

# **DevOps Model - Modelo de referencia para la adopción de DevOps en empresas de desarrollo de software**



**Jonathan Guerrero Astaiza**

Director:

**PhD. MSc. César Jesús Pardo Calvache**

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**

**Departamento de Sistemas**

**Grupo de I+D en Tecnologías de la información (GTI)**

Popayán, Diciembre de 2021

**JONATHAN GUERRERO ASTAIZA**

**DevOps Model - Modelo de referencia para la  
adopción de DevOps en empresas de  
desarrollo de software**

Tesis presentada a la Facultad de Ingeniería  
Electrónica y Telecomunicaciones de la  
Universidad del Cauca para la obtención del  
Título de

Magister en  
Computación

Director:

**PhD. MSc. César Jesús Pardo Calvache**

Popayán  
2021

# Nota de Aceptación

---

---

---

---

Presidente del jurado

---

Jurado

# Agradecimientos

*Gracias a Dios por brindarme su apoyo y compañía durante cada etapa mi vida.*

*Gracias a mi madre y tía quienes me apoyaron en la conclusión de mis estudios de ingeniería, gracias a los cuales me encuentro transitando los caminos actuales de mi profesión.*

*Gracias a mi esposa y mis hijas que me apoyan todo el tiempo en el alcance de mis objetivos.*

*Gracias al doctor César Jesús Pardo Calvache por su dirección, por su paciencia, por sembrar en mi una idea y ayudarme a durante todo el proceso para transformarla en una realidad.*

*Gracias a la doctora Elizabeth Suescun Monsalve por brindarme su experiencia, sus comentarios y aportes para mejorar el proceso de investigación durante y después de mi estancia de investigación en la universidad EAFIT.*

*Gracias a todas las personas que de una u otra forma pusieron su granito de arena en este proyecto.*

Jonathan Guerrero Astaiza  
Popayán, octubre 2021

# Resumen Estructurado

**Antecedentes:** DevOps ha surgido como una palabra de uso común en la industria del software de la cual existen muchas interpretaciones e implementaciones, sin embargo, no existe a la fecha una estandarización que le permita a la industria tener una guía base sobre la cual iniciar procesos de adopción de DevOps y disfrutar de los beneficios que si se encuentran muy bien documentados en la literatura existente.

**Objetivo:** Proponer un modelo de referencia que formalice un conjunto de elementos de proceso para la implementación de DevOps en las empresas de desarrollo de software mediante su identificación a través del análisis de la literatura existente.

**Métodos:** Se utilizaron los siguientes métodos de investigación: (i) el método de investigación- acción, (ii) el grupo focal, y (iii) el estudio de caso.

**Resultados:** (i) un mapeo sistemático de la literatura relacionado con la implementación de DevOps en empresas de desarrollo de software, (ii) una ontología de dominio (DevOps Ontology) que permite organizar el conocimiento relacionado con la adopción de DevOps en procesos de desarrollo de software de una manera genérica y formal, (iii) un modelo de referencia (DevOps Model), (iv) un método de valoración (DevOps Assessment Model), (v) una monografía con los resultados de la investigación, (vi) dos artículos publicados en revistas indexadas, (vii) una publicación en el evento internacional XIV jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento donde se obtuvo el reconocimiento al mejor paper del evento y (viii) la construcción de dos artículos adicionales, uno de los cuales describe la ontología desarrollada con una comparación con otras ontologías existentes y el otro artículo contiene los resultados del estudio de caso del modelo de referencia y valoración aplicado en una empresa de desarrollo de software.

**Conclusiones:** La solución propuesta puede ser utilizada como un referente para la identificación e incorporación de practicas en procesos de adopción de DevOps la cual evaluada mediante un grupo focal en el cual participaron expertos en DevOps. Esta propuesta adicionalmente soporta la valoración de estos procesos de adopción y fue usada con éxito mediante un estudio de caso en una organización desarrolladora de software. Como trabajo futuro se plantea la actualización del mapeo sistemático, agregar mas detalle al método de valoración de forma que permita la automatización del proceso de evaluación.

**Palabras clave:** Desarrollo de software, DevOps, Ontología, Modelo de referencia.

# Structured Abstract

**Background:** DevOps has emerged as a common use word in the software industry of which there are many interpretations and implementations, however, there is no standardization to date that allows the industry to have a base guide on which to start DevOps processes adoption and enjoy the benefits that are very well documented in the existing literature.

**Aim:** Propose a model reference that formalizes a set of process elements for the DevOps implementation in software development companies by identifying them through the analysis of existing literature.

**Methods:** The following research methods were used: (i) the research-action method, (ii) the focus group, and (iii) the case study.

**Results:** (i) a systematic mapping of the literature related to the DevOps implementation in software development companies, (ii) a domain ontology (DevOps Ontology) that allows organizing the knowledge related to the DevOps adoption in software development processes in a generic and formal way, (iii) a reference model (DevOps Model), (iv) an assessment method (DevOps Assessment Model), (v) a monograph with the research results, (vi) two published articles in indexed journals, (vii) a publication in the international event XIV Ibero-American Conference on Software Engineering and Knowledge Engineering where recognition was obtained for the best paper of the event and (viii) two additional articles, one of them describe the developed ontology with a comparison with other existing ontologies, and the other article shows the results of the case study of the reference and assessment model applied on an software development enterprise.

**Conclusions:** The proposed solution can be used as a reference for the identification and incorporation of practices in DevOps adoption processes, which is evaluated through a focus group in which DevOps experts participated. This proposal additionally supports the assessment of these adoption processes and was used successfully through a case study in a software development organization. As future work, the updating of the systematic mapping is proposed, adding more detail to the evaluation method in a way that allows the automation of the evaluation process.

**Keywords:** Software development, DevOps, Ontology, Reference model.

# Contenido

Capítulo 1.	Introducción .....	14
1.1.	Definición del problema .....	14
1.2.	Objetivos.....	17
1.1.1.	Objetivo General – OG.....	17
1.1.2.	Objetivos específicos – OE.....	17
1.3.	Metodología .....	18
1.1.3.	Ciclo conceptual. Análisis conceptual.....	18
1.1.4.	Ciclo metodológico. Definición de la solución.....	18
1.1.5.	Ciclo de evaluación. Evaluación de la solución.....	19
1.1.6.	Ciclo de documentación y socialización .....	19
1.4.	Resultados obtenidos .....	20
1.5.	Organización del documento .....	20
Capítulo 2.	Marco teórico y estado del arte .....	22
2.1.	Marco teórico .....	22
2.1.1.	DevOps.....	22
2.1.2.	Proceso software .....	22
2.1.3.	Modelo de referencia .....	22
2.1.4.	Ontología .....	23
2.1.5.	Goal Question Metric – GQM.....	23
2.2.	Estado del arte.....	23
2.2.1.	Protocolo de investigación .....	23
2.2.2.	Resultados del mapeo sistemático de la literatura .....	29
2.2.3.	Discusión.....	32
Capítulo 3.	Caracterización de elementos de proceso relacionados a DevOps .....	34
3.1.	Proceso para la identificación de los elementos de proceso en los estudios primarios.....	34
3.2.	Elementos de proceso relacionados a DevOps .....	36
3.2.1.	Procesos identificados .....	36
3.2.2.	Actividades identificadas .....	37
3.2.3.	Artefactos identificados .....	40
3.2.4.	Roles identificados.....	40

3.2.5.	Herramientas identificadas .....	41
3.3.	Resultados obtenidos .....	42
Capítulo 4.	DevOps Ontology.....	43
4.1.	Metodología .....	43
4.2.	Propósito .....	43
4.3.	Preguntas de competencia.....	44
4.4.	Integración de DevOps Ontology con otras ontologías .....	44
4.5.	Representación gráfica.....	45
4.6.	Conceptos y relaciones de DevOps Ontology.....	45
4.7.	Discusión de algunos conceptos y sus relaciones .....	47
4.8.	Evaluación de DevOps Ontology .....	48
4.8.1.	Evaluación teórica de DevOps Ontology .....	48
4.8.2.	Soportando la instanciación de las prácticas de integración y despliegue continuo propuestas por GitLab.....	49
4.8.3.	Soportando la consulta a DevOps Ontology a través de una aplicación móvil	50
4.8.4.	Respondiendo las preguntas de competencia usando SPARQL.....	51
4.9.	Comparación de ontologías y taxonomías propuestas en la literatura .....	52
4.10.	Conclusiones y trabajo futuro .....	52
Capítulo 5.	Modelo de referencia .....	54
5.1.	Valores de DevOps.....	55
5.2.	Principios de DevOps.....	56
5.3.	Dimensiones .....	57
5.4.	Prácticas.....	57
5.4.1.	<i>Prácticas fundamentales</i> .....	58
5.4.2.	Prácticas complementarias .....	61
5.5.	Relación entre valores, principios, dimensiones y prácticas propuestas.....	62
5.6.	DevOps Assessment Model - Método de valoración de la adopción de DevOps en EDS	64
5.6.1.	Propósito del modelo de valoración .....	64
5.6.2.	Objetivos específicos del modelo de valoración .....	64
5.6.3.	Componentes del modelo de valoración .....	64
5.6.4.	Roles del proceso de valoración.....	70
5.6.5.	Proceso de valoración .....	70



Capítulo 6.	Evaluación de DevOps Model.....	71
6.1.	Evaluación de DevOps Model usando un grupo focal.....	71
6.1.1.	Planteamiento de la investigación .....	71
6.1.2.	Diseño del grupo de discusión (Reclutamiento).....	73
6.1.3.	Conducción de la sesión de debate .....	74
6.1.4.	Captura de información.....	74
6.1.5.	Análisis de la información y reporte de resultados.....	75
6.1.6.	Acciones de mejora .....	79
6.2.	Estudio de caso .....	80
6.2.1.	Diseño del estudio de caso .....	80
6.2.2.	Sujetos del estudio .....	81
6.2.3.	Procedimiento de campo y recolección de datos .....	81
6.2.4.	Análisis de resultados del estudio de caso .....	86
6.2.5.	Análisis de validez.....	89
6.2.6.	Limitaciones del estudio de caso.....	89
Capítulo 7.	Conclusiones y trabajo futuro .....	90
7.1.	Análisis de los objetivos de investigación.....	90
7.1.1.	Objetivos específicos – OE.....	90
7.1.2.	Objetivo general – OG .....	92
7.2.	Publicaciones .....	92
7.3.	Conclusiones.....	93
7.4.	Trabajo Futuro .....	94
Capítulo 8.	Bibliografía.....	96
Anexo A.	Mapeo sistemático.....	102
Anexo B.	Resultados cuestionario grupo focal.....	118
Anexo C.	Documentación estudio de caso .....	125
	Resultados generales.....	133
	Resultados individuales .....	136
	Análisis de resultados del estudio de caso .....	143

# Lista de figuras

Figura 3-1. Proceso para la identificación de los elementos de proceso en los estudios primarios.....	35
Figura 3-2. Subproceso registrar información.....	36
Figura 3-3. Subproceso armonizar elementos de proceso.....	36
Figura 4-1. Representación gráfica de DevOps Ontology usando UML. ....	45
Figura 4-2. Instanciación de la práctica de integración y despliegue continuo propuesto por GitLab usando DevOps Ontology.....	50
Figura 4-3. Visualización de elementos de DevOps Ontology usando una aplicación móvil. ....	50
Figura 5-1. Resumen de los elementos de la propuesta. ....	54
Figura 5-2. Elementos que conforman en DevOps Model. ....	55
Figura 5-3. Prácticas fundamentales propuestas. ....	58
Figura 5-4. Prácticas complementarias propuestas. ....	61
Figura 6-1. Consolidado de respuestas a preguntas P1-P13. ....	76
Figura 6-2. Grado de cumplimiento de principios de DevOps. ....	84
Figura 6-3. Grado de cumplimiento de dimensiones de DevOps.....	84

# Lista de tablas

Tabla 2-1. Preguntas de Investigación.....	24
Tabla 2-2. Cadena de búsqueda básica usada en el mapeo sistemático. ....	25
Tabla 2-3. Cuestionario y puntuación de los criterios de evaluación de calidad. ....	26
Tabla 2-4. Evaluación de los estudios de acuerdo con los criterios de evaluación de calidad. .....	26
Tabla 2-5. Esquema de clasificación. ....	27
Tabla 2-6. Contribuciones de los estudios primarios a cada una de las preguntas de investigación.....	27
Tabla 2-7. Resultados de búsqueda y análisis de estudios encontrados. ....	28
Tabla 2-8. Elementos de proceso encontrados por estudio. ....	30
Tabla 3-1. Actividades para la identificación de los elementos de proceso en los estudios primarios.....	34
Tabla 3-2. Procesos encontrados en los estudios primarios.....	37
Tabla 3-3. Actividades encontradas en los estudios primarios. ....	37
Tabla 3-4. Roles encontrados en los estudios primarios.....	41
Tabla 3-5. Herramientas tecnológicas que soportan áreas / procesos de DevOps. ....	42
Tabla 3-6. Cantidad de elementos de proceso encontrados por estudio primario. ....	42
Tabla 4-1. Preguntas de competencia definidas para DevOps Ontology.....	44
Tabla 4-2. Glosario de conceptos de DevOps Ontology.....	46
Tabla 4-3. Relaciones de DevOps Ontology.....	47
Tabla 4-4. Preguntas de competencia expresadas en SPARQL. ....	51
Tabla 4-5. Comparación de ontologías.....	52
Tabla 5-1. Principios propuestos en DevOps Model. ....	56
Tabla 5-2. Dimensiones propuestas en DevOps Model. ....	57
Tabla 5-3. Prácticas fundamentales propuestas en DevOps Model. ....	58
Tabla 5-4. Prácticas complementarias DevOps Model.....	61
Tabla 5-5. Relación entre valores y principios propuestos en DevOps Model. ....	63
Tabla 5-6. Relación entre principios y prácticas fundamentales propuestas en DevOps Model.....	63
Tabla 5-7. Relación entre dimensiones y prácticas propuestas en DevOps Model. ....	64
Tabla 5-8. Preguntas asociadas a las prácticas de DevOps. ....	65
Tabla 5-9. Métricas definidas en el método de valoración. ....	67
Tabla 5-10. Escala de calificación definida en el método de valoración.....	67
Tabla 5-11. Roles definidos en el método de valoración. ....	70
Tabla 6-1. Estructura del protocolo definido para el grupo focal.....	72
Tabla 6-2. Elementos para la realización del grupo focal.....	72
Tabla 6-3. Perfil profesional de los participantes del grupo focal.....	73
Tabla 6-4. Organización del grupo focal.....	74
Tabla 6-5. Escala de Likert usada en el cuestionario.....	74
Tabla 6-6. Cuestionario de evaluación usado en el grupo focal. ....	75
Tabla 6-7. Conteo de respuestas a preguntas P1-P13.....	75

Tabla 6-8. Respuestas a las preguntas abiertas. ....	79
Tabla 6-9. Acciones de mejora definidas para DevOps Model.....	79
Tabla 6-10. Pregunta de investigación estudio de caso. ....	80
Tabla 6-11. Preguntas específicas investigación estudio de caso. ....	80
Tabla 6-12. Tiempos invertidos en ejecución de la valoración. ....	82
Tabla 6-13. Resumen del grado de cumplimiento de prácticas fundamentales.....	83
Tabla 6-14. Grado de cumplimiento de valores de DevOps.....	85
Tabla 6-15. Oportunidades de mejora encontradas en el estudio de caso. ....	85
Tabla 6-16. Relación de esfuerzo del estudio de caso. ....	86
Tabla 6-17. Relación de costos del estudio de caso. ....	87
Tabla 6-18. Cuestionario de evaluación estudio de caso. ....	87

# Lista de ecuaciones

Ecuación 1. Métrica de cumplimiento de práctica fundamental. ....	67
Ecuación 2. Métrica de cumplimiento de principio de DevOps. ....	68
Ecuación 3. Métrica de cumplimiento de dimensión de DevOps. ....	68
Ecuación 4. Métrica de cumplimiento de valor de DevOps. ....	69
Ecuación 5. Métrica de cumplimiento implementación de DevOps.....	69

# Capítulo 1. Introducción

En este capítulo se describe el problema de investigación abordado, objetivos generales y específicos, la metodología utilizada y detalle de los resultados obtenidos.

## 1.1. Definición del problema

El ciclo de vida de un producto software involucra la intervención de dos áreas diferentes: (i) el área de desarrollo y (ii) el área de infraestructura, también llamada Tecnologías de la Información (TI) u operaciones. El área de desarrollo es la responsable del: análisis, diseño y construcción del código fuente que es probado y que permite generar posteriormente una versión de un producto software [1], por su parte, el área de operaciones es la encargada de la configuración del hardware que se requiere con los servicios necesarios, esto, para poder llevar a cabo el despliegue de las versiones del producto software, versiones que son generadas por el área de desarrollo a entornos productivos para su uso por parte del cliente final, y adicionalmente; hacer seguimiento al comportamiento del producto software durante el tiempo que está en uso [2], [3].

Actualmente, existe un amplio portafolio de marcos y enfoques de trabajo que guían el proceso de desarrollo de un producto software, estos marcos pueden ser tradicionales o enfoques ágiles. Los enfoques tradicionales buscan imponer disciplina al proceso de desarrollo de software y de esa forma volverlo predecible y eficiente. Para conseguirlo, se soportan en un proceso detallado con énfasis en planeación, por otra parte, los enfoques ágiles son adaptativos – no predictivos- y orientados a las personas y no los procesos, lo que marca una gran diferencia respecto a los marcos tradicionales [4].

Los marcos y enfoques más representativos según el número de citas y explicaciones en libros de ingeniería como: Sommerville [5], Pressman [6] y el reconocimiento de la alta adopción en la industria según la última versión de la encuesta anual del estado de la agilidad del año 2020 [7], indican que los enfoques ágiles que más se destacan son: Scrum [8], Extreme Programming (XP) [9], Crystal Clear [10], Lean Software Development (LSD) [11], Adaptive Software Development (ASD) [12], Dynamic Systems Development Method (DSDM) [13], Feature Driven Development (FDD) [14], Agile Unified Process (Agile UP) [15], Kanban [16], entre otros. Por otro lado, entre los enfoques tradicionales más populares se encuentran: Rational Unified Process (RUP) [17], Microsoft Solutions Framework (MSF) [18], Capability Maturity Model Integration (CMMI) [19], Iconix [20], Model-Based Architecture and Software Engineering (MBASE) [21], entre otros. Desde la perspectiva del área de infraestructura, igualmente encontramos marcos de trabajo de facto, así como: ITIL [22] y COBIT [23], y estándares internacionales como la ISO/IEC 2000 [24], donde se proponen elementos, prácticas y/o actividades relacionadas con la administración de servicios de TI y la administración de procesos de mejora continua en los servicios de TI.

Sin embargo, los marcos de trabajo antes mencionados definen sus elementos, prácticas y/o actividades detalladas desde una de las dos áreas involucradas en el proceso de construcción

de un producto software, es decir: desde el área de desarrollo o el área de infraestructura, lo que ha creado una especie de muro divisor entre estas dos áreas afectando de manera significativa el proceso de desarrollo de un producto software desde su concepción hasta su entrega final [25], esto, debido a que los marcos propuestos definen flujos de trabajo, roles y artefactos sólo desde una perspectiva o área en particular, que muchas veces es complementaria de los objetivos y/o necesidades de la otra área involucrada, por ejemplo: las áreas de desarrollo implementan prácticas para ser más ágiles y estar disponibles a cambios en los requerimientos en cualquier momento, mientras que el área de infraestructura implementa prácticas para controlar la estabilidad en los requerimientos, ya que estos afectan de forma directa los procesos relacionados con el alistamiento de la infraestructura tecnológica [3].

Teniendo en cuenta lo anterior, actualmente está tomando gran fuerza el enfoque conocido como DevOps, que como su nombre lo indica, se enfoca en mejorar las relaciones entre el área de desarrollo (Dev) y el área de operaciones (Ops) [26]. Con la implementación de DevOps en una organización de software se espera: (i) eliminar las barreras existentes entre las áreas de desarrollo y las áreas de infraestructura [27] (ii) mejorar significativamente los plazos de lanzamiento al mercado (time to market) [28], (iii) favorecer la mejora y automatización de procesos [29], [30], mejora de la calidad del producto [31], tiempos de respuesta optimizados [31], entre otros.

En el reporte de resultados de la encuesta mundial acerca del estado de la agilidad del año 2020 [7], se recopila, analiza y presentan las respuestas de 1.121 personas encuestadas e involucradas en una amplia gama de empresas en el desarrollo de software ubicadas en Europa, Asia, América, África y Oceanía, en este reporte se muestra que las organizaciones de software encuestadas reconocen la importancia y beneficios de la transformación que produce DevOps. Por otra parte, también se puede observar que el 76% de los encuestados informó que debido a las necesidades, beneficios y éxitos manifestados por otras empresas, actualmente tienen planificado o están llevando a cabo la implementación de procesos basados en DevOps. Sin embargo, el 90% también manifiestan la importancia y necesidad cada vez mayor de entender e implementar DevOps y de formalizar y clarificar las prácticas de integración y entrega continua.

Hoy en día, las organizaciones de software desean obtener una mayor visibilidad del flujo de valor que se entrega a los clientes, así como acelerar la velocidad de entrega que se obtendría con la correcta adopción de DevOps [31], sin embargo, también hay otros interrogantes y desafíos que deben ser resueltos, como el de entender: *dónde, cómo y cuándo* es el mejor momento para comenzar la adopción de DevOps [32], esto, debido a que en la actualidad no existe una definición estándar o de facto internacionalmente aceptada de DevOps, por lo que se confunde su significado, generando una ambigüedad en su comprensión y aplicación [29], [33], incluso, favoreciendo la omisión en la gestión de desafíos tales como: (i) la resistencia al cambio [30], [32], (ii) cambios culturales [34], [35], (iii) cambios en estructuras y estándares organizacionales [29], [36], (iv) cambios en los procesos [37], (v) adquisición de más responsabilidades [32], [28], entre otros desafíos, para los que las organizaciones no están preparadas, y además, los avances en este sentido son pocos [32], con lo cual, se generan contextos donde su aplicación se puede llevar a cabo de manera informal, inadecuada

o incompleta precisamente por la falta de un marco o modelo general para su implementación.

A partir de un mapeo sistemático de la literatura llevado a cabo [38], con el objetivo de conocer el estado del arte acerca de DevOps, ha sido posible identificar algunos trabajos relacionados que describen soluciones que proponen marcos de trabajo, lenguajes de dominio específicos y algunas prácticas con las que se puede tener una percepción parcial de DevOps. A continuación, se mencionan algunos de ellos: Continuous Scrum presenta un marco de trabajo que adiciona fases a SCRUM para complementar labores de configuración y mantenimiento soportadas por prácticas DevOps [39], sin embargo, en esta propuesta no hay claridad frente roles y responsabilidades para la realización de las prácticas que propone, por otro lado, está la propuesta de un modelo de madurez presentado en [35], el cual define 5 niveles para las dimensiones de tecnología, procesos, personas y cultura, sin embargo, no aborda elementos como el contexto, políticas y aprovisionamiento relacionados con la arquitectura, asimismo, no se ha evaluado su idoneidad y utilidad en empresas medianas o grandes de la industria del desarrollo de software, en [34] se presenta un marco conceptual para soportar el cambio cultural en la adopción de DevOps sin detallar en los procesos, la tecnología, la arquitectura y los roles necesarios para la realización de todas las prácticas propuestas, en [40], [41] se presenta Rugby, un marco de trabajo basado en el proceso racional unificado (RUP) y Scrum para la implementación de modelos de despliegue continuo, sin embargo, no aborda los temas relacionados con los desafíos asociados a la cultura organizacional y su evaluación se remite solo a prácticas académicas, por lo que le falta validez en entornos reales de la industria del software, en [42] se propone DevOpSlang, el cual es un lenguaje específico de dominio para automatizar acciones relacionadas con DevOps, esta propuesta se centra principalmente en la automatización de tareas de configuración, pruebas y despliegue solamente, otros trabajos como en [25], [27]; se describen únicamente las experiencias particulares y lecciones aprendidas en la adopción de DevOps.

Teniendo en cuenta lo anterior, y luego de realizar un análisis detallado de las propuestas encontradas, se evidencia la falta de clarificación y formalización de DevOps en el desarrollo, despliegue y mantenimiento de productos y servicios de software, por ejemplo: la necesidad de estandarizar una definición formal de DevOps y una descripción detallada de las prácticas fundamentales y opcionales que podrían tenerse en cuenta al momento de usar este enfoque. Por otra parte, no se definen roles claros o las responsabilidades para nuevas actividades que se proponen, tampoco se describe la manera de cómo los nuevos elementos de proceso se integran exitosamente con los eventos y artefactos existentes de Scrum u otros marcos de trabajo o enfoques. Asimismo, no queda claro la manera como se abordarán los conflictos o problemas relacionados con la cultura organizacional, procesos, personas y herramientas. En este sentido, se hace justificable y necesario trabajar en la definición de un modelo de referencia que cubra los atributos fundamentales de DevOps luego de su identificación, caracterización y clasificación de acuerdo a un orden de importancia, a partir del cual, se espera que las empresa de desarrollo de software pueda construir sus propios procesos de adopción de DevOps y a partir del cual en un futuro se puedan también definir modelos de evaluación de la adopción de DevOps en las organizaciones software.



En este sentido, surgió la siguiente pregunta de investigación: *¿Qué recomendaciones, atributos y/o elementos fundamentales deberían considerarse para facilitar la correcta implementación y formalización de DevOps en las empresas de desarrollo de software?* Como respuesta a este interrogante se han desarrollado algunas iniciativas que intentan dar respuesta en parte, sin embargo, hay varios vacíos por abordar dada la significancia e importancia del área y su aplicación en la industria de software. Es así como, el objetivo de esta propuesta de investigación se enfoca en solucionar la ambigüedad en este tema de investigación por medio del diseño de una ontología que permita clarificar y homogeneizar los conceptos relacionados. Asimismo, y con base en dicha ontología, se propone un modelo de referencia que formaliza un conjunto de atributos de proceso fundamentales para tener en cuenta en la correcta implementación de DevOps en las empresas de desarrollo de software. El modelo de referencia propuesto fue evaluado a través de un grupo focal que permitió reducir la subjetividad y mejorar su pertinencia, asimismo, se llevó a cabo su aplicación en una empresa de software como estudio de caso. Las soluciones obtenidas con el desarrollo de este proyecto permitirán a la industria de software disponer de herramientas que faciliten llevar a cabo la mejora de los procesos de las organizaciones en términos de la comprensión de los elementos fundamentales a tener en cuenta para la correcta adopción de DevOps.

## **1.2. Objetivos**

A continuación, se presentan los objetivos de esta tesis de maestría de acuerdo con cómo fueron aprobados por el consejo de Facultad de Ingeniería Electrónica y Telecomunicaciones - FIET de la Universidad del Cauca según la resolución número 8.4.3-90.2/250 de 2019.

### **1.1.1. Objetivo General – OG**

Proponer un modelo de referencia que formalice un conjunto de elementos de proceso que sean fundamentales para tener en cuenta para la implementación de DevOps en las empresas de desarrollo de software mediante su identificación a través del análisis de la literatura existente.

### **1.1.2. Objetivos específicos – OE**

- Identificar y clasificar los elementos a tener en cuenta para la adopción de DevOps por medio de su identificación a través de un estudio de mapeo sistemático de la literatura que permitan clarificar y formalizar la implementación de este enfoque en la industria de software.
- Definir una ontología de dominio específico que permita entender y organizar el conocimiento relacionado con los términos y relaciones alrededor de la implementación de DevOps en empresas de desarrollo de software.
- Definir un modelo de referencia que facilite la adopción e implementación de DevOps y que esté basado en los elementos identificados y/o definidos a través de la realización del objetivo específico anterior.
- Evaluar el modelo de referencia a través de su aplicación en un Focus Group para obtener realimentación por parte de profesionales expertos en el área.

- Realizar un estudio de caso que permita evaluar el modelo de referencia propuesto a través de la definición de un proceso de DevOps para una empresa de software de la región.

### 1.3. Metodología

Para llevar a cabo la ejecución del proyecto propuesto, se utilizó el método Investigación Acción con múltiples ciclos de forma lineal [43], grupo focal [44], [45] y estudio de caso [46]. Teniendo en cuenta las fases y actividades planteadas por la metodología Investigación Acción, para el desarrollo de esta propuesta se llevaron a cabo 4 ciclos de investigación, a continuación, se describen los ciclos y las actividades llevadas a cabo de manera secuencial e incremental para el desarrollo del proyecto.

#### 1.1.3.Ciclo conceptual. Análisis conceptual

En esta fase se llevó a cabo la revisión sistemática del estado del arte actual relacionado a la adopción de DevOps [47], en escenarios de desarrollo de software. Asimismo, la realización de esta fase permitió identificar y estudiar las propuestas y soluciones existentes y los elementos sensibles para tener en cuenta para la definición de la solución. En esta fase se realizaron las siguientes actividades:

- **Revisión de la literatura existente:** Se realizó un mapeo sistemático de la literatura existente sobre los diferentes estudios relacionados con la adopción de DevOps, en escenarios de desarrollo de software.
- **Síntesis la literatura seleccionada:** a partir del estudio de la literatura se realizó una caracterización de elementos de procesos asociados a DevOps donde se logra identificar diferentes procesos, actividades, artefactos, roles y herramientas, considerando los retos a los que se enfrenta la adopción de DevOps, las soluciones propuestas y las brechas existentes, para los escenarios de desarrollo de software.

#### 1.1.4.Ciclo metodológico. Definición de la solución

En esta fase se armonizan los diferentes elementos de proceso (roles, prácticas, tareas, fases, etc.) y estándares para la adopción de DevOps con base en los resultados obtenidos del ciclo anterior para construir cada componente del modelo de referencia para la adopción de DevOps. En esta fase se desarrollaron las siguientes actividades:

- **Definición de la ontología DevOps Ontology:** Se definió una ontología de dominio específico donde se describen los conceptos y relaciones alrededor de los elementos asociados a procesos de adopción de DevOps.
- **Construcción del modelo de referencia DevOps Model:** Se desarrolló un modelo de referencia que formaliza elementos para la correcta adopción de DevOps en organizaciones involucradas en el desarrollo de software a partir de los conceptos

identificados en la ontología y la caracterización de elementos de procesos desarrollados en fases anteriores.

- **Construcción del modelo de valoración DevOps Assessment Model:** Se desarrolló un modelo de valoración de adopción de DevOps con el cual las empresas podrán medir el porcentaje de cumplimiento de valores, principios, dimensiones y prácticas fundamentales de acuerdo con el modelo de referencia DevOps Model.

