



Proceso para fomentar y apoyar la adopción de DevOps en PyMEs de software

Process to Encourage and Support the Adoption of DevOps in Software VSEs

Processo para incentivar e apoiar a adoção de DevOps em PMEs de software

Carlos Orozco¹

César Pardo²

Karen Zúñiga³

Samuel-Camilo Certuche⁴

Recibido: febrero de 2022

Aceptado: agosto de 2022

Para citar este artículo: Certuche S. C., Zúñiga, K., Pardo, C. y Orozco, C. (2022). Proceso para fomentar y apoyar la adopción de DevOps en PyMEs de software. *Revista Científica*, 45(3), 422-437. <https://doi.org/10.14483/23448350.19644>

Resumen

En este artículo se presenta un proceso para fomentar y apoyar la adopción de DevOps en pequeñas y medianas empresas de software. El proceso fue diseñado a través de la identificación de elementos sugeridos en la literatura por medio de un mapeo sistemático. Los elementos identificados fueron armonizados, comparados e integrados, dando como resultado un proceso claro, homogéneo y sin conflictos terminológicos compuesto por tres subprocesos, roles, actividades, artefactos, herramientas y flujos de proceso en BPMN como apoyo a la gestión de configuración, integración, despliegue y monitoreo continuo desde un enfoque fundamental y complementario de prácticas sugeridas. El proceso propuesto fue evaluado a través de un grupo focal conformado por expertos en DevOps, quienes consideraron que es pertinente, claro, completo

y aplicable en pequeñas y medianas empresas de software. El proceso describe un total de 82 actividades, 16 artefactos, 9 roles y recomienda 13 herramientas tecnológicas para apoyar la automatización de cada una de las actividades relacionadas con cada subproceso, que permitirá guiar a los profesionales y las empresas para disminuir la subjetividad en la comprensión y adopción de DevOps.

Palabras clave: desarrollo y operaciones, DevOps, integración continua, entrega continua, despliegue continuo, gestión de la configuración, monitoreo continuo

Abstract

This article presents a process to encourage and support the adoption of DevOps in small and medium software companies. The process was designed through the identification of elements suggested

1. Universidad del Cauca, Popayán, Colombia. Contacto: carlosorozco@unicauca.edu.co
2. Universidad del Cauca, Popayán, Colombia. Contacto: cpardo@unicauca.edu.co
3. Universidad del Cauca, Popayán, Colombia. Contacto: karenza@unicauca.edu.co
4. Universidad del Cauca, Popayán, Colombia. Contacto: scamiloc@unicauca.edu.co

and identified in the literature through systematic mapping. The identified elements were harmonized, compared and integrated, resulting in a clear, homogeneous process without terminological conflicts composed of three sub-processes, roles, activities, artifacts, tools and process flows in BPMN as support for configuration management, integration, continuous deployment and monitoring from a fundamental and complementary approach of suggested practices. The proposed process was evaluated through a focus group made up of DevOps experts, who considered that the process is pertinent, clear, complete and applicable in small and medium-sized software companies. In short, the process describes a total of 82 activities, 16 artifacts, 9 roles and recommends 13 technological tools to support the automation of each of the activities related to each thread, which will guide professionals, companies and stakeholders to reduce the ambiguity and subjectivity in the understanding and adoption of DevOps.

Keywords: development and operation, DevOps continuous integration, continuous deployment, continuous delivery, software configuration management, continuous monitoring

Resumo

Este artigo apresenta um processo para incentivar e apoiar a adoção de DevOps em pequenas e médias empresas de software. O processo foi desenhado por meio da identificação de elementos sugeridos e identificados na literatura por meio de mapeamento sistemático. Os elementos identificados foram harmonizados, comparados e integrados, resultando em um processo claro, homogêneo e sem conflitos terminológicos composto por três subprocessos, papéis, atividades, artefatos, ferramentas e fluxos de processos em BPMN como suporte para gerenciamento de configuração, integração, implantação contínua e acompanhamento a partir de uma abordagem fundamental e complementar das práticas sugeridas. O processo proposto foi avaliado por meio de um grupo focal formado por especialistas em DevOps, que consideraram o processo pertinente, claro, completo e aplicável em pequenas e médias empresas de software. Em suma, o processo descreve um

total de 82 atividades, 16 artefatos, 9 papéis e recomenda 13 ferramentas tecnológicas para apoiar a automação de cada uma das atividades relacionadas a cada segmento, o que orientará profissionais, empresas e stakeholders a reduzir a subjetividade na compreensão e adoção do DevOps.

Palavras-chaves: desenvolvimento e operação, DevOps, gerenciamento de configuração de software, integração contínua, implantação contínua, entrega contínua, monitoramento contínuo

Introducción

El desarrollo de soluciones software involucra dos áreas inicialmente separadas: (i) el área de desarrollo (Dev - Development), la cual se enfoca en prácticas, actividades, roles y tareas que se asocian a la implementación del producto final, así como actividades de diseño, análisis, implementación, control de calidad, despliegue y puesta en producción, entre otros ([Conradi y Fuggetta, 2002](#)), y (ii) el área de operaciones (Ops - Operations), la cual da solución a las tareas que involucran gestión, implementación y monitoreo de los aspectos relacionados con la infraestructura que soporta los proyectos que desarrollan las empresas ([Jabbari et al., 2016](#)). Actualmente es posible encontrar diferentes enfoques, modelos y marcos de referencia que apoyan el área de desarrollo de software (Dev), estas soluciones se pueden clasificar en: tradicionales y ágiles, las cuales se enfocan en dar solución a aspectos relacionados con el ciclo de vida de los productos o servicios software desde la perspectiva del desarrollo. Entre los enfoques tradicionales más utilizados se encuentran: CMMI ([CMMI Institute, 2018](#)), RUP ([Rational Software, 2020](#)), Modelo en cascada (Waterfall) ([Royce, 1987](#)), modelo en espiral ([Boehm, 1988](#)), diseño rápido de aplicaciones (RAD) ([Martin, 1991](#)), entre otros; asimismo, entre los enfoques y marcos ágiles más conocidos y utilizados se encuentran: Scrum ([Schwaber y Sutherland, 2017](#)), Lean Software ([Poppendieck y Poppendieck, 2003](#)), Test Driven Development (TDD) ([Beck, 2002](#)), Extreme Programming (XP)

([Beck y Andres, 2000](#)), Crystal Clear ([Cockburn, 2004](#)), Adaptive Software Development ([Highsmith, 2013](#)), Dynamic Systems Development Method ([Stapleton, 1997](#)), entre otros. Por otro lado, también existen procesos, marcos de trabajo de facto y soluciones utilizadas para apoyar el ciclo de vida de la infraestructura y la operación (Ops), entre los más conocidos se encuentran: Information Technology Infrastructure Library (ITIL) ([Hochstein et al., 2005](#)), Control Objectives for Information and Related Technologies (COBIT) ([Young et al., 2004](#)), estándares internacionales como la norma ISO/IEC 29110 ([Galván-Cruz et al., 2020](#)), y la norma ISO/IEC 20000 ([ISO/IEC, 2013](#)). Las soluciones mencionadas tienen como objetivo proponer los elementos relacionados con la administración de servicios de tecnologías de la información (TI).

Aunque se pueden observar soluciones que permitan potenciar tanto el proceso de desarrollo (Dev) como las operaciones (Ops), cada solución define prácticas, elementos y actividades desde su propia perspectiva, sin contemplar la realización de las dos áreas de manera integrada. Lo anterior surge debido a que, si bien las dos áreas son aplicadas en paralelo por las empresas, estas tienen objetivos diferentes, por ejemplo: (i) el desarrollo busca la entrega de productos funcionales de manera ágil y continua mediante la entrega constante de valor a los clientes ([Conradi y Fuggetta, 2002](#)), y (ii) las operaciones buscan establecer marcos de trabajo que permitan definir, controlar, monitorear y mantener la integridad de los procesos relacionados con la infraestructura tecnológica que soporta el proceso de desarrollo ([Stapleton, 1997](#)).

