

DevOps y agilidad: Una revisión sistemática

DevOps and agility: A systematic review

Arturo Ricardo Alejos Yamamoto  ORCID, Cesar Luza montero  ORCID, Gloria

Maritza Valencia Vivas  ORCID

RESUMEN

Ágil y DevOps son dos metodologías que pueden ayudar a las organizaciones a abordar la complejidad de gestionar los requisitos y solicitudes de los clientes, reducir el tiempo de lanzamiento y mejorar la calidad del software al incorporar una mayor automatización y pruebas en el proceso de desarrollo. Ágil se centra en el desarrollo de software incremental e iterativo, mientras que DevOps se centra en la automatización integral del desarrollo y la entrega de software. El principal objetivo del presente trabajo fue el de realizar una revisión sistemática de literatura acerca de la influencia de DevOps en el desarrollo de proyectos utilizando metodologías ágiles, así como la influencia de ambos en la calidad del software, la entrega continua, así como su importancia en la seguridad de software. Para tal efecto se realizaron búsquedas en las bases de datos Scopus, Springer, ScienceDirect y Researchgate. Después de aplicar los criterios de inclusión-exclusión se seleccionaron un total de 47 artículos. Los resultados de la revisión nos muestran que la adopción generalizada de DevOps en proyectos ágiles ha experimentado un crecimiento sustancial convirtiéndose ambas en prácticas más comunes. Al analizar la calidad del software, se encontró que las características de automatización, colaboración y medición en la metodología DevOps están estrechamente ligadas a la calidad y seguridad en el desarrollo de software ágil.

Palabras claves

DevOps, Agilidad, Integración continua, Despliegue Continuo.

ABSTRACT

Agile and DevOps are two methodologies that can help organizations address the complexity of managing customer requirements and requests, reduce time to release, and improve software quality by incorporating greater automation and testing into the development process. Agile focuses on incremental and iterative software development, while DevOps focuses on end-to-end automation of software development and delivery. The main objective of this work was to carry out a systematic review of literature about the influence of DevOps in the development of projects using agile methodologies, as well as the influence of both on software quality, continuous delivery, as well as their importance in software security. For this purpose, searches were carried out in the Scopus, Springer, ScienceDirect and Researchgate databases. After applying the inclusion-exclusion criteria, a total of 47 articles were selected. The results of the review show us that the widespread adoption of DevOps in agile projects has experienced substantial growth, both becoming more common practices. When analyzing software quality, it was found that the automation, collaboration and measurement features in the DevOps methodology are closely linked to quality and security in agile software development.

Keywords

DevOps, Agility, Continuous Integration, Continuous Deployment.

INTRODUCCIÓN

El ámbito del desarrollo de software está en constante evolución a consecuencia de nuevas propuestas de metodologías (Masud et al., 2022). En los últimos años, hay un creciente interés por las metodologías ágiles (Krasteva & Ilieva, 2020) y el enfoque DevOps (Erich et al., 2017). La filosofía de las primeras es entregar versiones funcionales del software en iteraciones cortas y luego actualizar el software de acuerdo con los comentarios de los clientes (Shankarmani et al., 2012). DevOps es un esfuerzo organizacional colaborativo y multidisciplinario para automatizar la entrega continua de nuevas actualizaciones de software garantizando al mismo tiempo su corrección y confiabilidad (Leite et al., 2019).

La práctica de la entrega continua y el desarrollo continuo ha adquirido relevancia significativa en el ámbito del desarrollo de software y como resultado, es esencial considerarlas en la planificación estratégica. Además, se ha observado que las prácticas de DevOps enfocadas en la entrega continua y el desarrollo continuo impulsan la entrega ágil y frecuente de características adaptables, al mismo tiempo que garantizan la calidad del producto (Azad & Hyrynsalmi, 2023).

Ambas propuestas están relacionadas. DevOps puede entenderse como una evolución del desarrollo ágil, que fue concebido gradualmente a través de experiencias prácticas de implementación en el desarrollo de software (Almeida et al., 2022a). Considerando que el método ágil tiene su enfoque dirigido específicamente al desarrollo de software, y que DevOps pretende involucrar al área de desarrollo de software en la implementación y operación del software desarrollado, entonces, los procesos DevOps se pueden implementar dentro de los procesos Ágiles (Almeida et al., 2022a).

Se han identificado restricciones en la aplicación de metodologías ágiles y prácticas DevOps, las cuales abarcan la ausencia de colaboración entre los equipos de desarrollo, las áreas de prueba y entrega en diversos proyectos de software, así como la carencia de habilidades necesarias para llevar a cabo la implementación de dichas metodologías (Myklebust et al., 2020).

Las metodologías ágiles implican la incorporación de nuevas características en el sistema en cada entrega. Estas prácticas pueden incluir características completamente desarrolladas o presentar errores o fallos que impactan en la entrega (Mishra & Otaiwi, 2020). La influencia de la presión de calidad es un aspecto que incide en los procesos de desarrollo de software en diferentes fases, y DevOps se presenta como una solución propuesta para abordar dicha presión. La principal orientación de DevOps consiste en potenciar la velocidad, la frecuencia y la calidad de las implementaciones. Aunque hay extensas investigaciones bibliográficas sobre esta innovación en el desarrollo de software, se ha prestado escasa atención al impacto de DevOps en la calidad del software (Mishra & Otaiwi, 2020).

Los principios ágiles están firmemente arraigados en la cultura DevOps, la cual fomenta la comunicación y colaboración entre los equipos de Tecnologías de la información (TI); además, permite, adoptar una estrategia de lanzamientos más frecuentes, tanto en términos de incidencias como de mejoras, resulta beneficioso tanto para los desarrolladores como para las operaciones de TI, ya que contribuye a la estabilización del sistema. Las prácticas automatizadas, como las pruebas que mejoran la calidad y confiabilidad del código, junto con los despliegues, permiten realizar cambios de emergencia de manera más eficiente, lo que a su vez contribuye a la salud de la aplicación y a resolver incidentes de alto impacto de forma más rápida (Faustino et al., 2020).

Los paradigmas Ágiles y DevOps son compatibles, si es cierto ágil generó un modelo de entrega rápida alineado con las expectativas del cliente, DevOps lo optimizó,

indicando que ambas metodologías no solo se complementan, sino que ayudan a las empresas a afrontar cambios en un equipo. Así mismo enfatiza en el mismo contexto que las prácticas DevOps se centran en automatizar las diversas prácticas para entregar código de calidad en producción, así como la creación de entornos de prueba, indicando que la “Seguridad”, forma parte del término DevOps manejada como una etapa posterior al ciclo de vida de desarrollo de software. Donde las metodologías ágiles crean un espacio para realizar entregas parciales y DevOps crea el entorno adecuado para la gestión de estos procesos, así como una comunicación efectiva (Almeida et al., 2022b).