### 1.1.5.Ciclo de evaluación. Evaluación de la solución

La evaluación de la propuesta se llevó a cabo mediante el uso de un grupo focal, el cual estuvo conformado por expertos en DevOps y stakeholders y mediante la realización de un estudio de caso. En esta fase se llevaron a cabo las siguientes actividades:

- **Planificación del grupo focal:** Se llevó a cabo la capacitación, coordinación, organización y diseño del grupo focal.
- **Selección de la muestra:** Se llevó a cabo el proceso de selección de personas que participaran en el grupo focal, teniendo en cuenta su experiencia y conocimiento en el área de gestión de proyectos de software.
- **Ejecución del grupo focal:** Se ejecutó el grupo focal teniendo en cuenta la planificación y diseño planteado (ver Sección 6.1.2).
- **Observación y reflexión:** Se recogieron los datos sobre la ejecución e intervención del grupo focal y posteriormente se generó un reporte como resultado de la reflexión y el análisis de los datos obtenidos durante la ejecución del grupo focal, para obtener conclusiones que permitieran llevar a cabo la retroalimentación y evaluación del aprendizaje obtenido.
- **Planificación y selección empresa para estudio de caso:** Se llevó a cabo la coordinación, organización y diseño del estudio de caso teniendo en cuenta la experiencia y conocimiento en el área de gestión de proyectos de software de la empresa seleccionada.
- **Ejecución del estudio de caso:** Se ejecutó el estudio de caso teniendo en cuenta la planificación y diseño planteado en las actividades previas.
- **Observación y reflexión estudio de caso:** Se recogieron los datos sobre la ejecución e intervención del estudio de caso y posteriormente se generó un reporte como resultado de la reflexión y el análisis de los datos obtenidos durante la ejecución del estudio de caso, para obtener conclusiones que permitieran llevar a cabo la retroalimentación y evaluación del aprendizaje obtenido.

### 1.1.6.Ciclo de documentación y socialización

Este ciclo se llevó a cabo de manera transversal con el fin de organizar toda la documentación y socialización de los resultados obtenidos durante la ejecución del proyecto. En esta fase se espera llevaron a cabo las siguientes actividades:

- **Estancia de investigación:** Se realizó una pasantía en el grupo de investigación I+D+I en tecnologías de la información y las comunicaciones – GIDITIC de la

universidad EAFIT bajo la tutoría de la PhD. Elizabeth Suescún. Esta pasantía permitió socializar los resultados parciales del proyecto y realizar los ajustes sobre la propuesta de la ontología DevOps Ontology.

- **Elaboración artículos:** Se elaboraron seis artículos, de los cuales tres fueron ya publicados en revistas indexadas, uno ya fue aprobado y se encuentra en proceso de publicación por parte de los editores de la revista, y dos más se encuentran en proceso de escritura, uno describiendo el modelo de referencia propuesto y otro detallando el modelo de evaluación, para su posterior envío a revistas indexadas.
- **Elaboración de la monografía:** Se elaboró de la monografía y los Anexos que resultaron durante la realización del trabajo de grado.
- **Sustentación del proyecto:** Se presentaron y sustentaron los resultados obtenidos durante el desarrollo del proyecto.

## 1.4. Resultados obtenidos

Los resultados obtenidos con la realización del presente trabajo de maestría son los siguientes:

- Monografía de trabajo de grado, el cual se estructura de acuerdo con los elementos definidos en la siguiente Sección.
- Prototipo de aplicación móvil para la consulta de los conceptos y relaciones definidos por medio de la ontología DevOps Ontology.
- Artículos publicados en revistas indexadas (ver Sección 7.2).

## 1.5. Organización del documento

Los siguientes capítulos de este documento están organizados de la siguiente forma:

- **Capítulo 2.** Presenta el marco teórico, el cual incluye los conceptos básicos relacionados con la temática del proyecto, y el estado del arte el cual se describe por medio de un mapeo sistemático de la literatura.
- **Capítulo 3.** Presenta la caracterización de elementos de proceso obtenidos desde la revisión sistemática de la literatura, en este capítulo se presentan los procesos, actividades, artefactos, roles y herramientas identificados.
- **Capítulo 4.** Presenta la ontología DevOps Ontology, describiendo la metodología utilizada, el propósito, preguntas de competencia, su representación gráfica, la evaluación de la ontología realizada y su comparación con otras ontologías y taxonomías existentes.
- **Capítulo 5.** Presenta el modelo de referencia DevOps Model el cual se describe desde los valores de DevOps, los principios de DevOps, las dimensiones asociadas y el conjunto de prácticas fundamentales y complementarios, y este capítulo también presenta el modelo de valoración DevOps Assessment Model que se construye a partir de DevOps Model y el paradigma GQM.
- **Capítulo 6.** Presenta la evaluación de DevOps Model, indicando el procedimiento realizado para la aplicación de un grupo focal y lo correspondiente para la realización del estudio de caso en una organización de desarrollo de software.

- **Capítulo 7.** Presenta las conclusiones y trabajo futuro.
- **Capítulo 8.** Presenta la bibliografía.
- **Anexos:** Los cuales se dividen en:
  - **Anexo A:** Presenta el detalle del mapeo sistemático de la literatura realizado.
  - **Anexo B:** Presenta los cuestionarios de evaluación diligenciados por los participantes del grupo focal.
  - **Anexo C:** Presenta la documentación del estudio de caso: (i) carta de aceptación de participación, (ii) instrumento de evaluación diligenciado y (iii) informe de resultados entregado a la organización.
  - **Anexo D:** Modelo de carta de participación en el estudio de caso.

# Capítulo 2. Marco teórico y estado del arte

En este capítulo se presentan los conceptos más relevantes usados durante la realización de este proyecto, lo que constituye el marco teórico de la propuesta, también se presenta el mapeo sistemático usado como instrumento para proporcionar un estado del arte actualizado donde se sintetizan los trabajos relacionados, propuestas e iniciativas en el proceso de adopción de DevOps en empresas de desarrollo de software.

## 2.1. Marco teórico

A continuación, se presentan algunos conceptos importantes que facilitarán el proceso de lectura del documento y permitirán mejorar la comprensión de la propuesta presentada.

### 2.1.1. Software

Según [5], software es un término que asocia programas que se ejecutan en una computadora con sus configuraciones y documentación. En el contexto de esta investigación extendemos esta definición a todos los productos y servicios que son ejecutados en un dispositivo del cliente junto a su configuración y documentación.

### 2.1.2. DevOps

Según [48], DevOps es: “una metodología de desarrollo dirigida a cerrar la brecha entre el desarrollo y las operaciones. Enfatiza en la comunicación, colaboración, integración continua, el aseguramiento de la calidad y la entrega con el despliegue automatizado, utilizando un conjunto de prácticas de desarrollo”.

### 2.1.3. Proceso software

En el contexto de la ingeniería de software un proceso de software es un enfoque adaptable que permite que el equipo de desarrollo busque y elija el conjunto apropiado de acciones y tareas para el trabajo [6].

### 2.1.4. Modelo de referencia

Según [49], un modelo de referencia es “un modelo conceptual genérico que formaliza prácticas recomendadas para cierto dominio”. Las prácticas recomendadas o también conocidas como mejores prácticas indican la forma de abordar un problema y pueden ser replicadas y adaptadas por una organización en un contexto similar [50]. Un modelo de referencia incluye un conjunto de conceptos, axiomas y relaciones propios de un dominio particular del problema, y es independiente de estándares científicos, tecnologías, implementaciones, o de cualquier otro detalle concreto [51].

### **2.1.5.Ontología**

En [52] se define la ontología como la especificación explícita de una conceptualización, donde la conceptualización hace referencia a los objetos, conceptos y otras entidades cuya existencia se asume en un área de interés y las relaciones existentes entre ellas. Las ontologías permiten aclarar la estructura del conocimiento en un dominio, ayudan a reducir la ambigüedad de conceptos y términos y permiten a distintas personas o sistemas compartir conocimiento acerca de un dominio [53].

### **2.1.6.Goal Question Metric – GQM**

Según [54], GQM es un paradigma para desarrollar y mantener un programa de métricas que plantea como principio básico; que la medición debe ser realizada siempre orientada hacia un objetivo. Este paradigma propone un modelo de medición en tres niveles, donde primero se define un objetivo, luego se refina por medio de preguntas, y finalmente se definen unas métricas que intentan dar información para responder las preguntas que se hayan formulado.

## **2.2. Estado del arte**

Con el fin de brindar al lector una visión clara del estado del arte relacionado con la propuesta presentada en este documento, en esta Sección se presenta el mapeo sistemático de la literatura realizado, el cual fue presentado en [38], [55], y pueden ser consultados con mayor detalle en el Anexo A.

### **2.2.1.Protocolo de investigación**

Un mapeo sistemático de la literatura es un método para recopilar y categorizar información existente sobre un tema de investigación. Este mapeo sistemático se llevó a cabo siguiendo las pautas presentadas en los siguientes estudios: [47], [56]–[59] y tiene como propósito filtrar toda la información relevante relacionada con los procesos de adopción de DevOps y generar los insumos requeridos para realizar una caracterización de la información existente. El mapeo sistemático se realizó en tres etapas: planeación, ejecución y documentación. En los siguientes apartados se presenta un resumen de la planificación, ejecución y análisis de los resultados.

#### *2.2.1.1. Etapa de planificación*

En la etapa de planificación se realizaron las siguientes actividades: (i) definición de preguntas de investigación, (ii) definición de la estrategia de búsqueda, (iii) definición de los criterios de selección de estudios primarios, (iv) definición de los criterios de evaluación de la calidad, (v) definición de la estrategia de extracción de datos y (vi) Selección de los métodos de síntesis.

A continuación, en los siguientes apartados se abordan con mayor detalle cada una de las actividades realizadas.

### 2.2.1.2. Preguntas de investigación

Con el fin de lograr el objetivo del mapeo sistemático e identificar posibles brechas existentes para proponer nuevas áreas de investigación, se han establecido las preguntas de investigación que se muestran en la Tabla 2-1.

Tabla 2-1. Preguntas de Investigación.

<b>Id.</b>	<b>Pregunta de investigación</b>	<b>Motivación</b>
P1	¿De acuerdo con la literatura analizada que se entiende por DevOps?	Se busca una aproximación a una definición de DevOps estándar en el entorno del software.
P2	¿Qué tipo de instrumentos o métodos de investigación son usados en la literatura analizada?	Determinar los instrumentos o métodos de investigación más usados en la resolución de preguntas de investigación respecto a DevOps.
P3	¿Qué tipo de soluciones han sido propuestas para simplificar la adopción de DevOps en empresas de desarrollo de software?	Determinar el estado del arte relacionado con la adopción de DevOps, y establecer el trabajo por realizar o mejorar. Se entiende por solución aquella propuesta que presenta un enfoque, modelo, metodología, técnica u otros.
P4	¿Cuáles elementos de procesos se han considerado para aplicar DevOps?	Examinar las propuestas, su enfoque, completitud y aplicabilidad. Elemento de proceso: toda aquella sugerencia a nivel de tareas, actividades, artefactos, plantillas y roles.
P5	¿En qué tipo de proyectos de desarrollo de software se usa DevOps?	Determinar los tipos de desarrollo de proyectos de software: standalone, web o móvil, sobre los que se ha realizado la adopción de DevOps.
P6	¿Qué herramientas tecnológicas son usadas para simplificar la adopción de DevOps?	Conocer las herramientas tecnológicas que soportan algún tipo de actividad para facilitar la adopción de DevOps.
P7	¿Cuáles son los beneficios y desafíos que conlleva la adopción de DevOps?	Determinar los beneficios y desafíos a los que se enfrentan las empresas que llevan a cabo la adopción de DevOps.

Acrónimos utilizados: **Id.**: identificador de la pregunta.

Considerando los trabajos relacionados, es posible observar que; si bien existen diferentes propuestas que intentan abordar algunos de los desafíos en las empresas de desarrollo de software, no existen estudios o propuestas que consideren los posibles escenarios multimodales que se pueden generar en este tipo de ambientes.

### 2.2.1.3. Estrategia de búsqueda

Para la búsqueda, se utilizaron combinaciones de los conectores lógicos “AND” y “OR” en las palabras clave identificadas. Se utilizaron las bases de datos: Scopus, Google Scholar, SpringerLink e IEEE Xplore, en las que se introdujo la cadena de búsqueda adaptada para cada uno de los buscadores. Asimismo, se utilizaron estudios proporcionados por expertos, que han sido clasificados como literatura gris. La Tabla 2-2 presenta la cadena de búsqueda básica formada a partir de un conjunto de palabras clave utilizadas para encontrar respuesta a las preguntas de investigación. Se consideró toda la información publicada desde enero de 2009 (fecha en la que se acuñó por primera vez el término DevOps en la industria según [60]) hasta junio de 2020. Este período de tiempo ha sido validado durante la ejecución del mapeo sistemático, y la mayoría de las publicaciones relevantes sobre la adopción de DevOps en empresas de desarrollo de software se encuentran en los últimos seis años (2014 - 2020).



Tabla 2-2. Cadena de búsqueda básica usada en el mapeo sistemático.

<b>Cadena de búsqueda básica</b>
(“agile“ OR “agile approach“ OR “agile process“) AND (DevOps OR “continuous delivery“ OR “continuous release“ OR “continuous deployment“) AND (adopt OR integrate OR integration OR integrating) AND (“software development“).

#### 2.2.1.4. Criterios de selección de estudios primarios

Los estudios recolectados fueron evaluados considerando los siguientes aspectos: el título, resumen y palabras clave, para determinar si estaban o no incluidos entre los estudios relevantes. Luego, se analizaron en detalle para seleccionar los estudios primarios. Se incluyeron aquellos estudios que cumplieron al menos uno de los siguientes criterios de inclusión:

- Estudios en inglés que se refieren a la adopción de DevOps en empresas de desarrollo de software.
- Estudios publicados entre 2009 y 2020 en revistas, conferencias, congresos o talleres de prestigio con revisión por pares.

Por otro lado, no se tuvieron en cuenta los estudios que cumplieran alguno de los siguientes criterios de exclusión:

- Estudios duplicados.
- Estudios cuya principal contribución no esté relacionada con la adopción de DevOps en empresas de desarrollo de software.
- Estudios que contemplan el tema de manera superficial.
- Tipos de estudios de debate, o disponibles únicamente en forma de resúmenes o presentaciones.

#### 2.2.1.5. Criterios de evaluación de calidad

Para medir la calidad de los estudios seleccionados y determinar los estudios relevantes sobre la adopción de DevOps en empresas de desarrollo de software, se desarrolló un cuestionario con un sistema de puntuación de tres valores (-1, 0 y +1) siguiendo las sugerencias establecidas en [58]. Los criterios de calidad establecidos para evaluar los estudios primarios se explican en [55]. En la Tabla 2-3 se presentan las cuestiones a considerar para realizar el proceso de evaluación de cada estudio y la puntuación que se le debe asignar de acuerdo con las posibles respuestas. En la Tabla 2-4 se presentan los resultados de la evaluación de los estudios según los criterios de evaluación de la calidad. La suma de la puntuación de cada estudio corresponderá a la puntuación final (obteniendo un valor entre -6 y +6). Estas puntuaciones no se utilizarán para excluir un estudio específico del mapeo sistemático, sino que, en caso de obtener una mala puntuación, se utilizarán para buscar estudios más relevantes, que tendrán más relevancia en futuras investigaciones.

Tabla 2-3. Cuestionario y puntuación de los criterios de evaluación de calidad.

No.	Cuestión	Puntuación asignada a las posibles respuestas		
		+1	0	-1
C1	El estudio contempla una definición clara de DevOps.	Si	Parcialmente	No
C2	El estudio propone un conjunto de elementos de proceso descritos de manera detallada para llevar a cabo la adopción de DevOps.	Si	Parcialmente	No
C3	El estudio valida la propuesta de adopción de DevOps en Empresas de Desarrollo de Software.	A través de un estudio de caso	A través de un grupo focal	No validado
C4	El estudio expone de manera clara y detallada los resultados obtenidos tras aplicar la propuesta de adopción de DevOps.	Si	Parcialmente	No
C5	El estudio ha sido publicado en una revista, conferencia o congreso relevante.	Muy relevante (JCR mayor que 2.0)	Relevante (JCR entre 1.0 y 2.0)	No relevante (JCR es menor que 1.0 o no aparece en la lista)
C6	El estudio ha sido citado por otros autores (según el índice de citas de Google Scholar).	Por más de cinco autores	Entre uno y cinco autores.	No ha sido citado.

Acrónimos utilizados: No.: Número.

Tabla 2-4. Evaluación de los estudios de acuerdo con los criterios de evaluación de calidad.

Ref.	Criterio						Puntuación
	C1	C2	C3	C4	C5	C6	
[39]	1	1	1	0	-1	0	2
[35]	1	1	1	0	-1	1	3
[32]	0	0	1	-1	-1	1	0
[29]	1	-1	1	-1	-1	1	0
[30]	0	0	-1	-1	-1	0	-3
[61]	0	0	-1	-1	0	1	-1
[27]	1	-1	-1	-1	-1	1	-2
[37]	-1	-1	-1	-1	0	1	-3
[42]	0	0	1	0	-1	1	1
[33]	0	-1	-1	-1	-1	1	-3
[34]	0	0	-1	-1	-1	0	-3
[62]	0	-1	0	-1	0	-1	-1
[36]	1	0	-1	-1	-1	1	-1
[48]	1	-1	-1	-1	-1	1	-2
[63]	0	0	1	-1	-1	1	0
[40]	-1	1	1	0	-1	1	1
[41]	-1	0	1	-1	-1	1	-1
[64]	0	1	1	0	-1	-1	0
[65]	-1	0	0	-1	-1	-1	-4
[28]	0	-1	1	-1	-1	0	-2
[66]	1	0	-1	0	0	-1	-1
[67]	0	0	-1	-1	-1	-1	-4
[68]	0	1	1	0	0	0	2
[69]	1	1	-1	-1	0	1	1
[70]	1	1	-1	-1	0	1	1
[71]	0	-1	-1	-1	0	0	-3

Acrónimos utilizados: **Ref.**: referencia.

### 2.2.1.6. Estrategia de extracción de datos

Para asegurar la aplicación de los mismos criterios de extracción de datos para todos los estudios seleccionados y simplificar su clasificación, se utilizó un conjunto de posibles respuestas para cada una de las preguntas de investigación definidas, como se muestra en la Tabla 2-5.

Tabla 2-5. Esquema de clasificación.

Pregunta	Respuestas
P1. ¿Según la literatura analizada qué se entiende por DevOps?	a. Brecha entre desarrollo y operaciones, b. Cultura de colaboración y comunicación, c. Integración, entrega y despliegue continuo, d. Aseguramiento de la calidad, e. Automatización, f. Otros
P2. ¿Qué tipo de instrumentos o métodos de investigación se realizan en la literatura analizada?	a. Mapeo sistemático/Revisión de la literatura, b. Caso de estudio, c. Encuesta/entrevista, d. Propuestas, e. Otros
P3. ¿Qué tipo de soluciones han sido propuestas para facilitar la adopción de DevOps en las Empresas de Desarrollo de Software?	a. Enfoques, b. Modelo, c. Metodología, d. Técnica, e. Marco de trabajo, f. Otros, g. No se propone una solución concreta
P4. ¿Cuáles son los elementos de proceso que se han tenido en cuenta para aplicar DevOps?	a. Roles, b. Prácticas, c. Actividades, e. Plantillas, f. Artefactos, g. Otros
P5. ¿En qué tipos de proyectos de desarrollo de software se utiliza DevOps?	a. Standalone, b. Web, c. Móvil, e. No se clarifica el tipo de proyecto
P6. ¿Qué herramientas tecnológicas son utilizadas para facilitar la adopción de DevOps?	a. Herramientas para la integración, entrega y despliegue continuo, b. Herramientas para fomentar la comunicación y colaboración, c. Otros
P7. ¿Qué beneficios y desafíos conlleva la adopción de DevOps?	a. Económicos, b. Mejora procesos, c. Mejora la colaboración y comunicación, e. Aseguramiento de la calidad, f. Satisfacción del cliente, g. Mejora de la productividad, h. Resistencia al cambio, i. Otros

### 2.2.1.7. Métodos de síntesis

La información de los estudios primarios seleccionados fue extraída y estructurada de la siguiente manera: identificación (título, publicación, autores, referencia), resumen, descripción (metodología, tipo de propuesta, tipo de evaluación de la propuesta) y aspectos destacados. La Tabla 2-6 muestra la relación de los estudios primarios con las preguntas de investigación que respondió cada uno.

Tabla 2-6. Contribuciones de los estudios primarios a cada una de las preguntas de investigación.

Ref.	Pregunta						
	P1	P2	P3	P4	P5	P6	P7
[39]	X	X	X	-	-	-	-
[35]	X	X	X	X	-	X	X
[32]	X	-	X	X	-	X	X
[29]	X	X	X	-	-	-	X
[30]	X	X	X	X	X	X	X
[61]	X	X	X	X	-	-	X
[27]	X	X	X	-	-	-	X
[37]	-	X	X	-	-	-	X
[42]	X	-	X	-	-	-	-
[33]	X	-	X	-	-	-	X
[34]	X	X	X	X	-	-	X
[62]	X	-	X	-	-	-	-

[36]	X	X	X	X	-	X	X
[48]	X	X	X	-	X	-	-
[63]	X	X	X	X	X	X	-
[40]	-	X	X	X	X	X	-
[41]	-	X	X	X	X	X	-
[64]	X	X	X	X	-	X	-
[65]	-	X	X	-	-	-	-
[28]	X	X	X	-	-	-	X
[66]	X	X	X	X	X	X	X
[67]	X	X	X	X	-	X	-
[68]	X	X	X	X	-	X	-
[69]	X	X	X	X	X	-	-
[70]	X	X	X	X	-	X	-
[71]	X	X	X	-	-	-	-

Acrónimos utilizados: **Ref.**: referencia.

#### 2.2.1.8. Calendario del mapeo sistemático

El mapeo sistemático se realizó en el transcurso de tiempo comprendido entre enero de 2019 y abril de 2019. Luego, se realizó su debida actualización a junio de 2020; ambas versiones del mapeo sistemático han sido publicadas en [55] y [38] respectivamente. La versión más reciente y actualizada puede consultarse en el Anexo A.

#### 2.2.1.9. Etapa de ejecución

Se realizaron cinco iteraciones, una iteración cero donde se realizó una revisión inicial de literatura gris que constaba de 10 artículos, esta revisión permitió conocer en profundidad los conceptos a los que se suele asociar DevOps. Con el fin de afinar la final cadena de búsqueda, luego se realizó una nueva iteración para cada fuente de búsqueda establecida. La Tabla 2-7 muestra el número total de estudios: encontrados, relevantes, repetidos y primarios, adquiridos en las fuentes de búsqueda de Scopus, Google Scholar, Springer e IEEE Xplore.

Tabla 2-7. Resultados de búsqueda y análisis de estudios encontrados.

No.	Fuente de datos	Encontrados	Relevantes	Relevantes repetidos (iteraciones anteriores)	Primarios seleccionados
0	Literatura gris	10	9	0	7
1	Scopus	522	10	7	3
2	Google Scholar	219	7	4	6
3	Springer	487	8	4	4
4	IEEE Xplore	42	3	3	3
<b>Total</b>		1280	37	18	23

Acrónimos utilizados: **No.**: número

## 2.2.2. Resultados del mapeo sistemático de la literatura

En este apartado se muestran los resultados obtenidos para cada una de las preguntas de investigación definidas, todas las respuestas encontradas tienen sus referencias respectivas para facilitar su estudio posterior en profundidad por parte del lector e interesados.

### 2.2.2.1. Pregunta P1: Según la literatura analizada, ¿qué se entiende por DevOps?

Se encontraron diferentes definiciones para DevOps en la literatura analizada, parece que aún no se ha alcanzado una definición o un entendimiento común de lo que implica DevOps [48], por lo que algunas definiciones pueden estar caracterizando a DevOps desde una perspectiva muy específica [48]. Al analizar cada definición, se identificaron las características y/o aspectos comunes (en adelante conceptos) que los autores han considerado para definir DevOps. Los conceptos comunes que se han identificado en las definiciones son: (i) Colaboración y comunicación, (ii) Desarrollo y operaciones, (iii) Cubrir la brecha, (iv) Aseguramiento de la calidad, (v) Automatización, (vi) Integración, entrega y despliegue continuo y (vii) Conjunto de prácticas.

Según los estudios analizados, también se observó que DevOps representa: un enfoque [61], [68], [69], un movimiento cultural [39], [71], un conjunto de prácticas [29], [30], un paradigma emergente [42], [70], una mezcla de dos palabras [33], un fenómeno [34], un tema interdisciplinario [32], un conjunto de principios [62], un marco conceptual [36], una metodología de desarrollo [67], una palabra artificial [63], una filosofía [64] y un neologismo [27]. De lo anterior, se puede evidenciar que DevOps se ha interpretado de varias formas porque es un tema relativamente nuevo.

### 2.2.2.2. Pregunta P2: ¿Qué tipo de investigación se realiza en la literatura analizada?

Los estudios seleccionados en esta investigación se pueden dividir en dos categorías: (i) estudios con propuestas concretas sobre la adopción de DevOps [28], [29], [34], [39]–[41], [63], [64], [66], [70] y (ii) revisiones sistemáticas de la literatura [27], [30], [35], [36], [61], [65], [69], [71]. En cuanto a la primera categoría, se encontró que el 90% de los estudios ([9], [12], [19], [23]–[26], [28], [29]) registrados sobre la adopción de DevOps utilizan estudios de caso para validar sus propuestas, estos Casos de estudio se han realizado en un rango de uno a cinco años en su aplicación dependiendo del tamaño del proyecto. Por otro lado, de estos estudios el 77% ([9], [12], [19], [23]–[25], [28]) complementan los hallazgos con encuestas y/o entrevistas para determinar el impacto a nivel de cultura y organización. En cuanto a la segunda categoría, solo el 38% ([27], [35], [65]) de ellos utilizan sus hallazgos para proponer, aunque sea de forma conceptual, alguna solución para la implementación de DevOps con algún marco ágil.

### 2.2.2.3. Pregunta P3: ¿Qué tipo de soluciones han sido propuestas para la adopción de DevOps?

El 20% ([35], [39], [40], [42], [70]) de la literatura analizada propone soluciones concretas para guiar a la adopción de DevOps en empresas de desarrollo de software. Por otro lado, el

80% ([27], [28], [48], [61]–[69], [29], [71], [30], [32]–[34], [36], [37], [41]) de los estudios analizados, no proponen una solución concreta para la adopción de DevOps en empresas de desarrollo de software, pero describen su experiencia en la adopción de DevOps. Es posible observar que el 56% ([27], [28], [61], [62], [66], [70], [30], [32]–[37], [39]) identifica beneficios y desafíos, además, el 60% ([29], [32], [65], [67]–[70], [34], [35], [39], [41], [61]–[64]) proponen un conjunto de elementos de proceso que podrían considerarse para la adopción de DevOps. Del mismo modo, el 40% ([30], [35], [36], [40], [41], [63], [64], [66], [68], [70]) mencionan las herramientas tecnológicas que podrían respaldar las prácticas para adoptar DevOps. Es importante mencionar que la información detallada sobre los beneficios y desafíos identificados en la literatura analizada se encuentra disponible en [55].

2.2.2.4. *Pregunta P4: ¿Cuáles son los elementos de proceso que se han tenido en cuenta para aplicar DevOps?*

Luego de analizar los estudios primarios, se pudo observar que el 56% ([30], [32], [66]–[68], [70], [34]–[36], [40], [41], [61], [63], [64]) proponen o mencionan elementos de proceso, los cuales fueron categorizados de acuerdo con los elementos de proceso propuestos en la Ontología de Modelos Procesos-referenciales - PrMO [72]. PrMO define conceptos y sus propias definiciones y relaciones para elementos de proceso tales como: proceso, actividad, productos de trabajo, rol y herramienta. La ontología también define otros conceptos para los elementos del proceso, como la categoría del proceso y las tareas, sin embargo, dado que no se encontraron tales elementos del proceso, no se mencionan en este trabajo. En la Tabla 2-8 se presentan algunos elementos del proceso identificados en cada uno de los estudios seleccionados, en este sentido, la columna Actividad y Herramienta muestran la cantidad de actividades y herramientas, respectivamente. La tabla completa con todas las actividades identificadas se encuentra en el Anexo B. A partir de los estudios primarios seleccionados, se identificó que el 46% de ellos proponen procesos que apoyan a DevOps, de los procesos que apoyan a DevOps se encontraron: integración continua, entrega continua, supervisión continua, implementación continua, aseguramiento de la calidad, proceso de configuración, enfoque de DevOps Framework. Asimismo, el 38% de los estudios proponen actividades, el 8% productos de trabajo, el 38% roles, el 69% herramientas tecnológicas y el 31% proponen otros elementos del proceso, por ejemplo: áreas o dimensiones. Se debe considerar que un estudio puede proponer varios elementos de proceso y que estos pueden soportar diferentes procesos. La Tabla 2-8 muestra solo el número de herramientas propuestas por cada autor porque las herramientas tecnológicas que soportan DevOps se abordan en profundidad en la pregunta de investigación P6, Tabla 3-5.

Tabla 2-8. Elementos de proceso encontrados por estudio.

Ref.	Elemento de proceso				
	Proceso	Act.	Producto	Rol	Her.
[35]	ND	ND	Lista de impedimentos, Modelo de madurez de DevOps, Lista de herramienta de DevOps usada en la compañía.	ND	45
[32]	ND	ND	ND	Gerente de desarrollo de software, analista de pruebas y desarrollador con enfoque DevOps, Gerente de sistemas	4

[30]	CD, DC, QA	11	ND	ND	19
[61]	ND	ND	ND	Ingeniero, equipo y departamento DevOps, desarrollador con enfoque DevOps, Gerente de sistemas	ND
[34]	ND	24	ND	ND	ND
[36]	Medición, automatización	19	ND	ND	2
[63]	DC	2	ND	ND	2
[40]	ND	ND	ND	Desarrollador con enfoque DevOps, Product Owner, Líder de proyecto, líder de equipo, gerente de programa, gerente de despliegue	1
[41]	CI, CD, CS	ND	ND	Equipo DevOps	1
[64]	PC, DFA	10	ND	ND	4
[66]	ND	ND	ND	ND	14
[67]	CD, CD, DC	3	ND	ND	5
[68]	ND	ND	ND	Product Owner, Arquitecto software, Ingeniero de producción, gerente de lanzamiento	ND
[70]	CI, CS, DC	ND	ND	ND	29

Acrónimos utilizados: **Ref**: referencia; **Act**: Actividad; **Her**: Herramienta; **ND**: No disponible; **CI**: integración continua; **CD**: entrega continua; **CS**: supervisión continua; **DC**: despliegue continuo; **QA**: aseguramiento de la calidad; **PC**: proceso de configuración; **DFA**: Enfoque de marco de trabajo DevOps

#### 2.2.2.5. Pregunta P5: ¿En qué tipos de proyectos de desarrollo de software (standalone, web o móvil) se utiliza DevOps?

El 100% de los estudios seleccionados que describen proyectos de desarrollo de software con evidencia de avances o propuestas en la adopción de DevOps hacen uso de servicios Cloud, por ejemplo: servicios de integración continua, monitoreo, entre otros [28], [29], [34], [39]–[41], [63], [64], [66] y no se evidencian estudios donde se proponga la creación de servicios de soporte a prácticas relacionadas con la adopción de DevOps sobre redes locales. Solo en [46] se describen artefactos y prácticas específicas para la adopción de DevOps en entornos de desarrollo móvil, el resto de las propuestas se centran en entornos Web.