En el año 2009 en el evento DevOps Days, Debois sugirió por primera vez el término DevOps ([Debois, 2009](#)) con el objetivo de cerrar la brecha existente entre Dev y Ops, con el propósito de unificar el desarrollo y la operación del software en un proceso integrado, continuo y estructurado, derribando las barreras entre el área de desarrollo y la de operaciones, desde entonces esta filosofía ha sido ampliamente extendida en la industria de

software ([Virmani, 2015](#)). Inicialmente, DevOps fue concebido con el propósito de potenciar el proceso de desarrollo de software a través de la automatización del ciclo de vida de desarrollo mediante el uso de prácticas relacionadas con integración continua ([Shahin et al., 2017](#)), pruebas automatizadas ([Michelsen, 2014](#)), entrega y despliegue continuo ([Diel et al., 2016](#)), gestión de la configuración ([Orozco et al., 2020](#)). Por otra parte, de acuerdo con [Shahin et al. \(2017\)](#), DevOps trae consigo otra serie de beneficios relacionados con aspectos como: (i) aumento en la tasa de funcionalidades puestas en producción, (ii) despliegues y lanzamientos tempranos en intervalos cortos de tiempo, (iii) mejora en la colaboración y la comunicación de los miembros del equipo, (iv) pruebas con clientes reales ([ISO/IEC, 2019](#)).

A partir de los resultados obtenidos tras realizar un mapeo sistemático de la literatura ([Guerrero et al., 2020](#)), fue posible identificar diferentes aproximaciones y esfuerzos que proponen soluciones para apoyar el uso de DevOps en empresas de software, algunos de ellos son: (i) un marco de trabajo para apoyar procesos de gestión de la configuración ([Debois, 2009](#)), (ii) un modelo ágil basado en prácticas de entrega continua ([Shahin et al., 2017](#)), (iii) modelos para evaluar el grado de adopción de DevOps en empresas de software ([Jones et al., 2016](#); [Orozco-Garcés et al., 2022](#)), (iv) una metodología para aplicar DevOps en la nube ([Diel et al., 2016](#)). Además, fue posible identificar estudios que describen elementos de proceso como roles, artefactos, actividades y tareas relacionadas con DevOps. Sin embargo, los elementos de proceso identificados en la literatura son difusos y no son definidos a través de un marco de referencia claro que permita clarificar o formalizar los elementos de proceso presentes en la literatura a través de una solución genérica. Como consecuencia es posible observar una tendencia recurrente hacia el uso incompleto, incorrecto y desarticulado de las prácticas de DevOps relacionadas con calidad, gestión de la configuración e integración, despliegue y entrega y monitoreo continuo.

En este sentido, aunque aplicar DevOps trae consigo beneficios a las empresas de software, aún sigue siendo un desafío su aplicación debido a: (i) falta de definiciones estandarizadas y estructuradas que permitan establecer un estado del conocimiento claro y homogéneo en torno a DevOps ([ISO/IEC, 2013](#)), (ii) un alto grado de heterogeneidad en los mecanismos utilizados para aplicar prácticas específicas de DevOps como calidad, gestión de la configuración, integración, entrega y despliegue continuo, (iii) falta de procesos claros que permitan guiar de manera sistemática y organizada el conjunto de actividades, roles, tareas y responsables relacionados con las diferentes etapas involucradas en la definición y aplicación de DevOps, (iv) falta de claridad en la aplicación e institucionalización de las prácticas asociadas a DevOps, y (v) conflictos derivados del uso inadecuado, informal y poco organizado de DevOps ([Virmani, 2015](#)). Teniendo en cuenta lo anterior, en este artículo se presenta un proceso que fue desarrollado con el propósito de describir de manera clara y detallada los elementos fundamentales para facilitar la implementación de las prácticas de gestión de configuración, integración, entrega, despliegue y monitoreo continuo en empresas de software que aplican DevOps, el cual ha sido evaluado por expertos en la temática a través de un grupo focal. Además de la presente introducción, la estructura del artículo es la siguiente; la segunda sección presenta el análisis del estado del arte actual en torno a definición, conceptualización y uso de DevOps; la tercera sección presenta el proceso propuesto como resultado de realizar un análisis y una armonización de los elementos identificados en la literatura; la cuarta sección describe el protocolo y los resultados obtenidos tras aplicar un grupo focal como método cualitativo de estudio realizado con el propósito de identificar el grado de aplicabilidad, completitud, comprensibilidad e idoneidad de la propuesta; finalmente, la quinta sección presenta las conclusiones y el trabajo futuro.

Metología

Análisis del estado del arte

Con el propósito de conocer el estado del arte en torno a DevOps, se llevó a cabo un mapeo sistemático de la literatura, el cual puede ser consultado con mayor detalle en [Guerrero et al. \(2020\)](#). Como resultado, fue posible observar que se han realizado avances y esfuerzos significativos en la definición y la especificación de DevOps, así como el establecimiento de procesos, modelos, técnicas y herramientas que buscan describir un marco de trabajo para aplicar esta filosofía. De acuerdo con el análisis de los resultados obtenidos, estos se pueden clasificar en: (i) *estudios exploratorios*, que proponen definiciones, beneficios y desafíos relacionados con DevOps, (ii) *soluciones metodológicas*, que proponen modelos, procesos, técnicas y marcos de trabajo para apoyar el uso de DevOps y (iii) *herramientas*, que presentan o comparan soluciones tecnológicas diseñadas para sustentar diferentes prácticas de DevOps así como: integración, entrega y despliegue continuo, entre otros. A continuación, se presenta el detalle de los estudios identificados durante el análisis de los resultados.

Estudios exploratorios

Debido a que DevOps es un concepto relativamente nuevo en la industria, es importante preguntarse: ¿qué se entiende por DevOps? Como resultado, se identificó que no existe un consenso general que permita determinar un entendimiento común sobre DevOps ([Hemon et al., 2020](#)). A partir del análisis de los resultados, se observó que DevOps es presentado como: un enfoque ([Samarawickrama y Perera, 2017](#)), un movimiento cultural ([Lwakatare et al., 2015](#)), un conjunto de prácticas ([Bou Ghantous y Gill, 2017](#); [Riungu et al., 2016](#)), un paradigma ([Gupta et al., 2017](#)), una mezcla de dos palabras ([Erich et al., 2014](#)), un fenómeno ([Bucena y Kirikova, 2017](#)), un tema interdisciplinario ([Jabbari et al.,](#)

2016), un conjunto de principios (Elberzhager et al., 2017), un marco conceptual (Krusche et al., 2014), una metodología de desarrollo y filosofía (Benguria et al., 2018), una palabra artificial (Nagarajan y Overbeek, 2018) y un neologismo (de França et al., 2016). Por otro lado, los resultados del análisis del estado del arte permitieron identificar diferentes beneficios y desafíos que enfrentan las empresas que aplican DevOps. Con relación a los beneficios fue posible observar lo siguiente: (i) mejores tiempos de respuesta (*time to market*), (ii) optimización de procesos internos como resultado de cerrar la brecha entre el desarrollo y la operación y (iii) mayor satisfacción de los clientes. No obstante, también fue posible identificar desafíos relacionados con: (i) resistencia al cambio en las organizaciones, (ii) poca o nula comprensión del significado de DevOps (iii), poca colaboración entre desarrollo y operaciones, y (iv) resistencia a cambiar la cultura organizacional de la empresa. El detalle de los beneficios y desafíos puede ser consultado en Guerrero et al. (2020).