Si es cierto existen numerosos estudios respecto a implementaciones realizadas con DevOps y metodologías ágiles, cada una de estas se preocupa de manera aislada en la revisión de DevOps y la agilidad en seguridad o en la calidad o en la iteración y entrega continua, siendo muy pocos los estudios que han tratado la influencia de ambas tanto en la mejora de la calidad como en la seguridad y su rápida iteración y entrega continua.

El objetivo de esta revisión sistemática es establecer la influencia que tiene DevOps y las metodologías ágiles en la calidad, seguridad y entrega continua si es cierto intenta identificar diversas sus influencias en cada uno de estos puntos existen diferentes opiniones de diversos autores en el hecho de establecer una definición común respecto a DevOps y su influencia en las metodologías ágiles. Se han identificado restricciones en la aplicación de metodologías ágiles y prácticas DevOps, las cuales abarcan la ausencia de colaboración entre los equipos de desarrollo, las áreas de prueba y entrega en diversos proyectos de software, así como la carencia de habilidades necesarias para llevar a cabo la implementación de dichas metodologías (Myklebust et al., 2020).

Desde esta perspectiva, creemos que la revisión sistemática puede resultar altamente beneficiosa, ya que resume el conocimiento actual en relación con DevOps y agilidad, así como su impacto en aspectos como la calidad, seguridad, la entrega continua y la rápida iteración de entrega de software.

Las siguientes secciones se encuentra estructurado de la siguiente manera: La sección 2 de este artículo se describe los antecedentes y trabajos relacionados al desarrollo ágil de software y DevOps. La sección 3 describe la metodología utilizada para la revisión sistemática. La sección 4 expone los resultados obtenidos después de realizar el análisis detallado de los artículos revisados y por último se plantean algunos temas de Discusión y las Conclusiones.

MATERIALES Y MÉTODOS

Material

- **Desarrollo ágil de software**

El manifiesto ágil se propuso establecer principios para mejorar los enfoques de desarrollo de software existentes (Beck et al., 2001). La agilidad se destinó a resolver muchos de los problemas que se crearon en organizaciones intensivas en desarrollo con enfoques estructurados; además, el desarrollo de software ágil, que surgió como una práctica evolutiva de los enfoques estructurados existentes, abogó por incrementos iterativos de desarrollo de ciclo corto (Wesley, 2003).

Las metodologías ágiles proporcionan un contexto donde las personas pueden enfrentar desafíos complejos y adaptativos, al mismo tiempo que generan de manera productiva y creativa productos de máximo valor (Schwaber & Sutherland, 2017).

- **DevOps**

DevOps es una cultura, un proceso y una tecnología que promueve una colaboración estrecha entre el desarrollo de software y las operaciones e integra diversas disciplinas

para agilizar el desarrollo y la implementación de software, uniendo el desarrollo de software (Dev), los administradores de sistemas (Ops), así como el control de calidad y la comunidad de usuarios, a través de procesos alineados y herramientas de software. Su objetivo es mejorar la calidad del software, acelerar el desarrollo y la entrega, y fomentar la cooperación entre los equipos involucrados (Yarlagadda & Teja, 2018) (Spinellis & Diomidis, 2012) (Hernantes et al., 2015).

Dentro de la cultura de DevOps se abarcan elementos como la agilidad, la integración y el despliegue continuo, la automatización, la colaboración, la medición continua, el control de calidad, la resiliencia, el intercambio y la transparencia. Estos aspectos se pueden ampliar aún más para incluir la colaboración basada en la empatía, el respeto, la confianza, la responsabilidad, la alineación de incentivos y la comunicación abierta (Masombuka & Mnkandla, 2018) (Pinheiro Luz et al., 2019).

Metodología

- **Preguntas de revisión**

El diseño utilizado para la Revisión Sistemática de Literatura se ha basado en los lineamientos de (Kitchenham & Charters, 2007). Se consideran las fases de planificación, desarrollo y resultados. Estas fases también se han utilizados en las revisiones de literatura relacionadas con el desarrollo ágil realizadas por (Krasteva & Ilieva, 2020) y con el DevOps realizadas por (Teixeira et al., 2020). En la fase de planificación se plantean preguntas de investigación y se define el protocolo de revisión. En la fase de desarrollo se aplica el protocolo de revisión y seleccionan los artículos de acuerdo con los criterios de inclusión y exclusión definidos en el protocolo. Finalmente, en la fase de resultados se presentan los resultados del análisis de los artículos, que incluye tendencias estadísticas relevantes.

- **Planificación**

Las preguntas de investigación que guían el diseño de la revisión son:

P1 ¿Influye DevOps y las metodologías Ágiles en la calidad del software?

P2 ¿Contribuye DevOps a la entrega continua y la iteración rápida en un proyecto Ágil?

P3 ¿Influye DevOps en los proyectos de desarrollo Ágil?

P4 ¿Influye DevOps y las metodologías Ágiles en la Seguridad del software?

Para responder a estas preguntas se ha realizado una búsqueda de artículos de revistas indexadas en Scopus, Springer, SienceDirect y ResearchGate, considerando la siguiente cadena de búsqueda:

("Devops" AND "agile methodology") OR ("continuous delivery" AND "agile methodology") OR ("continuous integration" AND "agile methodology")

Además, se han considerado 4 criterios de inclusión y 4 de exclusión (Tabla 1).

Criterios de inclusión	Criterios de exclusión
1. Tipo documento: Artículo 2. Tipo de fuente: Journal y Conferencias 3. Idioma: inglés 4. Periodo: desde 2018	1. No presentan intersección de métodos ágiles e integración continua 2. No presentan intersección de métodos ágiles y despliegue continuo 3. No presentan intersección de métodos ágiles y DevOps 4. No responde al menos a una pregunta de investigación

Tabla 1. Criterios de inclusión y exclusión

- **Desarrollo**

Usando la cadena de búsqueda sobre las bases de datos establecidas en la planificación se obtuvieron potenciales documentos (Tabla 2).