#### 2.2.2.6. Pregunta P6: ¿Qué herramientas tecnológicas son utilizadas para facilitar la adopción de DevOps?

Se identificaron un total de 60 herramientas tecnológicas que apoyan los procesos involucrados en la adopción de DevOps; para clasificarlos, se utilizó el catálogo definido en [30]. 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). Las herramientas que fueron identificadas, su clasificación en un área/proceso de DevOps y referencias, se abordan en detalle más adelante en la Sección 3.2.5.

#### 2.2.2.7. Pregunta P7: ¿Qué beneficios y desafíos conlleva la adopción de DevOps?

En la literatura analizada se observa que la adopción de DevOps en las empresas de desarrollo de software conlleva ciertos beneficios, entre ellos: (i) tiempo de respuesta (time to market) que toma un producto desde que es concebido hasta que sale a la venta: [27]–[30], [32], [37], (ii) para cerrar la brecha entre Dev y Ops: [27]–[30], [32], (iii) mejora de la calidad del producto: [32], [36], [37], [66] y (iv) mejora de la satisfacción del cliente: [27], [28], [37]. Además de los beneficios que conlleva la adopción de DevOps, se identificaron ciertos desafíos que dificultan este proceso de adopción, estos incluyen: (i) resistencia al cambio por parte de las personas: [28]–[30], [32], [34], [35], (ii) carencia de comprensión del significado de DevOps: [29], [33]–[35], (iii) falta de colaboración entre desarrollo y operaciones: [29], [33], [35], [61] y (iv) para cambiar la cultura de la empresa: [29], [32], [34], [35]. La lista detallada de beneficios y desafíos que se identificaron en la literatura analizada se encuentra en [55].

### 2.2.3. Discusión

En esta Sección se presenta un análisis de los resultados obtenidos del mapeo sistemático realizado, con el fin de identificar las mejoras que se pueden realizar sobre las propuestas encontradas.

#### 2.2.3.1. Observaciones principales

El objetivo de este mapeo sistemático fue conocer el estado del arte actual de la adopción de DevOps en las empresas de desarrollo de software. En este sentido, tras analizar los resultados, se puede deducir que según el análisis de lo que representa DevOps en los estudios analizados, “neologismo” es el término que caracterizaría y clasificaría la mayoría de las definiciones. Además, se observa que este neologismo se refiere al movimiento cultural al que una empresa debe adaptarse para adoptar DevOps. Además, está claro que DevOps no es solo un conjunto de prácticas, sino que está relacionado con toda sugerencia a nivel de tareas, actividades, artefactos, plantillas y roles. Por tanto, se considera más apropiado utilizar el término “elementos de proceso”. Dado que existen definiciones que no toman en cuenta algunos de los conceptos comunes identificados, se propone la siguiente definición de DevOps: *“DevOps es un neologismo que surge de la combinación de las palabras Desarrollo y Operaciones. Es un movimiento cultural que permite cerrar la brecha entre los desarrolladores y el personal de operaciones. DevOps está relacionado con un conjunto de elementos de proceso que fomentan la colaboración y la comunicación, la automatización, el control de calidad y la integración, la entrega y despliegue continuo”*.

Una gran parte de las propuestas para la adopción de DevOps se centra en las prácticas de integración continua, despliegue continuo y automatización de procesos. Sin embargo, las propuestas definidas no están muy aclaradas, por lo que aún queda un largo camino por recorrer en la definición de estándares para la apropiación contextualizada de este movimiento cultural que impacta proyectos de software desde diferentes perspectivas como arquitectura, cultura, personas, procesos y herramientas.



En cuanto a los elementos de proceso para apoyar DevOps, se ha observado que estos no están claramente definidos, algunos de ellos solo se identifican, pero no se describen a fondo, incluso, las definiciones de los elementos de proceso son ambiguas, y en la mayoría de los casos, incompletas. Los autores proponen actividades y / o tareas que implican la adopción de DevOps, pero no explican con detenimiento cómo implementarlas ni qué roles se encargan de ejecutarlas. En cuanto a los roles, solo se menciona que es necesario tener conocimientos avanzados sobre DevOps, sin embargo, no profundizan en sus responsabilidades asignadas, la etapa en la que participan, entre otros. También se pudo observar que los autores definen los elementos del proceso de acuerdo con su experiencia y no a un estándar o modelo. Además, no se evidenció en los estudios analizados un análisis detallado de las actividades específicas que se deben realizar en cada proceso apoyado por DevOps. Tampoco en los casos de estudio realizados, se evidenció una mejora notable en la productividad lograda mediante la aplicación de DevOps en las empresas de casos de estudio.

#### *2.2.3.2. Limitaciones del mapeo sistemático*

La limitación de acceso a los motores de búsqueda académicos representa el estado del arte de la investigación sobre la adopción de DevOps en empresas de desarrollo de software. La inclusión de estudios solo en inglés puede significar que no se consideren estudios relevantes en otros idiomas, sin embargo, el inglés es el idioma más utilizado para publicar estudios sobre este tema. Además, aunque este mapeo sistemático ha arrojado resultados relevantes, sirven como punto de partida para una versión posterior del mapeo sistemático que aquí se presenta.

#### *2.2.3.3. Importancia para la investigación y la práctica*

Las observaciones de este mapeo sistemático son de gran importancia para aquellos investigadores que planean investigar DevOps, y más específicamente en el tema de la adopción de DevOps en empresas de desarrollo de software. Para los investigadores es un área de gran interés, ya que, como se ha señalado, es un campo nuevo, en el que se trabaja poco. Los investigadores, gracias al desarrollo en este campo, podrán crear guías que permitan la adopción, implementación y control de DevOps correctamente, o mejorar las prácticas existentes que las empresas están adoptando actualmente para implementar DevOps, para que las buenas prácticas de DevOps en empresas de desarrollo de software se puedan consolidar. Finalmente, las organizaciones se verán muy beneficiadas por el avance de este campo, pues hasta el momento, algunas organizaciones han desarrollado e implementado DevOps según sus propios criterios los cuales han considerado oportunos, sin embargo, esto se ha llevado a cabo sin una evaluación detallada de los resultados obtenidos en términos de los beneficios obtenidos.

# Capítulo 3. Caracterización de elementos de proceso relacionados a DevOps

Este capítulo presenta la caracterización de los elementos de proceso relacionados con DevOps, que fueron identificados por medio del mapeo sistemático de la literatura presentado en la Sección anterior (ver Sección 2.2). Esta caracterización es la base para la definición de la ontología y modelo de referencia para la adopción de DevOps propuestos en este trabajo, los cuales formalizan los elementos de proceso que se consideran fundamentales y que se deben tener en cuenta para una correcta implementación de DevOps en las empresas de desarrollo de software según lo encontrado en la literatura.

En el análisis de los estudios en el estado del arte fue posible identificar que varios trabajos proponen dimensiones similares para clasificar los elementos de proceso relacionados con DevOps, por ejemplo: cultura, herramientas, personas, procesos [33], [35], [36], entre otros, por lo tanto, se decidió tomar como referencia la clasificación definida en [35], por ser la más completa y detallada. Sin embargo, se evidenció que respecto a las actividades propuestas no había homogeneidad en los estudios analizados en cuanto a nombre y descripción se refiere, por lo cual, se decidió analizar la posibilidad de integrar las actividades que fueran similares. Para realizar esta integración, se siguió un proceso de comparación de elementos de proceso, el cual será descrito a continuación en la Sección 3.1 de este capítulo. En la Sección 3.2 se presentan los elementos de proceso identificados.

Por otra parte, también se encontraron elementos de proceso que no eran clasificados en una categoría específica, pero que se consideraron importantes para la propuesta en lo referente a: la gestión de la configuración, integración, entrega, despliegue y monitoreo continuo.

## 3.1. Proceso para la identificación de los elementos de proceso en los estudios primarios

Para la identificación de versiones más representativas, completas y relevantes de los elementos de proceso que serán la base de este trabajo, se realizaron las actividades descritas en la Tabla 3-1.

Tabla 3-1. Actividades para la identificación de los elementos de proceso en los estudios primarios.

<b>Id.</b>	<b>Actividad</b>	<b>Descripción</b>
A1	Seleccionar artículos	Se seleccionaron los artículos que presentaban elementos de proceso de cualquier tipo, según la ontología de referencia [29]. Además, se tuvieron en cuenta aquellos elementos propuestos en estos estudios que, aunque no cumplen con los conceptos definidos en la ontología, se consideraron relevantes para la definición de la propuesta.
A2	Registrar información	En este subproceso se encuentran las actividades que permiten clasificar los elementos de proceso identificados y registrar su información.
A2.1	Clasificar elementos de proceso	Clasificar elementos de proceso: en esta actividad se verifica la conformidad de los elementos de proceso, identificados en los estudios primarios, con los

		conceptos y/o descripción de la ontología de referencia [29]. Se verifica que dichos elementos de proceso cumplan con las definiciones dadas por la ontología de referencia para las siguientes categorías: (i) CP: Categoría de proceso, (ii) PR: Proceso, (iii) A: Actividad, (iv) T: Tarea, (v) PD: Producto, (vi) R: Rol, (vii) H: Herramienta, y (viii) O: Otro.
A2.2	Registrar información de los elementos de proceso	En esta actividad se crea un documento con los datos más representativos del elemento de proceso, opcionalmente, se añaden algunas observaciones que ayuden a entenderlo, y su(s) respectiva(s) referencia(s).
A3	Armonizar elementos de proceso	En este subproceso se encuentran las actividades que permiten realizar el debido tratamiento a los elementos de proceso dependiendo de si se identifican elementos de proceso similares.
A3.1	Identificar si hay elementos de proceso similares	Una vez identificados los elementos de proceso propuestos en cada uno de los estudios primarios seleccionados, se busca elementos de proceso similares mencionados por distintos autores.
N/A	Condición “¿Tiene similares?”	Aquí se presenta una bifurcación que altera el flujo de proceso, ya que dependiendo de si el elemento es similar a otros identificados, se realiza la actividad A3.2, de lo contrario el elemento de proceso se descarta para una posible integración con otros elementos de proceso actividad A3.3.
A3.2	Integrar los elementos de proceso	Se procede a realizar la integración de los elementos de proceso utilizando elementos de la teoría de conjuntos tales como: intersección, complemento y unión. El proceso de comparación se hizo uno a uno entre elementos de proceso del mismo tipo, es decir, actividades con actividades, herramientas con herramientas, etc. En cuanto a la comparación entre actividades, esta se realizó buscando actividades con nombres y/o definiciones similares. De modo que, si había actividades que se complementaban entre sí, se integraba su descripción y se ajustaba el nombre, obteniendo así una sola actividad más completa.
A3.3	Descartar elementos de proceso	Aquí se descartan los elementos de proceso de los cuales no se encontraron similares. Por lo tanto, estos elementos de proceso no sufren modificación alguna y se registran en el documento de resultados (ver Sección 3.2).
A4	Registrar resultados	Se asigna un nombre y descripción al elemento de proceso resultante, el cual debe ser lo más completo o significativo posible.

Acrónimos utilizados: **Id.**: identificador, **N/A**: no aplica.

En la Figura 3-1 se resume el proceso general aplicado para la identificación de los elementos de proceso en los estudios primarios. Adicionalmente, en la Figura 3-2 se describen las actividades de clasificar los elementos de proceso y registrar su información en la tabla de resultados que corresponden al subproceso de Registrar información. En la Figura 3-3 se observan las actividades de armonización de los elementos de proceso identificados.

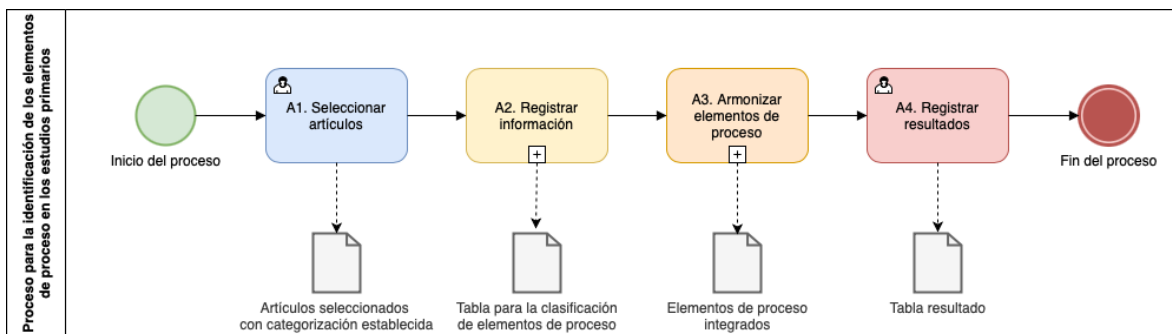


Figura 3-1. Proceso para la identificación de los elementos de proceso en los estudios primarios.

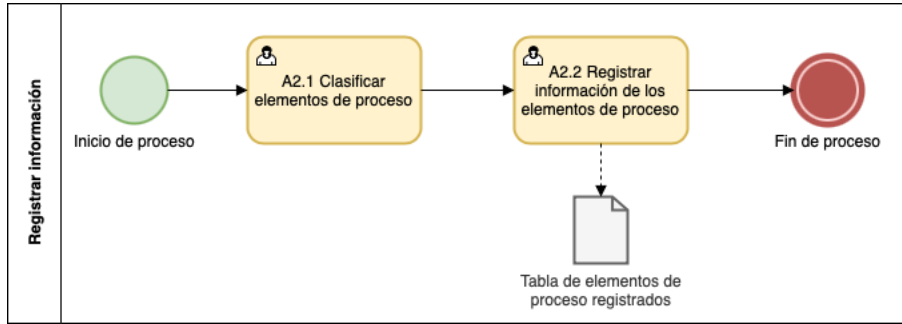


Figura 3-2. Subproceso registrar información.

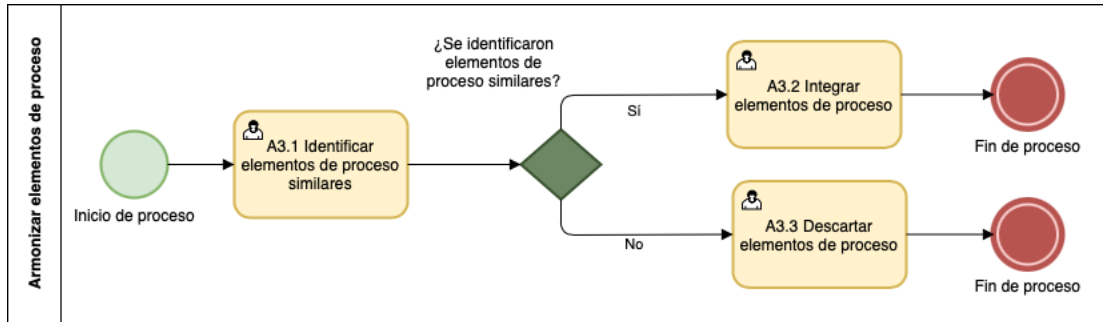


Figura 3-3. Subproceso armonizar elementos de proceso.

## 3.2. Elementos de proceso relacionados a DevOps

Al aplicar el proceso presentado anteriormente, se lograron identificar varios estudios que proponen o mencionan elementos de proceso relacionados con la implementación o adopción de DevOps en empresas de desarrollo de software. Con los resultados obtenidos se observó que algunos estudios categorizan estos elementos de proceso en dimensiones como se sugiere en [35], y otros estudios realizan clasificaciones por áreas como en [36], [37].

A continuación, se detalla la caracterización y clasificación de elementos de proceso realizada, la cual usa como referencia la ontología de elementos de proceso definida en [72], y que sugieren los siguientes elementos de proceso: categoría de proceso, proceso, actividad, tarea, producto, rol y herramienta.

### 3.2.1. Procesos identificados

Para este trabajo se usa la definición de *proceso* descrita en [72], donde se describe el proceso como “un conjunto de políticas, estructuras organizativas, tecnologías; procedimientos, propósitos, objetivos y productos de trabajo que se necesitan para diseñar, desarrollar, implementar y mantener un producto software”.

Teniendo en cuenta esta definición, se lograron detectar 9 procesos, los cuales son presentados en la Tabla 3-2, junto con la descripción y su referencia.

Tabla 3-2. Procesos encontrados en los estudios primarios.

No.	Proceso	Descripción	Ref.
1	Proceso de Configuración	Un proceso de configuración que detalla el control de versiones, la base de conocimiento, los ambientes, plataformas, monitoreo y transiciones	[64]
2	DevOps Framework Approach	Un proceso enfocado a los requerimientos en DevOps menciona las actividades: Identificar el requerimiento, implementar el requerimiento y desplegar el requerimiento	[64]
3	Integración Continua	Los miembros de un equipo integran su trabajo con frecuencia, por lo general, cada persona se integra al menos diariamente, lo que lleva a múltiples integraciones por día.	[37], [67]
4	Entrega continua	Lanzamientos frecuentes mediante la ayuda de un proceso de CI. Impone un proceso de desarrollo ágil para reducir el esfuerzo y la duración de las pruebas y el despliegue.	[30], [37], [67]
5	Supervisión continua	La capa de monitoreo en la tubería recopila una serie de métricas para evaluar el desempeño de la tubería del CD	[37]
6	Aseguramiento de la calidad	ND	[30]
7	Despliegue continuo	Hace uso de una tubería de despliegue e indica qué tipos de herramientas soportan las fases del despliegue (commit, pruebas automatizadas, pruebas exploratorias y producción)	[30], [63], [67]
8	Medición	Incluye actividades como contratar personas con experiencia en técnicas de medición o revisar las métricas utilizadas para medir el desempeño tanto de desarrollo como de operaciones	[36]
9	Automatización	Incluye actividades como hacer uso de la nube o el desarrollo dirigido por pruebas	[36]

Acrónimos utilizados: **No.**: número, **ND**: No disponible.

### 3.2.2. Actividades identificadas

Para este trabajo se usa la definición de *actividad* descrita en [72]: “comprende un conjunto de tareas o acciones que se utilizan para producir y mantener dispositivos, así como para lograr los objetivos del proceso. La actividad incluye los procedimientos, estándares, políticas y objetivos para crear y modificar un conjunto de productos de trabajo”.

Se consideraron como actividades las que cumplieran con la definición de la ontología usada como referencia y que además fueran presentadas como actividades por el autor del estudio. Se encontraron un total de 106 actividades en los estudios primarios analizados, actividades que estaban relacionadas con los procesos mencionados en el apartado anterior. A continuación, en la Tabla 3-3 se listan todas las actividades con su respectivo nombre y referencia.

Tabla 3-3. Actividades encontradas en los estudios primarios.

No.	Actividad	Ref.
1	Identificar requerimiento.	[64]
2	Implementar requerimiento.	
3	Desplegar requerimiento.	
4	Establecer control de versiones.	
5	Establecer base de conocimiento.	
6	Identificar ambientes.	
7	Suministrar ambientes.	

8	Configurar plataformas.	
9	Establecer monitoreo y realimentación.	
10	Configurar transiciones.	
11	Cree entornos limitados de desarrollo para una implementación mínima de código.	
12	Automatice la implementación de sandboxes a través de la canalización de desarrollo.	
13	Proporcione un sistema de colaboración continua en tiempo real.	
14	Automatice los sandboxes de prueba para que se ejecuten junto con los sandboxes de desarrollo.	
15	Realice pruebas Retro-QA en cajas de arena de compilación.	
16	Mantenga la varianza (código / calidad / comportamiento) entre el desarrollo y la producción al mínimo.	
17	Use las herramientas DevOps para automatizar la implementación, compilación, prueba, actualización, sincronización, implementación continua de código sandbox.	
18	Los desarrolladores deben poder acceder a los informes de incidentes de operaciones de TI y sincronizarse con las operaciones para mejorar la capacidad de soporte del proyecto.	
19	Los informes de prueba (generados automáticamente por una herramienta DevOps, unidades de prueba Sandbox, pruebas de calidad) deben compartirse entre Desarrolladores y Operaciones.	
20	Los registros de monitoreo (generados por herramientas DevOps o registros de monitoreo Retro-QA) deben compartirse entre Desarrollo y Operaciones.	[30]
21	El equipo de DevOps sincroniza servicios críticos como transacciones, rendimiento, tiempo de actividad, calendario de implementación, costos de tiempo de ejecución, control de versiones, y alcance del proyecto.	
22	El equipo de DevOps utiliza el repositorio central para el control de versiones y la sincronización del código fuente de la aplicación.	
23	Las implementaciones de lanzamiento de aplicaciones deben estar completamente automatizadas a través de la canalización de desarrollo.	
24	El equipo de DevOps debe proporcionar visibilidad general sobre el alcance del proyecto y liberar el tiempo para las partes interesadas.	
25	El equipo de DevOps debe proporcionar autoservicio y gestión de recursos de la plataforma (Cloud, Híbrido y Servidor) a las partes interesadas.	
26	El equipo de DevOps debe proporcionar parámetros de implementación seguros para evitar una carga de trabajo excesiva en la infraestructura.	
27	Discuta cómo la nueva forma de trabajar agregará valor a la unidad de negocio desintermediada (especialmente a los equipos de Operaciones).	
28	Comuniqué cómo la nueva forma de trabajar beneficiará al empleado individual.	
29	Reclutar a los empleados por la nueva forma de trabajar	
30	Articular cómo será el nuevo futuro y el rol del empleado individual en ese futuro.	
31	Invite a expertos líderes de la industria a evangelizar la nueva metodología dentro de las organizaciones, es decir, personas influyentes líderes en la industria.	
32	Motivar y animar a los empleados a asistir a las principales conferencias, eventos y reuniones de la industria.	
33	Crear oportunidades para establecer redes dentro de la industria, es decir, organizar reuniones, etc.	
34	Cree un día de exhibición dentro de la organización donde los empleados puedan mostrar a la gerencia y a otros lo que lograron.	
35	Ofrecer libros y otro material de capacitación para empleados.	
36	Crear una cultura de dominio y recompensar a los empleados en consecuencia.	
37	Comuniqué cómo la nueva forma de trabajar beneficiará al empleado individual.	[34]
38	Comuniqué cómo la nueva forma de trabajar agregará valor al negocio.	
39	Involucrar a todos al tomar decisiones de adopción.	
40	Identificar los silos y solicitar que justifiquen su existencia.	
41	Reorganizar equipos alrededor de flujos de valor.	
42	Continuar monitoreando los silos formados.	
43	Formar gremios que fomenten la comunicación a través de silos.	
44	Ajustar los objetivos para medir el flujo a través de la corriente de valor.	
45	Crear la visión del estado futuro y comunicarla.	
46	Insertar la estrategia de futuro en los objetivos de desempeño del liderazgo.	
47	Implementar a pequeña escala y mostrar los beneficios.	
48	Incorporar comentarios de la gerencia.	
49	Seguimiento del progreso frente a los objetivos estratégicos.	
50	Desarrollar marcos de toma de decisiones. Cambio de puestos de trabajo a roles	

51	Revise la prohibición de la política de acceso a la red para permitir que el personal vea videos de capacitación durante las horas de trabajo. Crea una academia de informática. Fomentar la adquisición de insignias digitales.	
52	Crea un gremio para impulsar el cambio cultural y el liderazgo de servicio. Dirección ejecutiva y alta dirección al liderazgo de servicio. [Se puede desarrollar un programa personalizado].	
53	Permitir la retroalimentación regular. Desarrolle una guía de retroalimentación para los líderes de equipo y atégase al liderazgo de servicio.	
54	Considerar el cambio cultural tanto desde la perspectiva del conductor como de los participantes.	
55	Cuando surgen situaciones excepcionales, uno tiene que ser flexible para abandonar temporalmente los valores de DevOps.	
56	Se debe implementar un proceso para integrar los cambios una vez que pase la crisis.	
57	Deje que los equipos decidan qué herramientas utilizan en función de sus propias habilidades y experiencia.	
58	Fomentar la transparencia entre el personal de desarrollo y de operaciones.	
59	Contratación de personas con el conocimiento correcto de automatización es importante para apoyar a DevOps	
60	Utilizar el almacenamiento en la nube para almacenar grandes archivos. - Procesar trabajos asíncronos utilizando colas.	
61	Use la plataforma como servicio en lugar de la infraestructura como servicio.	
62	Utilice las sesiones de usuario para equilibrar la carga del servidor de aplicaciones.	
63	Utilice un servicio de entrega de correo electrónico basado en la nube.	
64	Utilice un servicio de registro basado en la nube.	
65	Utilice una herramienta de monitoreo de usuarios en tiempo real.	
66	Test Driven Development (TDD): los desarrolladores primero escriben un caso de prueba automatizado antes de escribir el código de producción real.	
67	Behavioral Driven Development (BDD): similar a TDD, la prueba es una descripción del comportamiento de un programa.	
68	Contratar empleados con experiencia en técnicas de medición como el modelado de hilos y el análisis de riesgos.	
69	Reconsiderar la forma tradicional de medir el desempeño de los empleados para el personal de desarrollo y operaciones, porque conduce a la fricción. El personal de desarrollo se mide en función de las características entregadas. Esto fomenta un alto ritmo de entrega. El personal de operaciones se mide en base a la estabilidad del sistema. Esto fomenta un ritmo de entrega más lento. Esta situación paradójica conduce a mucha fricción.	[36]
70	Para facilitar el intercambio, los desarrolladores y las operaciones deben tratar de hacer que su documentación sea comprensible para ambas partes. Contratación de empleados que tienen experiencia tanto en desarrollo como en operaciones facilita esto.	
71	Lo que también facilita el intercambio de información entre el personal de desarrollo y de operaciones es cuando los desarrolladores obtienen requisitos con respecto a las operaciones mediante el estudio de varios recursos.	
72	Tanto los requisitos del producto como del proceso de operación deben considerarse en el proceso de ingeniería de requisitos.	
73	Utilizar diversos servicios de terceros ofrecidos mediante la computación en la nube, en lugar de las bibliotecas convencionales.	
74	La comunicación con servicios de terceros se puede probar mediante las operaciones dirigidas por comportamiento.	
75	Monitorización de usuarios en tiempo real es un patrón que se puede usar para detectar los problemas a los que se enfrenta el usuario final en una etapa temprana.	
76	Realizar cambios estructurales para facilitar la colaboración en toda la organización.	
77	Combinar DevOps con CMMI e ITIL.	
78	DevOps se puede incorporar a los métodos existentes, mientras que también hay modelos de procesos empresariales más grandes que incluyen DevOps, como Disciplined Agile Delivery.	
79	Creación de un grupo de apoyo central que pueda dar recomendaciones y consejos con respecto a la transformación de DevOps.	
80	Realizar procedimientos de prueba lo antes posible en el proceso de desarrollo.	
81	Automatizar las pruebas por sí mismos.	
82	Construir una medición común del progreso y el rendimiento, de modo que, en última instancia, todos los miembros del equipo de DevOps sigan el mismo panel de control de medidas de desempeño.	[68]

83	Invocar al servidor de integración continua cada vez que alguien hace commit, para que se construya la aplicación y se generen los artefactos (guardados y reusados después), si falla notifica el programador.	[63]
84	Testing automatizado.	
85	Testing exploratorio.	
86	Desplegar a producción.	
87	Detectar impedimentos al flujo de desarrollo de software.	[35]
88	Establecer el nivel existente de madurez de DevOps.	
89	Obtener la lista priorizada de prácticas de DevOps.	
90	Identificar las herramientas existentes en la organización.	
91	Elegir objeto de adopción de DevOps.	
92	Identificar las métricas relacionadas al objeto de adopción de DevOps.	
93	Procesar las primeras fases de la adopción de DevOps.	
94	Iniciar la siguiente fase de adopción de DevOps.	[40]
95	Construir la aplicación.	
96	Realizar pruebas unitarias.	
97	Empaquetar la aplicación.	
98	Configurar ambiente de pruebas.	
99	Desplegar binarios a ambiente de pruebas.	
100	Configurar ambiente de producción.	[37]
101	Desplegar binarios a ambiente de producción	
102	Compilación automatizada para detectar errores de integración lo más pronto.	
103	Realizar pruebas.	
104	Crear una instancia de una tubería de CD.	
105	Realizar mediciones en la tubería.	
106	Aprender de dichas mediciones para mejorar aún más la tubería.	

Acrónimos utilizados: **No.:** Número, **Ref.:**Referencia.

### 3.2.3.Artefactos identificados

Usando la definición de *artefacto* descrita en [72]: “es el conjunto de componentes a desarrollar, entregar y mantener en un proyecto. Los artefactos pueden ser de entrada o de salida; obligatorios u opcionales”. Se encontró un solo estudio [35] que propone un proceso de adopción de DevOps de forma gradual, cuyo nivel de detalle llega hasta la descripción de artefactos, los cuales son: (i) lista de categorías de impedimentos para la adopción de DevOps, (ii) modelo de madurez de DevOps, (iii) lista de herramientas que soportan prácticas de DevOps y (iv) lista de herramientas que ya esté usando la empresa de desarrollo de software que se encuentra en el proceso de adopción.

### 3.2.4.Roles identificados

Según la definición de *rol* en [72]: “describe un conjunto o grupo de responsabilidades, deberes y habilidades requeridas para realizar una actividad específica”. Se encontraron un total de 16 roles entre los que se incluyen roles pertenecientes a metodologías o marcos de trabajo tradicional en el proceso de desarrollo de software con algunas variaciones para adoptar prácticas de DevOps, y también se encontraron roles descritos en marcos de trabajo ágiles como Scrum. En la Tabla 3-4 se presentan los roles encontrados, junto a su descripción, origen (desarrollo de software tradicional o ágil) y referencia.



Tabla 3-4. Roles encontrados en los estudios primarios.

No.	Rol	Descripción	Origen	Ref.
1	Equipo DevOps	Se hace cargo de la aplicación (sistema) durante toda su vida útil, es decir, durante el desarrollo y la operación. Compuesto por ingenieros DevOps (expertos en el tema entrevistados a través de una encuesta).	DSTAD	[64]
2	Gerente de desarrollo de ND software con conocimientos en DevOpS	ND	DSTAD	[32]
3	Desarrollador con ND enfoque DevOps	ND	DSTAD	
4	Analista de pruebas con ND enfoque DevOps	ND	DSTAD	
5	Administrador de sistema (equipo de operaciones de TI)	ND	DST	
6	Product Owner	Desempeña un doble papel, como representante de las necesidades del cliente, pero también ofrecen un rol operativo real que vincula el negocio con la gestión del proyecto. Optimizar el valor de lo que producen los equipos de desarrollo	Scrum	
7	Team Manager	Incluyen diferentes profesiones: líder del equipo de desarrollo / operación o supervisión, y gerente de integración de calificación. Los PM definen y gestionan un proyecto de TI desde la concepción hasta la entrega para obtener un resultado óptimo para los requisitos del cliente de calidad, rendimiento, costo, tiempo y seguridad. y los evaluadores son responsables de la producción, operaciones, monitoreo de incidentes, soporte y soporte al usuario.	DST	[68]
8	Arquitecto	Participa en el desarrollo de archivos arquitectónicos y en la producción de aplicaciones.	DST	
9	Ingeniero de producción	Realiza las pruebas como Operaciones, realiza las tareas de supervisión. Para un Ingeniero de Producción o un Administrador de Sistemas, las habilidades técnicas / duras incluirían el uso de herramientas específicas para monitoreo y análisis de registros.	DST	
10	Gerente de lanzamiento	Responsable de los procesos de implementación	DST	
11	Ingeniero / equipo / departamento DevOps	ND	DSTAD	[61]
12	Project Leader / Team Leader	Tienen labores semejantes al Scrum Master	DST	[40]
13	Desarrollador	ND	DST	
14	Cliente	ND	DST	
15	Program Manager	ND	DST	
16	Release Manager	ND	DST	

Acrónimos utilizados: **DST**: Desarrollo de software tradicional, **DSTAD**: Desarrollo de software tradicional adaptado a DevOps, **Ref.**: referencia, **ND**: no disponible, **No.**: Número

### 3.2.5. Herramientas identificadas

De acuerdo con [72], una *herramienta* se puede definir como: “aquello que automatiza la ejecución de determinadas actividades” y se identificaron un total de 60 herramientas tecnológicas que apoyan procesos u actividades involucradas con la adopción de DevOps. En la Tabla 3-5 se muestra el área o proceso en la que se categorizó la herramienta y el respectivo nombre de las herramientas.