Soluciones metodológicas

La falta de una terminología común que estandarice e institucionalice el conocimiento en torno a DevOps trajo consigo la definición de soluciones que buscan apoyar el uso de DevOps desde diferentes perspectivas como: (i) *proceso*, mediante la caracterización de soluciones que describen los lineamientos o pasos a seguir para guiar la adopción de DevOps, (ii) *marco de trabajo*, a través de estudios que documentan la experiencia descrita tras aplicar DevOps en estudios de caso específicos y (iii) *prácticas*, a través de estudios que buscan identificar, caracterizar y clasificar el conjunto de prácticas presentes en el ciclo de vida de la adopción de DevOps. Se observó que los estudios relacionados se enfocan en prácticas muy específicas como integración, despliegue y entrega continua. El detalle de los estudios identificados por cada categoría puede ser consultado en Guerrero et al., (2020).

Herramientas

En la actualidad, la industria de software busca establecer estrategias que permitan potenciar la adopción de DevOps a través del uso de herramientas tecnológicas (Orozco-Garcés et al., 2022). En particular, el uso de DevOps involucra de manera directa el uso de herramientas tecnológicas que tienen como objetivo automatizar aspectos relacionados con integración, entrega y despliegue continuo. De acuerdo con el análisis de los resultados del mapeo sistemático, se identificó un total de 60 herramientas tecnológicas que apoyan los procesos involucrados en la adopción de DevOps. Se observó que las herramientas más utilizadas son: (i) Jenkins (reportado en el 70 % de los estudios), (ii) Puppet (reportado en el 50 % de los estudios), (iii) Docker (reportado en el 40 % de los estudios) y (iv) Github (reportado en el 40 % de los estudios). El detalle de todas las herramientas identificadas y área/proceso puede ser consultado en <https://bit.ly/3vssZ5o>.

Brechas existentes

A partir del análisis del estado del arte, se observó que: (i) DevOps trae grandes beneficios a las empresas en términos de esfuerzo, alcance, costo y tiempo (Erich et al., 2014; Chen, 2017; Jones et al., 2016; Lwakatare et al., 2019), sin embargo, (ii) no existe un consenso que permita identificar de manera clara *qué es DevOps* (Guerrero et al., 2020) además, (iii) existe heterogeneidad en la naturaleza de las soluciones propuestas para apoyar el uso de DevOps en empresas de software (Guerrero et al., 2020). Como resultado, se ha identificado una fuerte tendencia hacia la definición de soluciones que no están sustentadas en estándares o marcos de referencia que puedan ser adoptados de manera transversal por las empresas, resultando en soluciones parciales a través de elementos de proceso que no están claramente definidos, por lo cual, prácticas, actividades, tareas, y roles presentes en los estudios son descritos de manera incompleta, difusa y, en la mayoría de los casos, ambigua.

Resultados

Después de llevar a cabo el análisis de los estudios relacionados, fue posible observar que se han propuesto diferentes aproximaciones para adoptar DevOps. Sin embargo, actividades, prácticas, roles y tareas propuestos en la literatura cuentan con un alto grado de heterogeneidad (Guerrero *et al.*, 2020), por lo cual, se realizó un proceso de armonización que permitió identificar de manera ordenada, estructurada y clara el conjunto de elementos de proceso que se deben considerar para implementar DevOps en medianas y pequeñas empresas de software. El protocolo aplicado para la identificación de elementos de proceso de DevOps fue realizado a través de las actividades descritas en la [Tabla 1](#).

Como resultado del proceso de armonización se identificó un total de 14 procesos, 106 actividades, 16 roles y 60 herramientas. Adicionalmente, los elementos de proceso identificados en la literatura fueron categorizados de acuerdo con su impacto en las cuatro dimensiones propuestas en

[Bucena y Kirikova \(2017\)](#) relacionadas con: *cultura, personas, procesos y tecnología*. A partir de estas dimensiones, se obtuvo 60 herramientas en la dimensión de *tecnología*, 18 actividades y 7 procesos en la dimensión de *procesos*, 27 actividades en la dimensión de *cultura* y 12 actividades, 3 procesos y 13 roles en la dimensión de *personas*. Los elementos de proceso categorizados de acuerdo con las cuatro dimensiones pueden ser consultados en detalle a través del enlace: <https://bit.ly/3MwjCYi>.

Descripción del proceso sugerido para DevOps

A continuación, se presenta el proceso definido con el propósito de apoyar la adopción de DevOps en medianas y pequeñas empresas de software. El proceso armoniza e integra los elementos de proceso presentes en la literatura en una estructura común y homogénea. El proceso propone tres subprocesos: (i) gestión de la configuración (GC), (ii) integración, entrega y despliegue continuo (conocidos por sus siglas en inglés como CI/CD) y (iii) monitoreo continuo (MC). Además, el proceso describe un total

Tabla 1. Actividades llevadas a cabo para la armonización de elementos de proceso

Etapa	Actividades
Identificación	<p>1. Selección de estudios: en esta actividad se realizó el análisis del estado del arte que permitió identificar los estudios relacionados que proponen, describen o presentan elementos de proceso para DevOps.</p> <p>2. Clasificación de elementos de proceso: en esta actividad se clasificaron los elementos presentes en la literatura en una de las siguientes categorías: (i) proceso, (ii) actividad, (iii) tarea, (iv) producto, (v) rol, (vi) herramienta y (vii) otro. Las categorías utilizadas fueron aplicadas de acuerdo con las definiciones descritas en Pardo <i>et al.</i> (2012)</p> <p>3. Registro de información: en esta actividad se documentó la información más representativa de cada elemento de proceso, así como: nombre, descripción, categoría, observaciones y referencia.</p>
Armonización	<p>En esta actividad se llevó a cabo un proceso de armonización de múltiples modelos a través de la aplicación del proceso HPROCESS propuesto en Pardo Calvache (2012), el cual se llevó a cabo mediante las siguientes actividades:</p> <p>1. Identificación de la terminología común: esta actividad fue realizada durante el análisis del estado del arte y tuvo por objetivo identificar cada uno de los elementos de proceso propuestos por diferentes autores.</p> <p>2. Homogeneización de elementos de proceso: en esta actividad se caracterizó cada proceso, actividad, artefacto, rol y herramienta relacionados con los elementos de proceso presentes en la literatura.</p> <p>3. Comparación de elementos de proceso: en esta actividad se identificaron las diferencias terminológicas y conceptuales relacionadas con cada uno de los elementos de proceso obtenidos durante la actividad de homogeneización.</p> <p>4. Integración de elementos de proceso: en esta actividad se aplicó teoría de conjuntos con el propósito de determinar un conjunto de definiciones y descripciones comunes para identificar de manera unívoca cada uno de los elementos de proceso en una estructura común. Las actividades realizadas incluyen tareas de intersección, complemento y unión.</p>
Reporte	<p>1. Documentación de los resultados: en esta actividad se documentó cada uno de los elementos de proceso resultantes del proceso de armonización.</p>