Base de datos	Cantidad de potenciales documentos
Scopus	358
Springer	599
ScienceDirect	309
ResearchGate	410
Total	1676

Tabla 2. Cantidad de Potenciales documentos

Seguidamente se aplicó los criterios de inclusión exclusión en basa a la declaración PRISMA. La figura 1 presenta el diagrama de flujo Prisma que describe la aplicación de los criterios de inclusión y exclusión.

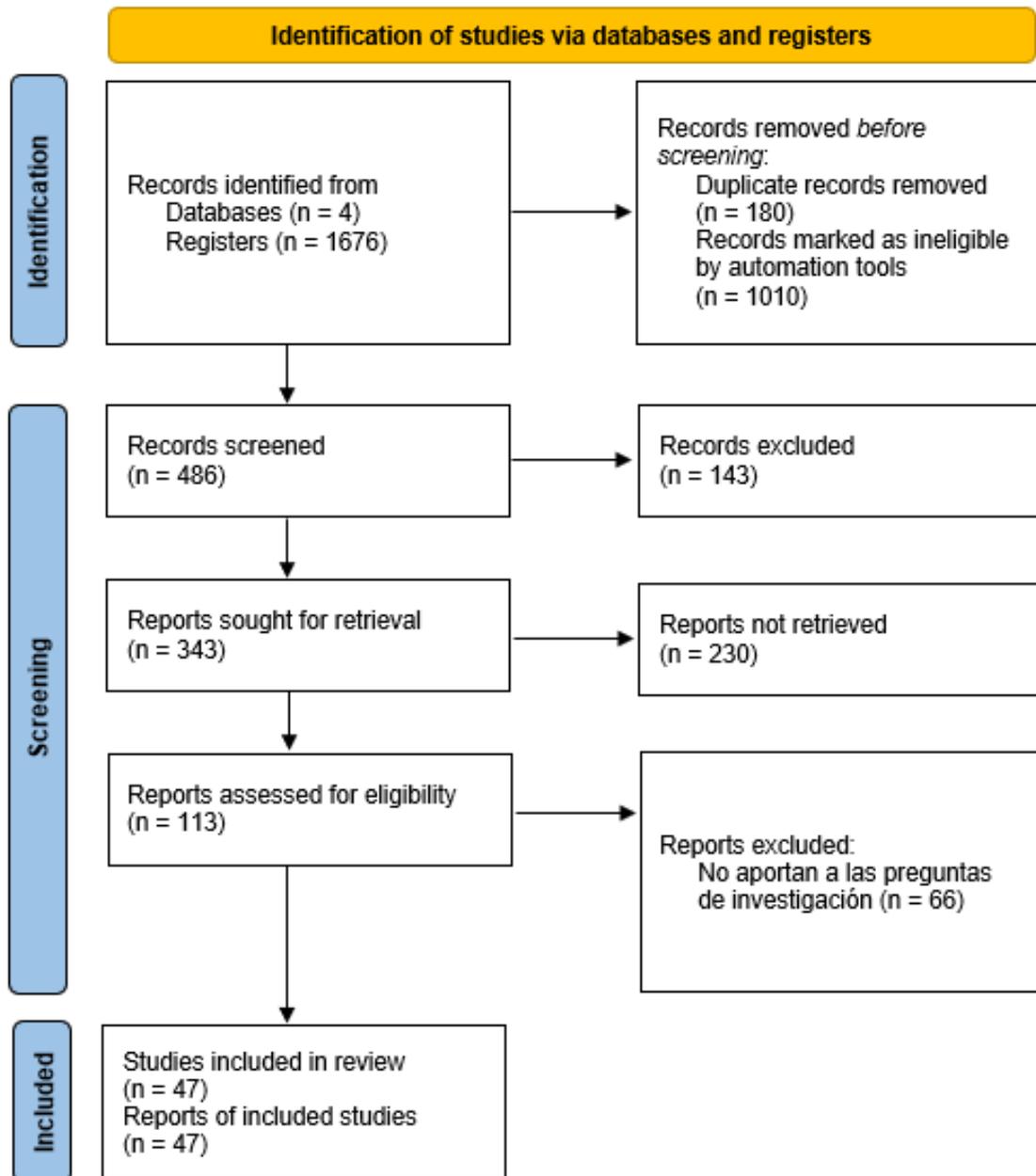


Figura 1 Diagrama de Flujo Prisma

Tabla 4. Artículos y su contribución a las preguntas

Resultados

Del total de artículos revisados a profundidad se observa que un 79% (37 artículos), indican que DevOps y las metodologías ágiles influyen en la calidad de software, un 100% (47 artículos) indican que DevOps Contribuye a la entrega continua y la iteración rápida en un proyecto ágil, un 100% (47 artículos) indican que DevOps influye de manera importante en los proyectos ágiles de desarrollo de software y un 38% (18 artículos) indica que tanto DevOps como las metodologías ágiles contribuyen en la Seguridad de software (figura 2)



Figura 2: Respuestas positivas por pregunta}

La cantidad de artículos seleccionados para su análisis en profundidad son 47. En la tabla 3 se presenta los artículos agrupados por su contribución en responder a las preguntas.