Tabla 3-5. Herramientas tecnológicas que soportan áreas / procesos de DevOps.

No.	Área / Proceso	Herramienta
1	Gestión de control de fuente (Repositorio)	Github, Bitbucket, GitLab, Mercurial, Apache Subversion, SonarQube, Maven
2	Integración continua y orquestación	Codship, Travis CI, Jenkins, Atlassian bamboo, Circle CI, TeamCity, Rancher, Gitflow
3	IaaS/PaaS	Heroku
4	Infraestructura como código	Amazon cloudformation
5	Monitoreo	Nagios, New Relic, AWS CloudWatch, Apache Kafka, Zabbix, Splunk, AppDynamics, Bugzilla, Track & TesTrack, MantisBT, Assembla, Kinesis
6	Gestión y manejos de bases de datos	MongoDB, DBMaestro, LiquiBase, RedGate
7	Registro/Seguridad	Loggly, Papertrail
8	Testing	Cucumber, Junit, Selenium, TestComplete, Jmeter
9	Colaboración	Slack, HipChat, PagerDuty
10	Containerización	Docker, Mercurial, Bitbucket, Rocker, Vagrant
11	Gestión de configuración	Docker, Puppet, Ansible, Cheff, Vagrant, SaltStack
12	Intercambio de conocimientos	Crowdbase, Nuclion, Confluence; Trello
13	Planificación	Clarizen, Confluence, Asana

Acrónimos utilizados: **No.:** Número.

### 3.3. Resultados obtenidos

Usando como base el análisis de los documentos seleccionados como estudios primarios y la ontología de referencia para la armonización de modelos de referencia de procesos PrMo presentada en [72], fue posible categorizar los elementos de proceso. En la Tabla 3-6 se muestra un resumen de la cantidad de procesos, actividades, productos y recursos encontrados. Adicionalmente, en la tabla se presenta la columna “Otro” para indicar la cantidad de elementos encontrados en cada estudio que no fueron posible categorizarlos de acuerdo con las definiciones presentes en la ontología.

Tabla 3-6. Cantidad de elementos de proceso encontrados por estudio primario.

No.	Ref.	Elemento de proceso					
		Proceso	Actividad	Producto	Recurso		Otro
					Rol	Herramienta	
1	[61]	ND	ND	ND	3	ND	ND
2	[64]	ND	ND	ND	ND	ND	ND
3	[41]	3	ND	ND	1	1	ND
4	[30]	3	11	ND	ND	19	2
5	[34]	ND	24	ND	ND	ND	4
6	[63]	1	2	ND	ND	2	ND
7	[32]	ND	ND	ND	4	4	ND
8	[36]	2	19	ND	ND	2	1
9	[40]	1	ND	ND	5	1	ND
10	[35]	ND	ND	4	ND	45	4
11	[67]	3	2	ND	ND	5	ND
12	[68]	ND	ND	ND	5	ND	ND
13	[66]	ND	ND	ND	ND	14	ND
14	[70]	3	ND	ND	ND	29	ND

Acrónimos utilizados: **No.:** número, **Ref:** referencia, **ND:** no disponible.

# Capítulo 4. DevOps Ontology

En este capítulo se presenta la ontología denominada DevOps Ontology, la cual es una ontología de dominio que permite organizar el conocimiento relacionado con la adopción de DevOps en procesos de desarrollo de software de una manera genérica y formal, permitiendo que el conocimiento pueda ser reutilizado y compartido por los interesados en esta área. Asimismo, DevOps Ontology facilitará la evaluación del nivel de apropiación de DevOps por las organizaciones a partir de la identificación de las relaciones existentes entre los procesos de desarrollo de software, los principios de DevOps y valores de DevOps propuestos.

Teniendo en cuenta el estado del arte relacionado a las ontologías y trabajos existentes para soportar la adopción de prácticas en empresas de desarrollo de software (ver Sección 2.2), se establecieron los siguientes objetivos para la definición de DevOps Ontology: (i) averiguar e identificar sinónimos, homónimos, inconsistencias y conflictos en la terminología asociada a DevOps, (ii) integración de los conceptos encontrados a partir del análisis de literatura analizada y su clasificación (ver Capítulo 3). A continuación, se describe DevOps Ontology en términos de: (i) la metodología utilizada para su construcción, (ii) el propósito de la ontología, (iii) preguntas de competencia, (iv) integración con otras ontologías, (v) representación gráfica, (vi) conceptos y relaciones definidas y (vii) su evaluación.

## 4.1. Metodología

Después de analizar diferentes trabajos como: Gestión del conocimiento basada en ontologías [73], METHONTOLOGY [74], Un enfoque de traducción a las especificaciones de ontologías portátiles [75], Modelado de procesos de negocio basado en la ontología y la lógica de primer orden [76] y REFSENO: un formalismo de representación para ontologías de ingeniería de software [77], se decidió utilizar REFSENO para el desarrollo de DevOps Ontology. Esta selección tuvo en cuenta los siguientes criterios: (i) REFSENO fue diseñada específicamente para la definición de ontologías en el área del desarrollo de software por medio de constructores que permiten definir conceptos, atributos y relaciones, (ii) REFSENO es una adaptación de METHONTOLOGY, la cual es ampliamente utilizada para la definición de ontologías en diferentes contextos y (iii) la forma de representación propuesta en REFSENO es menos compleja lo cual favorece su interpretación por personas que no están familiarizadas con la lógica de predicados de primer orden o similares. En la Sección 2.1.5 del marco teórico se detalla mejor los conceptos relacionados a la definición de ontologías y los formalismos asociados.

## 4.2. Propósito

El propósito de DevOps Ontology es permitir a los interesados en la industria del desarrollo de software conocer los conceptos y relaciones que deberían estar presentes en un proceso de adopción de DevOps sin importar el marco de trabajo base o infraestructura tecnológica a implementar. DevOps Ontology puede ser usada desde las siguientes perspectivas:

- Académica: porque todos los términos y relaciones definidas en la ontología pueden ser usados por los interesados en aprender o investigar el tema de adopción de DevOps.
- Industrial: ya que las empresas de desarrollo de software tendrán un instrumento para instanciar sus procesos de adopción, que les permitirá determinar los conceptos y relaciones presentes en el contexto de DevOps. Asimismo, sirve como base para la construcción de un instrumento de valoración y evaluación de adopción de DevOps.

### 4.3. Preguntas de competencia

DevOps Ontology permite responder a las preguntas de competencia (PC) presentes en la Tabla 4-1.

Tabla 4-1. Preguntas de competencia definidas para DevOps Ontology.

<b>Id.</b>	<b>Preguntas de competencia</b>
PC1	¿Cuál es la relación entre principios de DevOps y prácticas de DevOps?
PC2	¿Cuáles son los principales elementos que componen una práctica de DevOps?
PC3	¿Cuál es la relación entre productos generados y las prácticas de DevOps?
PC4	¿Cuál es la relación entre los productos generados y los principios de DevOps?
PC5	¿Cuál es la relación entre las tareas y los roles de DevOps?
PC6	¿Cuál es la relación entre productos y roles de DevOps?
Acronimos utilizados: <b>Id.</b> : Identificador	

### 4.4. Integración de DevOps Ontology con otras ontologías

Considerando que DevOps Ontology es un instrumento para instanciar procesos de adopción de DevOps en empresas de desarrollo de software, es necesario incluir conceptos relacionados con procesos de software y medición de software. En este sentido, se realizó una integración con la ontología “Ontology of Process-reference Models (PrMO)” propuesta en [72], que clarifica y establece los elementos principales en enfoques basados en procesos. También se integró con algunos conceptos de la sub ontología “Software Measurement Ontology (SMO)” presente en [78]. SMO establece y clarifica elementos clave en la definición de medidas de software y terminología relacionada con las acciones de medición de software.

## 4.5. Representación gráfica

En la Figura 4-1 se puede observar la representación gráfica de DevOps Ontology, sus términos y relaciones existentes con las ontologías PrMO y SMO usando la representación de UML (lenguaje de modelado unificado)

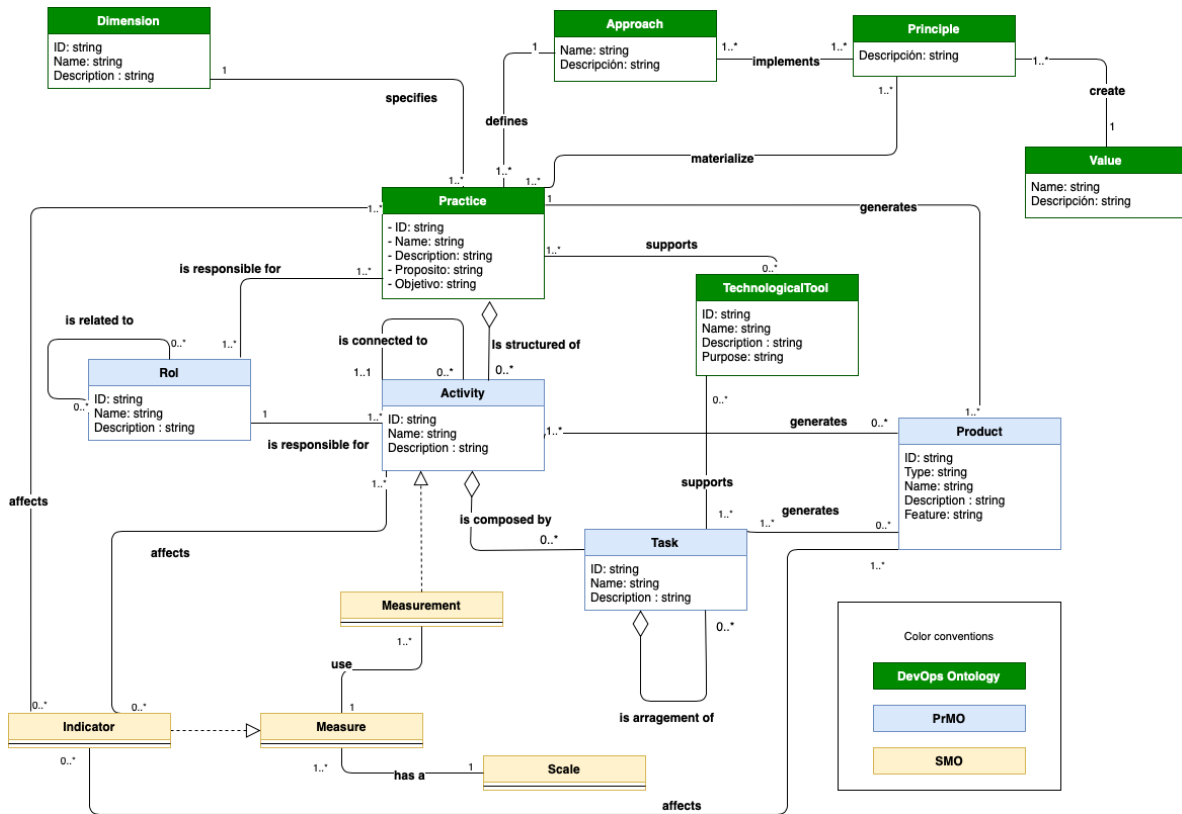


Figura 4-1. Representación gráfica de DevOps Ontology usando UML.

## 4.6. Conceptos y relaciones de DevOps Ontology

En la Tabla 4-2 se presenta cada una de las definiciones y conceptos incluidos en la ontología organizados de la siguiente manera: la primera columna presenta el concepto usado en la ontología, la segunda columna muestra el súper concepto al que pertenece, la tercera columna describe la definición del término en la ontología, y la cuarta columna muestra la fuente de donde el término fue adoptado o adaptado. Adicionalmente, la Tabla 4-3 presenta las relaciones existentes entre los conceptos de la Tabla 4-2. Las descripciones presentadas en la cuarta columna de la Tabla 4-2 puede ser una de las siguientes:

- **Tomado de [fuente]**, si el término ha sido definido en otro trabajo y es clave para la definición usada en la ontología y dicho término no ha sido modificado o adaptado.
- **Definido desde [fuente]**, si el término no fue definido en la fuente explícitamente, pero la información encontrada en ella permite establecer la definición usada en esta ontología.

- **Citado de [fuente]**, si el término ha sido citado en la fuente y esta no es la fuente original pero dicho término no ha sido modificado.
- **Fuente propia**, si el término ha sido definido en esta tesis o se le ha dado una definición particular para esta ontología.

Tabla 4-2. Glosario de conceptos de DevOps Ontology.

Término	Súper Concepto	Definición	Fuente
Activity	Concept	Comprende un conjunto de tareas o acciones que se utilizan para producir y mantener dispositivos, así como para lograr los objetivos del proceso. La actividad incluye los procedimientos, estándares, políticas y objetivos para crear y modificar un conjunto de productos de trabajo.	Tomado de [72]
Approach	Concepto	Hace referencia a los enfoques de desarrollo de software que definen un conjunto practicas con las que buscan fomentar el desarrollo de software con calidad	Fuente propia
Dimension	Concept	Áreas empresariales en las que se agrupan prácticas y procesos.	Definido desde [33]
Indicator	Measure	Una medida que se deriva de otras medidas que utilizan un modelo de análisis como enfoque de medición.	Tomado de [78]
Measure	Concept	El enfoque de medición definido y la escala de medición.	Tomado de [78]
Measurement	Concept	Conjunto de operaciones que tienen por objeto determinar el valor de un resultado de medición, para un atributo dado de una entidad, utilizando un enfoque de medición	Tomado de [78]
Metric	Concept	Permiten medir el éxito del proceso de adopción.	Tomado de [35]
Value	Concept	Cualidades y/o fundamentos que deben estar presente en los integrantes de un equipo de trabajo. colaboración, automatización, medición y seguimiento.	Fuente propia
Practice	Concept	Las prácticas materializan las ideas y valores fundamentales que caracterizan a DevOps en las actividades de desarrollo y operaciones diarias.	Tomado de [27]
Principle	Concept	Base de ideales, fundamentos o reglas desde las cuales surgen las ideologías.	Definido desde [27][33]
Product	Concept	El conjunto de artefactos que se desarrollarán, entregarán y mantendrán en un proyecto se denomina producto. Los productos pueden ser del tipo de entrada o salida; obligatorio u opcional. En la mayoría de los casos, los productos son artefactos tangibles consumidos, producidos o modificados por tareas.	Tomado de [72]
Role	Concept	Describe un conjunto o grupo de responsabilidades, deberes y habilidades requeridas para realizar una actividad específica.	Tomado de [72]
Scale	Concept	Un conjunto de valores con propiedades definidas.	Tomado de [78]
Task	Concept	Elemento de proceso que define el trabajo realizado por roles. Una tarea está asociada con los productos de entrada y salida.	Tomado de [72]
Tool	Resource	Las herramientas son una parte importante del enfoque DevOps. Están estrechamente relacionados con diferentes prácticas de DevOps, como la automatización, el monitoreo o la configuración integrada, y ayudan a introducir estas prácticas.	Tomado de [35]

Tabla 4-3. Relaciones de DevOps Ontology.

Nombre	Conceptos	Descripción
<i>affects</i>	Activity - Indicator	Una actividad puede afectar o no uno o varios indicadores. Un indicador puede ser afectado por una o varias actividades.
<i>affects</i>	Practice - Indicator	Una práctica puede afectar o no uno o varios indicadores. Un indicador puede ser afectado por una o varias prácticas.
<i>affects</i>	Product - Indicator	Un producto puede afectar o no uno o varios indicadores. Un indicador puede ser afectado por uno o varios productos.
<i>defines</i>	Policy - practice	Una política define una o varias prácticas. Una práctica es definida por una o varias políticas.
<i>generates</i>	Practice - Product	Una práctica genera uno o varios productos. Un producto es generado por una práctica.
<i>generates</i>	Activity - Product	Una actividad genera o no uno o varios productos. Un producto es generado por una o varias actividades.
<i>generates</i>	Task - Product	Una tarea genera o no uno o varios productos. Un producto es generado por una o varias tareas.
<i>has</i>	Scale - Measure	Una medida tiene una escala. Una escala sirve para definir una o más medidas.
<i>is Arrangement of</i>	Task - Task	Una tarea es un arreglo o no de una o múltiples tareas.
<i>is associated to</i>	Policy - Indicator	Una política está asociada con uno o varios indicadores. Un indicador está asociado o no con una o varias políticas.
<i>is composed of</i>	Activity - Task	Una actividad está compuesta o no por una o varias tareas. Una tarea hace parte de una actividad.
<i>is connected to</i>	Activity - Activity	Una actividad está conectada o no con una o varias actividades.
<i>is related to</i>	Rol - Rol	Un rol está relacionado o no con uno o varios roles.
<i>is responsible for</i>	Rol - Practice	Un rol es responsable de una o varias prácticas. Una práctica es responsabilidad de uno o varios roles.
<i>is responsible for</i>	Rol - Activity	Un rol es responsable de una o varias actividades. Una actividad es responsabilidad de un rol.
<i>is structured of</i>	Practice - Activity	Una práctica está estructurada o no por una o varias actividades. Una actividad es parte de una práctica.
<i>materialize</i>	Practice - Principle	Una práctica materializa uno o varios principios. Un principio es materializado por una o varias prácticas.
<i>specifies</i>	Dimension - Practice	Una dimensión especifica una o varias prácticas. Una práctica es especificada por una dimensión.
<i>supports</i>	Technological Tool - Task, practice	Una herramienta tecnológica soporta una o varias tareas. Una tarea es soportada o no por una o varias herramientas tecnológicas.
<i>use</i>	Measure - Measurement	Cada medición usa una medida. Una medida puede ser usada en una o varias mediciones.

## 4.7. Discusión de algunos conceptos y sus relaciones

Algunos de los términos usados en DevOps Ontology merecen una discusión especial. Como tal, en esta Sección se amplía la definición y análisis de algunos términos como: principios, prácticas y dimensión.

**Valor:** Corresponde a los fundamentos sobre los que se origina la esencia de DevOps, por ejemplo: los elementos más comunes encontrados en la literatura sugieren que la *colaboración* [33] debe existir como valor entre las diferentes áreas involucradas en el proceso de desarrollo de software, además, que esta debe ser reforzada mediante: el intercambio de información, la ampliación de las habilidades y el cambio de responsabilidades entre los equipos de trabajo, en segundo lugar, encontramos el valor de la *automatización* [33], elemento crucial para permitir que diferentes tipos de tareas sean provistas y desplegadas de forma repetitiva y rápida, en tercer lugar el valor de la *medición*

[33], la cual es importante durante todo el proceso ya que por medio de la incorporación de métricas se facilita el incremento de la eficiencia en el desarrollo de productos y finalmente el valor del *monitoreo* [33] que ayuda a determinar la asignación apropiada de recursos con el objetivo de detectar, reportar y corregir problemas que pueden ocurrir durante o después de cualquier tipo de actualización del sistema.

**Práctica:** Permite materializar los principios por medio de la realización de acciones puntuales siendo las prácticas de integración, construcción, despliegue, configuración y monitoreo continuo las más visibilizadas en la mayor parte de los trabajos encontrados [38].

**Dimensión:** Hace referencia a todas aquellas áreas empresariales donde se puede clasificar todo el conjunto de prácticas y sus elementos relacionados; entre las que encontramos: los procesos, las personas, la cultura y la tecnología [35].

## 4.8. Evaluación de DevOps Ontology

De acuerdo con [79], la evaluación de una ontología se debe enfocar principalmente en la evaluación de tres elementos por parte del usuario relacionadas con el rendimiento y la definición correcta, estos elementos son: *consistencia*, *completitud* y *concisión*. Por ejemplo, la consistencia en la evaluación se presenta si es posible inferir conocimiento desde la ontología, la completitud sólo puede evaluarse probando la inexistencia de incompletitud determinando si las definiciones individuales están bien establecidas y si todo lo que se supone debe estar presente en la ontología está presente o puede inferirse. Adicionalmente, la ontología será concisa si no es redundante y no tienen definiciones innecesarias.

La ontología resumida y presentada en este documento se implementó en el lenguaje de ontología OWL [80] utilizando Protégé [81] y el razonador HermiT [82] para evaluar la consistencia de la estructura de la ontología (ontología completa disponible en: <https://stanford.io/38TisW7>). Una vez finalizado el análisis, se pudo comprobar que DevOps Ontology no presenta inconsistencias en su estructura. Posteriormente, DevOps Ontology fue aplicado en tres casos de implementación que resultaron exitosos y que se mencionan a continuación: (i) una validación teórica en la cual se evidenció cómo los conceptos y relaciones asociadas a la ontología son usados para describir las particularidades que se podrían encontrar en la descripción de un proceso de adopción de DevOps, (ii) la instanciación de las prácticas de integración y despliegue continuo propuesta por la empresa GitLab y (iii) el desarrollo de una aplicación móvil que realiza consultas a DevOps Ontology utilizando el protocolo y lenguaje de Consulta RDF (SPARQL) para visualizar la relación entre prácticas y principios de DevOps. Además, se evaluó la capacidad de DevOps Ontology para responder las preguntas de competencia utilizando SPARQL. Finalmente, se analizaron los aportes de la ontología propuesta a la comunidad académica y la industria mediante su comparación con ontologías existentes para DevOps.

### 4.8.1. Evaluación teórica de DevOps Ontology

El primer caso de implementación involucra la narración de un ejemplo teórico de aplicación de algunos de los conceptos incluidos en DevOps Ontology. Los conceptos utilizados



aparecen en cursiva. El modelo de implementación de las prácticas de integración y despliegue continuo de la empresa Gitlab [83] da vida al *valor* de DevOps llamado automatización, que determina los *principios* de entrega temprana y continua, automatización de procesos y despliegue en el menor tiempo posible. Estos *principios* cobran vida con las *prácticas* de integración y despliegue continuos, definidas en el pipeline de procesos de la empresa, que se categorizan en la *dimensión* de procesos y tecnología. A partir de la *práctica* de la integración continua se desarrollan las *actividades* de codificación, revisión, aprobación, automatización de la compilación del código, automatización de pruebas y automatización del despliegue. Estas dos últimas *actividades* también forman parte de la *práctica* de despliegue continuo y generan como *producto* los respectivos despliegues para revisión y despliegue para producción. En cuanto a la *actividad* de codificación, se describen las siguientes *tareas* clave que se realizan directamente a través de la *herramienta tecnológica* Gitlab y que son responsabilidad de quien asume el *rol* de desarrollador: creación de nuevas ramas, envío de mejoras al código, envío de código cambios y operaciones de merge.

#### **4.8.2. Soportando la instanciación de las prácticas de integración y despliegue continuo propuestas por GitLab**

En el segundo caso de implementación, se realizó una instanciación de los elementos propuestos por GitLab para las prácticas de integración y despliegue continuo. Para esta instanciación, se utilizaron los siguientes conceptos de DevOps Ontology: *valor*, *principio*, *práctica*, *dimensión*, *actividad*, *herramienta tecnológica*, *rol*, *tarea* y *producto*. En la Figura 4-2 se muestra la instancia creada por DevOps Ontology, donde se puede ver que las prácticas de integración y despliegue continuo se crean a partir del valor DevOps de automatización e involucran las dimensiones de tecnología y procesos. Adicionalmente, a partir de la práctica de la integración continua, se estructuran tres actividades principales: codificación, automatización de compilación y automatización de pruebas. Dentro de la actividad de codificación, que es responsabilidad de los desarrolladores, se detallan las tareas de creación de nuevas ramas, envío de correcciones de código, envío de modificaciones de código y procesos de merge en los repositorios del proyecto.

Por el lado de la práctica del despliegue continuo, se estructuran actividades de automatización de pruebas y automatización de despliegue más exhaustivas, generando como salida versiones funcionales desplegadas en entornos de revisión o producción, según las necesidades de la organización. En esta instanciación se puede ver que DevOps Ontology proporciona los conceptos y relaciones necesarios para representar los elementos más importantes de las prácticas de integración y despliegue continuo propuestas por Gitlab.

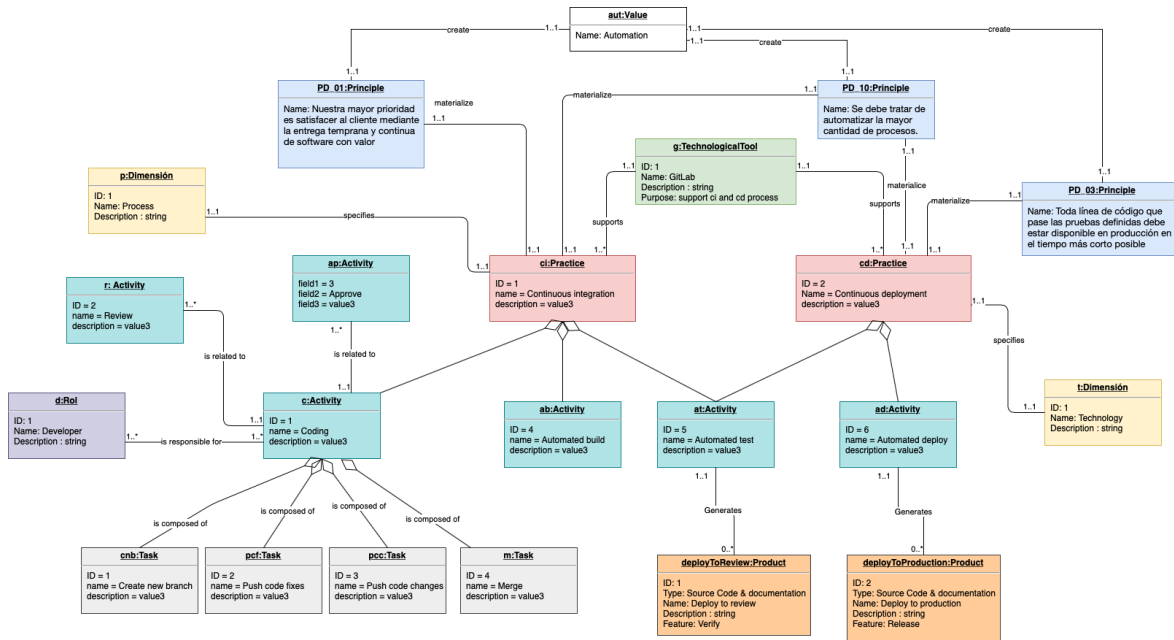


Figura 4-2. Instanciación de la práctica de integración y despliegue continuo propuesto por GitLab usando DevOps Ontology.

### 4.8.3. Soportando la consulta a DevOps Ontology a través de una aplicación móvil

El tercer caso de implementación está relacionado con el desarrollo de una aplicación móvil que extrae la información desde la ontología generada y facilita a los usuarios la visualización de los conceptos y relaciones definidas y que dan soporte a cada una de las consultas de competencia definidas. En la Figura 4-3 se pueden observar algunas de las pantallas de la aplicación móvil, como el menú de navegación que permite interactuar con los conceptos y relaciones definidas en la ontología, la pantalla de inicio, y la descripción detallada de conceptos. La información que se presenta es el resultado del uso de consultas SPARQL enviadas a la DevOps Ontology.

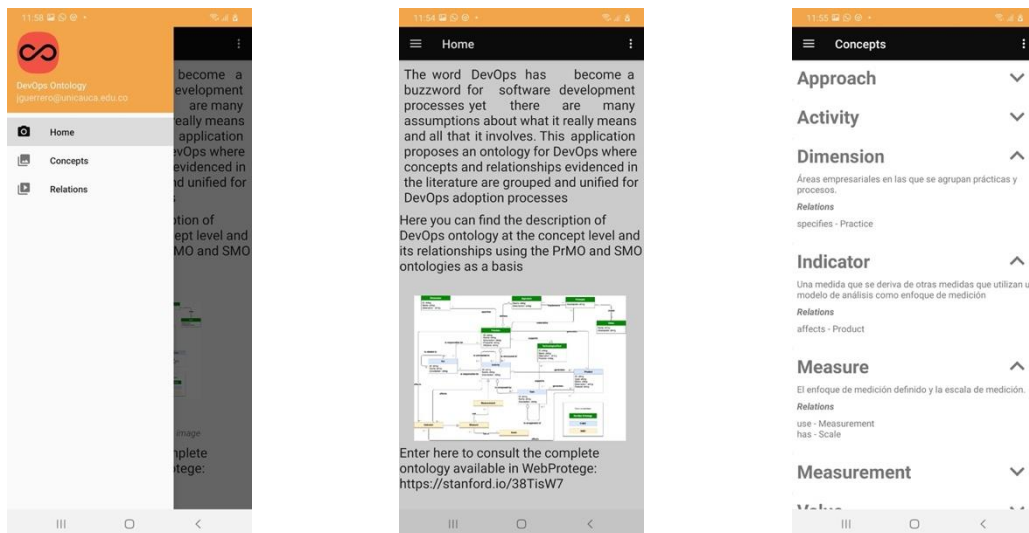


Figura 4-3. Visualización de elementos de DevOps Ontology usando una aplicación móvil.

#### 4.8.4. Respondiendo las preguntas de competencia usando SPARQL

Para evaluar el cumplimiento del propósito de la ontología, se realizó la traducción de cada una de las preguntas de competencia en el lenguaje de consultas SPARQL. Los resultados permitieron comprobar que DevOps Ontology es capaz de responder todas las preguntas de competencia propuestas en la parte inicial del desarrollo. En la Tabla 4-4 se pueden observar cada una de las preguntas de competencia expresadas en SPARQL y los resultados obtenidos. A partir de las preguntas de competencia definidas, se encontraron las siguientes respuestas desde la ontología: (i) las prácticas de DevOps materializan principios; (ii) los elementos relacionados más importantes a una práctica de DevOps son los roles, principios, indicadores, herramientas tecnológicas y dimensiones; (iii) las dimensiones especifican prácticas las cuales a su vez generan productos; (iv) un producto puede ser generado por una práctica, las prácticas están materializando principios; (v) los roles son responsables de las actividades y las actividades están compuestas por tareas; y (vi) los productos se generan desde la ejecución de prácticas, actividades o tareas que son responsabilidad de un rol particular.