de 82 actividades, 16 artefactos, 9 roles y recomendación 13 herramientas tecnológicas para apoyar la automatización de cada una de las actividades relacionadas con cada subproceso. La figura 1 presenta de forma resumida relación e interacción entre cada uno de los subprocesos que conforman el proceso. Cada uno de los subprocesos se describe de acuerdo con un conjunto de herramientas (He), artefactos (Ar), roles (R), actividades fundamentales (A_f) y actividades complementarias u opcionales (A_c), en resumen, se puede observar que el subproceso de gestión de la configuración (GC) sugiere: 14 actividades fundamentales (A_f), 15 actividades complementarias (A_c), 2 roles (R), 7 artefactos (Ar) y algunas herramientas tecnológicas para apoyar actividades de GC como Terraform, Chef, Ansible. Para el subproceso que apoya integración, entrega y despliegue continuo (CI/CD) se sugieren: 32 actividades fundamentales (A_f), 17 actividades complementarias (A_c), 5 roles (R), 6 artefactos (Ar) y algunas herramientas tecnológicas para apoyar actividades de CI/CD como GitHub, GitLab, SonarQube, TestNG, Selenium, Jenkins, Azure DevOps, Docker, Slack, Jira. Finalmente, para el subproceso de monitoreo continuo (MC) se sugieren: 4 actividades complementarias (A_c), 2 roles (R), 3 artefactos (Ar) y algunas herramientas tecnológicas para apoyar actividades de MC como Nagios, Zabbix, Zenoss. En las siguientes subsecciones se describe el proceso en términos de propósito, subprocesos, roles propuestos, artefactos, actividades y herramientas de apoyo. El proceso fue definido con el objetivo de apoyar la implementación de las prácticas de gestión de configuración, integración, despliegue y

monitoreo continuo en pequeñas y medianas empresas de software. En este sentido, el propósito del proceso es guiar a las empresas desarrolladoras de software en la adopción de DevOps mediante la definición de un conjunto claro de elementos de proceso, así como actividades, roles, artefactos y herramientas, todo esto a través de un proceso único, integrado y continuo, que habilite a la organización para responder de manera efectiva y ágil entregando valor y mejorando la calidad de sus productos. El proceso fue diseñado siguiendo los lineamientos de la guía para la definición de procesos en organizaciones desarrolladoras de software propuesta en [Quintana \(2017\)](#). El proceso se compone de tres subprocesos, los cuales se describen en la [Tabla 2](#). El flujo de cada una de las actividades relacionadas con el proceso se realizó utilizando la notación para el modelado de procesos de negocio BPMN ([Dijkman et al., 2011](#)) y apoyado en la herramienta Bizagi (s.f.). La descripción completa del proceso presenta de manera detallada los subprocesos de gestión de la configuración (GC), integración, entrega y despliegue continuo (CI/CD) y monitoreo continuo (MC), y puede ser consultado en la guía electrónica: <https://bit.ly/3L4p0Bf>.

Roles propuestos: en la [Tabla 3](#) se presenta el nombre y la descripción de los 9 roles que participan en la ejecución de las diferentes actividades del proceso.

Artefactos: el proceso sugiere un total de 16 artefactos (entrada/salida), la [Tabla 4](#) presenta el nombre de los artefactos sugeridos, la descripción puede ser consultada a través del enlace: <https://bit.ly/3LUhWYD>.

Tabla 2. Subprocesos

Subproceso	Acronimo	Descripción
Gestión de la configuración	GC	En este subproceso se establece la configuración que permite crear, configurar y desplegar una tubería de integración, entrega, despliegue y monitoreo continuo en un entorno de trabajo.
Integración, entrega y despliegue continuo	CI/CD	En este subproceso se recomienda cómo llevar a cabo la ejecución de la tubería de integración, entrega y despliegue continuo.
Monitoreo continuo	MC	En este subproceso se recomiendan las actividades para realizar seguimiento continuo al rendimiento de la aplicación software que se encuentra en producción.

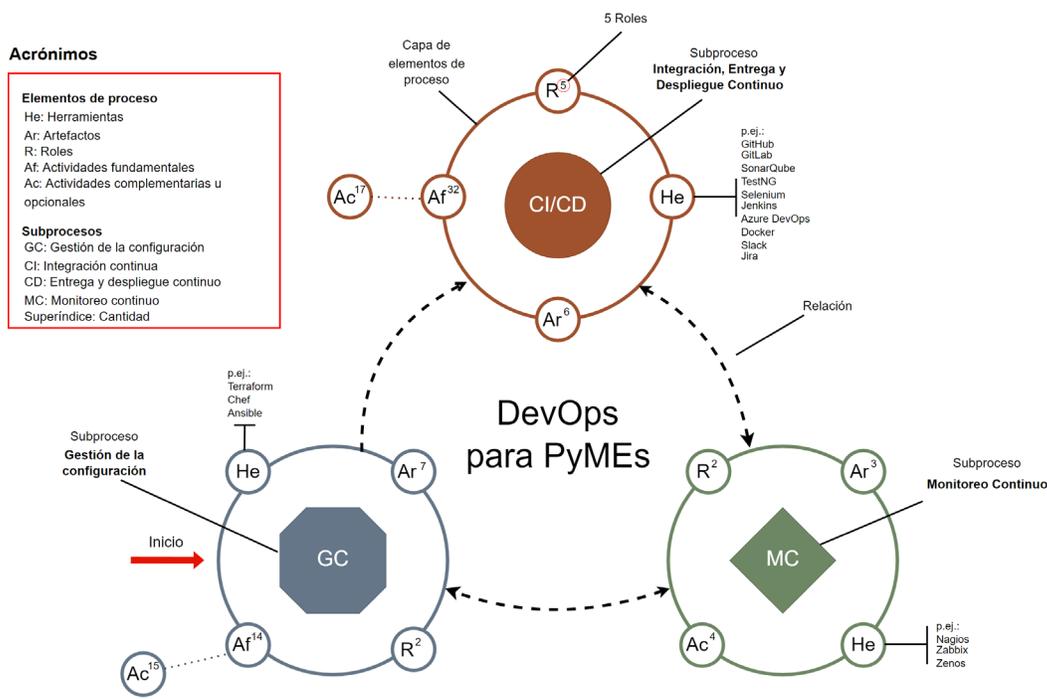


Figura 1. Proceso para apoyar la implementación de DevOps

Tabla 3. Roles propuestos

Rol	Responsabilidades
Administrador de sistemas	Implementa, configura, mantiene, monitoriza, documenta y asegura el correcto funcionamiento de un sistema informático o algún aspecto de este, por ejemplo, el rendimiento.
Arquitecto de software	Traduce los requisitos funcionales y no funcionales hacia una solución técnica. Identifica estilos arquitectónicos y tecnologías que sean apropiados para resolver un problema.
Coordinador de liberaciones	Autoriza y realiza los despliegues al entorno de producción.
Desarrollador de software	Convierte la especificación del sistema en código fuente ejecutable utilizando uno o más lenguajes de programación.
Facilitador	Lidera al equipo de trabajo en la gestión ágil del proyecto para que alcance sus objetivos hasta llegar a la etapa final de cada iteración. Es el moderador de las reuniones y establece soluciones en conjunto para eliminar los obstáculos que dificultan el desarrollo ágil del equipo.
Ingeniero de pruebas	Ejecuta pruebas en diversos componentes y funciones para detectar problemas técnicos y permite garantizar que los clientes obtengan productos funcionales de alta calidad.
Ingeniero DevOps	Guía a la empresa en la adopción de DevOps mediante prácticas y herramientas tecnológicas que soportan integración, entrega, despliegue y monitoreo continuo. Permite cerrar la brecha y mejora la coordinación y colaboración entre los roles que antes estaban aislados (desarrollo y operaciones de TI).
Líder de desarrollo	Lidera los procesos del equipo de desarrollo de software. En algunas EDS puede denominarse líder técnico, haciendo referencia al desarrollador de software más experimentado del equipo.
Usuario final	Software o persona que utiliza de manera directa la aplicación software del cliente.