Pregunta	Referencia
P1	(Surace, Chabaud, & Fenouillet, 2018) (Srinivasan, Eppinger, & Joglekar, 2019) (Sorgalla, Wizenty, Rademacher, Sachweh, & Zündorf, 2021) (Sabau, Hacks, & Steffens, 2021) (Rinker et al., 2023) (Raj & Sinha, 2020) (Pardo, Guerrero, Suescún, & Researcher, 2022) (Paasivaara, Behm, Lassenius, & Hallikainen, 2018) (Oluyisola, Bhalla, Sgarbossa, & Strandhagen, 2022) (Myklebust et al., 2020) (Masud et al., 2022) (Muñoz, Negrete, & Arcilla-Cobián, 2021) (Mosser, Pulgar, & Reinharz, 2022) (Mishra & Otaiwi, 2020) (Maroukian & Gulliver, 2022) (Kovynyov, Buerck, & Mikut, 2021) (Klotins, Gorschek, Sundelin, & Falk, 2022) (Hernández, Moros, & Nicolás, 2023) (Hemon, Lyonnet, Rowe, & Fitzgerald, 2019) (Grünewald, 2021) (Granlund, Stirbu, & Mikkonen, 2021) (Girolamo, Klug, Luginin, & Martinez, 2018) (Ghantous & Gill, 2019) (Fukuzaki, Liu, & Butler, 2022) (Forbrig & Dittmar, 2019) (Faustino et al., 2020) (Ebert & Hochstein, 2023) (Donca et al., 2022) (Casale et al., 2020) (Ghantous & Gill, 2021) (Baron & Louis, 2023) (Barbareschi, Barone, Carbone, & Casola, 2022) (Azad & Hyrynsalmi, 2023) (Ashenden & Ollis, 2022) (Alt et al., 2020) (Alnamlah, Alshathry, Alkassim, & Jamail, 2021) (Almeida et al., 2022)
P2	(Wiedemann, Wiesche, Gewalt, & Krcmar, 2020) (Wagenaar, Overbeek, Lucassen, Brinkkemper, & Schneider, 2018) (Valdivia-Bedregal, Bedregal-Alpaca, & Castañeda-Huaman, 2021) (Surace et al., 2018) (Srinivasan et al., 2019) (Sorgalla et al., 2021) (Shi et al., 2022) (Sabau et al., 2021) (Rinker et al., 2023) (Reyes, Jimeno, & Villanueva-Polanco, 2023) (Raj & Sinha, 2020) (Premchand, Sandhya, & Sankar, 2019) (Pardo et al., 2022) (Paasivaara et al., 2018) (Oluyisola et al., 2022) (Myklebust et al., 2020) (Masud et al., 2022) (Muñoz et al., 2021) (Mosser et al., 2022) (Mishra & Otaiwi, 2020) (Maroukian & Gulliver, 2022) (Kovynyov et al., 2021) (Kohyarnejadfad, Aloise, Azhari, & Dagenais, 2022) (Klotins et al., 2022) (Hernández et al., 2023) (Hemon et al., 2019) (Grünewald, 2021) (Granlund et al., 2021) (Girolamo et al., 2018) (Ghantous & Gill, 2019) (Fukuzaki et al., 2022) (Forbrig & Dittmar, 2019) (Faustino et al., 2020) (Emmelmann, Corici, Eichhorn, Hauswirth, & Magedanz, 2022) (Ebert & Hochstein, 2023) (Donca et al., 2022) (Modica & Foschini, 2023) (Casale et al., 2020) (Ghantous & Gill, 2021) (Baron & Louis, 2023) (Barbareschi et al., 2022) (Azad & Hyrynsalmi, 2023) (Ashenden & Ollis, 2022) (Angara, Prasad, & Sridevi, 2018) (Alt et al., 2020) (Alnamlah et al., 2021) (Almeida et al., 2022)

P3	(Wiedemann et al., 2020) (Wagenaar et al., 2018) (Valdivia-Bedregal et al., 2021) (Surace et al., 2018) (Srinivasan et al., 2019) (Sorgalla et al., 2021) (Shi et al., 2022) (Sabau et al., 2021) (Rinker et al., 2023) (Reyes et al., 2023) (Raj & Sinha, 2020) (Premchand et al., 2019) (Pardo et al., 2022) (Paasivaara et al., 2018) (Oluyisola et al., 2022) (Myklebust et al., 2020) (Masud et al., 2022) (Muñoz et al., 2021) (Mosser et al., 2022) (Mishra & Otaiwi, 2020) (Maroukian & Gulliver, 2022) (Kovynyov et al., 2021) (Kohyarnejadfarid et al., 2022) (Klotins et al., 2022) (Hernández et al., 2023) (Hemon et al., 2019) (Gr̄unewald, 2021) (Granlund et al., 2021) (Girolamo et al., 2018) (Ghantous & Gill, 2019) (Fukuzaki et al., 2022) (Forbrig & Dittmar, 2019) (Faustino et al., 2020) (Emmelmann et al., 2022) (Ebert & Hochstein, 2023) (Donca et al., 2022) (Modica & Foschini, 2023) (Casale et al., 2020) (Ghantous & Gill, 2021) (Baron & Louis, 2023) (Barbareschi et al., 2022) (Azad & Hyrynsalmi, 2023) (Ashenden & Ollis, 2022) (Angara et al., 2018) (Alt et al., 2020) (Alnamlah et al., 2021) (Almeida et al., 2022)
P4	(Valdivia-Bedregal et al., 2021) (Surace et al., 2018) (Sabau et al., 2021) (Reyes et al., 2023) (Myklebust et al., 2020) (Mishra & Otaiwi, 2020) (Maroukian & Gulliver, 2022) (Klotins et al., 2022) (Hernández et al., 2023) (Gr̄unewald, 2021) (Granlund et al., 2021) (Ebert & Hochstein, 2023) (Casale et al., 2020) (Baron & Louis, 2023) (Barbareschi et al., 2022) (Azad & Hyrynsalmi, 2023) (Ashenden & Ollis, 2022) (Almeida et al., 2022)

Tabla 3: Autores y sus aportes a la revisión

En La tabla 5 se muestra qué artículos se incluyeron y su contribución a las preguntas de revisión.

Autor	P1	P2	P3	P4
(Wiedemann et al., 2020)		X	X	
(Wagenaar et al., 2018)		X	X	
(Valdivia-Bedregal et al., 2021)		X	X	X
(Surace et al., 2018)	X	X	X	X
(Srinivasan et al., 2019)	X	X	X	
(Sorgalla et al., 2021)	X	X	X	
(Shi et al., 2022)		X	X	
(Sabau et al., 2021)	X	X	X	X
(Rinker et al., 2023)	X	X	X	
(Reyes et al., 2023)		X	X	X
(Raj & Sinha, 2020)	X	X	X	
(Premchand et al., 2019)		X	X	
(Pardo et al., 2022)	X	X	X	
(Paasivaara et al., 2018)	X	X	X	
(Oluyisola et al., 2022)	X	X	X	
(Myklebust et al., 2020)	X	X	X	X
(Mosser et al., 2022)	X	X	X	
(Munoz et al., 2021)	X	X	X	
(Mosser et al., 2022)	X	X	X	
(Mishra & Otaiwi, 2020)	X	X	X	X
(Maroukian & R. Gulliver, 2020)	X	X	X	X
(Kovynyov et al., 2021)	X	X	X	
(Kohyarnejadfard et al., 2022)		X	X	
(Klotins et al., 2022)	X	X	X	X
(Hernández et al., 2023)	X	X	X	X
(Hemon et al., 2019)	X	X	X	
(Grünwald, 2022)	X	X	X	X
(Granlund et al., 2021)	X	X	X	X
(Di Girolamo, 2018)	X	X	X	
(Ghantous & Gill, 2019)	X	X	X	
(Fukuzaki et al., 2022)	X	X	X	
(Forbrig & Dittmar, 2019)	X	X	X	
(Faustino et al., 2020)	X	X	X	
(Emmelmann et al., 2022)		X	X	
(Ebert & Hochstein, 2023)	X	X	X	X
(Donca et al., 2022)	X	X	X	
(Di Modica & Foschini, 2023)		X	X	
(Casale et al., 2020)	X	X	X	X
(Bou Ghantous & Gill, 2021)	X	X	X	
(Baron & Louis, 2023)	X	X	X	X
(Barbareschi et al., 2022)	X	X	X	X
(Azad & Hyrynsalmi, 2023)	X	X	X	X
(Ashenden & Ollis, 2020)	X	X	X	X