Tabla 4-4. Preguntas de competencia expresadas en SPARQL.

IPC	Consulta en SPARQL	Resultado
PC1	<i>SELECT ?e1 ?rel WHERE { ?rel rdfs:domain ?e1 . ?rel rdfs:range devops:Principle . FILTER (?e1 IN (devops:Practice )) }</i>	<i>Practice materialize Principle</i>
PC2	<i>SELECT ?PracticeElement WHERE { ?r rdfs:domain devops:Practice . ?r rdfs:range ?PracticeElement }</i>	<i>Policy -Rol - Principle- Product – Indicator -TechnologicalTool – Activity - Dimension</i>
PC3	<i>SELECT ?e1 ?rel1 ?e2 ?rel2 WHERE { ?rel1 rdfs:domain ?e1 . ?rel1 rdfs:range ?e2. ?rel2 rdfs:domain ?e2. ?rel2 rdfs:range devops:Product. FILTER(?e1 IN (devops:Dimension))}</i>	<i>Dimension specifies Practice Practice generates Product</i>
PC4	<i>SELECT ?e1 ?rel1 ?e2 ?rel2 WHERE { ?rel1 rdfs:domain ?e1 . ?rel1 rdfs:range ?e2.</i>	<i>Product isGenerate By Practice Practice materialize Principle</i>
PC5	<i>SELECT ?e1 ?rel1 ?e2 ?rel2 WHERE { ?rel1 rdfs:domain ?e1 . ?rel1 rdfs:range ?e2. ?rel2 rdfs:domain ?e2. ?rel2 rdfs:range devops:Task. FILTER(?e1 IN (devops:Rol))}</i>	<i>Rol isResponsibleFor Activity Activity isComposedOf Task</i>
PC6	<i>SELECT ?e1 ?rel WHERE { ?rel rdfs:domain ?e1 . ?rel rdfs:range devops:Principle . FILTER (?e1 IN (devops:Practice )) }</i>	<i>Practice materialize Principle</i>

Acrónimos utilizados: **IPC**: Identificador de preguntas de competencia, **PCX**: Pregunta de competencia.

## 4.9. Comparación de ontologías y taxonomías propuestas en la literatura

En la Tabla 4-5 se presenta la comparación entre DevOps Ontology con las ontologías y taxonomías que fueron identificadas durante la revisión de la literatura. Para realizar esta tarea, se aplicaron los siguientes criterios de evaluación: (C1) La ontología puede ser aplicada a cualquier proceso de DevOps presente en una empresa, (C2) Los conceptos incluyen los principios y valores definidos por DevOps, (C3) puede ser instanciada e integrada a ontologías existentes, (C4) la ontología fue definida a través de un lenguaje formal, (C5) la ontología está disponible en un repositorio de acceso público.

De acuerdo con la comparación, se observa que las ontologías identificadas en la literatura no tienen el objetivo de soportar DevOps y están enfocados en procesos dedicados a aspectos relacionados a procesos de desarrollo, enfoques tradicionales y/o ágiles para el desarrollo de software o para la evaluación de elementos muy particulares de DevOps. Además, algunas de las ontologías no cuentan con acceso público que garanticen su disponibilidad y permitan realizar procesos de realimentación y evaluación.

Tabla 4-5. Comparación de ontologías.

Ontología	Ref.	Criterios				
		C1	C2	C3	C4	C5
Composable DevOps	[25]	X		X	X	
DevOps Ontology		X	X	X	X	X
Taxonomía	Ref.	Criterios				
		C1	C2	C3	C4	C5
Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process	[84]	X			X	
A taxonomy of software delivery performance profiles: Investigating the effects of DevOps practices	[85]	X	X		X	
Prioritization Based Taxonomy of DevOps Security Challenges Using PROMETHEE	[86]	X			X	
Prioritization Based Taxonomy of DevOps Challenges Using Fuzzy AHP Analysis	[87]	X			X	
An Empirical Taxonomy of DevOps in Practice	[88]		X	X		
DevOps Ontology		X	X	X	X	X

Acrónimos utilizados: **Ref:** Referencia.

## 4.10. Conclusiones y trabajo futuro

En este trabajo se presentó una ontología semi-formal para el dominio de los elementos involucrados en el proceso de adopción de DevOps en empresas de desarrollo de software llamada DevOps Ontology. La ontología propuesta establece un vocabulario que permite organizar parte del conocimiento encontrado en la literatura actual de forma genérica y formal, reduciendo la inconsistencia, integridad y ambigüedad de los conceptos y términos presentes en este dominio de investigación. Adicionalmente, DevOps Ontology ha sido definida teniendo en cuenta los siguientes aspectos: se usó el formalismo REFSENO que utiliza la representación en UML y se utilizó el lenguaje OWL para su construcción para

facilitar su uso por la comunidad académica y las organizaciones de la industria del software interesadas.

Por otra parte, el objetivo principal de DevOps Ontology es proporcionar una solución preliminar para la discusión de los términos, conceptos y relaciones identificadas en este dominio de investigación, el cual inicialmente ha sido aplicado en tres casos de implementación: (i) validación teórica, (ii) instanciación de las prácticas de integración y despliegue continuo propuesta por la empresa GitLab, una de las plataformas líderes en el mercado mundial para el soporte de este tipo de prácticas y (iii) la implementación de una aplicación móvil para visualizar la relación entre principios y prácticas a través de consultas SPARQL. Además, fue posible responder las preguntas de competencia que fueron establecidas en el propósito de la ontología a través del lenguaje de consulta SPARQL.

Los resultados obtenidos en el desarrollo de DevOps Ontology se utilizarán para abordar algunos trabajos futuros en el siguiente orden: (i) se espera evaluar y mejorar DevOps Ontology a través de su aplicación en estudios de caso en organizaciones dedicadas a la construcción de productos de software, (ii) utilizar la ontología como base conceptual para la definición de un modelo de referencia para la adopción y evaluación de DevOps en empresas de desarrollo de software y focalizar los esfuerzos en la (iii) automatización de la etapa de evaluación de la adopción de DevOps por medio de mecanismos de inferencia que usen la ontología y que permitan a auditores tener como insumo valoraciones previas que permitan cotejar resultados de evaluaciones manuales. Por tanto, como trabajo futuro, el siguiente paso de este proyecto de investigación se centrará en el desarrollo de algoritmos que permitan mejorar y ampliar la capacidad del proceso de evaluación mediante su automatización.

# Capítulo 5. Modelo de referencia

En este capítulo se presenta el modelo de referencia DevOps Model, el cual es un modelo para la adopción de DevOps en las empresas de desarrollo de software. El modelo de referencia se estructura a través de 4 elementos: (i) valores; (ii) principios; (iii) dimensiones y (iv) prácticas.

Los elementos definidos en DevOps Model surgieron a partir de la revisión sistemática de la literatura que se presenta en el apartado 2.2 y la creación de una ontología para procesos de adopción de DevOps llamada DevOps Ontology descrita en el capítulo Capítulo 4, la cual proporciona conceptos, definiciones y relaciones existentes alrededor del dominio del conocimiento relacionado con DevOps. Adicionalmente este modelo de referencia se complementa con un modelo de valoración que es descrito en detalle en la Sección 5.6. El resumen de los elementos que componen esta propuesta se puede observar en Figura 5-1.

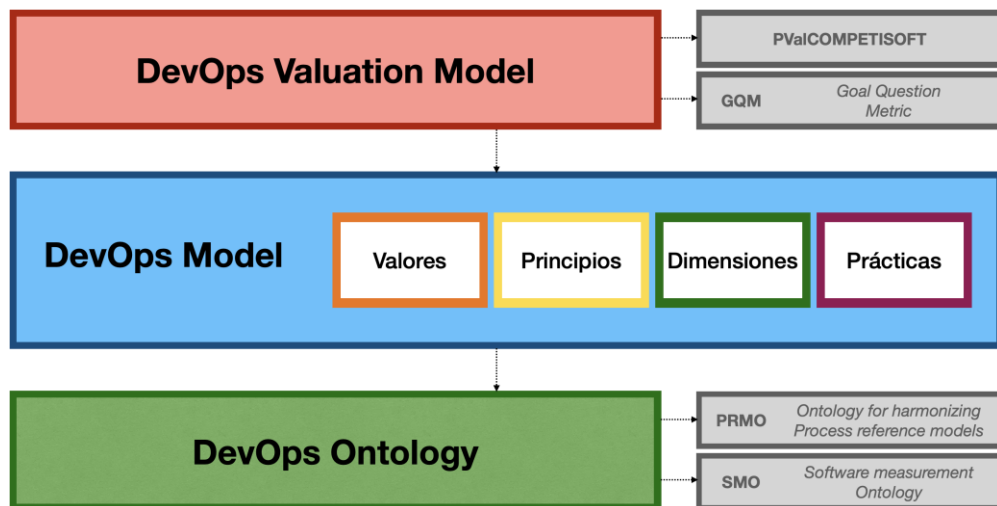


Figura 5-1. Resumen de los elementos de la propuesta.

El modelo de referencia es flexible, de modo que puede ser adaptado y posibilita la creación de procesos e indicadores de acuerdo con las características y necesidades de las organizaciones relacionadas con la industria del software que deseen adelantar procesos de adopción de DevOps. En la Figura 5-2 se observa un resumen de los elementos que componen a DevOps Model, como se puede observar, el centro del modelo está conformado por los cuatro valores propuestos para DevOps, luego, en la siguiente capa observamos los 12 principios adaptados para DevOps, luego, en la tercera capa encontramos las 4 dimensiones propuestas que categorizan el conjunto de prácticas fundamentales y complementarias que están en la última capa.

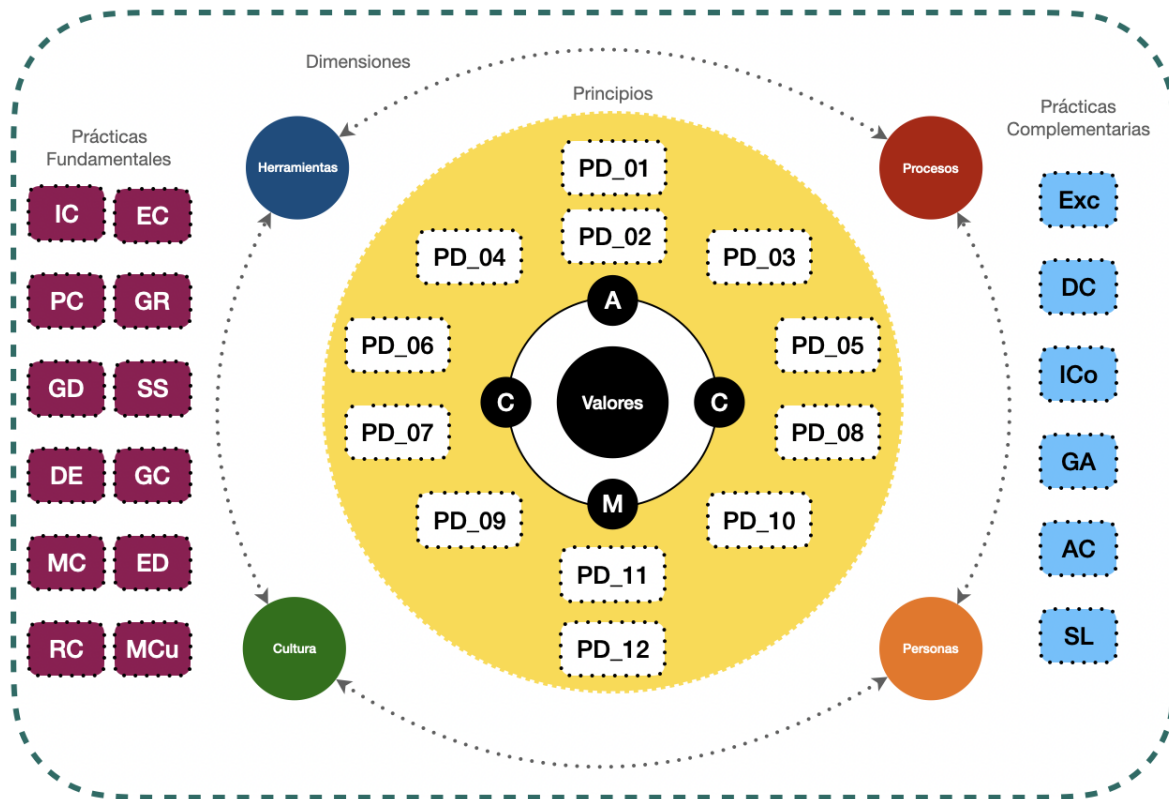


Figura 5-2. Elementos que conforman en DevOps Model.

En las siguientes Secciones, se describirán en detalle los elementos propuestos: valores, principios, dimensiones y prácticas que componen a DevOps Model.

## 5.1. Valores de DevOps

Las empresas de desarrollo de software; actualmente buscan en sus procesos acelerar la entrega de valor sin disminuir la calidad de los productos/servicios que desarrollan, por lo que se han venido incorporando marcos de trabajo asociados al agilísimo [89], por ejemplo: Scrum [90] y Scrumban [91], entre otros. Marcos de trabajo en los que encontramos como elemento clave los valores ágiles, los cuales representan los principales atributos que debe tener un proceso de desarrollo de software para ser considerado ágil de acuerdo a lo definido en el *Manifiesto Ágil* [92], siguiendo esta estructura; y con el objetivo de fomentar e informar la importancia de DevOps, se proponen cuatro (4) valores para DevOps; los cuales están basados en los valores ágiles [92], y se pueden considerar como la evolución natural de los estos y las necesidades actuales del mercado 20 años después de su definición.

El primer valor es la **automatización**; es indispensable priorizar la automatización de procesos sobre la realización de procesos manuales siempre que esto repercuta en mejoras al equipo de trabajo. Todo lo que se pueda automatizar se debe automatizar. El segundo valor es la **colaboración**; el trabajo en equipo es clave, aunque existan roles o áreas de trabajo diferentes, cualquier acción realizada por un individuo beneficiará o afectará a otro. El tercer valor es la **medición**; la mejora continua sólo es realizable cuando se generan datos continuos

que permitan tomar mejores decisiones. El cuarto valor es la **comunicación**; la comunicación y la transparencia es requerida en todos los niveles de la organización y con el cliente.

## 5.2. Principios de DevOps

Los principios son el segundo elemento definido en DevOps Model, estos establecen las reglas o normas de carácter general que deben orientar las acciones de las organizaciones de software en el marco de un proceso de desarrollo de software que contempla el uso de DevOps. La definición de los principios para DevOps propuestos tienen como base los principios definidos en el *Manifiesto Ágil* [92], los hallazgos encontrados en la categorización de elementos de proceso relacionados con DevOps que se presenta en el Capítulo 3 y la ontología desarrollada que se presenta en el Capítulo 4.

En la Tabla 5-1 se pueden observar la relación de los principios ágiles definidos en [92] y la actualización de los principios propuestos para DevOps Model. Aunque la mayoría de los principios fueron reescritos se puede observar la excepción en los principios PD\_01 y PD\_07; ya que para DevOps la construcción de software funcional que satisfaga las necesidades del cliente con entregas cada vez más tempranas y continuas; sigue siendo la principal medida de progreso.

Tabla 5-1. Principios propuestos en DevOps Model.

Id	Principio Ágil	Principio DevOps Model
PD_01	Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.	Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
PD_02	Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.	Todo cambio en los requisitos en cualquier etapa del desarrollo debe ser conocido y acordado en conjunto con representantes de todas las áreas de trabajo.
PD_03	Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible	Toda línea de código que pase las pruebas definidas debe estar disponible en producción en el tiempo más corto posible.
PD_04	Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.	Los responsables de negocio, desarrolladores e infraestructura trabajan juntos de forma cotidiana durante todo el proyecto.
PD_05	Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.	Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, brindarles herramientas que permitan automatizar sus procesos y confiarles la ejecución del trabajo.
PD_06	El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.	Siempre se debe buscar el medio de comunicación más eficiente y efectivo entre todos los miembros del equipo.
PD_07	El software que funciona es la medida principal de progreso.	El software que funciona es la medida principal de progreso.
PD_08	Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.	A mayor velocidad de entregas en producción, mayor velocidad en la recepción del feedback y posibilidad de mejora.
PD_09	La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.	El equipo debe tener el conocimiento y habilidades necesarias para adaptarse a los cambios rápidamente.







PD_10	La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.	Se debe tratar de automatizar la mayor cantidad de procesos.
PD_11	Las mejores arquitecturas, requisitos y diseños emergen de equipos auto organizados.	Se deben aprovechar las habilidades y visión de cada individuo en el equipo de trabajo para la construcción de mejores arquitecturas, requisitos y diseños.
PD_12	A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.	Todo aprendizaje se debe documentar y comunicar. Esto permite ajustar y perfeccionar las acciones del equipo.

Acrónimos utilizados: **Id**: Identificador; **PD\_XX**: Principio DevOps.

### 5.3. Dimensiones

Las dimensiones propuestas en DevOps Model, son los elementos que caracterizan cada una de las actividades, roles, productos, prácticas y herramientas requeridas para la implementación efectiva de los principios y valores propuestos para DevOps. La definición de las dimensiones tiene en cuenta las clasificaciones propuestas en otros trabajos así como [33], [63] y [35]. A continuación, en la Tabla 5-2 se presenta un identificador de dimensión, el nombre de la dimensión, su descripción y la simbología asociada para su asociación con otros elementos del modelo propuesto.

Tabla 5-2. Dimensiones propuestas en DevOps Model.

Id	Dimensión	Descripción	Simbología
DD_01	Herramientas	Esta dimensión relaciona las herramientas tecnológicas que dan soporte a los procesos, prácticas, tareas o productos desarrollados en el marco del desarrollo de software en todas sus fases bajo un enfoque de DevOps.	
DD_02	Procesos	Esta dimensión relaciona todo el conjunto de políticas, estructuras organizativas, procedimientos, propósitos, objetivos y productos de trabajo requeridos durante todo el ciclo de vida de un producto de software que se deben tener en cuenta para la adopción de DevOps.	
DD_03	Cultura	Esta dimensión tiene en cuenta todo lo relacionado con conocimientos, ideas, tradiciones y costumbres que caracterizan una organización, puntualmente una organización dedicada al desarrollo de productos en la industria del software.	
DD_04	Personas	Esta dimensión se refiere a todo el conjunto de características a nivel de los individuos que conforman un equipo de trabajo.	

Acrónimos utilizados: **Id**: Identificador; **DD\_#**: Dimensión DevOps.

### 5.4. Prácticas

DevOps Model sugiere 18 prácticas, de las cuales 12 se proponen como esenciales o fundamentales, es decir, se requiere de su implementación para cumplir con la mayoría de los principios definidos en el modelo. Además, sugiere 6 prácticas adicionales que se

presentan como complementarias, estas prácticas pueden mejorar los resultados que se pueden obtener con las prácticas fundamentales, pero requieren de una mayor inversión por parte de las organizaciones. Cada una de estas prácticas está agrupada en alguna o varias de las dimensiones de DevOps y se detallan con una descripción, propósito, artefactos esperados y dimensiones impactadas.

A continuación, se describen las prácticas fundamentales:

### 5.4.1. Prácticas fundamentales

Las organizaciones tienen características, políticas internas, recursos y objetivos diferentes, por lo que es importante establecer del conjunto prácticas disponibles; cuáles son vitales y claves cuando se aborda un proceso de adopción de DevOps; y además, cuáles podrían servir como complemento si se tienen las facilidades para implementarlas.

En la Figura 5-3 se puede visualizar cada una de las prácticas fundamentales propuestas con su respectivo identificador, nombre de la práctica y dimensiones asociadas.



Figura 5-3. Prácticas fundamentales propuestas.

En la Tabla 5-3 se presentan las prácticas fundamentales propuestas, cada una estructurada con su respectivo: identificador, nombre de la práctica, descripción, propósito y artefactos esperados.

Tabla 5-3. Prácticas fundamentales propuestas en DevOps Model.

Id	Detalle prácticas fundamentales
IC	<b>Nombre:</b> Integración continua
	<b>Descripción:</b> Práctica de desarrollo de software que requiere que los desarrolladores integren código en un repositorio compartido muchas veces al día [30]. Cada nueva versión del proyecto se verifica luego mediante una compilación automatizada, lo que permite a los equipos detectar errores de manera más fácil y rápida [30].
	<b>Propósito:</b> Proporcionar retroalimentación rápida para que; si se introduce un defecto en la base del código, este se pueda identificar y corregir lo antes posible.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Habilitar un sistema de control de versiones</li> <li>2. Acelerar el proceso de integración de código</li> <li>3. Detectar errores de integración de forma temprana</li> <li>4. Visibilizar constantemente de versiones compiladas para proceso de prueba, demostración o lanzamiento</li> </ol>
<b>Artefactos esperados:</b> Flujo de trabajo para integración continua, nueva actualización del producto software [93].	
EC	<b>Nombre:</b> Entrega continua

	<p><b>Descripción:</b> Práctica que permite que el código fuente desarrollado que haya pasado todas las pruebas de validación este listo para un entorno de preproducción tan pronto como sea posible.</p> <p><b>Propósito:</b> Producir software en ciclos cortos, asegurando que el software se pueda lanzar de manera confiable en cualquier momento.</p> <p><b>Objetivos Específicos:</b></p> <ol style="list-style-type: none"> <li>1. Implementar código de forma predecible</li> <li>2. Crear con bajo riesgo releases del producto ejecutados bajo demanda.</li> <li>3. Reducción de los tiempos para la creación y entrega de cambios incrementales en el software.</li> </ol> <p><b>Artefactos esperados:</b> Nueva actualización del producto software, flujo de trabajo para entrega continua [93].</p>
PC	<p><b>Nombre:</b> Pruebas continuas</p> <p><b>Descripción:</b> Esta práctica incorpora pruebas de código continuas, preprogramadas y automatizadas a medida que se escribe o actualiza el código de la aplicación. Estas pruebas pueden acelerar la entrega del código a producción.</p> <p><b>Propósito:</b> Evaluar la calidad del software en cada paso del proceso de entrega continua mediante pruebas tempranas y pruebas frecuentes.</p> <p><b>Objetivos Específicos:</b></p> <ol style="list-style-type: none"> <li>1. Habilitar ambiente de pruebas.</li> <li>2. Integrar pruebas a procesos de entrega y despliegue continuo.</li> <li>3. Analizar resultados de las pruebas.</li> <li>4. Enviar informes de defectos.</li> <li>5. Descubrir defectos de manera temprana.</li> </ol> <p><b>Artefactos esperados:</b> Pruebas automatizadas, informe de ejecución de pruebas.</p>
GR	<p><b>Nombre:</b> Gestión de requisitos</p> <p><b>Descripción:</b> Práctica encargada de apoyar la gestión de requisitos en herramientas para hacer más fácil rastrear, probar, analizar, visualizar y comunicar a las partes interesadas los requisitos de un proyecto.</p> <p><b>Propósito:</b> Operar con requisitos claros, realistas y acordados.</p> <p><b>Objetivos Específicos:</b></p> <ol style="list-style-type: none"> <li>1. Proceso de gestión de requisitos (obtenido, analizado, documentado y gestionado)</li> <li>2. Análisis de requerimientos</li> <li>3. Priorización de requisitos</li> <li>4. Trazabilidad de requisitos hacia elementos de trabajo</li> <li>5. Evaluar el impacto de los cambios</li> </ol> <p><b>Artefactos esperados:</b> Documentos de especificación de requisitos / product backlog</p>
GD	<p><b>Nombre:</b> Gestión de datos</p> <p><b>Descripción:</b> Práctica que se ocupa de seleccionar, obtener, mantener y usar los datos de forma segura y eficiente.</p> <p><b>Propósito:</b> Aumentar la velocidad de los cambios en las bases de datos, monitorear constantemente el impacto de los cambios en el proceso de DevOps y replicarlos al lugar donde se requieran.</p> <p><b>Objetivos Específicos:</b></p> <ol style="list-style-type: none"> <li>1. Incrementar la velocidad de entrega de cambios en la base de datos</li> <li>2. Reducir el riesgo de pérdida de datos durante el proceso de despliegue</li> <li>3. Sincronización de cambios en la aplicación y la base de datos</li> <li>4. Establecer sistemas de control de versiones de la base de datos</li> <li>5. Preservar y proteger los datos críticos de la empresa</li> </ol> <p><b>Artefactos esperados:</b> Base de datos, diccionario de datos, modelos de datos.</p>
SS	<p><b>Nombre:</b> Supervisión de la seguridad</p> <p><b>Descripción:</b> Práctica encargada de integrar las pruebas y los controles de seguridad en el control de calidad, las operaciones y el trabajo de desarrollo diarios.</p> <p><b>Propósito:</b> Mejorar la entrega de software y el rendimiento de la organización.</p> <p><b>Objetivos Específicos:</b></p> <ol style="list-style-type: none"> <li>1. Gestión de identidad y acceso</li> <li>2. Gestión de privilegios</li> <li>3. Hacer cumplir la política y la gobernanza</li> <li>4. Cortafuegos / gestión unificada de amenazas</li> <li>5. Gestión de vulnerabilidades en todo el flujo de trabajo de DevOps</li> </ol> <p><b>Artefactos esperados:</b> Políticas de seguridad de la información, control de acceso, claves y copias de seguridad.</p>

DE	<b>Nombre:</b> Dirección estratégica
	<b>Descripción:</b> Práctica encargada de movilizar personas y recursos hacia el uso de ambientes DevOps, hacer que las cosas ocurran sin sobrevalorar los beneficios que las herramientas pueden ofrecer a los equipos de trabajo, generar acción al interior de la organización.
	<b>Propósito:</b> Alinear todos los esfuerzos de la organización en torno a un objetivo.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Formular e implantar la estrategia</li> <li>2. Desarrollo y movilización de los recursos y capacidades</li> <li>3. Coordinación de diferentes recursos para que estén disponibles</li> <li>4. Establecer la visión, misión y filosofía</li> <li>5. Creación de valor</li> </ol>
	<b>Artefactos esperados:</b> Objetos estratégicos, misión y visión, plan de acción.
GC	<b>Nombre:</b> Gestión de la configuración
	<b>Descripción:</b> Práctica de controlar y administrar cambios en el software utilizando el control de versiones de una manera estándar y repetible.
	<b>Propósito:</b> Mantener la integridad y validez de los productos desarrollados durante todas las etapas del ciclo de vida del producto.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Identificar los productos que serán tratados como elementos de configuración</li> <li>2. Apoyar la evaluación de las solicitudes de cambio y documentar los resultados de control de cambios</li> <li>3. Mantener la validez de la configuración y la precisión del sistema de gestión de la configuración</li> </ol>
	<b>Artefactos esperados:</b> Plan de gestión de la configuración, registros de control de cambios, código fuente, ejecutables, especificación de requerimientos.
MC	<b>Nombre:</b> Monitoreo y observabilidad continua
	<b>Descripción:</b> Práctica de monitorear, alertar y tomar medidas de manera proactiva en áreas clave para brindar a los equipos visibilidad del estado de la aplicación durante todo el ciclo de vida de un producto software.
	<b>Propósito:</b> Práctica esencial de DevOps que se encarga del monitoreo de todo tipo de recurso, personal y procesos para encontrar errores lo antes posible y determinar por qué suceden.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Mejorar la transparencia y la visibilidad de las operaciones de red y TI</li> <li>2. Identificar causas de error en el funcionamiento y rendimiento del software</li> <li>3. Aplicar soluciones adecuadas a errores identificados antes de que se produzcan daños importantes en el tiempo de actividad y los ingresos</li> <li>4. Rastrear el comportamiento del usuario</li> </ol>
	<b>Artefactos esperados:</b> Registro de eventos y rastreos, métricas.
ED	<b>Nombre:</b> Educación en torno a DevOps
	<b>Descripción:</b> Práctica que impulsa la entrega de software y rendimiento organizacional mejores.
	<b>Propósito:</b> Contribuye al rendimiento de la entrega de software con las siguientes características: mayor frecuencia de implementación, plazos de entrega reducidos para los cambios, tiempo para restablecer el servicio y cambio en la tasa de fallas, cultura de equipo sólida [94].
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Mejorar la calidad y velocidad del equipo en su flujo de trabajo</li> </ol>
	<b>Artefactos esperados:</b> Plan de capacitación.
RC	<b>Nombre:</b> Realimentación continua e innovación
	<b>Descripción:</b> Práctica encargada de mantener de forma permanente la socialización y/o documentación de aprendizajes del equipo de trabajo.
	<b>Propósito:</b> Comunicación y aprendizaje permanente.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Optimizar los procesos asociados al flujo de trabajo.</li> </ol>
	<b>Artefactos esperados:</b> Plan de realimentación, cuestionarios de realimentación.
MCu	<b>Nombre:</b> Medición de la cultura
	<b>Descripción:</b> Práctica que permite medir las capacidades que impulsan la entrega de software y rendimiento organizativo para determinar acciones de mejora.
	<b>Propósito:</b> Conocer el impacto de la cultura sobre los resultados en el proceso de adopción de DevOps en la empresa.

	<b>Objetivos Específicos:</b> 1. Identificar comportamientos en la cultura organizacional. 2. Fomentar prácticas organizacionales que favorezcan la confianza y el flujo de trabajo.
	<b>Artefactos esperados:</b> Cuestionarios para el personal.

Acrónimos utilizados: Id: Identificador.

## 5.4.2. Prácticas complementarias

En la Figura 5-4 se puede visualizar cada una de las prácticas complementarias propuestas con su respectivo identificador, nombre de la práctica y dimensiones asociadas.



Figura 5-4. Prácticas complementarias propuestas.

Adicionalmente, en la Tabla 5-4 se especifica cada una de estas prácticas con su respectivo identificador, nombre de la práctica, descripción, propósito y artefactos esperados.

Tabla 5-4. Prácticas complementarias DevOps Model.

<b>Id</b>	<b>Detalle prácticas complementarias</b>
<b>DC</b>	<b>Nombre:</b> Despliegue continuo
	<b>Descripción:</b> Práctica que se esfuerza por automatizar el despliegue del software a entornos de producción sin intervención humana.
	<b>Propósito:</b> Asegurar que se necesita un esfuerzo mínimo para desplegar código en entornos productivos.
	<b>Objetivos Específicos:</b> 1. Establecer una tubería de despliegue continuo 2. Establecer prácticas de contenerización o virtualización. 3. Reducción de dependencias entre miembros del equipo (Dev -Op - QA)
	<b>Artefactos esperados:</b> Scripts de aprovisionamiento de la infraestructura y despliegue [93].
<b>ICo</b>	<b>Nombre:</b> Infraestructura como código
	<b>Descripción:</b> Práctica encargada de automatizar el aprovisionamiento de la infraestructura necesaria para la construcción de un producto software en cualquier fase, utilizando lenguajes descriptivos o de alto nivel para codificar procesos de implementación y aprovisionamiento más versátiles y adaptables.
	<b>Propósito:</b> Permitir a los desarrolladores o equipos de operaciones administrar, monitorear y aprovisionar recursos automáticamente.
	<b>Objetivos Específicos:</b> 1. Los usuarios pueden definir, aprovisionar y administrar los recursos que necesitan, sin necesidad de que el personal de TI lo haga por ellos. 2. Creación y uso de archivos de definición donde se especifiquen los elementos de la infraestructura y cómo deben configurarse. 3. Gestionar archivos de definiciones para cada código de producto en un sistema de control de versiones.