Tabla 4. Artefactos de entrada y salida del proceso

No	Nombre del artefacto
1	Archivo de configuración de la imagen del contenedor
2	Archivo de configuración de la infraestructura
3	Archivo que empaqueta la aplicación software
4	Documento de requisitos de infraestructura
5	Imagen construida del contenedor
6	Lista de dispositivos a monitorear continuamente
7	Lista de herramientas CI/CD que se deben instalar y configurar
8	Lista de proveedores de la infraestructura
9	Lista de métricas para medir la calidad del código
10	Lista de servicios a monitorear continuamente
11	Reporte de resultados de la compilación del código
12	Reporte de resultados de las pruebas de integración
13	Reporte de resultados de las pruebas funcionales automatizadas
14	Reporte de resultados de las pruebas unitarias
15	Reporte de resultados del análisis de la calidad del código
16	Reporte de resultados negativos durante el monitoreo continuo

Actividades: como resultado de la caracterización, se obtuvo un total de 46 actividades fundamentales y 22 actividades complementarias. Las actividades fundamentales (A_f) garantizan que una empresa aplique los subprocesos de gestión de configuración, integración, entrega y despliegue continuo de manera adecuada. Por otro lado, las actividades complementarias (A_c) ayudan a potenciar el proceso de monitoreo continuo y establecer las políticas necesarias para la implementación de contenedores en los proyectos llevados a cabo por una empresa que aplica DevOps. El análisis del estado del arte y el proceso de armonización permitió identificar que si bien las actividades relacionadas con monitoreo continuo e implementación de contenedores ayudan a mejorar considerablemente el proceso de aplicación de DevOps, el uso de contenedores depende en gran medida de las características particulares de cada empresa, por lo cual, su aplicación se considera complementaria u opcional. Debido a limitaciones de espacio, en la [Tabla 5](#) se presentan las actividades propuestas para el subproceso de gestión de la configuración, asimismo, se presenta su clasificación como: actividad fundamental (A_f) o actividad complementaria (A_c). Las actividades correspondientes a

la dimensión *personas* están identificadas mediante la letra *P* seguida de un número, por ejemplo: P1 y P3. Por otra parte, las actividades correspondientes a la dimensión *tecnología* están identificadas mediante la letra *T* seguida de un número, por ejemplo: T1 y T3. El listado total de actividades se puede consultar en el enlace: <https://bit.ly/37Uids>. La guía de implementación de las actividades fundamentales y complementarias puede ser consultada en el enlace: <https://bit.ly/3L4p0Bf>.

El flujo de cada una de las actividades relacionadas con el proceso se realizó utilizando la notación para el modelado de procesos de negocio BPMN ([Dijkman et al., 2011](#)). En la [Figura 2](#) se presenta un extracto de aplicación del subproceso monitoreo continuo. El proceso completo puede ser consultado con mayor detalle a través del enlace: <https://bit.ly/3L4p0Bf>.

Herramientas de apoyo tecnológico: en la [Tabla 6](#) se presentan las herramientas tecnológicas que podrían apoyar la adopción del proceso. En total se identificaron 60 herramientas que soportan las diferentes actividades relacionadas con áreas y procesos presentes en el ciclo de vida de DevOps, entre los que se encuentran: gestión y control de código fuente, integración continua y orquestación, monitoreo, gestión de la configuración, planeación. El listado completo de herramientas y su relación con cada área o proceso se puede consultar en <https://bit.ly/3vssZ5o>.

Evaluación de la propuesta mediante grupo focal

El proceso fue sometido a una evaluación mediante un grupo focal, conformado por expertos de la academia y la industria que cuentan con experiencia en definición, adopción e implementación de DevOps. El procedimiento para llevar a cabo el grupo focal siguió los lineamientos definidos en [Kontio et al. \(2008\)](#) que sugiere cinco fases: (i) planeación de la investigación, (ii) reclutamiento, (iii) moderación de la sesión de debate, (iv) análisis y reporte de los resultados, (v) identificación de acciones de mejora.

Tabla 5. Actividades relacionadas con el proceso de gestión de la configuración

Nombre e identificador de la actividad	Grupo	
	A _i	A _c
P1. Elegir las herramientas para automatizar la tubería CI/CD	x	
P2. Adquirir los proveedores de infraestructura	x	
P3. Adquirir la herramienta de orquestación CI/CD	x	
P4. Adquirir la herramienta IaC		x
P5. Configurar la herramienta de orquestación CI/CD para que soporte a la herramienta IaC		x
P6. Crear archivo de configuración de infraestructura que soportará la tubería CI/CD		x
P7. Crear una tarea para leer el archivo de configuración de la infraestructura y ejecutarlo		x
P8. Ejecutar la tarea que lee y ejecuta el archivo de configuración de la infraestructura		x
T1. Clonar el repositorio donde se encuentran los archivos de configuración de la infraestructura		x
T2. Solicitar la ejecución del archivo de configuración de la infraestructura		x
T3. Conectarse con el proveedor de infraestructura		x
T4. Proveer la infraestructura		x
P9. Inicializar el repositorio del proyecto	x	
P10. Crear las ramas del repositorio del proyecto	x	
P11. Configurar la herramienta de orquestación /CD	x	
P12. Establecer la ruta del repositorio del proyecto y los permisos para acceder a él	x	
P13. Activar un disparador para detectar cambios en el repositorio cada cierto tiempo	x	
P14. Establecer el lenguaje de programación que será analizado	x	
P15. Establecer las métricas de calidad del código	x	
P16. Establecer los dispositivos a monitorear		x
P17. Establecer los servicios que se van a monitorear		x
P18. Indicar la ruta de las pruebas automatizadas a ejecutar	x	
P19. Crear un registro para almacenar las imágenes del software		x
P20. Crear el archivo de tareas para construir los contenedores		x
P21. Crear el clúster de contenedores		x
P22. Definir la configuración del despliegue		x
P23. Crear o programar las tareas que debe ejecutar cada herramienta de la tubería CI/CD	x	
P24. Configurar el medio de notificación de la ejecución de cada actividad de la tubería CI/CD	x	
P25. Gestionar el acceso y la seguridad de las herramientas DevOps	x	

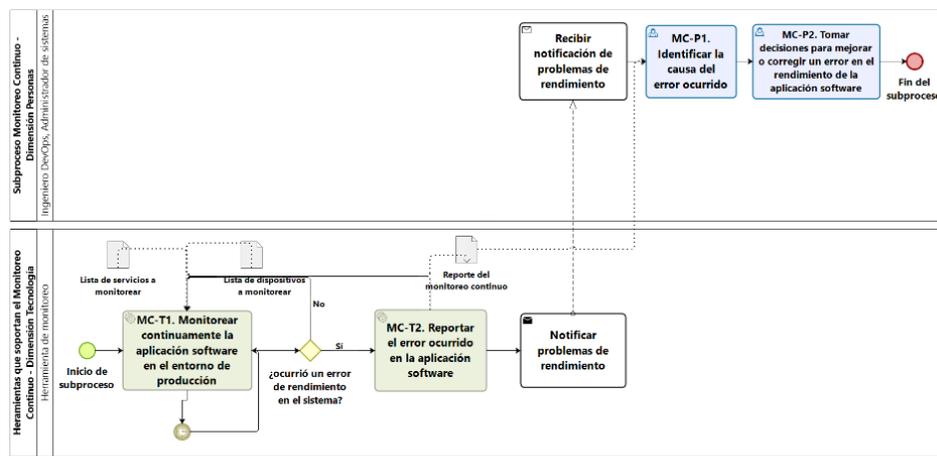


Figura 2. Extracto del subproceso de monitoreo continuo

Tabla 6. Herramientas tecnológicas identificadas

Herramienta	Definición
Automatización de pruebas	Permite codificar y ejecutar las pruebas que se realizan a la aplicación software, por ejemplo, pruebas unitarias, pruebas de integración, pruebas funcionales.
Compilación del código	Permite automatizar la compilación y el empaquetamiento del código para su posterior despliegue.
Contenerización	Permite virtualizar y encapsular entornos de trabajo a medida. Permite desplegar aplicaciones software utilizando contenedores.
Gestión de calidad del código	Permite evaluar la calidad del código programado.
Gestión de comunicación	Permite notificar a cada miembro del equipo de trabajo el resultado de la ejecución de cada una de las actividades de la tubería CI/CD.
Gestión del código	Permite gestionar las versiones del código fuente o los archivos de un proyecto.
Monitoreo	Permite monitorear de manera continua el funcionamiento de la aplicación software que se encuentra en producción.
Orquestación CI/CD	Permite crear y ejecutar las actividades que automatizan integración, entrega y despliegue continuo.
Orquestación de contenedores	Permite gestionar un grupo de contenedores. Además, permite realizar despliegues automatizados.
Herramienta IaC	Permite crear, configurar y mantener la infraestructura como código.