(Angara et al., 2018)		X	X	
(Alt et al., 2020)	X	X	X	
(Alnamlah et al., 2021)	X	X	X	
(Almeida et al., 2022a)	X	X	X	X

Tabla 4: Autores y sus aportes a la revisión

A continuación, se exponen los resultados relacionados con cada una de las preguntas de investigación definidas en la revisión sistemática, con citas apropiadas para mejorar su comprensión.

- **Pregunta P1:** de acuerdo a la literatura analizada ¿Influye DevOps y las metodologías Ágiles en la calidad del software?

De acuerdo a los artículos revisados se encontró que, tanto las metodologías Ágiles como DevOps influyen en la gestión del alcance, la gestión de la calidad y la estimación, lo cual impacta las prácticas de gestión de proyectos (Srinivasan et al., 2019), así también se indica que DevOps surge como un enfoque para ayudar a las organizaciones a automatizar, optimizar costos, aumentar la rentabilidad, mejorar la estabilidad del proceso de desarrollo de software y la capacidad de respuesta de las organizaciones, y crear un proceso de desarrollo y lanzamiento ágil, mejorando la calidad del producto (Pardo et al., 2022).

De los artículos analizados se considera que la Integración continua (CI) de DevOps, es una buena práctica conocida y reportada en otras transformaciones ágiles a gran escala, por lo que representó una inversión significativa para que la CI funcionara cuando se trató de enfrentar los “desafíos de garantía de calidad” que se refiere a los elementos que se adaptan a las pruebas no funcionales, la falta de pruebas automatizadas y la ambigüedad de los requisitos afecta el control de calidad (Paasivaara et al., 2018) (Donca et al., 2022). Tanto Agile y DevOps han logrado avances revolucionarios en la ingeniería de software. Estas metodologías mejoran enormemente la calidad del software (Masud et al., 2022). En el desarrollo de software basado en métodos ágiles, donde DevOps integra técnicas a fin de disminuir el tiempo de entrega y la utilización de prácticas de monitoreo a fin de garantizar la calidad a través de la retroalimentación (Munoz et al., 2021).

Así también los autores señalan que, DevOps combina el desarrollo de software (ágil) y las operaciones de TI para ofrecer software de alta calidad, gracias a un ciclo de retroalimentación donde los "operadores" alimentan a los "desarrolladores" y viceversa (Mosser et al., 2022); por lo que es inevitable evolucionar hacia soluciones simples, rápidas e innovadoras, brindando opciones de instalación e implementaciones rápidas, flexibles y de calidad, así como gobernanza ágil de datos esperando eficiencia y calidad (Di Girolamo, 2018). Debido a que DevOps se caracteriza por su enfoque en la automatización de la calidad y la entrega continua (Forbrig & Dittmar, 2019). Así también las organizaciones adoptan DevOps y metodologías ágiles con el objeto de entregar software más rápido y con mayor calidad de software que los enfoques tradicionales (Alt et al., 2020).

- **Pregunta P2:** de acuerdo a la literatura analizada ¿Contribuye DevOps a la entrega continua y la iteración rápida en un proyecto ágil?

De los artículos analizados se indica que las prácticas DevOps aseguran una respuesta rápida a las necesidades del mercado, un desarrollo ágil de funciones y una integración continua de funcionalidades (Di Modica & Foschini, 2023); también

indica que influye en la reducción del tiempo de lanzamiento al mercado y los requisitos cambiantes (Angara et al., 2018). Así también DevOps integra las ventajas del desarrollo de software ágil para reaccionar rápidamente a las demandas de los clientes, ampliando la agilidad a operaciones como la arquitectura de software las responsabilidades y el conocimiento (Wiedemann et al., 2020).

Luego del análisis de los artículos encontramos que los requisitos de las aplicaciones distribuidas a menudo no se comprenden completamente o requieren cambios continuos durante la producción, se fomenta el uso de métodos ágiles y DevOps en el desarrollo y operación (Shi et al., 2022). Como resultado, hemos observado una mejora general en la productividad, permitiendo que los desarrolladores dediquen más tiempo a la innovación y creación de soluciones nuevas (Premchand et al., 2019).

- **Pregunta P3:** de acuerdo a la literatura analizada ¿Influye DevOps en los proyectos de desarrollo Ágil?

Luego del análisis de los artículos seleccionados, se pudo observar que cuanto más opera un equipo ágil de acuerdo con DevOps, más beneficios obtiene de sus propios artefactos como en el caso de un script de implementación y/o una lista de verificación de lanzamiento (Wagenaar et al., 2018). La arquitectura de microservicios impulsa la adopción de DevOps al requerir procesos de entrega continua, basados en contenedores y una plataforma de orquestación promueve el desarrollo ágil (Sorgalla et al., 2021).

Así también de la revisión tenemos que la agilidad se presenta como una necesidad para crear soluciones desde cero y el enfoque de DevOps ha surgido como una preferencia para gestionar el ciclo de desarrollo de código de forma continua y controlada, siguiendo un flujo de versiones (Oluyisola et al., 2022) y las organizaciones, continúan utilizando un enfoque ágil para el desarrollo de software utilizan DevOps porque proporciona un conjunto de herramientas, como el control de versiones y la integración continua, aumentando su rendimiento y productividad (Munoz et al., 2021); así como el desarrollo basado en pruebas, la automatización y la entrega continua (Kovynyov et al., 2021).

DevOps y los enfoques ágiles, como las arquitecturas de microservicios y la integración continua, se han vuelto necesarias dada la creciente necesidad de soluciones flexibles y escalables (Kohyarnjadfard et al., 2022). La relación entre DevOps y la agilidad favorece un modelo de gestión de proyectos ágil que permita la automatización y la integración continua (Ghantous & Gill, 2019).