	<b>Artefactos esperados:</b> Documento de configuración de entornos.
GA	<b>Nombre:</b> Gestión del acceso con privilegios
	<b>Descripción:</b> Práctica que integra políticas, procesos y herramientas para proteger, administrar y supervisar el acceso, los usuarios y las credenciales con privilegios.
	<b>Propósito:</b> Proteger la infraestructura y aplicaciones utilizadas, gestionar el negocio de manera eficiente y mantener la confidencialidad de los datos sensibles y la infraestructura crítica.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Gestionar cuentas con privilegios.</li> <li>2. Supervisar la actividad con privilegios.</li> <li>3. Controlar el acceso de usuarios con privilegios.</li> <li>4. Controlar y proteger las cuentas de infraestructura.</li> <li>5. Proteger las credenciales de las aplicaciones de terceros.</li> </ol>
	<b>Artefactos esperados:</b> Política de contraseñas, política de gestión de roles y usuarios.
AC	<b>Nombre:</b> Aprendizaje continuo
	<b>Descripción:</b> Práctica que facilita el aprender nuevas habilidades y conocimientos de forma continua.
	<b>Propósito:</b> Mantener la competitividad, favorecer la innovación.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Formalizar el proceso de aprendizaje.</li> <li>2. Establecer objetivos de aprendizajes dentro de los equipos.</li> <li>3. Iniciar un ecosistema de coaching peer-to-peer</li> </ol>
	<b>Artefactos esperados:</b> Plan de capacitación.
ExC	<b>Nombre:</b> Experimentación continua
	<b>Descripción:</b> Práctica encargada de motivar la participación y generación de nuevas ideas de forma independiente, escribir y cambiar especificaciones durante el desarrollo.
	<b>Propósito:</b> Favorecer la innovación y el aprendizaje.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Fomentar el debate sobre distintas perspectivas.</li> <li>2. Priorizar el proceso de toma de decisiones.</li> </ol>
	<b>Artefactos esperados:</b> Talleres de ideación.
SL	<b>Nombre:</b> Satisfacción laboral
	<b>Descripción:</b> Estado emocional positivo del personal respecto a la empresa y de las labores que realiza en ella.
	<b>Propósito:</b> Impactar positivamente el rendimiento y compromiso del equipo de trabajo.
	<b>Objetivos Específicos:</b> <ol style="list-style-type: none"> <li>1. Implementar sistemas de recompensas que promuevan la confianza y la colaboración.</li> <li>2. Involucrar al equipo en procesos de la empresa.</li> <li>3. Compartir un sistema de valores y objetivos.</li> </ol>
	<b>Artefactos esperados:</b> Cuestionarios para captar la percepción y opinión de las personas.

Acrónimos utilizados: **Id:** Identificador.

## 5.5. Relación entre valores, principios, dimensiones y prácticas propuestas

DevOps Model además de representar de manera individual los elementos esenciales para la construcción de procesos y/o indicadores para la adopción de DevOps también guarda coherencia a nivel de conjunto, así entonces las prácticas fundamentales materializan los principios de DevOps definidos y estos principios a su vez son creados para dar respuesta a los valores.

En la Tabla 5-5 se muestra la relación de cada uno de los valores de DevOps y su relación con cada uno de los doce principios. En la Tabla 5-6 se muestra la relación de los principios de DevOps y las prácticas fundamentales que las materializan.

Tabla 5-5. Relación entre valores y principios propuestos en DevOps Model.

		Valores DevOps			
		Automatización	Colaboración	Medición	Comunicación
Principio DevOps	PD_01	X			X
	PD_02			X	X
	PD_03	X			
	PD_04		X		X
	PD_05	X	X		
	PD_06				X
	PD_07			X	
	PD_08	X	X		X
	PD_09		X		
	PD_10	X			
	PD_11		X		X
	PD_12			X	X

Tabla 5-6. Relación entre principios y prácticas fundamentales propuestas en DevOps Model.

		Prácticas propuestas en DevOps Model																	
		IC	E	PC	G	G	S	D	G	M	E	R	MC	D	IC	G	A	Ex	S
		C	C	R	R	D	S	E	C	C	D	C	C	o	A	C	C	L	
Principios DevOps	PD_01	X	X	X		X	X		X	X			X	X	X				
	PD_02	X	X	X	X	X	X	X	X	X	X		X	X		X	X	X	
	PD_03				X						X	X	X				X	X	X
	PD_04	X	X	X	X	X	X	X	X	X		X		X	X	X	X	X	X
	PD_05										X	X	X				X	X	X
	PD_06				X	X			X	X		X						X	X
	PD_07	X	X	X								X		X	X				
	PD_08	X	X	X						X		X		X	X				
	PD_09				X	X			X	X	X	X					X	X	X
	PD_10	X	X	X	X	X			X					X	X				
	PD_11				X	X		X				X					X	X	X
	PD_12							X				X					X		

La relación entre dimensiones y prácticas propuestas en DevOps Model se muestran en la Tabla 5-7, en la cual se puede observar que la gran mayoría de prácticas están relacionadas con la dimensión DD\_04 con un total de 16 prácticas, lo cual confirma que un proceso de desarrollo basado en DevOps está centrado en las personas. En segundo lugar, se encuentran las dimensiones DD\_01 y DD\_02, con un total de 15 cada una, lo cual refleja la importancia de los procesos y las herramientas. Finalmente, la dimensión DD\_03 está vinculada a 7 prácticas.

Tabla 5-7. Relación entre dimensiones y prácticas propuestas en DevOps Model.

		Prácticas propuestas en DevOps Model																	
		IC	E C	PC	G R	G D	S S	D E	G C	M C	E D	R C	MC u	D C	IC o	G A	A C	Ex C	S L
Dimensió n	DD_01	X	X	X		X	X		X	X				X	X	X			
	DD_02	X	X	X	X	X	X	X	X	X	X		X	X		X	X	X	
	DD_03				X						X	X	X				X	X	X
	DD_04	X	X	X	X	X	X	X	X	X				X	X	X	X	X	X

## 5.6. DevOps Assessment Model - Método de valoración de la adopción de DevOps en EDS

A continuación se presenta en detalle el método de valoración del proceso de adopción de DevOps en EDS con relación a su: (i) propósito, (ii) objetivos específicos, (iii) componentes, (iv) roles y (v) el proceso de valoración. Para la construcción del proceso de valoración se usaron las actividades establecidas en PvalCOMPETISOFT [95].

### 5.6.1. Propósito del modelo de valoración

El modelo de valoración tiene como objetivo valorar el porcentaje de adopción de DevOps teniendo en cuenta un conjunto de preguntas que permiten conocer el porcentaje de cumplimiento de las prácticas fundamentales, principios, dimensiones y valores propuestos por DevOps Model descritos en las Secciones anteriores. El reporte de valoración que se genera a partir de la aplicación de este método de valoración permite a las EDS encontrar oportunidades de mejora relacionadas con los procesos de DevOps.

### 5.6.2. Objetivos específicos del modelo de valoración

Este método de valoración de DevOps tiene los siguientes objetivos:

- Brindar elementos necesarios para llevar a cabo una valoración del porcentaje de implementación y adopción de DevOps en los procesos de desarrollo de software en EDS.
- Brindar elementos que permitan identificar oportunidades de mejora relacionadas con los procesos de adopción de DevOps en EDS.
- Permitir la valoración continua de la implementación y adopción de DevOps en EDS.

### 5.6.3. Componentes del modelo de valoración

Para la construcción de los componentes del modelo de valoración de DevOps en EDS se siguieron las recomendaciones planteadas por el paradigma Goal, Question, Metric – GQM [96], el cual propone: (i) un nivel conceptual (Goal), (ii) un nivel operacional (Question) y (iii) un nivel cuantitativo (Metric). Respecto al nivel conceptual, la meta definida fue la implementación de los valores, principios, dimensiones y prácticas definidas en DevOps Model. En el nivel operacional, se definió un conjunto de preguntas asociadas a cada uno de los objetivos específicos de las prácticas fundamentales y un instrumento de valoración tipo cuestionario. Finalmente, para el nivel operacional, se definió un conjunto de métricas que a



partir de las respuestas a las preguntas del instrumento de valoración, permiten conocer el grado de cumplimiento de las prácticas, dimensiones, principios y valores de DevOps definidos en DevOps Model. A continuación, se describen las preguntas, el instrumento de valoración y las métricas propuestas.

### 5.6.3.1. Preguntas asociadas a las prácticas fundamentales

El modelo de valoración define un conjunto de preguntas obtenidas a partir del análisis de los objetivos específicos asociados a cada una de las prácticas fundamentales definidas en DevOps Model. Las preguntas se definieron usando una escala nominal con posibles valores entre: “SI” o “NO”. En la Tabla 5-8 se pueden observar las preguntas incluyendo su identificador y práctica fundamental asociada.

Tabla 5-8. Preguntas asociadas a las prácticas de DevOps.

IdPrac	Práctica DevOps	Preguntas asociadas a las prácticas de DevOps	
		IdPreg	Pregunta
IC	Integración Continua	IP_01	¿Se tiene un sistema de control de versiones?
		IP_02	¿El equipo tiene un repositorio definido para integración de código?
		IP_03	¿Equipos de desarrollo, operaciones y calidad tienen acceso al sistema de control de versiones?
		IP_04	¿Se ejecutan pruebas unitarias antes de la integración?
		IP_05	¿La compilación de código está automatizada?
		IP_06	¿Existe un servicio que crea y ejecuta pruebas unitarias automáticas en los cambios realizados en el código?
		IP_07	¿Es posible visualizar fallos durante el proceso de integración?
EC	Entrega continua	IP_08	¿Los cambios en el código se envían a un entorno de preproducción?
PC	Pruebas continuas	IP_09	¿Existe un entorno de pruebas?
		IP_10	¿Se realizan pruebas en el proceso de integración?
		IP_11	¿Se realizan pruebas unitarias?
		IP_12	¿Se generan pruebas funcionales a partir de historias de usuario o requisitos?
		IP_13	¿Existen pruebas sobre las APIs?
		IP_14	¿Se realiza un plan de pruebas de acuerdo con el sprint o iteración?
		IP_15	¿Existen pruebas de rendimiento?
GR	Gestión de requisitos	IP_16	¿Existe documentación o registros sobre incidentes en la ejecución de pruebas?
		IP_17	¿Existe un repositorio donde se visualicen los requisitos?
		IP_18	¿La información de los requisitos es siempre accesible a las personas interesadas?
		IP_19	¿Existe un proceso de priorización de requisitos?
GD	Gestión de datos	IP_20	¿Es posible hacer seguimiento a cada requisito?
		IP_21	¿Existe claridad sobre quién usa los datos y desde dónde?
		IP_22	¿Hay un protocolo, procedimiento, proceso o similar para el acceso a datos?
		IP_23	¿Hay un proceso claro de gestión de datos heredados?
		IP_24	¿Hay un modelo de datos y diccionario de datos?
		IP_25	¿Se han establecido mecanismos para realizar backups?
		IP_26	¿Se toman medidas sobre el uso de los datos?
		IP_27	¿Existen sistemas de seguridad que protejan los datos?
SS	Supervisión de la seguridad	IP_28	¿Existe un sistema de recuperación de los datos?
		IP_29	¿Las prácticas de seguridad están alineadas con el flujo de trabajo del equipo?
		IP_30	¿Se identifican vulnerabilidades?
		IP_31	¿Existen políticas claras sobre el uso de herramientas o software de terceros?

		IP_32	¿Existen políticas claras sobre la creación y asignación de privilegios en sistemas software y en hardware?
		IP_33	¿Hay una política de seguridad?
		IP_34	¿Hay políticas de control de acceso?
DE	Dirección estratégica	IP_35	¿Existe un plan estratégico?
		IP_36	¿Se gestionan y evalúan los resultados obtenidos?
		IP_37	¿Se realizan procesos de vigilancia tecnológica?
		IP_38	¿Es claro el mercado, la posición actual de la empresa en el mercado y la competencia existente?
GC	Gestión de la configuración	IP_39	¿Existe una base de datos donde se identifican todos los activos de TI?
		IP_40	¿Se gestiona la información correspondiente a las configuraciones?
		IP_41	¿Se realiza la gestión de incidentes, problemas y cambios?
		IP_42	¿Se realiza un control y gestión de versiones del software?
MC	Monitoreo y observabilidad continua	IP_43	¿Se verifica el estado y salud de los servidores usados?
		IP_44	¿Existe un proceso definido para hacer seguimiento a los logs de la aplicación?
		IP_45	¿Se usan herramientas de monitoreo de la infraestructura?
		IP_46	¿Se usan herramientas de monitoreo de la red?
		IP_47	¿Existen mecanismos para alertar o realimentar al equipo sobre problemas en la aplicación en entornos productivos?
		IP_48	¿Existen mecanismos para obtener información sobre la experiencia de los usuarios de la aplicación en entornos de producción?
ED	Educación en torno a DevOps	IP_49	¿Existe un plan de capacitación relacionado a temas de DevOps?
		IP_50	¿El equipo de trabajo tiene los recursos necesarios para iniciar el proceso de capacitación en temas de DevOps?
RC	Realimentación continua e innovación	IP_51	¿Existen mecanismos para apoyar y compartir conocimiento entre los integrantes del equipo de trabajo?
		IP_52	¿Se le permite al equipo implementar nuevas formas de trabajo?
Mcu	Medición de la cultura	IP_53	¿Existen procesos que permitan medir la cultura organizacional?
		IP_54	¿Se identifican e implementan oportunidades de mejora a todo el personal?
		IP_55	¿Hay formas de reconocimiento y/o estímulo personal?

Acrónimos utilizados: **IdPrac**: Identificar de práctica, **IdPreg**: Identificador de pregunta

A partir de las preguntas descritas anteriormente, se definió un instrumento de valoración tipo cuestionario donde el evaluador asigna un valor de “SI” o “NO” a cada una de las preguntas de acuerdo con la evidencia que se encuentre del cumplimiento en el proceso de desarrollo. La evidencia puede ser recolectada por medio de la observación directa en la ejecución de los procesos o prácticas de desarrollo y la recopilación de las opiniones de las personas responsables y del personal involucrado en su ejecución. Para obtener una valoración de “SI” en el instrumento de valoración en una de las preguntas definidas, se debe tener evidencia de su cumplimiento total, es decir, en caso de que el cuestionamiento se cumpla parcialmente deberá asignar un valor de “NO” y usar el campo “Comentarios” del instrumento para describir el estado actual de la implementación de los elementos relacionados a la pregunta. En el Anexo C, es posible observar el instrumento de valoración.

### 5.6.3.2. Métricas

El método de valoración define cinco métricas que permiten conocer de manera cuantitativa el grado de adopción de DevOps en los procesos de desarrollo del software en EDS a partir del cumplimiento de las preguntas asociadas a las prácticas fundamentales de DevOps Model. En la Tabla 5-9 es posible observar las métricas definidas en el método de valoración las cuales permiten conocer: (i) el grado de cumplimiento de las prácticas fundamentales, (ii) el grado de cumplimiento de los principios de DevOps, (iii) el grado de cumplimiento de las dimensiones de DevOps, (iv) el grado de cumplimiento de los valores de DevOps, y (v) el

grado de implementación de DevOps en EDS. Todas las métricas se derivan de las respuestas obtenidas en el instrumento de valoración.

Tabla 5-9. Métricas definidas en el método de valoración.

<b>Id.</b>	<b>Nombre</b>	<b>Descripción</b>	<b>Unidad de medida</b>
<i>gPF</i>	Grado de cumplimiento de la práctica fundamental.	Representa el grado de cumplimiento de una práctica fundamental de DevOps Model.	Porcentaje
<i>gPD</i>	Grado de cumplimiento del principio de DevOps.	Representa el grado de cumplimiento de un principio de DevOps Model.	Porcentaje
<i>gDD</i>	Grado de cumplimiento de la dimensión de DevOps.	Representa el grado de cumplimiento de una dimensión de DevOps Model.	Porcentaje
<i>gVD</i>	Grado de cumplimiento del valor de DevOps.	Representa el grado de cumplimiento de un Valor de DevOps Model.	Porcentaje
<i>gID</i>	Grado de implementación de DevOps.	Representa el grado de implementación de DevOps según DevOps Model.	Porcentaje

Acrónimo utilizado: **Id.**: Identificador.

Para la calificación del grado de cumplimiento de las prácticas, principios, dimensiones, valores y el grado de adopción de DevOps en EDS, se utilizó como referente la escala definida en la norma ISO/IEC 15504. A continuación, en la Tabla 5-10 se describe la escala de calificación.

Tabla 5-10. Escala de calificación definida en el método de valoración.

<b>Acrónimo</b>	<b>Rango</b>	<b>Grado de cumplimiento</b>
NA	El grado de cumplimiento o grado de adopción de DevOps está entre el 0% y 15%.	No alcanzado
PA	El grado de cumplimiento o grado de adopción de DevOps está entre el 16% y 50%.	Parcialmente alcanzado
AA	El grado de cumplimiento o grado de adopción de DevOps está entre el 51% y 85%.	Ampliamente alcanzado
CA	El grado de cumplimiento o grado de adopción de DevOps está entre el 86% y 100%.	Completamente alcanzado

A continuación, se describe en detalle cada una de las métricas propuestas, además se presenta un ejemplo de aplicación para facilitar su comprensión:

- **Grado de cumplimiento de la práctica fundamental – gPF**

Esta métrica permite conocer en qué porcentaje se está cumpliendo con una determinada práctica fundamental en el proceso de desarrollo de software. Se calcula usando la Ecuación 1.

Ecuación 1. Métrica de cumplimiento de práctica fundamental.

$$gPF = \frac{nPP}{nTP} \times 100$$

Donde:

*nPP* = Número de preguntas asociadas a la práctica fundamental con un valor de “SI”.

*nTP* = Número total de preguntas asociadas a la práctica fundamental.

Por ejemplo, la práctica Dg – Dirección estratégica tiene asociadas cuatro preguntas (IP\_34, IP\_35, IP\_36 y IP\_37). En este caso, suponiendo que tres de estas preguntas sean evaluadas con un valor de “SI”, se tendría que el grado de cumplimiento de la práctica sería de  $(3/4)*100 = 75\%$ , Es decir: ampliamente alcanzado.

- **Grado de cumplimiento de principio de DevOps – gPD**

Esta métrica permite conocer en qué porcentaje se está cumpliendo con determinado principio de DevOps en el proceso de desarrollo de software. Su cálculo corresponde al promedio del grado de cumplimiento de las prácticas fundamentales asociadas a este principio de DevOps.

Ecuación 2. Métrica de cumplimiento de principio de DevOps.

$$gPD = \frac{\sum gPFp}{nTPp}$$

Donde:

$gPFp$  = grado de cumplimiento de cada práctica fundamental asociada al principio de DevOps.

$nTPp$  = Número total de prácticas fundamentales asociadas al principio de DevOps.

Por ejemplo, el principio de DevOps PD\_01 (Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor) tiene asociados 6 prácticas fundamentales (IC – Integración continua, EC – Entrega continua, PC – Pruebas continuas, GD – Gestión de datos, SS – Supervisión de la seguridad, GC – Gestión de la configuración y MC – Monitoreo y observabilidad continua). En este caso suponiendo un grado de cumplimiento de cada práctica correspondiente a: IC: 71%, EC: 100%, PC: 57%, GD: 88%, SS: 60%, GC: 50% y MC: 67%, se tendría un grado de cumplimiento del principio de DevOps PD\_01 de 70%, es decir: ampliamente alcanzado.

- **Grado de cumplimiento de la dimensión de DevOps – gDD**

Esta métrica permite conocer en qué porcentaje se está cumpliendo con una determinada dimensión de DevOps en el proceso de desarrollo de software. Su cálculo corresponde al promedio del grado de cumplimiento de las prácticas fundamentales asociadas a esta dimensión de DevOps.

Ecuación 3. Métrica de cumplimiento de dimensión de DevOps.

$$gDD = \frac{\sum gPFd}{nTPFd}$$

Donde:

$gPFd$  = grado de cumplimiento de cada práctica fundamental asociada a la dimensión de DevOps.

$nTPFd$  = Número total de prácticas fundamentales asociadas a la dimensión de DevOps.

Por ejemplo, la dimensión DD\_03 (Cultura) tiene asociadas 4 prácticas (GR, ED, RC, Mcu). En este caso, suponiendo un grado de cumplimiento de cada práctica fundamental correspondiente a: GR: 75%, ED: 50%, RC: 50% y Mcu: 33%, se tendría un grado de cumplimiento de la dimensión de devOps DD\_03 de 52%, es decir: ampliamente alcanzado.

- **Grado de cumplimiento de valor de DevOps – gVD**

Esta métrica permite conocer en qué porcentaje se está cumpliendo con un determinado valor de DevOps en el proceso de desarrollo de software. Su cálculo corresponde al promedio del grado de cumplimiento de los principios de DevOps asociados a ese valor de DevOps.

Ecuación 4. Métrica de cumplimiento de valor de DevOps.

$$gVD = \frac{\sum gPDv}{nTPDv}$$

Donde:  $nTPDv$  = Número total de principios de DevOps asociados al valor de DevOps.  
 $gPDv$  = grado de cumplimiento de cada principio de DevOps asociado al valor de DevOps.

Por ejemplo, el valor de DevOps de Automatización tiene asociados 5 principios de DevOps (PD\_01, PD\_03, PD\_05, PD\_08 y PD\_10). En este caso, suponiendo un grado de cumplimiento de cada principio de DevOps correspondiente a : PD\_01: 60%, PD\_03: 72%, PD\_05: 68%, PD\_08: 33% y PD\_10: 54%, se tendría un grado de cumplimiento del valor de DevOps Automatización de 57%, es decir: ampliamente alcanzado.

- **Grado de implementación de DevOps – gID.**

Esta métrica permite conocer en qué porcentaje se está cumpliendo con el proceso de adopción de DevOps en EDS según DevOps Model. Su cálculo corresponde al promedio del grado de cumplimiento de los valores de DevOps.

Ecuación 5. Métrica de cumplimiento implementación de DevOps.

$$gID = \frac{\sum gVD}{nTVD}$$

Donde:  $nTVD$  = Número total valores de DevOps.  
 $gVD$  = grado de cumplimiento de valor de DevOps.

Por ejemplo, suponiendo un grado de cumplimiento de cada valor de DevOps correspondientes a Automatización: 45%, Colaboración: 62%, Medición: 47% y Comunicación 44%, se tendría un porcentaje de implementación de DevOps del 50%, es decir: parcialmente alcanzado.

## 5.6.4. Roles del proceso de valoración

En la Tabla 5-11 se pueden observar los roles definidos en el método de valoración. Es importante tener en cuenta que un rol puede ser asumido por varias personas y una persona puede asumir varios roles.

Tabla 5-11. Roles definidos en el método de valoración.

<b>Id.rol</b>	<b>Nombre</b>	<b>Características</b>
EV	Evaluador	Conocimiento del método de valoración, capacidad para aplicar el instrumento de evaluación y analizar los datos recolectados.
RP	Responsable del proceso	Conocimiento del proceso de desarrollo de software en la EDS.

Acrónimos utilizados: **Id.rol**: Identificador del rol.

## 5.6.5. Proceso de valoración

El proceso de valoración adopta las actividades propuestas por el método de evaluación PvalCOMPETISOFT [95] las cuales se resumen a continuación:

- **Planificación de la valoración:** El evaluador describe la organización donde se realizará la valoración, la fecha de realización, las actividades a realizar y el tiempo a emplear.
- **Ejecución de la valoración:** El evaluador recolecta la información que se requiere para diligenciar el instrumento de evaluación. Esta información se obtiene por medio de la observación directa del proceso de desarrollo en una EDS y mediante la recolección de opiniones de las personas involucradas o responsables de estos procesos.
- **Generación de resultados:** Una vez se diligencia el instrumento de evaluación, el evaluador utiliza las métricas descritas anteriormente para obtener el grado de adopción de DevOps, el grado de cumplimiento de las prácticas fundamentales, de las dimensiones, principios y valores de DevOps propuestos en DevOps Model.
- **Reporte de resultados:** Una vez se obtienen los resultados de las métricas, el evaluador presenta un reporte de resultados al responsable del proceso. En este reporte se incluyen resultados cuantitativos, gráficas de grado de cumplimiento de prácticas, principios, dimensiones y valores, y oportunidades de mejora detectadas.

# Capítulo 6. Evaluación de DevOps Model

En este capítulo, se presenta la evaluación de DevOps Model a través de un grupo focal y un estudio de caso. En la Sección 6.1, se describe el grupo focal realizado con el fin de evaluar el modelo de referencia para la adopción de DevOps. A partir de los resultados obtenidos en el grupo focal, se encontraron oportunidades de mejora que orientaron los ajustes realizados al modelo y así obtener una versión final del modelo el cual se presentó en el Capítulo 5. En segundo lugar, se presenta el estudio de caso realizado en una organización dedicada al desarrollo de software, el cual evaluó la utilidad, comprensibilidad, y practicidad de modelo de referencia para la adopción de DevOps.

## 6.1. Evaluación de DevOps Model usando un grupo focal

En esta Sección se presentan los resultados del grupo focal aplicado a la primera versión de DevOps Model. Un grupo focal es una técnica de investigación que permite captar la opinión de un grupo particular de individuos respecto a un tema puntual y por medio del cual se pueden obtener datos cualitativos, que en nuestro caso particular permitieron encontrar oportunidades de mejora para redefinir el modelo propuesto. Para la realización del grupo focal se tuvieron en cuenta las directrices definidas en [97]:

- Planeamiento de la investigación: se establece el contenido y el procedimiento a ser aplicado al debate de los participantes.
- Diseño del grupo de discusión (Reclutamiento): se definen las estrategias de selección de los participantes del grupo focal.
- Conducción de la sesión de debate (Moderación): se ejecutan los procedimientos establecidos en la fase de planeación con el fin de generar la sesión de debate y capturar las opiniones y comentarios de los participantes.
- Análisis de la información y reporte de resultados: se realiza el análisis de tipo cuantitativo y/o cualitativo utilizando estadística descriptiva o métodos de tipo cuantitativo.

En las siguientes Secciones se presenta el detalle de los resultados obtenidos en cada una de las fases descritas anteriormente.

### 6.1.1. Planteamiento de la investigación

En esta fase se definen los objetivos del grupo focal, el objetivo de investigación y se preparan los materiales y procedimientos a seguir por el grupo investigador para el grupo focal.

1. **Objetivo del grupo focal:** conocer la opinión y percepción de los profesionales con experiencia en procesos de adopción de DevOps en entornos de desarrollo de

software acerca del grado de aceptación o rechazo en los siguientes elementos: (i) idoneidad, (ii) completitud, (iii) comprensibilidad del modelo y (iv) aplicabilidad en la industria del desarrollo de software.

2. **Objetivo de investigación:** realizar la evaluación del modelo de referencia propuesto con el fin de encontrar oportunidades de mejora y generar una nueva versión del modelo si es necesario.
3. **Preparación de materiales y procedimientos a seguir por parte del grupo investigador:** en esta actividad se definen los elementos, procedimientos y técnicas a ser usadas durante la ejecución del grupo focal, entre las que se encuentran: (i) estructura del protocolo del grupo focal, (ii) instrumentos y métodos a emplear en el grupo focal, (iii) socialización y formalización de documentos a los participantes, (iv) definición de métodos de captura y registro de información y (v) definición de los métodos de análisis de la información obtenida en el debate.
4. **Protocolo del grupo focal:** A continuación, en la Tabla 6-1 se presenta el protocolo definido para el grupo focal

Tabla 6-1. Estructura del protocolo definido para el grupo focal.

No.	Elemento	Descripción
1	Agenda de trabajo	Documento donde se especifican las actividades a realizar por cada participante durante la aplicación del grupo focal.
2	Cuestionario	Documento que contiene un conjunto de preguntas clave que permitirán obtener información relevante para la evaluación de la propuesta durante el grupo focal.
3	Estructura del protocolo	Documento que indica el protocolo utilizado para la aplicación del grupo focal.
4	Propuesta a evaluar	Documento que contiene la descripción de la propuesta que será evaluada.

Acrónimos utilizados: No: Número

5. **Elementos necesarios para llevar a cabo el grupo focal:** en la Tabla 6-2 se presentan los elementos utilizados para la realización del grupo focal.

Tabla 6-2. Elementos para la realización del grupo focal.

Elemento	Descripción
Fecha de realización	Fecha en la que se realizará el grupo focal.
Hora de inicio	Hora exacta en la que se dará inicio al grupo focal.
Duración	Duración aproximada del grupo focal.
Lugar	Lugar donde se realizará el grupo focal.
Tema por tratar	Tema que se tratara durante el grupo focal
Moderador	Nombre de la persona encargada de asegurar la intervención de los participantes y verificación de la agenda propuesta para el grupo focal.
Supervisor	Persona encargada de recopilar la información relevante.
Relator	Persona encargada de la exposición del tema del grupo focal.
Participantes	Personas encargadas de evaluar la propuesta presentada.
Objetivo del grupo focal	Objetivo principal de la realización del grupo focal.
Objetivo de investigación	Objetivos relacionados con las actividades realizadas en el grupo focal.

6. **Métodos de captura y registro de información:** El grupo focal contó con un relator para su ejecución, el cual tomó nota de los comentarios, apreciaciones y sugerencias relevantes de cada participante. También, se entregó a cada participante de forma anticipada un documento con la propuesta para su análisis preliminar y un



cuestionario. Además, se realizó un registro de audio y video de la sesión del grupo focal, con el fin de tener material extra de apoyo.

7. **Métodos de análisis de la información:** posterior a la realización del grupo focal, se realizó un análisis estadístico de la información recolectada en los cuestionarios y un análisis cualitativo de las observaciones y oportunidades de mejora registradas.

### 6.1.2. Diseño del grupo de discusión (Reclutamiento)

En esta fase se definen las estrategias de selección de los participantes del grupo focal. Esta actividad estuvo a cargo del grupo investigador e incluyó las siguientes actividades:

1. Definición del perfil del participante: para la realización de esta actividad se definieron los siguientes criterios de selección:
  - Tener actividad en la industria del software y/o el entorno académico como profesor o estudiante.
  - Tener conocimiento en prácticas de despliegue de productos a producción o prácticas asociadas a DevOps.
  - Conocimientos sobre marcos de trabajo ágiles o tradicionales y su aplicación en la industria del software.
  - Profesionales con experiencia en la industria del software.
2. Identificación de los participantes: tomando como referencia los criterios de selección previamente definidos se logró identificar los posibles participantes del grupo focal. El grupo conformado estuvo compuesto por profesionales con experiencia y conocimiento en diferentes áreas de la ingeniería de software, específicamente en procesos de adopción de prácticas asociadas a DevOps. En la Tabla 6-3 se presenta la descripción del perfil profesional de los participantes.

Tabla 6-3. Perfil profesional de los participantes del grupo focal.