Planeación

En esta fase se definió el objetivo del grupo focal (OGF) y el objetivo de investigación (OI). Adicionalmente, se prepararon los materiales y procedimientos a seguir por parte del grupo investigador. El OGF estuvo orientado a conocer la opinión y la percepción de profesionales con experiencia en integración, entrega, despliegue y monitoreo continuo del desarrollo de software acerca del grado de aceptación o rechazo en los siguientes aspectos: (i) idoneidad, (ii) completitud, (iii) facilidad de comprensión y entendimiento, (iv) aplicabilidad en pequeñas y medianas EDS, esto para disminuir la subjetividad de la propuesta a partir de la opinión de expertos. El OI estuvo enfocado a evaluar el proceso de DevOps e identificar posibles oportunidades de mejora que permitieran generar una nueva versión del proceso si fuera necesario. Los materiales preparados fueron: agenda de trabajo cuestionario, estructura del protocolo y propuesta a evaluar.

Reclutamiento

En esta fase se definieron las estrategias de selección de los participantes del grupo focal. Asimismo, esta fase estuvo a cargo del grupo investigador

e incluyó la definición del perfil del participante: (i) estar activo en la industria software o en un entorno académico; (ii) tener conocimientos avanzados sobre enfoques ágiles y su aplicación en la industria o (iii) profesionales con mínimo dos años de experiencia en la industria de software realizando actividades de integración, entrega, despliegue y monitoreo continuo. Teniendo en cuenta los criterios, se realizó la invitación a 16 potenciales participantes, de los cuales aceptaron participar 8.

Moderación

La sesión de debate tuvo una duración de dos horas y fue coordinada por un moderador y un relator, quienes formaron parte del grupo investigador. El orden y la secuencia de la agenda de trabajo a desarrollar durante la sesión fueron enviados previamente por correo electrónico a los participantes. El relator fue el encargado de anotar percepciones, comentarios y sugerencias de los participantes, por otro lado, el moderador tuvo la responsabilidad de organizar la participación de los profesionales, así como de motivar su participación. En esta sesión también se pidió a los participantes diligenciar un cuestionario conformado por un total de 18 preguntas, 16 preguntas definidas mediante

una escala de evaluación discreta de cinco valores, la cual tuvo en cuenta los niveles de conformidad propuestos en la escala de [Likert \(1932\)](#) de la siguiente manera: (1 punto) muy insatisfecho(a), (2 puntos) poco satisfecho(a), (3 puntos) algo satisfecho(a), (4 puntos) satisfecho(a) y (5 puntos) muy satisfecho(a). Adicionalmente, se definieron dos preguntas abiertas que permitieron a los participantes proponer ajustes al proceso y realizar comentarios adicionales de acuerdo con sus observaciones. En la [Tabla 7](#) se presentan las preguntas realizadas, las preguntas 1 a 5 se enfocaron en

evaluar la idoneidad, las preguntas 6 a 8 la aplicabilidad de la propuesta en pequeñas y medianas empresas desarrolladoras de software, las preguntas 9 a 11 en la completitud, las preguntas 12 a 15 en la comprensibilidad y las preguntas 17 y 18 en todas las anteriores variables.

Análisis y reporte de los resultados

La [Figura 3](#) presenta de manera gráfica los resultados obtenidos para las preguntas discretas 1 a 16. De acuerdo con los resultados, fue posible

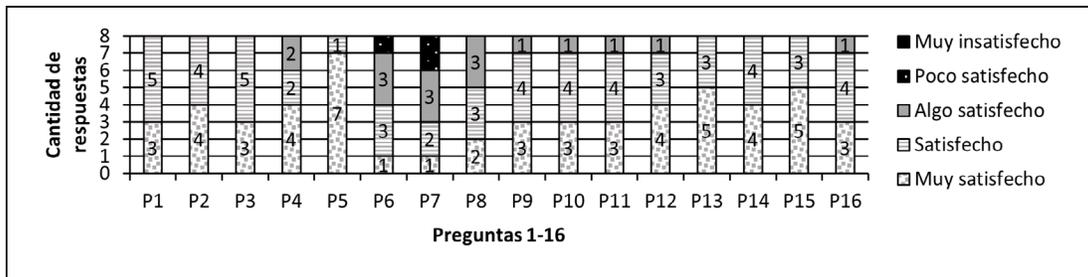


Figura 3. Respuesta a las preguntas 1-16

Tabla 7. Preguntas diligenciadas por los participantes

Id	Pregunta
Preguntas cerradas	
P1	¿Considera que los subprocesos permiten la implementación de un pipeline de DevOps en una EDS en relación con: (i) gestión de la configuración, (ii) integración, entrega y despliegue continuo y (iii) monitoreo continuo en el desarrollo de software?
P2	¿Considera que el subproceso <i>gestión de la configuración</i> cumple con el objetivo propuesto?
P3	¿Considera que el subproceso <i>integración, entrega y despliegue continuo</i> cumple con el objetivo propuesto?
P4	¿Considera que el subproceso <i>monitoreo continuo</i> cumple con el objetivo propuesto?
P5	¿Considera que las herramientas sugeridas apoyan la ejecución de las actividades descritas en los subprocesos propuestos?
P6	De acuerdo con su experiencia, ¿considera que los elementos definidos (actividades, roles, artefactos y herramientas tecnológicas) son apropiados y pueden aplicarse con éxito en un proyecto de desarrollo de software de una pequeña o mediana EDS?
P7	¿Considera que los roles propuestos pueden ser asumidos por los recursos humanos disponibles en una pequeña EDS?
P8	¿Considera que los roles propuestos pueden ser asumidos por los recursos humanos disponibles en una mediana EDS?
P9	¿Considera que los elementos definidos en el subproceso <i>gestión de la configuración</i> son suficientes?
P10	¿Considera que los elementos definidos en el subproceso <i>integración, entrega y despliegue continuo</i> son suficientes?
P11	¿Considera que los elementos definidos en el subproceso <i>monitoreo continuo</i> son suficientes?
P12	¿Considera que la plantilla utilizada para organizar y describir los subprocesos es de fácil comprensión?
P13	¿Considera que el subproceso <i>gestión de la configuración</i> y los elementos que lo describen son de fácil comprensión?
P14	¿Considera que el subproceso <i>integración, entrega y despliegue continuo</i> y los elementos que lo describen (actividades, roles y artefactos) son de fácil comprensión?
P15	¿Considera que el subproceso <i>monitoreo continuo</i> y los elementos que lo describen son de fácil comprensión?
P16	¿Encuentra útiles los diagramas de BPMN presentados para facilitar la comprensión de los subprocesos propuestos?
Preguntas abiertas	
P17	¿Considera que se deben agregar, eliminar o modificar elementos (actividades, roles, artefactos) de la propuesta?
P18	¿Tiene algún comentario adicional acerca de los subprocesos propuestos?

