability testing in DevOps. Journal of Software: Evolution and Process, 35(3) doi:10.1002/smr.2298
23. S. Dupont et al., "Incremental Common Criteria Certification Processes using DevSecOps Practices," 2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Vienna, Austria, 2021, pp. 12-23, doi: 10.1109/EuroSPW54576.2021.00009.
24. Fandi Gunawan and Eko K. Budiardjo. 2021. &nbsp;A Quest of Software Process Improvements in DevOps and Kanban: A Case Study in Small Software Company. In 2021 The 4th International Conference on Software Engineering and Information Management (ICSIM 2021). Association for Computing Machinery, New York, NY, USA, 39–45. <https://doi.org/10.1145/3451471.3451478>
25. Zeller, M. (2021). Towards continuous safety assessment in Context of DevOps doi:10.1007/978-3-030-83906-2\_11
26. Moyón, F., Soares, R., Pinto-Albuquerque, M., Mendez, D., & Beckers, K. (2020). Integration of security standards in DevOps pipelines: An industry case study doi:10.1007/978-3-030-64148-1\_27
27. Christophe Ponsard and Valery Ramon. 2022. Survey of automation practices in model-driven development and operations. In Proceedings of the Fourth International Workshop on Bots in Software Engineering (BotSE '22). Association for Computing Machinery, New York, NY, USA, 14–17. <https://doi.org/10.1145/3528228.3528405>
28. P. Haindl, R. Plösch and C. Körner, "An Extension of the QUAMOCO Quality Model to Specify and Evaluate Feature-Dependent Non-Functional Requirements," 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Kallithea, Greece, 2019, pp. 19-28, doi: 10.1109/SEAA.2019.00012.
29. P. Haindl, R. Plösch and C. Kömer, "An Operational Constraint Language To Evaluate Feature-Dependent Non-Functional Requirements," 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Portoroz, Slovenia, 2020, pp. 34-42, doi: 10.1109/SEAA51224.2020.00017.
30. Wieringa, R., Maiden, N., Mead, N. et al. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Eng 11, 102–107 (2006). <https://doi.org/10.1007/s00766-005-0021-6>.

[31] Ankur Kumar, Mohammad Nadeem, and Mohammad Shameem. 2022. Assessing the Maturity of DevOps Practices in Software Industry: An Empirical Study of HELENA2 Dataset. In Proceedings of the 26th International Conference on Evaluation and Assessment in Software Engineering (EASE '22). Association for Computing Machinery, New York, NY, USA, 428–432. <https://doi.org/10.1145/3530019.3531335>