- **Pregunta P4:** de acuerdo a la literatura analizada ¿Influye DevOps y las metodologías Ágiles en la Seguridad del software?

De la revisión sistemática tenemos que tanto DevOps y los métodos ágiles han sido ampliamente adoptados en la industria del software debido a que integran la seguridad durante todo el ciclo de vida de una aplicación (Hernández et al., 2023). El enfoque de seguridad ágil es un habilitador, para el desarrollo ágil junto a los procesos DevOps y enfatiza que mejora la comunicación entre Operadores, Usuarios y Desarrolladores y facilita las actualizaciones del software (Reyes et al., 2023).

De los artículos analizados se considera que las metodologías ágiles y el movimiento DevOps enfatiza la colaboración entre desarrollo y operaciones, aplicando estas técnicas también en la infraestructura. Además, la canalización de actividades de pruebas, integración, rendimiento y seguridad es una parte importante en esta capacidad (Sabau et al., 2021). Como parte de las metodologías ágiles la integración continua y la implementación continua, que forman parte del ciclo DevOps ha

ayudado a las organizaciones a crear software de manera eficaz, eficiente y de calidad (Reyes et al., 2023).

Así también indicar que las prácticas de integración y entrega continua permiten identificar defectos y problemas de calidad de manera temprana así también requiere practicar la seguridad en toda la tubería DevOps combinando los principios ágiles con una amplia automatización y eliminando los obstáculos organizativos (Klotins et al., 2022).

DISCUSIÓN

El análisis sistemático actual sobre DevOps y agilidad muestra un marcado interés en DevOps, lo que indica que los enfoques ágiles están ganando terreno en el desarrollo de software.

Se reconoce que DevOps es un tema de discusión relevante y se percibe como un factor de respaldo para las metodologías ágiles, revelando una conexión significativa entre ambas.

El desarrollo y las operaciones se han abordado como dos dominios separados. Sin embargo, de acuerdo con este estudio, existe un beneficio en explorar la unión de ambas perspectivas, ya que, en la industria, numerosas organizaciones están optando por la integración de desarrollo y operaciones.

Un problema descubierto en la literatura es la falta de una definición compartida concreta de DevOps. Por ejemplo, lo definen como un marco conceptual, otros autores lo sindicaron como una cultura o como una framework, mientras que otros lo ven como un conjunto de habilidades.

CONCLUSIONES

A través de la revisión sistemática de la literatura podemos inferir la influencia que tiene DevOps en los proyectos de desarrollos ágiles, así también como ambos benefician a los usuarios finales y desarrolladores de software brindando productos de calidad, seguros y con la rapidez esperada.

DevOps junto con las metodologías ágiles son un cambio importante en el desarrollo de sistemas de información. DevOps reduce la brecha entre los desarrolladores, las operaciones y el usuario final, lo que permite la detección temprana de problemas y las metodologías ágiles apoya las entregas de productos en ciclos cortos e iterativos de desarrollo.

Al introducir DevOps dentro de las metodologías ágiles se proporciona productos de software en ciclos más cortos de tiempo. DevOps representa un cambio cultural en el ámbito de TI, enfocándose en la entrega rápida de servicios de TI a través de la adopción de prácticas ágiles y ajustadas en un enfoque orientado al usuario final. Tanto la agilidad como DevOps nos proporcionan un enfoque que nos garantiza la calidad del producto mediante la colaboración tanto de los equipos de desarrollo como de producción, lo que nos permite entregar aplicaciones de software basadas en principios ágiles de manera más segura.

Referencias Bibliográficas.

- Almeida, F., Simões, J., & Lopes, S. (2022a). Exploring the Benefits of Combining DevOps and Agile. *Future Internet*, 14(2), 63. <https://doi.org/10.3390/fi14020063>
- Almeida, F., Simões, J., & Lopes, S. (2022b). Exploring the Benefits of Combining DevOps and Agile. *Future Internet*, 14(2), 63. <https://doi.org/10.3390/fi14020063>
- Alnamlah, B., Alshathry, S., Alkassim, N., & Mohd Jamail, N. S. (2021). The necessity of a lead person to monitor development stages of the DevOps pipeline. *Indonesian Journal of Electrical Engineering and Computer Science*, 21(1), 348. <https://doi.org/10.11591/ijeecs.v21.i1.pp348-353>
- Alt, R., Leimeister, J. M., Priemuth, T., Sachse, S., Urbach, N., & Wunderlich, N. (2020). Software-Defined Business: Implications for IT Management. *Business & Information Systems Engineering*, 62(6), 609-621. <https://doi.org/10.1007/s12599-020-00669-6>
- Angara, J., Prasad, S., & Sridevi, G. (2018). Towards Benchmarking User Stories Estimation with COSMIC Function Points—A Case Example of Participant Observatio. *International Journal of Electrical and Computer Engineering (IJECE)*, 8(5), 3076. <https://doi.org/10.11591/ijece.v8i5.pp3076-3083>
- Ashenden, D., & Ollis, G. (2020). Putting the Sec in DevSecOps: Using Social Practice Theory to Improve Secure Software Development. *New Security Paradigms Workshop 2020*, 34-44. <https://doi.org/10.1145/3442167.3442178>
- Azad, N., & Hyrynsalmi, S. (2023). DevOps critical success factors—A systematic literature review. *Information and Software Technology*, 157, 107150. <https://doi.org/10.1016/j.infsof.2023.107150>
- Barbareschi, M., Barone, S., Carbone, R., & Casola, V. (2022). Scrum for safety: An agile methodology for safety-critical software systems. *Software Quality Journal*, 30(4), 1067-1088. <https://doi.org/10.1007/s11219-022-09593-2>
- Baron, C., & Louis, V. (2023). Framework and tooling proposals for Agile certification of safety-critical embedded software in avionic systems. *Computers in Industry*, 148, 103887. <https://doi.org/10.1016/j.compind.2023.103887>
- Beck, K., Beedle, M., van Bennekum, A., Cunningham, W., Fowler, M., & Grenning, J. (2001). *Manifesto for Agile Software Development*.
- Bou Ghantous, G., & Gill, A. Q. (2021). Evaluating the DevOps Reference Architecture for Multi-cloud IoT-Applications. *SN Computer Science*, 2(2), 123. <https://doi.org/10.1007/s42979-021-00519-6>
- Casale, G., Artač, M., Van Den Heuvel, W.-J., Van Hoorn, A., Jakovits, P., Leymann, F., Long, M., Papanikolaou, V., Presenza, D., Russo, A., Srirama, S. N., Tamburri, D. A., Wurster, M., & Zhu, L. (2020). RADON: Rational decomposition and orchestration for serverless computing. *SICS Software-Intensive Cyber-Physical Systems*, 35(1-2), 77-87. <https://doi.org/10.1007/s00450-019-00413-w>
- Di Girolamo, G. (2018, mayo 28). SSA Data Systems evolution towards hybrid Architecture. *2018 SpaceOps Conference*. 2018 SpaceOps Conference, Marseille, France. <https://doi.org/10.2514/6.2018-2594>
- Di Modica, G., & Foschini, L. (2023). A Survey on the Use of Lightweight Virtualization in I4.0 Manufacturing Environments. *Journal of Network and Systems Management*, 31(2), 37. <https://doi.org/10.1007/s10922-023-09725-4>