observar que los participantes consideran que los subprocesos propuestos para la caracterización del proceso cumplen con su objetivo de manera adecuada, indicando que cada uno de los subprocesos definen un conjunto claro de roles, artefactos, actividades, herramientas y tareas a través de la aplicación del conjunto de buenas prácticas propuestas por el modelo BPMN, por lo cual, el proceso cumple con los criterios de idoneidad, comprensibilidad y completitud. Por otro lado, fue posible identificar aspectos de mejora relacionados con la aplicabilidad de la propuesta en pequeñas y medianas empresas de software. De acuerdo con la percepción de los participantes, la primera versión del proceso requería de una cantidad de roles mayor a la que podría asumir una empresa con limitaciones de infraestructura y talento humano. Además, el análisis de las respuestas a las preguntas abiertas permitió conocer comentarios, observaciones, sugerencias y percepciones de cada uno de los participantes. Como resultado, fue posible identificar acciones de mejora en relación con la forma en la cual se definieron los flujos de trabajo para cada uno de los subprocesos. Finalmente, gracias a las observaciones y sugerencias de los participantes, se obtuvo una versión refinada y más liviana del proceso, el cual comprende solo el conjunto de elementos necesarios y suficientes que pueda aplicar una empresa mediana o pequeña sin incurrir en sobreesfuerzos. El detalle de las acciones de mejora se puede consultar en el enlace <https://bit.ly/3MIF7oX>.

Limitaciones

Durante el grupo focal se presentaron algunas limitaciones y las soluciones sugeridas fueron: (i) aunque todos los invitados tenían experiencia aplicando DevOps, no todos tenían el mismo nivel de conocimiento sobre la temática, por lo tanto, para evitar tener poca participación durante la sesión, se envió a cada participante el material de lectura dos semanas antes; (ii) para realizar el grupo focal se requería mínimo la participación de

seis personas, por lo tanto, se decidió invitar a un total de 16 personas para mitigar el riesgo de cantidad de participación mínima; (iii) si bien existía un formato y una agenda predefinidos, por falta de experiencia no era tan fácil para el moderador tener control sobre el estilo de discusión de los participantes menos activos, esto fue corregido por el investigador más experimentado tan pronto como fue detectado. Adicionalmente, el moderador invitaba a cada uno de los participantes a intervenir durante la sesión. Con esto se logró obtener la participación y retroalimentación de todos los participantes del grupo focal; (iv) algunos comentarios que realizaban los participantes hacían referencia a temas que estaban por fuera del alcance del proyecto de investigación, por lo que cada uno de ellos fue aclarado rápidamente, lo cual permitió continuar con la discusión de la propuesta sin perder de vista el objetivo principal del grupo focal; y (v) el grupo focal fue realizado en época de pandemia, por lo tanto, la sesión se realizó de manera virtual, se solicitó el permiso de los asistentes para grabar la sesión.

Conclusiones y trabajo futuro

En este artículo se presentó un proceso que busca apoyar la adopción de DevOps en pequeñas y medianas empresas de software, a través de la definición de un conjunto claro de roles, actividades y artefactos en una estructura formal y unificada. Como resultado, el proceso sugiere un total de 46 actividades fundamentales enfocadas en apoyar el ciclo de vida de las prácticas de gestión de configuración, integración, entrega, despliegue y monitoreo continuo en medianas y pequeñas empresas de software. Además, propone un total de 22 actividades complementarias enfocadas en apoyar y facilitar la adopción de contenedores para facilitar el despliegue de aplicaciones software.

El proceso fue el resultado de armonizar los resultados obtenidos tras aplicar un mapeo sistemático de la literatura, el cual permitió conocer,

entender, caracterizar y clasificar los elementos de proceso de DevOps sugeridos por la academia y la industria. La armonización de los elementos identificados en la literatura permitió clasificar, categorizar y definir los elementos necesarios para la definición del proceso propuesto a través de una estructura clara, genérica, homogénea y sin conflictos terminológicos. El proceso propuesto sugiere elementos que permiten apoyar la gestión de configuración, integración, entrega, despliegue y monitoreo continuo en las pequeñas y medianas empresas de software. Por otro lado, la caracterización del proceso permitió identificar que si bien muchos de los beneficios de aplicar DevOps se basan en el uso de herramientas tecnológicas para automatizar tareas relacionadas con las prácticas abordadas por el proceso propuesto, también se requiere que los miembros de los equipos de trabajo cuenten con ciertas habilidades blandas (*soft skills*) que permitan a los equipos ser colaborativos, comunicativos, cooperativos, con alta capacidad para experimentar, aprender continuamente y estar abiertos a los cambios en la cultura organizacional.

En relación con la evaluación de la propuesta, el grupo focal permitió comprobar que el proceso: (i) tiene una estructura consistente y permite identificar los elementos que forman parte de la adopción de la gestión de configuración, integración, entrega, despliegue y monitoreo continuo, (ii) puede ser implementado tanto por una pequeña como mediana empresa de software, (iii) permite la implementación de actividades relacionadas con la gestión de configuración, integración, entrega, despliegue y monitoreo continuo, y (iv) se presenta de una manera que facilita su comprensión por parte del lector. Además, a partir del grupo focal se pudo evidenciar que las guías electrónicas BPMN permitieron facilitar a los lectores la comprensión del proceso. El uso de un estándar para la representación gráfica del proceso permitió reducir la ambigüedad en la descripción de los posibles escenarios, caminos o flujos que conlleva la adopción del proceso. Finalmente, el proceso de

investigación permitió identificar diferentes brechas de trabajo futuro, las cuales se presentan a continuación: (i) es necesario actualizar el estado del conocimiento en torno a DevOps mediante la aplicación de nuevos mapeos sistemáticos de la literatura, (ii) inclusión de nuevas actividades que involucren otras áreas presentes en DevOps como la ingeniería de requisitos, la experimentación y aprendizaje continuo, y (iii) llevar a cabo estudios de caso aplicados en empresas activas en la industria para validar el proceso mediante una evaluación práctica.

Referencias

- Beck, K. (2002). *Test-Driven Development: By Example*. Addison-Wesley Professional.
- Beck, K., Andres, C. (2000). *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional.
- Benguria, G., Alonso, J., Etxaniz, I., Orue-Echevarria, L., Escalante, M. (2018). Agile development and operation of complex systems in multi-technology and multi-company environments: Following a DevOps approach. En: X. Larrucea, I. Santamaria, R. O'Connor, R. Messnarz (Eds.), *Systems, Software and Services Process Improvement* (pp. 15-27). Springer. https://doi.org/10.1007/978-3-319-97925-0_2
- Bizagi. (s.f.). Bizagi. <https://www.bizagi.com/es>
- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), 61-72.
- Bou Ghantous, G., Gill, A. (2017). DevOps: Concepts, practices, tools, benefits and challenges. *PACIS 2017 Proceedings*, 96.
- Bucena, I., Kirikova, M. (2017). Simplifying the DevOps Adoption Process. *BIR Workshops*, 1-15.
- Chen, L. (2017). Continuous delivery: overcoming adoption challenges. *Journal of Systems and Software*, 128, 72-86. <https://doi.org/10.1016/j.jss.2017.02.013>
- CMMI Institute. (2018). *Capability Maturity Model Integration for Development*.
- Cockburn, A. (2004). *Crystal Clear: A Human-Powered Methodology for Small Teams*. Pearson Education.