<b>Id.</b>	<b>Ocupación</b>	<b>Estudios</b>	<b>Experiencia</b>
1	Analista de Calidad de Software	<ul style="list-style-type: none"> <li>• Ingeniera de sistemas</li> <li>• Especialización</li> </ul>	Líder de calidad de software, proceso de integración continua. 2 años de experiencia en DevOps
2	Analista de pruebas	<ul style="list-style-type: none"> <li>• Ingeniero de sistemas</li> </ul>	1 año de experiencia en DevOps
3	Arquitecto de Software	<ul style="list-style-type: none"> <li>• Pregrado</li> </ul>	Desarrollo backend en tecnología Java Analista desarrollador, 2 años de experiencia en DevOps
4	Arquitecto Cloud	<ul style="list-style-type: none"> <li>• Ingeniero de sistemas</li> </ul>	Docente en asignaturas de implementación de DevOps, Arquitecto Cloud en PSL desde 2012.
5	Ingeniero de requisitos	<ul style="list-style-type: none"> <li>• Ingeniera de sistemas</li> </ul>	1 año de experiencia en DevOps
6	Vicepresidente de Ingeniería	<ul style="list-style-type: none"> <li>• Ingeniero de sistemas</li> <li>• Especialista en gerencia de proyectos</li> <li>• Maestría en ingeniería</li> </ul>	Líder de equipo de arquitectura Certificado Azure DevOps Experto 2 años de experiencia en DevOps
7	Docente	<ul style="list-style-type: none"> <li>• Ingeniero de sistemas</li> <li>• Maestría</li> </ul>	Docente catedrático en el área de software. Investigador en área de ingeniería de software. 3 años de experiencia en marcos de trabajo ágiles.

Acrónimos utilizados: **Id:** Identificador

### 6.1.3. Conducción de la sesión de debate

La sesión de debate tuvo una duración de dos horas, la cual fue coordinada por un moderador que siguió la secuencia determinada en la Tabla 6-4.

Tabla 6-4. Organización del grupo focal.

Nº	Descripción
1	Bienvenida a los participantes
2	Presentación del grupo investigador, objetivos del grupo focal y de investigación
3	Presentación de los participantes
4	Presentación de DevOps Model
5	Discusión de la propuesta por parte de los participantes
6	Realización de la encuesta
7	Agradecimiento a los participantes
8	Finalización del grupo focal

Acronimos utilizados: N°: Número

### 6.1.4. Captura de información

Para el proceso de captura de información se contó con un relator quien fue la persona encargada de estar atento de tomar nota de las observaciones y comentarios realizados por los participantes del grupo focal.

Además, como complemento a las observaciones realizadas por los participantes, se les solicitó responder un cuestionario al final de la sesión de discusión de la propuesta. El cuestionario estaba estructurado en dos partes. La primera parte del cuestionario permitió recolectar información de cada participante: (i) nombre completo, (ii) rol desempeñado por el participante en el lugar de trabajo, (iii) años de experiencia investigando y/o aplicando DevOps y (iv) estudios. La segunda parte del cuestionario se enfocó en conocer la opinión de los participantes sobre la propuesta. Las primeras 13 preguntas, se diseñaron para ser respondidas mediante una escala Likert de 5 puntos detallada en la Tabla 6-5.

Tabla 6-5. Escala de Likert usada en el cuestionario.

Valor	Descripción
1	Muy mal, muy insatisfecho (a)
2	Mal, poco satisfecho(a)
3	Bien, suficiente, adecuado, algo satisfecho(a)
4	Bastante bien, adecuado, satisfecho(a)
5	Muy bien, muy adecuado, muy satisfecho(a)

Las preguntas formuladas, permitieron medir la comprensibilidad (4 preguntas), aplicabilidad (1 pregunta), idoneidad (3 preguntas) y completitud (2 preguntas) del modelo propuesto. También se definieron preguntas que medían dos variables al tiempo, las cuales fueron: la idoneidad y completitud (1 pregunta) y la idoneidad y aplicabilidad (2 preguntas). Por último, se incluyeron 2 preguntas abiertas que permitieron a los participantes proponer ajustes al modelo y realizar comentarios adicionales. A continuación, en la Tabla 6-6, se puede observar las preguntas propuestas. En el Anexo C, se presentan los cuestionarios diligenciados por los participantes.

Tabla 6-6. Cuestionario de evaluación usado en el grupo focal.

Aspecto por evaluar	Id.	Pregunta
Comprensibilidad	P1	¿Considera que los valores de DevOps propuestos son de fácil comprensión?
	P2	¿Considera que los principios de DevOps propuestos son de fácil comprensión?
	P3	¿Considera que las dimensiones propuestas son de fácil comprensión?
	P4	¿Considera que las prácticas propuestas son de fácil comprensión?
Aplicabilidad	P5	De acuerdo con su experiencia: ¿Considera que los elementos definidos en el modelo de referencia son apropiados y pueden aplicarse con éxito en un proyecto de desarrollo de software?
Idoneidad	P6	¿Considera que los valores propuestos capturan la esencia principal de DevOps?
	P7	De acuerdo con su experiencia: ¿Considera que los principios definidos para DevOps son apropiados e interpretan correctamente las dinámicas actuales de la industria del Software?
Complejidad	P8	De acuerdo con su experiencia: ¿Considera que los principios definidos para DevOps son suficientes?
Idoneidad / Complejidad	P9	¿Considera que las dimensiones definidas para DevOps agrupan correctamente los principios y prácticas definidas?
Idoneidad / Aplicabilidad	P10	¿Considera que las prácticas fundamentales propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?
Idoneidad / Aplicabilidad	P11	¿Considera que las prácticas complementarias propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?
Complejidad	P12	¿Considera que las prácticas propuestas son suficientes para un proceso de adopción de DevOps?
Idoneidad	P13	¿Considera que el modelo propuesto sirve de referencia para proyectos de mejora que tienen por objetivo la definición de procesos relacionados con DevOps?
<b>Preguntas abiertas</b>		
Id.	Pregunta	
P14	¿Considera que se deben agregar, eliminar o modificar elementos (valores, principios, dimensiones o prácticas) de la propuesta?	
P14	¿Tiene algún comentario adicional acerca del modelo de referencia propuesto?	

Acrónimos utilizados: **Id.**: Identificador

### 6.1.5. Análisis de la información y reporte de resultados

Una vez realizado el grupo focal, se llevó a cabo el análisis de los aportes realizados por los participantes de la sesión de debate de la propuesta y los resultados del cuestionario de evaluación diligenciado al final de la sesión. A continuación, se presentan las actividades realizadas para este proceso de análisis de la información obtenida en el grupo focal.

#### 6.1.5.1. Análisis de las preguntas cerradas

Para las preguntas P1 a P13, se realizó un conteo de las respuestas de cada participante. En la Tabla 6-7, se presenta el conteo de las respuestas para cada opción de la escala de Likert (1, 2, 3, 4, 5) en las preguntas P1 a P13. En la Figura 6-1 se muestra una gráfica con los resultados consolidados.

Tabla 6-7. Conteo de respuestas a preguntas P1-P13.

Id.	Nivel de conformidad				
	1	2	3	4	5
P1	0	0	0	3	4
P2	0	0	0	3	4
P3	0	0	0	3	4
P4	0	0	1	5	1

P5	0	0	2	4	1
P6	0	0	0	4	3
P7	0	0	1	2	4
P8	0	0	2	2	3
P9	0	0	0	3	4
P10	0	0	2	3	2
P11	0	0	1	3	3
P12	0	0	0	4	3
P13	0	0	1	3	3

Acrónimos utilizados: **Id**: Identificador

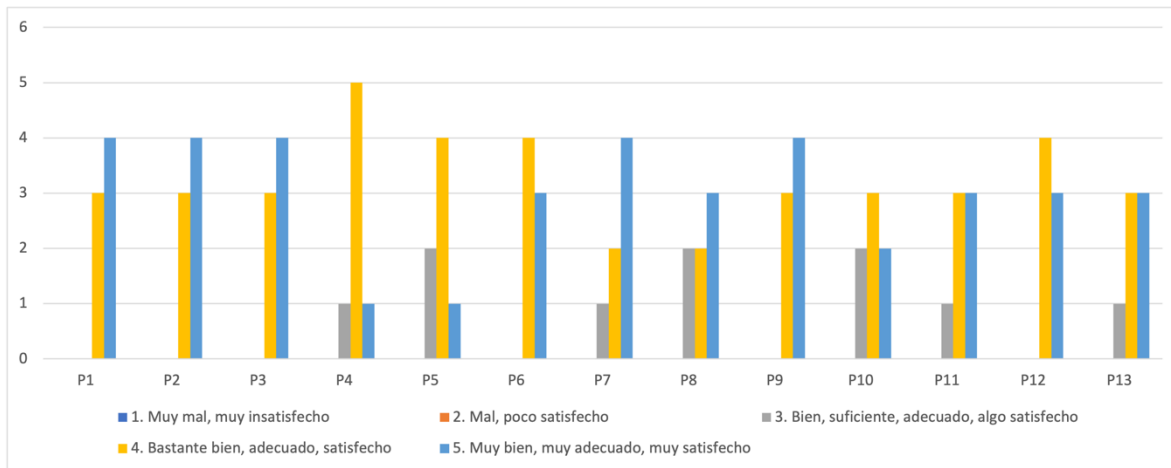


Figura 6-1. Consolidado de respuestas a preguntas P1-P13.

A continuación se presenta un análisis de las respuestas a las preguntas agrupadas según el aspecto a evaluar: Comprensibilidad, aplicabilidad, idoneidad y completitud.

- **Comprensibilidad**

- En P1 se preguntó a los participantes si consideraban que los valores de DevOps propuestos son de fácil comprensión. Tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P2 se preguntó a los participantes si consideraban que los principios de DevOps propuestos son de fácil comprensión. Tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P3 se preguntó a los participantes si consideraban que las dimensiones propuestas son de fácil comprensión. Tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P4 se preguntó a los participantes si consideraban que las prácticas propuestas son de fácil comprensión. Un (1) participante respondió usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, cinco (5) participantes respondieron con la opción “Bastante bien, adecuado,

satisfecho(a)”, y un (1) participante respondió usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.

- **Aplicabilidad:**

- En P5 se preguntó a los participantes si de acuerdo con su experiencia consideraban que los elementos definidos en el modelo de referencia son apropiados y pueden aplicarse con éxito en un proyecto de desarrollo de software. Dos (2) participantes respondieron usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, cuatro (4) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y un (1) participante respondió usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P10 se preguntó a los participantes si consideraban que las prácticas fundamentales propuestas pueden aplicarse con éxito en un proceso de desarrollo de software. Dos (2) participantes respondieron usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y dos (2) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P11 se preguntó a los participantes si consideraban que las prácticas complementarias propuestas pueden aplicarse con éxito en un proceso de desarrollo de software. Un (1) participante respondió usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.

- **Idoneidad:**

- En P6 se preguntó a los participantes si consideraban que los valores propuestos capturan la esencia principal de DevOps. Cuatro (4) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P7 se preguntó a los participantes si de acuerdo con su experiencia consideraban que los principios definidos para DevOps son apropiados e interpretan correctamente las dinámicas actuales de la industria del Software. Un (1) participante respondió usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, dos (2) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- En P9 se preguntó a los participantes si consideraban que las dimensiones definidas para DevOps agrupan correctamente los principios y prácticas definidas. Tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.

- En P10 se preguntó a los participantes si consideraban que las prácticas fundamentales propuestas son apropiadas en un proceso de desarrollo de software. Dos (2) participantes respondieron usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y dos (2) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
  - En P11 se preguntó a los participantes si consideraban que las prácticas complementarias propuestas son apropiadas en un proceso de desarrollo de software. Un (1) participante respondió usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
  - En P13 se preguntó a los participantes si consideraban que el modelo propuesto sirve de referencia para proyectos de mejora que tienen por objetivo la definición de procesos relacionados con DevOps. Un (1) participante respondió usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
- **Completitud:**
    - En P8 se preguntó a los participantes si de acuerdo con su experiencia consideraban que los principios definidos para DevOps son suficientes. Dos (2) participantes respondieron usando la opción “Bien, suficiente, adecuado, algo satisfecho(a)”, dos (2) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
    - En P9 se preguntó a los participantes si consideraban que las dimensiones definidas para DevOps agrupan correctamente los principios y prácticas definidas. Tres (3) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y cuatro (4) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.
    - En P12 se preguntó a los participantes si consideraban que las prácticas propuestas son suficientes para un proceso de adopción de DevOps. Cuatro (4) participantes respondieron con la opción “Bastante bien, adecuado, satisfecho(a)”, y tres (3) participantes respondieron usando la opción “Muy bien, muy adecuado, muy satisfecho(a)”.

Según los resultados anteriores, se puede observar que los participantes tuvieron una opinión favorable acerca de las comprensibilidad, aplicabilidad, idoneidad y completitud de DevOps Model.

### 6.1.5.2. Análisis de preguntas abiertas

Las preguntas P14 y P15 permitieron a los participantes proponer ajustes al modelo y realizar cambios adicionales. A continuación, en la Tabla 6-8 se presentan las respuestas de cada uno de los participantes.

Tabla 6-8. Respuestas a las preguntas abiertas.

Pregunta	PF.	Respuesta
P14: ¿Considera que se deben agregar, eliminar o modificar elementos (valores, principios, dimensiones o prácticas) de la propuesta?	PF1	Revisar si el despliegue continuo al menos en un nivel básico debe ser una práctica fundamental. Sería interesante establecer los requisitos previos que debe cumplir la organización para iniciar un proceso de implementación de DevOps.
	PF2	Yo le agregaría para la parte de integración continua alguna referencia de como incorporar el proceso de pruebas para que quede listo para pasar a producción, mirar si hay elementos de riesgos se puedan presentar.
	PF3	Como sugerencia se podría dejar la práctica "Supervisión de la seguridad" como complementaria, porque no bloquea la implementación de la propuesta pero sí permite mejorarla.
	PF4	No, en mi opinión es un trabajo bastante completo.
	PF5	No, todos los elementos presentados son claros.
	PF6	Tal vez hablar del shift left.
	PF7	Ninguno.
P15: ¿Tiene algún comentario adicional acerca del modelo de referencia propuesto?	PF1	Muy buen trabajo. Un aporte acerca de la gestión de requisitos: es ideal que sean cortos, completos, claros, realistas y acordados para evitar cuellos de botella desde el área de análisis.
	PF2	Excelente muy bien, felicitaciones. Muchos exitos.!
	PF3	La propuesta es robusta y muy bien fundamentada.
	PF4	Excelente propuesta y bien fundamentada.
	PF5	Buen trabajo.
	PF6	Ninguno.
	PF7	Ninguno.

Acrónimos utilizados: **PF**: Identificador del participante del grupo focal

### 6.1.6. Acciones de mejora

A partir de los resultados del grupo focal, se analizaron las sugerencias realizadas por los participantes con el fin de determinar si se considerarían como oportunidades de mejora para el modelo. A continuación, en la Tabla 6-9, por cada comentario recibido se describe el cambio realizado o la justificación de por qué no se realizó el cambio.

Tabla 6-9. Acciones de mejora definidas para DevOps Model.

Comentario	Acción de mejora / justificación
Revisar si el despliegue continuo al menos en un nivel básico debe ser una práctica fundamental. Sería interesante establecer los requisitos previos que debe cumplir la organización para iniciar un proceso de implementación de DevOps.	En efecto, la práctica de despliegue continuo es una práctica muy importante en la búsqueda de procesos más cortos de despliegue de productos o servicios a producción, impactando significativamente los retornos de inversión y procesos de feedback con el cliente. Sin embargo, esta práctica requiere adoptar herramientas y actividades de automatización, las cuales requieren un mayor nivel de entrenamiento para que no se produzca intervención humana para el proceso de despliegue. Por esta razón solo se especifica como práctica fundamental la práctica de entrega e integración continuas que permiten obtener buenos resultados en el despliegue, pero aun con el acompañamiento de personas. Por la anterior, la práctica de despliegue continuo se mantiene como práctica complementaria.
Yo le agregaría para la parte de integración continua alguna referencia de cómo incorporar el proceso de pruebas para que quede listo para	En DevOps Model se propone se la práctica de pruebas continuas, se considera que esta práctica cubre diferentes tipos de pruebas y en diferentes tiempos durante todo el ciclo de vida. DevOps Model intenta ser un modelo genérico esencial que se centra en la especificación de qué debe tener un proceso de

pasar a producción, mirar si hay elementos de riesgos se puedan presentar.	adopción de DevOps y por medio de esta, cada empresa con sus particularidades tenga la posibilidad de definir el cómo.
Como sugerencia se podría dejar la práctica "Supervisión de la seguridad" como complementaria, porque no bloquea la implementación de la propuesta pero sí permite mejorarla	Se considera que la práctica de supervisión de la seguridad debe ser una práctica fundamental, ya que DevOps favorece la implementación de entornos altamente productivos y ante el incremento de la velocidad de puesta en producción de un producto o servicio no se puede escatimar en aspectos de seguridad.
Tal vez hablar del shift left.	Shift left es un término asociado a la práctica de realizar procesos de pruebas rápidas y confiables lo antes posible en el ciclo de vida de la entrega de software. Consideramos que shift left está incluido dentro de la práctica de pruebas continuas definida como práctica fundamental en el modelo propuesto.
Un aporte acerca de la gestión de requisitos: es ideal que sean cortos, completos, claros, realistas y acordados para evitar cuellos de botella desde el área de análisis.	El aporte realizado es una muy buena sugerencia para tener en cuenta en la especificación de procesos o pipelines que detallen el cómo se debe realizar la práctica de gestión de requisitos lo cual no hace parte del alcance del proyecto.

## 6.2. Estudio de caso

El estudio de caso es un método de investigación detallado que sirve para describir, comparar, evaluar y comprender aspectos asociados a un problema de investigación. En esta Sección se presenta la evaluación de DevOps Model mediante un estudio de caso aplicado en una organización desarrolladora de software, para esto, se siguieron los lineamientos establecidos para la realización de estudios de caso en [98]. A continuación, se describe en detalle: (i) el diseño del estudio de caso, (ii) el sujeto de estudio, (iii) el procedimiento de campo y recolección de datos, (iv) el análisis de resultados, (v) el análisis de validez, y (vi) las limitaciones del estudio.

### 6.2.1. Diseño del estudio de caso

Con el fin de direccionar el estudio de caso, se definió la pregunta de investigación detallada en la Tabla 6-10.

Tabla 6-10. Pregunta de investigación estudio de caso.

Pregunta de investigación
¿DevOps Model permite de manera comprensible, útil y práctica la definición de un modelo de valoración de adopción de DevOps de acuerdo con las características de cada organización?

A partir de esta pregunta de investigación, se definieron las preguntas de investigación específicas detalladas en la Tabla 6-11.

Tabla 6-11. Preguntas específicas investigación estudio de caso.

Id	Pregunta
P1	¿Los componentes de DevOps Model (Valores, principios, dimensiones y prácticas) son de fácil comprensión?
P2	¿Los resultados obtenidos mediante la aplicación del modelo de valoración generado a partir de DevOps Model proporcionan información útil a las organizaciones para mejorar sus procesos de adopción de DevOps?
P3	¿La aplicación de DevOps Model requiere de un nivel de esfuerzo aceptable para las organizaciones?

Teniendo en cuenta la clasificación presentada en [98], el diseño de este estudio de caso fue holístico con una unidad de análisis, la cual correspondió a la aplicación del modelo de



valoración basado en DevOps Model en una organización dedicada al desarrollo de software. Las medidas utilizadas en el estudio de caso fueron: Las oportunidades de mejora identificadas por medio de DevOps Model y DevOps Assessment Model, el esfuerzo requerido para generar un diagnóstico utilizando DevOps Model y DevOps Assessment Model de acuerdo con la realidad de la empresa, y los beneficios percibidos por la organización con relación a DevOps Model.

### **6.2.2. Sujetos del estudio**

A continuación, se describen los criterios que se tuvieron en cuenta para seleccionar la organización donde se llevó a cabo el estudio de caso:

- Es una organización desarrolladora de software.
- La organización ya tiene implementadas prácticas asociadas a DevOps.
- La organización está interesada en evaluar y encontrar oportunidades de mejora en sus procesos de desarrollo de software.

A partir de los criterios descritos anteriormente, se seleccionó una organización desarrolladora de software que por razones de confidencialidad no se revelara su nombre pero se puede indicar que es de alcance internacional (LATAM), cuenta con 1300 empleados y 32 años de experiencia en la industria a la fecha, su principal enfoque está en la producción y mantenimiento de servicios en el área de contabilidad y también como proveedor de servicios de facturación electrónica. Con respecto a los modelos y/o estándares implementados en la organización se destaca el uso de prácticas ágiles del marco de trabajo Scrum, la metodología ShapeUp y la adaptación del modelo de tribus de Spotify.

### **6.2.3. Procedimiento de campo y recolección de datos**

En las siguientes sesiones se describen las actividades ejecutadas en la organización con el fin de aplicar DevOps Model: (i) planificación de la evaluación, (ii) ejecución de la evaluación, y (iii) generación y reporte de resultados.

#### *6.2.3.1. Planificación de la evaluación*

Con el objetivo de planificar la realización del estudio de caso, se estableció un primer contacto telefónico con el Vicepresidente de Ingeniería de la organización con el fin de exponer el objetivo del estudio, las actividades a realizar, aclarar las dudas, caracterizar la organización y establecer una fecha para realizar la reunión de inicio del estudio. Luego de este primer contacto, se definió una fecha y hora para la realización de una reunión de forma remota y síncrona usando la herramienta Google Meet como mediadora, la cual tuvo una duración de una hora y se desarrolló con la siguiente agenda:

- Presentación del proyecto y del equipo investigador.
- Presentación general del modelo de referencia DevOps Model.

- Presentación general del modelo de valoración DevOps Assessment Model.
- Presentación y firma del documento: Aceptación de participación en el estudio de caso (ver Anexo C).
- Asignación de roles. En este punto el Vicepresidente de Ingeniería asumió el rol de responsable del proceso y evaluador debido a su amplio conocimiento en el proceso de desarrollo de la organización, su amplio conocimiento de las prácticas y pipelines de DevOps adoptadas y a su interés en encontrar oportunidades de mejora que se puedan aplicar.
- Cronograma de evaluación: por motivos de disponibilidad de tiempo, se acordó una nueva reunión de asesoría en una fecha posterior usando la herramienta Google Meet o de la plataforma WhatsApp para clarificar dudas que se pudieran presentar durante la aplicación del instrumento de valoración en la organización y se acordó que la entrega de resultados se haría llegar al equipo investigador una semana después.
- Entrega del instrumento de evaluación y documentación de soporte. Se usó como instrumento un formulario de Google Forms que está disponible en <https://forms.gle/AGNB1MBJUa5cRT4P7> y las preguntas definidas en la Tabla 5-8.

#### 6.2.3.2. Ejecución de la evaluación

El evaluador seleccionó 6 colaboradores representantes de diferentes áreas claves en el proceso de adopción de prácticas de DevOps, los roles seleccionados por el evaluador según los cargos de la empresa fueron los siguientes: TechLead (líder técnico), SER (ingeniero de confiabilidad del sitio), SRE TechLead, Tech Lead de Arquitectura (líder técnico de arquitectura), QA Automation (Ingeniero de automatización de pruebas de calidad) e ingeniero DevOps. La selección de varios participantes en el proceso de valoración se ejecutó con el ánimo de realizar valoraciones desde diferentes puntos de vista e identificar la percepciones individuales y generales sobre los procesos de adopción de prácticas de DevOps de los cuales son responsables y sobre los que son responsables otros compañeros de los equipos de trabajo. Luego se contestaron las preguntas del instrumento de valoración proporcionado en formato de formulario de Google el cual permitía por cada pregunta contestar Si o No dependiendo de la percepción individual sobre el cumplimiento o no de la actividad y además podía agregar comentarios. El tiempo aproximado requerido para la evaluación fue variable para cada uno de los colaboradores involucrados, en la Tabla 6-12 se muestra un resumen de los tiempos invertidos por cada rol.

Tabla 6-12. Tiempos invertidos en ejecución de la valoración.

<b>Rol</b>	<b>Tiempo invertido</b>
TechLead	5 minutos
SRE	7 minutos
SRE Tech Lead	20 minutos
Tech Lead de Arquitectura	7 minutos
QA Automation	90 minutos
Ingeniero DevOps	120 minutos

### 6.2.3.3. Generación y reporte de resultados

Los datos obtenidos fueron muy variables y tenían como propósito dar una visión desde diferentes perspectivas sobre el estado actual de las prácticas de adopción de DevOps según DevOps Model. En esta actividad el evaluador remitió los resultados obtenidos por el instrumento al asesor quien usó esta información para crear un producto de trabajo denominado informe de evaluación. Los valores numéricos individuales y generales de las prácticas, principios, dimensiones y valores de DevOps evaluados se obtuvieron a partir de las especificaciones de medición y métricas propuestas en la Sección 5.6.

En la Tabla 6-13 se muestra la distribución y porcentaje del grado de cumplimiento individual y general de cada una de las doce prácticas fundamentales propuestas en DevOps Model, así como el valor del grado de cumplimiento correspondiente de acuerdo con la escala definida en base a la norma ISO/IEC 15504 y descrita en la Tabla 5-10. Los acrónimos utilizados para cada una de las prácticas corresponden a los identificadores definidos y detallados en la Tabla 5-3.

Tabla 6-13. Resumen del grado de cumplimiento de prácticas fundamentales.

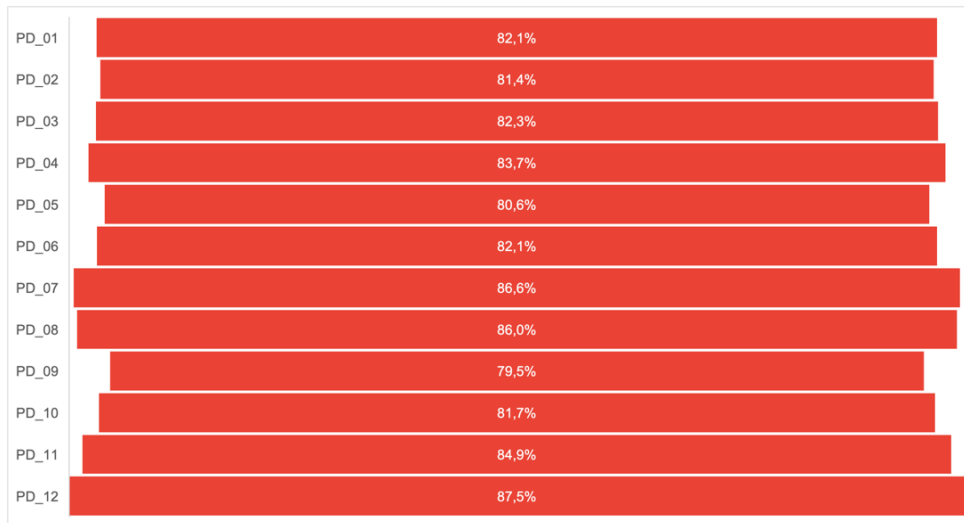
<b>IdPrac</b>	<b>ValC1</b>	<b>ValC2</b>	<b>ValC3</b>	<b>ValC4</b>	<b>ValC5</b>	<b>ValC6</b>	<b>Total Val</b>	<b>Grado Cumplimiento</b>
IC	100%	100%	86%	86%	71%	86%	88%	Completamente alcanzado
EC	100%	100%	100%	100%	100%	100%	100%	Completamente alcanzado
PC	88%	88%	100%	63%	13%	50%	67%	Ampliamente alcanzado
GR	100%	100%	100%	100%	50%	75%	88%	Completamente alcanzado
GD	88%	75%	75%	75%	75%	75%	77%	Ampliamente alcanzado
SS	83%	100%	83%	100%	100%	67%	89%	Completamente alcanzado
DE	100%	75%	100%	100%	50%	75%	83%	Ampliamente alcanzado
GC	100%	100%	50%	75%	50%	50%	71%	Ampliamente alcanzado
MC	67%	100%	100%	83%	83%	67%	83%	Ampliamente alcanzado
ED	100%	100%	100%	0%	50%	50%	67%	Ampliamente alcanzado
RC	100%	100%	100%	100%	100%	50%	92%	Completamente alcanzado
Mcu	100%	100%	100%	67%	100%	33%	83%	Ampliamente alcanzado

Acrónimos utilizados: **IdPrac**: Identificador de la práctica; **ValCX**: Valoración registrada por colaborador; **Total Val**: Valoración promedio de la empresa por práctica

Como se puede observar en la valoración realizada todas las prácticas definidas en DevOps Model tienen un nivel alto de cumplimiento en la organización, siendo las prácticas PC - pruebas continuas y la práctica ED - educación en torno a DevOps la de menor grado de cumplimiento con un 67% y la práctica con un grado de completamente alcanzada con un valor de 100% lo correspondiente a la práctica EC - entrega continua.

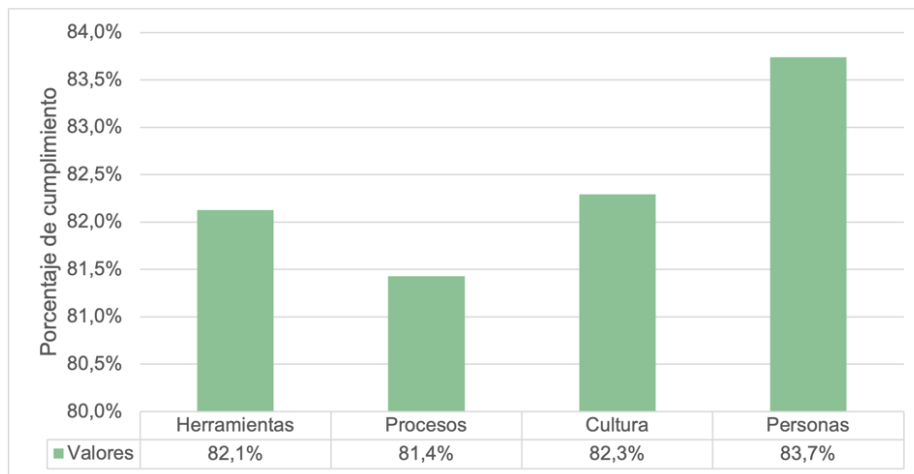
En la Figura 6-2 se puede observar el grado de cumplimiento general de cada uno de los principios definidos en DevOps Model, se puede observar que cada uno de ellos está por encima del 80%, lo que define un grado de cumplimiento *ampliamente alcanzado* según la escala definida. Los principios PD\_07, PD\_08 y PD\_12 alcanzaron un grado de cumplimiento *completamente alcanzado* por superar el umbral de 86%.

Figura 6-2. Grado de cumplimiento de principios de DevOps.



En la Figura 6-3 se puede observar el grado de cumplimiento general de cada una de las dimensiones definidas en DevOps Model, se puede observar que las cuatro dimensiones están en un nivel *ampliamente alcanzado*, siendo la dimensión de *Personas* la del porcentaje de cumplimiento más alto con el 83,7% y la dimensión de *procesos* con el más bajo porcentaje de cumplimiento con el 81,4%.

Figura 6-3. Grado de cumplimiento de dimensiones de DevOps.