- Donca, I.-C., Stan, O. P., Misaros, M., Gota, D., & Miclea, L. (2022). Method for Continuous Integration and Deployment Using a Pipeline Generator for Agile Software Projects. *Sensors*, 22(12), 4637. <https://doi.org/10.3390/s22124637>
- Ebert, C., & Hochstein, L. (2023). DevOps in Practice. *IEEE Software*, 40(1), 29-36. <https://doi.org/10.1109/MS.2022.3213285>
- Emmelmann, M., Corici, M., Eichhorn, F., Hauswirth, M., & Magedanz, T. (2022). Open 5G campus networks: Key drivers for 6G innovations. *E & i Elektrotechnik Und Informationstechnik*, 139(7), 589-600. <https://doi.org/10.1007/s00502-022-01064-7>
- Erich, F.M.A., Amri, C., & Daneva, M. (2017). A Qualitative Study of DevOps Usage in Practice. *J. Softw. Evol. and Proc.*
- Faustino, J., Pereira, R., & Alturas, B. (2020). Agile information technology service management with DevOps: An incident management case study. *Int. J. Agile Systems and Management*, 13(4).
- Forbrig, P., & Dittmar, A. (2019). Integrating HCD into BizDevOps by Using the Subject-Oriented Approach. En C. Bogdan, K. Kuusinen, M. K. Lárusdóttir, P. Palanque, & M. Winckler (Eds.), *Human-Centered Software Engineering* (Vol. 11262, pp. 327-334). Springer International Publishing. https://doi.org/10.1007/978-3-030-05909-5_21
- Fukuzaki, T., Liu, S., & Butler, M. (2022). DevFemOps: Enhancing maintainability based on microservices using formal engineering methods. *Connection Science*, 34(1), 2125-2138. <https://doi.org/10.1080/09540091.2022.2099347>
- Ghantous, G. B., & Gill, A. Q. (2019). An Agile-DevOps Reference Architecture for Teaching Enterprise Agile. *International Journal of Learning, Teaching and Educational Research*, 18(7), 128-144. <https://doi.org/10.26803/ijlter.18.7.9>
- Granlund, T., Stirbu, V., & Mikkonen, T. (2021). Towards Regulatory-Compliant MLOps: Oravizio's Journey from a Machine Learning Experiment to a Deployed Certified Medical Product. *SN Computer Science*, 2(5), 342. <https://doi.org/10.1007/s42979-021-00726-1>
- Grünwald, E. (2022). *Cloud Native Privacy Engineering through DevPrivOps* (Vol. 644, pp. 122-141). https://doi.org/10.1007/978-3-030-99100-5_10
- Hemon, A., Lyonnet, B., Rowe, F., & Fitzgerald, B. (2019). Conceptualizing the Transition from Agile to DevOps: A Maturity Model for a Smarter IS Function. En A. Elbanna, Y. K. Dwivedi, D. Bunker, & D. Wastell (Eds.), *Smart Working, Living and Organising* (Vol. 533, pp. 209-223). Springer International Publishing. https://doi.org/10.1007/978-3-030-04315-5_15
- Hernández, R., Moros, B., & Nicolás, J. (2023). Requirements management in DevOps environments: A multivocal mapping study. *Requirements Engineering*, 28(3), 317-346. <https://doi.org/10.1007/s00766-023-00396-w>
- Hernantes, J., Gallardo, G., & Serrano, N. (2015). IT Infrastructure-Monitoring Tools. *IEEE Software*, 32(4), 88-93.
- Kitchenham, B., & Charters, S. (2007). *Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3*. Keele University and University of Durham.
- Klotins, E., Gorschek, T., Sundelin, K., & Falk, E. (2022). Towards cost-benefit evaluation for continuous software engineering activities. *Empirical Software Engineering*, 27(6), 157. <https://doi.org/10.1007/s10664-022-10191-w>
- Kohyarnejadfar, I., Aloise, D., Azhari, S. V., & Dagenais, M. R. (2022). Anomaly detection in microservice environments using distributed tracing data analysis