- Conradi, H., Fuggetta, A. (2002). Improving software process improvement. *IEEE Software*, 19(4), 92-99. <https://doi.org/10.1109/MS.2002.1020295>
- de França, B., Jeronimo, H., Travassos, G. (2016). Characterizing DevOps by hearing multiple voices. En: *Proceedings of the 30th Brazilian Symposium on Software Engineering*, 53-62. <https://doi.org/10.1145/2973839.2973845>
- Debois, P. (2009). *Devopsdays - Organizing Guide*.
- Diel, E., Marczak, S., Cruzes, D. S. (2016). Communication challenges and strategies in distributed DevOps. En: *IEEE 11th International Conference on Global Software Engineering (ICGSE)*, 24-28. <https://doi.org/10.1109/ICGSE.2016.28>
- Dijkman, R., Hofstetter, J., Koehler, J. (2011). *Business Process Model and Notation*, LNBP, 89. Springer.
- Elberzhager, F., Arif, T., Naab, M., Süß, I., Koban, S. (2017). From agile development to DevOps: Going towards faster releases at high quality - Experiences from an industrial context. D. Winkler, S. Biffel, J. Bergsmann (Eds.), *Software Quality. Complexity and Challenges of Software Engineering in Emerging Technologies* (pp. 33-44). Springer. https://doi.org/10.1007/978-3-319-49421-0_3
- Erich, F., Amrit, C., Daneva, M. (2014). *Report: DevOps literature review*. <https://doi.org/10.13140/2.1.5125.1201>
- Galván-Cruz, S., Muñoz, M., Mejía, J., Laporte, C. Y., Negrete, M. (2020). Building a guideline to reinforce agile software development with the basic profile of ISO/IEC 29110 in very small entities. En: *International Conference on Software Process Improvement*, 20-37. https://doi.org/10.1007/978-3-030-63329-5_2
- Guerrero, J., Zúñiga, K., Certuche, C., Pardo, C. (2020). A systematic mapping study about DevOps. *Journal de Ciencia e Ingeniería*, 12(1), 48-62. <https://doi.org/10.46571/JCI.2020.1.5>
- Gupta, V., Kapur, P. K., Kumar, D. (2017). Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling. *Information and Software Technology*, 92, 75-91. <https://doi.org/10.1016/j.infsof.2017.07.010>
- Hemon, A., Lyonnet, B., Rowe, F., Fitzgerald, B. (2020). From agile to DevOps: Smart skills and collaborations. *Information Systems Frontiers*, 22(4), 927-945. <https://doi.org/10.1007/s10796-019-09905-1>
- Highsmith, J. (2013). *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. Addison-Wesley.
- Hochstein, A., Zarnekow, R., Brenner, W. (2005). ITIL as common practice reference model for IT service management: Formal assessment and implications for practice. En: *IEEE International Conference on e-Technology, e-Commerce and e-Service*, 704-710.
- ISO/IEC. (2013). *Sistemas de gestión la seguridad de la información*.
- ISO/IEC. (2019). *Calidad de los servicios TI*.
- Jabbari, R., Bin Ali, N., Petersen, K., Tanveer, B. (2016). What is DevOps?: A systematic mapping study on definitions and practices. En: *Proceedings of the Scientific Workshop Proceedings of XP2016*, 1-11. <https://doi.org/10.1145/2962695.2962707>
- Jones, S., Noppen, J., Lettice, F. (2016). Management challenges for DevOps adoption within UK SMEs. *Proceedings of the 2nd International Workshop on Quality-Aware Devops*, 7-11. <https://doi.org/10.1145/2945408.2945410>
- Kontio, J., Bragge, J., Lehtola, L. (2008). The focus group method as an empirical tool in software engineering. In: F. Shull, J. Singer, D. Sjøberg (Eds.), *Guide to Advanced Empirical Software Engineering* (pp. 93-116). Springer. https://doi.org/10.1007/978-1-84800-044-5_4
- Krusche, S., Alperowitz, L., Bruegge, B., Wagner, M. O. (2014). Rugby: an agile process model based on continuous delivery. *Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering*, 42-50. <https://doi.org/10.1145/2593812.2593818>
- Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, 22(140).
- Lwakatare, L., Kuvaja, P., Oivo, M. (2015). Dimensions of DevOps. In: C. Lassenius, T. Dingsøyr, M. Paasi-vaara (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 212-217). Springer. https://doi.org/10.1007/978-3-319-18612-2_19

- Lwakatare, L. E., Kilamo, T., Karvonen, T., Sauvola, T., Heikkilä, V., Itkonen, J., Kuvaja, P., Mikkonen, T., Oivo, M., Lassenius, C. (2019). DevOps in practice: A multiple case study of five companies. *Information and Software Technology*, 114, 217-230. <https://doi.org/10.1016/j.infsof.2019.06.010>
- Martin, J. (1991). *Rapid Application Development*. Macmillan Publishing Company.
- Michelsen, J. (2014). *Dysfunction Junction: A Pragmatic Guide to Getting Started with DevOps*.
- Nagarajan, A. D., Overbeek, S. J. (2018). A DevOps implementation framework for large agile-based financial organizations. In: H. Panetto, C. Debruyne, H. Proper, C. Ardagna, D. Roman, R. Meersman (Eds.), *On the Move to Meaningful Internet Systems. OTM 2018 Conferences* (pp. 172-188). Springer. https://doi.org/10.1007/978-3-030-02610-3_10
- Orozco, C., Pardo, C., Vásquez, S., Ordoñez, H., Suescún, E. (2020). An agile process to support software configuration management. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informação*, 2020(E32).
- Orozco-Garcés, C.-E., Pardo-Calvache, C.-J., Salazar-Mondragón, Y.-H. (2022). What is there about DevOps assessment? A systematic mapping. *Revista Facultad de Ingeniería*, 31(59), e13896. <https://doi.org/10.19053/01211129.v31.n59.2022.13896>
- Pardo, C., García, F., Pino, F. J., Piattini, M., Baldassarre, M. T. (2012). PrMO: An ontology of process-reference models. En: *Jornadas SISTEDES'2012*, 393-406.
- Pardo Calvache, C. J. (2012). *A Framework to Support the Harmonization between Multiple Models and Standards* [Tesis Doctoral]. Universidad de Castilla - La Mancha, España.
- Poppendieck, M., Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley Professional.
- Quintana, D. (2017). *Método para definir procesos en organizaciones desarrolladoras de software*. Universidad del Cauca.
- Rational Software. (2020). Best Practices for Software Development Teams. *Rational Unified Process*.
- Riungu, L., Mäkinen, S., Lwakatare, L., Tiihonen, J., Männistö, T. (2016). DevOps adoption benefits and challenges in practice: A case study. En: P. Abrahamsson et al. (Eds.), *Product-Focused Software Process Improvement* (pp. 590-597). Springer. https://doi.org/10.1007/978-3-319-49094-6_44
- Royce, W. W. (1987). Managing the development of large software systems. En: *Proceedings of the 9th International Conference on Software Engineering*, 328-338.
- Samarawickrama, S. S., Perera, I. (2017). Continuous scrum: A framework to enhance scrum with DevOps. En: *Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, 1-7. <https://doi.org/10.1109/ICTER.2017.8257808>
- Schwaber, K., Sutherland, J. (2017). The definitive guide to Scrum: The rules of the game. <https://www.scrum.org>
- Shahin, M., Babar, M. A., Zhu, L. (2017). Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices. *IEEE Access*, 5, 3909-3943. <https://doi.org/10.1109/ACCESS.2017.2685629>
- Stapleton, J. (1997). *DSDM, Dynamic Systems Development Method: The Method in Practice*. Cambridge University Press.
- Virmani, M. (2015). Understanding DevOps & bridging the gap from continuous integration to continuous delivery. En: *Fifth International Conference on the Innovative Computing Technology (INTECH 2015)*, 78-82.
- Young, J., Ridley, G., Carroll, P. (2004). COBIT and its utilization: A framework from the literature. En: *Proceedings of the 37th Annual Hawaii International Conference on System Sciences, 2004*.