- and NLP. *Journal of Cloud Computing*, 11(1), 25.
<https://doi.org/10.1186/s13677-022-00296-4>
- Kovynyov, I., Buerck, A., & Mikut, R. (2021). Design of transformation initiatives implementing organisational agility: An empirical study. *SN Business & Economics*, 1(6), 79. <https://doi.org/10.1007/s43546-021-00073-6>
- Krasteva, I., & Ilieva, S. (2020). *Adopting Agile Software Development methodologies in big data projects—a systematic literature review of experience reports* (pp. 2028-2033). IEEE International Conference on Big Data.
- Leite, L., Rocha, C., Kon, F., & Meirelles, P. (2019). A survey of DevOps concepts and challenges. *ACM Computing Surveys (CSUR)*, 1.35.
- Maroukian, K., & R. Gulliver, S. (2020). Leading DevOps Practice and Principle Adoption. *9th International Conference on Information Technology Convergence and Services (ITCSE 2020)*, 41-56.
<https://doi.org/10.5121/csit.2020.100504>
- Masombuka, T., & Mnkandla, E. (2018). *A DevOps collaboration culture acceptance model*. 279--285.
- Masud, S. M. R. A., Masnun, Md., Sultana, A., Sultana, A., Ahmed, F., & Begum, N. (2022). DevOps Enabled Agile: Combining Agile and DevOps Methodologies for Software Development. *International Journal of Advanced Computer Science and Applications*, 13(11).
<https://doi.org/10.14569/IJACSA.2022.0131131>
- Mishra, A., & Otaiwi, Z. (2020). DevOps and software quality: A systematic mapping. *Computer Science Review*, 38, 100308.
<https://doi.org/10.1016/j.cosrev.2020.100308>
- Mosser, S., Pulgar, C., & Reinhar, V. (2022). Modelling Agile Backlogs as Composable Artifacts to support Developers and Product Owners. *The Journal of Object Technology*, 21(3), 3:1. <https://doi.org/10.5381/jot.2022.21.3.a3>
- Munoz, M., Negrete, M., & Arcilla-Cobián, M. (2021). Using a platform based on the Basic profile of ISO/IEC 29110 to reinforce DevOps environments. *JUCS - Journal of Universal Computer Science*, 27(2), 91-110.
<https://doi.org/10.3897/jucs.65080>
- Myklebust, T., Stålhane, T., & Hanssen, G. K. (2020). Agile Safety Case and DevOps for the Automotive Industry. *Proceedings of the 30th European Safety and Reliability Conference and 15th Probabilistic Safety Assessment and Management Conference*, 4652-4657. https://doi.org/10.3850/978-981-14-8593-0_3495-cd
- Oluyisola, O. E., Bhalla, S., Sgarbossa, F., & Strandhagen, J. O. (2022). Designing and developing smart production planning and control systems in the industry 4.0 era: A methodology and case study. *Journal of Intelligent Manufacturing*, 33(1), 311-332. <https://doi.org/10.1007/s10845-021-01808-w>
- Paasivaara, M., Behm, B., Lassenius, C., & Hallikainen, M. (2018). Large-scale agile transformation at Ericsson: A case study. *Empirical Software Engineering*, 23(5), 2550-2596. <https://doi.org/10.1007/s10664-017-9555-8>
- Pardo, C. J., Guerrero, J., & Suescún Monsalve, E. (2022). DevOps model in practice: Applying a novel reference model to support and encourage the adoption of DevOps in a software development company as case study. *Periodicals of Engineering and Natural Sciences (PEN)*, 10(3), 221.
<https://doi.org/10.21533/pen.v10i3.3086>

- Pinheiro Luz, W., Pinto, G., & Bonifácio, R. (2019). *Adopting DevOps in the real world: A theory, a model, and a case study*. 157. <https://doi.org/10.1016/j.jss.2019.07.083>
- Premchand, A., Sandhya, M., & Sankar, S. (2019). Simplification of application operations using cloud and DevOps. *Indonesian Journal of Electrical Engineering and Computer Science*, 13(1), 85. <https://doi.org/10.11591/ijeecs.v13.i1.pp85-93>
- Raj, P., & Sinha, D. P. (2020). *Project Management In Era Of Agile And Devops Methodologies*. 9(01).
- Reyes, A., Jimeno, M., & Villanueva-Polanco, R. (2023). Continuous and Secure Integration Framework for Smart Contracts. *Sensors*, 23(1), 541. <https://doi.org/10.3390/s23010541>
- Rinker, F., Waltersdorfer, L., Meixner, K., Winkler, D., Lüder, A., & Biffel, S. (2023). Traceable Multi-view Model Integration: A Transformation Pipeline for Agile Production Systems Engineering. *SN Computer Science*, 4(2), 205. <https://doi.org/10.1007/s42979-022-01572-5>
- Sabau, A. R., Hacks, S., & Steffens, A. (2021). Implementation of a continuous delivery pipeline for enterprise architecture model evolution. *Software and Systems Modeling*, 20(1), 117-145. <https://doi.org/10.1007/s10270-020-00828-z>
- Schwaber, K., & Sutherland, J. (2017). *The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org.
- Shankarmani, R., Pawar, R., Mantha, S., & Babu, V. (2012). Agile methodology adoption: Benefits and constraints. *International Journal of Computer Applications*.
- Shi, Z., Ivankovic, V., Farshidi, S., Surbiryala, J., Zhou, H., & Zhao, Z. (2022). AWESOME: An auction and witness enhanced SLA model for decentralized cloud marketplaces. *Journal of Cloud Computing*, 11(1), 27. <https://doi.org/10.1186/s13677-022-00292-8>
- Sorgalla, J., Wizenty, P., Rademacher, F., Sachweh, S., & Zündorf, A. (2021). Applying Model-Driven Engineering to Stimulate the Adoption of DevOps Processes in Small and Medium-Sized Development Organizations: The Case for Microservice Architecture. *SN Computer Science*, 2(6), 459. <https://doi.org/10.1007/s42979-021-00825-z>
- Spinellis, & Diomidis. (2012). Don't Install Software by Hand. *IEEE Software*, 29(4), 86-87.
- Srinivasan, R., Eppinger, S. D., & Joglekar, N. (2019). The Structure of Devops in Product-Service System Development. *Proceedings of the Design Society: International Conference on Engineering Design*, 1(1), 3111-3120. <https://doi.org/10.1017/dsi.2019.318>
- Surace, C., Chabaud, P.-Y., & Leurent, M. (2018, mayo 28). Software development using Agile and DevOps approach: Lessons learned and technics in space projects. *2018 SpaceOps Conference*. 2018 SpaceOps Conference, Marseille, France. <https://doi.org/10.2514/6.2018-2570>
- Teixeira, D., Pereira, R., Henriques, T. A., Silva, M., & Faustino, J. (2020). A systematic literature review on DevOps capabilities and areas. 1-22.
- Valdivia-Bedregal, J., Bedregal-Alpaca, N., & Castañeda-Huaman, E. (2021). Private LTE Network Service Management Model, based on Agile Methodologies, for Big Mining Companies. *International Journal of Advanced Computer Science and Applications*, 12(4). <https://doi.org/10.14569/IJACSA.2021.0120451>

- Wagenaar, G., Overbeek, S., Lucassen, G., Brinkkemper, S., & Schneider, K. (2018). Working software over comprehensive documentation – Rationales of agile teams for artefacts usage. *Journal of Software Engineering Research and Development*, 6(1), 7. <https://doi.org/10.1186/s40411-018-0051-7>
- Wesley, A. (2003). *Lean software development: An agile toolkit*.
- Wiedemann, A., Wiesche, M., Gewalt, H., & Krcmar, H. (2020). Understanding how DevOps aligns development and operations: A tripartite model of intra-IT alignment. *European Journal of Information Systems*, 29(5), 458-473. <https://doi.org/10.1080/0960085X.2020.1782277>
- Yarlagadda, & Teja, R. (2018). *Understanding DevOps & bridging the gap from continuous integration to continuous delivery*. 5.