En laTabla 6-14 se muestran cada uno de los valores definidos en DevOps Model y el grado de cumplimiento alcanzado por la empresa según los datos registrados por los seis colaboradores incluidos en el proceso por el evaluador. Se puede observar que todos los valores tienen un grado de cumplimiento *Ampliamente alcanzado*, donde el valor de la *Medición* en DevOps tiene el porcentaje de cumplimiento más alto con un 85% y los valores de *Automatización* y *Colaboración* tiene los porcentajes de cumplimiento más bajos con un 83%.

Tabla 6-14. Grado de cumplimiento de valores de DevOps.

Valor DevOps	Porcentaje obtenido	Grado de cumplimiento
Automatización	83%	Ampliamente alcanzado
Colaboración	83%	Ampliamente alcanzado
Medición	85%	Ampliamente alcanzado
Comunicación	84%	Ampliamente alcanzado

Con base a los datos obtenidos y aplicando las métricas respectivas se obtuvo un porcentaje de 84% en el grado de implementación de DevOps de acuerdo con los elementos definidos en DevOps Model, lo cual categoriza a la empresa en un grado *Ampliamente alcanzado*, además, como parte del producto de trabajo denominado informe de evaluación se identificaron 8 oportunidades de mejora respecto a las prácticas fundamentales definidas en DevOps Model, las cuales se listan en la Tabla 6-15. Estos resultados obtenidos permiten a la empresa llevar a cabo la definición de un plan general de mejora que permita mejorar todo un conjunto de prácticas y actividades que le permitan lograr plenamente la implementación de DevOps.

Tabla 6-15. Oportunidades de mejora encontradas en el estudio de caso.

IdPrac	gPF	Oportunidad de mejora
IC	88%	La práctica de integración continua es la base para el cumplimiento de otras prácticas de DevOps como entrega, pruebas y despliegue continuo por lo cual se recomienda estandarizar en todos los proyectos los procesos de pruebas unitarias locales antes de enviar integración y adicionalmente incorporar pruebas de unidad automáticas en los cambios realizados repositorio de código compartido.
PC	67%	Se recomienda formalizar e informar el proceso de pruebas de la organización incluyendo las pruebas sobre las API's (unidad, funcionalidad, carga, confiabilidad, documentación) y pruebas de rendimiento. Como la dinámica de la industria es muy variable y es necesario la adaptación se recomienda abandonar planes de pruebas generales e incorporarse en planes de pruebas de acuerdo con cada sprint y formalizar la documentación o registros sobre incidentes en la ejecución de pruebas.
GD	77%	Estandarizar y formalizar el protocolo, procedimiento, proceso o similar para el acceso a datos. Informar en los proyectos que aplique información referente a los modelos de datos sobre los que se trabaja y especialmente donde existen datos heredados.
SS	89%	Implementar procesos que permitan la detección temprana de vulnerabilidades y actualizar políticas de seguridad de acuerdo con los elementos detectados.
DE	83%	Socializar el plan estratégico con los colaboradores.
GC	71%	A medida que se avanza en la construcción de procesos que permiten llevar más rápido a producción cualquier producto es indispensable contar con todos los elementos de backup que permitan la recuperación rápida de los sistemas no solo a nivel de código fuente sino a nivel de infraestructura tecnológica donde se tenga caracterizado los activos de TI y sus configuraciones. Adicionalmente formalizar el proceso de gestión y divulgación al interior de los equipos sobre los incidentes, problemas y cambios.
MC	83%	Se recomienda hacer especial seguimiento a los productos ya desplegados en producción, implementar actividades que permitan recoger información del funcionamiento del sistema desde los logs de la aplicación y sistemas de recepción de feedback sobre la experiencia de los usuarios de la aplicación desplegada.
ED	67%	Incluir temas relacionados a DevOps y sus prácticas en los planes de capacitación de los colaboradores.

Acrónimos utilizados: **IdPrac**: Identificador de la práctica fundamental; **gPF**: grado de cumplimiento de la práctica fundamental.

## 6.2.4. Análisis de resultados del estudio de caso

### 6.2.4.1. Esfuerzo y costo

En la Tabla 6-16 y la Tabla 6-17 se presenta el esfuerzo y los costos asociados al estudio de caso tomando en cuenta los roles implicados en el proceso y el tiempo dedicado por cada uno de estos roles para realizar el proceso de valoración de las prácticas fundamentales, principios, valores y dimensiones de acuerdo con el modelo de referencia DevOps Model. Las actividades tomadas en cuenta para realizar el cálculo del esfuerzo fueron las siguientes: (1) planificación de la valoración, (2) ejecución de la valoración, (3) actividad de aclaración de dudas y (4) actividad de generación y socialización de resultados.

Respecto a los costos, se usó como referencia los valores promedio por hora en dólares americanos de los roles implicados de acuerdo a las tarifas definidas en [99] y según cada actividad desarrollada. De acuerdo con los valores obtenidos de esfuerzo y costo se puede pensar que el proceso de valoración realizado es aplicable a pequeñas y medianas empresas, que quieran iniciar la construcción de procesos de adopción de DevOps y/o desean valorar el estado actual de sus procesos para posteriormente iniciar la construcción de procesos de mejora. Lo anterior debido a que el esfuerzo y costo es relativamente bajo, esto en parte a que el instrumento de valoración se puede diligenciar rápidamente si se tiene el conocimiento y la información requerida, eliminando cualquier efecto negativo o sobrecarga considerable en las actividades diarias de las personas involucradas. Además, los resultados de las métricas se calculan rápidamente desde la hoja de cálculo generada.

Tabla 6-16. Relación de esfuerzo del estudio de caso.

Roles	Tiempo invertido por actividades programadas medido en minutos				Esfuerzo total
	Planificación de la valoración	Ejecución de la valoración	Aclaración de dudas	Generación y socialización de resultados	
VP of Engineering	55	10	2	20	82
TechLead	0	7	0	0	7
SRE	0	7	0	0	7
SRE Tech Lead	0	20	0	0	20
Tech Lead de Arquitectura	0	7	0	0	7
QA Automation	0	90	0	0	90
Ingeniero DevOps	55	120	0	0	175
Total del esfuerzo del estudio de caso					388 minutos

### 6.2.4.2. Análisis del método de valoración DevOps Assessment y modelo de referencia DevOps Model.

Una vez se finalizó el proceso de generación y socialización de resultados, se realizó una encuesta al participante que asumió el rol de evaluador con el fin de conocer su percepción general del proceso realizado respecto a la comprensibilidad, utilidad y practicidad de DevOps Model y DevOps Assessment Model.

Tabla 6-17. Relación de costos del estudio de caso.

<b>Roles</b>	<b>Costo hora promedio por rol en dólares americanos según [99]</b>	<b>Total costo por rol en dólares americanos</b>
VP of Engineering	72.13	98.57
TechLead	19.74	2.30
SRE	40.15	4.68
SRE Tech Lead	40.15	13.38
Tech Lead de Arquitectura	72.36	8.44
QA Automation	25.94	38.91
Ingeniero DevOps	47.69	139.1
Costo total del estudio de caso		305.38 dólares

El cuestionario en su parte inicial permitió caracterizar la organización participante en el estudio de caso, mediante la definición de: (i) nombre de la empresa, (ii) número de empleados, (iii) cantidad de años en funcionamiento, (iv) descripción del mercado y alcance de la empresa, (v) modelos y/o estándares usados actualmente en la empresa.

En la segunda parte del cuestionario el esfuerzo se enfocó en capturar la percepción del participante sobre DevOps Model y DevOps Assessment Model. En esta parte se aplicaron 17 preguntas diseñadas para ser respondidas utilizando la siguiente escala de Likert: 1- Muy mal, muy insatisfecho(a), 2 - Mal, poco satisfecho(a), 3 - Bien, suficiente, adecuado, algo satisfecho(a), 4 - Bastante bien, adecuado, satisfecho(a) y 5 - Muy bien, muy adecuado, muy satisfecho(a).

Con estas preguntas se busca medir la comprensibilidad (4 preguntas), la aplicabilidad (3 preguntas), la idoneidad (3 preguntas), la completitud (4 preguntas), idoneidad y completitud (1 pregunta) y idoneidad y aplicabilidad (2 preguntas) del método propuesto. Por último, se incluyeron 2 preguntas abiertas donde el participante podrá proponer ajustes al modelo de referencia y de valoración propuestos y realizar comentarios adicionales. A continuación, en la Tabla 6-18 se observa cada una de las preguntas realizadas y las respuestas proporcionadas por el representante de la organización quien estuvo ejerciendo el rol de encargado y además de evaluador del proceso durante el estudio de caso.

Tabla 6-18. Cuestionario de evaluación estudio de caso.

<b>Aspecto por evaluar</b>	<b>Id.</b>	<b>Pregunta</b>	<b>Resp.</b>
Comprensibilidad	P1	¿Considera que los valores de DevOps propuestos son de fácil comprensión?	5
	P2	¿Considera que los principios de DevOps propuestos son de fácil comprensión?	5
	P3	¿Considera que las dimensiones propuestas son de fácil comprensión?	5
	P4	¿Considera que las prácticas propuestas son de fácil comprensión?	5
Aplicabilidad	P5	De acuerdo con su experiencia: ¿Considera que los elementos definidos en el modelo de referencia son apropiados y pueden aplicarse con éxito en un proyecto de desarrollo de software?	5
	P16	¿Considera que el esfuerzo requerido para la aplicación del modelo de referencia es apropiado para una empresa de desarrollo de software?	5
	P17	¿Considera que el esfuerzo requerido para la aplicación del modelo de valoración es apropiado para una empresa de desarrollo de software?	5
Idoneidad	P6	¿Considera que los valores propuestos capturan la esencia principal de DevOps?	5
	P7	De acuerdo con su experiencia: ¿Considera que los principios definidos para DevOps son apropiados e interpretan correctamente las dinámicas actuales de la industria del Software?	5

	P13	¿Considera que el modelo propuesto sirve de referencia para proyectos de mejora que tienen por objetivo la definición de procesos relacionados con DevOps?	5
Completitud	P8	De acuerdo con su experiencia: ¿Considera que los principios definidos para DevOps son suficientes?	5
	P12	¿Considera que las prácticas propuestas son suficientes para un proceso de adopción de DevOps?	5
	P14	¿Considera que el modelo de valoración generado desde el modelo de referencia brinda elementos necesarios para llevar a cabo una valoración del porcentaje de implementación y adopción de DevOps?	5
	P15	¿Considera que el modelo de valoración generado desde el modelo de referencia brinda elementos que permitan identificar oportunidades de mejora relacionadas con los procesos de adopción de DevOps?	5
Idoneidad / Completitud	P9	¿Considera que las dimensiones definidas para DevOps agrupan correctamente los principios y prácticas definidas?	5
Idoneidad / Aplicabilidad	P10	¿Considera que las prácticas fundamentales propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?	5
Idoneidad / Aplicabilidad	P11	¿Considera que las prácticas complementarias propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?	5
<b>Preguntas abiertas</b>			
<b>Id.</b>	<b>Pregunta</b>		
P18	¿Considera que se deben agregar, eliminar o modificar elementos (valores, principios, dimensiones o prácticas) de la propuesta? <b>Respuesta:</b> ninguno		
P19	¿Tiene algún comentario adicional acerca del modelo de referencia propuesto? <b>Respuesta:</b> Muy completo y muy alineado con el mercado el modelo, nos permitió ver algunos detalles que debemos afinar para tener una mejor completitud en DevOps		

Acrónimos utilizados: **Id:** Identificador; **Resp:** Respuesta

A partir del análisis de las respuestas del cuestionario de evaluación descrito anteriormente, fue posible obtener las siguientes conclusiones con respecto a DevOps Model y DevOps Assessment Model:

- Respecto a la comprensibilidad se puede considerar que el modelo de referencia DevOps Model y el modelo de valoración asociado DevOps Assessment Model no presenta elementos muy complejos de entender lo cual favorece su entendimiento y posterior aplicación y los resultados generados también son de fácil comprensión, por lo que no se requiere un nivel avanzado de conocimiento por parte de las organizaciones para identificar los valores, principios, dimensiones y prácticas requeridos para la adopción de DevOps.
- DevOps Model y DevOps Assessment Model son elementos muy prácticos para la organización ya que son aplicables independiente de los modelos, estándares o marcos de trabajo implementados en la organización y adicionalmente el esfuerzo medido en tiempo y los costos asociados son razonables
- Los resultados obtenidos de la aplicación del modelo de valoración permiten a la organización identificar de manera objetiva el estado actual de la adopción de prácticas, principios, valores y dimensiones de DevOps y determinar a partir de estos datos planes de mejora que le permitan según su conveniencia mejorar procesos de adopción de DevOps.



### 6.2.5. Análisis de validez

A continuación, se describe el análisis de validez de constructo, la validez externa y de confiabilidad respecto a los resultados obtenidos en el estudio de caso.

- **Validez de constructo:** Para conservar la validez del constructo de modo que las medidas obtenidas en el estudio representen los aspectos que realmente se quieren medir y tengan coherencia con las preguntas de investigación, se realizó una reunión inicial donde se contextualizó el proyecto explicando el modelo de referencia y el método de valoración, el plan de trabajo para el estudio de caso y se resolvieron inquietudes. Adicionalmente, se proporcionó documentación detallada de apoyo para el evaluador para que pudiera profundizar en la propuesta y resolver dudas que se pudieran generar en el camino.
- **Validez externa:** No es posible generalizar los resultados del estudio de caso, ya que el mismo solo se realizó en una empresa con características que pueden ser particulares del modelo de negocio y posición en el mercado latinoamericano, sin embargo, sí es posible replicar este estudio en organizaciones de software mientras se sigan los criterios de selección sugeridos en la selección del sujeto de estudio, el procedimiento de campo y el proceso de recolección de datos establecidos en el diseño del estudio de caso.
- **Confiabilidad:** Para lograr independencia entre los resultados obtenidos y los investigadores que llevan a cabo el estudio de caso, se definieron las siguientes actividades: (i) definición de los criterios para la selección del sujeto de estudio, (ii) definición del procedimiento de campo y de recolección de datos, (iii) elaboración de documentación detallada del modelo de referencia y su modelo de valoración para ser usada por la organización en el estudio de caso, y (iv) soporte permanente al evaluador y participantes del estudio de caso durante la fase de ejecución.

### 6.2.6. Limitaciones del estudio de caso

Las limitaciones encontradas en la realización del presente estudio de caso se listan a continuación:

- Debido a la aplicación del estudio de caso en una sola empresa de desarrollo de software, se imposibilita la capacidad de generalizar los resultados obtenidos, por lo cual se hace necesario su aplicación en más empresas de desarrollo de software con características diversas.
- Existe un posible sesgo en la realización del estudio de caso respecto a la subjetividad del evaluador en la aplicación del instrumento de valoración y la posible subjetividad en la interpretación de los resultados por parte del equipo investigador.

# Capítulo 7. Conclusiones y trabajo futuro

A continuación, se presentan y evalúan los resultados obtenidos con el desarrollo de la presente investigación. Inicialmente, se describen las evidencias que permiten verificar el cumplimiento de los objetivos de investigación. Luego, se presentan los artículos desarrollados y publicados a partir de la realización de actividades dentro del proceso desarrollado en esta investigación. Finalmente, se presentan las conclusiones de la investigación y los trabajos futuros.

## 7.1. Análisis de los objetivos de investigación

En este apartado, se listan cada uno de los objetivos planteados en esta investigación y las sesiones dentro de este documento donde se puede evidenciar el trabajo realizado para dar cumplimiento a cada uno de ellos.

### 7.1.1. Objetivos específicos – OE

- **OE1:** Identificar y clasificar los elementos a tener en cuenta para la adopción de DevOps por medio de su identificación a través de un estudio de mapeo sistemático de la literatura que permitan clarificar y formalizar la implementación de este enfoque en la industria de software.

Para lograr el cumplimiento de este objetivo se realizó un mapeo sistemático de la literatura sobre los trabajos relacionados con procesos de adopción de DevOps en empresas de desarrollo de software, el cual se presenta en la Sección 2.2. Luego se realizó un análisis de los trabajos encontrados por medio de una caracterización de elementos de proceso asociados a la adopción de DevOps, donde se lograron identificar procesos, actividades, artefactos, roles, herramientas, beneficios y desafíos. Esta caracterización se presenta en el Capítulo 3.

- **OE2:** Definir una ontología de dominio específico que permita entender y organizar el conocimiento relacionado con los términos y relaciones alrededor de la implementación de DevOps en empresas de desarrollo de software.

A partir de la caracterización de elementos de proceso, se construyó DevOps Ontology, la cual es una ontología de dominio que permite organizar el conocimiento relacionado con la adopción de DevOps en empresas de desarrollo de software, esta ontología se detalla el Capítulo 4.

- **OE3:** Definir un modelo de referencia que facilite la adopción e implementación de DevOps y que esté basado en los elementos identificados y/o definidos a través de la realización del objetivo específico anterior.

A partir del mapeo sistemático realizado, la caracterización de procesos y la definición de la ontología se construye DevOps Model, el cual es un modelo de referencia para la adopción de DevOps en empresas de desarrollo de software, el cual está detallado en el Capítulo 5. DevOps Ontology tiene cuatro componentes fundamentales: (i) valores de DevOps, (ii) principios de DevOps, (iii) dimensiones de DevOps, y (iv) prácticas fundamentales y complementarias. Los valores de DevOps representan los principales atributos que debe tener un proceso de desarrollo de software en el marco de la adopción de DevOps, los principios establecen normas o reglas de carácter general que orientan las acciones de las organizaciones, las dimensiones caracterizan las prácticas y las prácticas materializan todas las actividades requeridas en el contexto de la adopción de DevOps integrando adicionalmente tareas, roles, productos, herramientas tecnológicas, entre otros.

- **OE4:** Evaluar el modelo de referencia a través de su aplicación en un Focus Group para obtener realimentación por parte de profesionales expertos en el área.

En la Sección 6.1 se presentan las actividades realizadas y los resultados obtenidos producto de la realización de un grupo focal con el que se realiza el primer proceso de evaluación de la propuesta y a partir del cual se encontraron oportunidades de mejora que permitieron refinar del modelo de referencia propuesto.

Teniendo en cuenta que DevOps Model es un modelo de referencia, este no describe el proceso requerido para realizar una valoración. En este sentido, en la Sección 5.6 se presenta DevOps Assessment Model el cual es un método de valoración de la adopción de DevOps en empresas de desarrollo de software. Este método define un proceso de valoración aplicando el paradigma GQM sobre DevOps Model y basado en PvalCOMPETISOFT, el cual describe en detalle la forma de realizar evaluaciones de procesos y es fácilmente adaptable a cualquier instrumento de evaluación para evaluar características específicas de los procesos de una organización.

- **OE5:** Realizar un estudio de caso que permita evaluar el modelo de referencia propuesto a través de la definición de un proceso de DevOps para una empresa de software de la región.

En la Sección 6.2, se presenta la evaluación de DevOps Model y DevOps Assessment Model mediante un estudio de caso en una empresa de desarrollo de software. En dicho estudio se planteó verificar si los componentes de DevOps Model eran de fácil comprensión, si los resultados de la aplicación del modelo de valoración derivado de DevOps Model proporcionaba información útil a las organizaciones para mejorar sus procesos de adopción de DevOps y si el esfuerzo requerido para su aplicación era aceptable para las empresas desarrolladoras de software.

## 7.1.2. Objetivo general – OG

**OG:** Proponer un modelo de referencia que formalice un conjunto de elementos de proceso que sean fundamentales para tener en cuenta para la implementación de DevOps en las empresas de desarrollo de software mediante su identificación a través del análisis de la literatura existente.

En el Capítulo 5, se describe DevOps Model que es un modelo de referencia para la adopción de DevOps en empresas de desarrollo de software, donde se formalizan un conjunto de elementos de proceso fundamentales para la implementación de DevOps a partir de la caracterización de elementos de procesos definida en el Capítulo 3 y la definición de una ontología descrita en el Capítulo 4. Adicionalmente, con la definición del modelo de valoración de la Sección 5.6 se logra ampliar el alcance del modelo de referencia y lo convierte en una versión inicial de un Framework para soportar el proceso de adopción de DevOps en empresas de desarrollo de software.

## 7.2. Publicaciones

Los resultados relacionados con el mapeo sistemático de la literatura, la caracterización de elementos de proceso y la definición de la ontología para la adopción de DevOps en empresas de desarrollo de software, fueron publicados en revistas indexadas.

- Artículo presentado en el evento internacional XIV Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento IIISIC'19 en Costa Rica donde obtuvo el reconocimiento de mejor paper del evento con la siguiente referencia: J. Guerrero, C. Certuche, K. Zúñiga, and C. Pardo, "What is there about DevOps? Preliminary Findings from a Systematic Mapping Study," 2019.
- Artículo publicado en revista indexada categoría B según PUBLINDEX de Colciencias con la siguiente referencia: J. Guerrero, C. Certuche, K. Zúñiga, and C. Pardo, "Tendencias en DevOps: un mapeo sistemático de la literatura," 2020.
- Artículo publicado en revista indexada categoría C según PUBLINDEX de Colciencias con la siguiente referencia: J. Guerrero, K. Zúñiga, C. Certuche, and C. Pardo, "A systematic mapping study about DevOps," *J. Cienc. e Ing.*, vol. 12, no. 1, pp. 48–62, 2020, doi: <https://doi.org/10.46571/JCI.2020.1.5>.
- Artículo enviado a una revista indexada categoría Q2 con la descripción de la ontología DevOps Ontology.
- Artículo enviado del modelo de referencia y artículo del modelo de evaluación en el que se presentan los resultados del estudio de caso.

### 7.3. Conclusiones

- El mapeo sistemático de la literatura sobre los procesos de adopción de DevOps en empresas de desarrollo de software permitió evidenciar: (i) la inexistencia de homogeneidad respecto a la definición de DevOps, y (ii) falta de estandarización respecto a conceptos, definiciones y relaciones asociadas a un proceso de adopción de DevOps.
- La caracterización de elementos de procesos siguiendo el modelo de referencia PrMO a partir de un mapeo sistemático de la literatura, permitió categorizar 9 procesos, 106 actividades, 4 productos y gran cantidad de recursos entre los que se destacan 16 roles y 60 herramientas que fueron claves para la organización del conocimiento existente.
- La definición de DevOps Ontology establece un vocabulario que permite organizar el conocimiento encontrado en la literatura de forma genérica y formal, reduciendo la inconsistencia, integridad y ambigüedad de los conceptos, términos y relaciones dentro del contexto de investigación.
- El uso de la metodología REFSENO fue importante para la definición de la ontología DevOps Ontology, la cual es fácil de interpretar para el entorno de la Ingeniería de Software a través de su representación en lenguaje UML y la forma de presentación tabular de los conceptos y relaciones.
- Las diferentes estrategias de evaluación de la ontología DevOps Ontology permitió comprobar que: (i) tiene una estructura consistente que permite la instanciación de diferentes propuestas de procesos y prácticas de DevOps en la industria del software, (ii) puede ser utilizada como base conceptual para la definición de un modelo de referencia para la adopción y evaluación de DevOps en empresas de desarrollo de software, (iii) puede ser usada como fuente de información para una aplicación móvil que use el protocolo y lenguaje de consulta RDF (SPARQL) y (iv) es capaz de responder la preguntas de competencia definidas en la etapa de diseño. Sin embargo, queda como trabajo futuro incluir algoritmos de inferencia que permitan generar un sistema autónomo en la valoración del cumplimiento de adopción de DevOps.
- DevOps Model se construyó a partir de varios elementos: (i) un mapeo sistemático de la literatura entorno a procesos de adopción de DevOps en la industria del desarrollo de software, (ii) el análisis a partir de la caracterización de los elementos de procesos encontrados, (iii) la ontología DevOps Ontology, y (iv) los valores y principios ágiles definidos en el manifiesto ágil, por lo anterior el modelo de referencia propuesto es un documento que parte de referencias altamente aceptadas por la comunidad de desarrollo de software.
- La realización de un grupo focal para la evaluación el modelo de referencia DevOps Model, permitió reunir a diferentes expertos en el área de ingeniería de software involucrados en procesos de adopción de DevOps, los cuales desde sus puntos de vista y experiencias en la industria compartieron observaciones y sugerencias que permitieron identificar oportunidades de mejora a la propuesta, las cuales generaron ajustes para obtener una versión mejorada de la propuesta.
- El estudio de caso diseñado permitió identificar que el modelo de referencia y el modelo de valoración propuestos son de fácil comprensión, no requieren roles con niveles técnicos avanzados, es independiente de los modelos, estándares o marcos de

trabajo implementados en la organización y el esfuerzo requerido para su aplicación en función del tiempo y esfuerzos son totalmente razonables.

- Aunque el estudio de caso fue diseñado tratando de conseguir validez de constructo, validez externa y confiabilidad, tiene las siguientes limitaciones: (i) no es posible generalizar los resultados para toda la industria del software ya que se requiere de un mayor número de aplicaciones con empresas desarrolladores de software con diferentes características, como por ejemplo, empresas con marcos de trabajo tradicionales, marcos de trabajo híbridos, enfoques escalados y equipos distribuidos, y (ii) la existencia de un posible sesgo debido a la subjetividad del evaluador en la aplicación de los instrumentos de valoración y en la interpretación de los resultados generados por parte del equipo investigador.
- Como respuesta a la pregunta de investigación formulada al principio de esta investigación: “¿*Qué recomendaciones, atributos y/o elementos fundamentales deberían considerarse para facilitar la correcta implementación y formalización de DevOps en las empresas de desarrollo de software?*”, se obtuvo como resultado el “DevOps Model - Modelo de referencia para la adopción de DevOps en empresas desarrolladores de software”, el cual consta de dos capas: (i) conceptual y (ii) metodológica. En la capa conceptual, define un modelo de referencia que describe un conjunto de valores, principios, dimensiones y un conjunto de prácticas fundamentales y complementarias. Las relaciones descritas surgieron a partir de la definición de DevOps Ontology, una ontología de dominio que permite organizar el conocimiento relacionado con la adopción de DevOps en procesos de desarrollo de software. En la capa metodológica, define el método de valoración DevOps Assessment Model que describe el proceso necesario para valorar el cumplimiento en la adopción de DevOps en empresas de desarrollo de software. Adicionalmente se proporciona un prototipo de una aplicación móvil que permite la visualización de conceptos y relaciones definidas en DevOps Ontology y que son el soporte sobre el que se fundamenta la propuesta.

## 7.4. Trabajo Futuro

A continuación, se presentan los trabajos futuros relacionados con esta investigación:

- **Actualización del mapeo sistemático:** Debido a que este fue realizado en las primeras etapas de construcción del proyecto y se alimentó con la literatura existente hasta mediados del año 2020, por lo cual es necesario la realización de una actualización que permita encontrar nuevas propuestas y líneas de investigación relacionadas al tema de adopción de DevOps en empresas de desarrollo de software.
- **Actualización de DevOps Ontology:** Es necesario ampliar la evaluación realizada a la ontología con la instanciación de más prácticas de DevOps, ya sea a partir de propuestas de otras organizaciones reconocidas en implementación de prácticas de DevOps o por medio de estudios de caso en organizaciones que desee instanciar sus procesos internos. Lo anterior permitirá verificar la necesidad de extender o no la ontología a más conceptos y relaciones.
- **Realización de nuevos estudios de caso:** Debido a que en esta investigación solo se realizó un estudio de caso con una empresa desarrolladora de software, es necesario

hacer el mismo proceso con otras organizaciones con diversas características y tamaños, lo cual permitirá identificar tendencias y eventualmente generalizar los resultados obtenidos.

- **Automatización de la evaluación:** El modelo de valoración generado a partir de DevOps Model es un proceso manual basado en instrumentos tipo cuestionarios. Una mejora significativa hacia el futuro sería la implementación de algoritmos que permitan la realización del proceso de valoración de forma periódica y automatizada.
- **Extender el modelo de valoración:** Es necesario ampliar el conjunto de objetivos, preguntas y métricas desde la perspectiva de procesos específicos para el desarrollo de cada una de las prácticas fundamentales y complementarias propuestas para que el modelo pase de una valoración a una evaluación concreta.

# Capítulo 8. Bibliografía

- [1] M. Snapp and J. Vinopal, “DevOps is Bigger than IT: Driving Digital Transformation in Libraries.,” Columbus, 2020.
- [2] Stackify, “Defining the Ops in DevOps,” *Divvy Up DevOps Tasks*, 2020. <http://bit.ly/2ZawgTY> (accessed Sep. 12, 2021).
- [3] Dev2ops.org, “What is DevOps?,” 2021. <https://bit.ly/1pB9cF8> (accessed Sep. 12, 2021).
- [4] M. Fowler, “The New Methodology,” 2005. <https://bit.ly/38ZgGBR> (accessed Sep. 12, 2021).
- [5] I. Sommerville, *Software engineering*, [9a ed.]. Boston: Addison-Wesley, 2010.
- [6] R. S. Pressman, *Ingeniería del software: un enfoque práctico*, Séptima ed. México: McGraw-Hill, 2010.
- [7] Digital.ai, “14th Annual State of Agile Report,” 2020. [Online]. Available: <https://stateofagile.com/#ufh-i-615706098-14th-annual-state-of-agile-report/7027494>.
- [8] H. Takeuchi and I. Nonaka, “The new new product development game,” *Harv. Bus. Rev.*, vol. 64, pp. 137–146, 1986.
- [9] K. Beck and E. Gamma, “Extreme Programming Explained: Embrace Change,” *Addison Wesley*, 2000. <https://bit.ly/2L6KPFa> (accessed Sep. 12, 2021).
- [10] A. Cockburn, “Crystal Clear: A Human-Powered Methodology for Small Teams.,” *Pearson Education*, 2004. <https://bit.ly/32e6jFx> (accessed Sep. 12, 2021).
- [11] M. P. and T. Poppendieck, “Lean Software Development: An Agile Toolkit,” *The Agile Software Development Series*, 2003. <https://bit.ly/2xB3tFf> (accessed Sep. 12, 2021).
- [12] J. A. Highsmith, “Adaptive Software Development: A Collaborative Approach to Managing Complex Systems,” vol. 12, 2000.
- [13] J. Stapleton, “Dynamic Systems Development Method: The Method in Practice.,” 1997.
- [14] S. Palmer and J. Felsin, “A Practical Guide to Feature Driven Development,” 2002.
- [15] S. W. Ambler, “Agile Unified Process,” 2005. <https://bit.ly/3tza0DV> (accessed Sep. 12, 2021).
- [16] Oxosoft, “KanBan Fundamentals.,” 2021. <https://bit.ly/38Y7wWa> (accessed Sep. 12, 2021).
- [17] D. Gornik, “Rational Unified Process,” 2003. [Online]. Available: <https://ibm.co/2Xafysf>.
- [18] L. Geoffrey, C. Derick, and R. Allison, “Microsoft Solutions Framework v3 Overview,” 2003.
- [19] Carnegie Mellon University, “CMMI® for Development, Version 1.2,” 2006.
- [20] I. ICONIX Software Engineering, “Iconix,” 2016. <http://www.iconixsoftware.net> (accessed Sep. 12, 2021).
- [21] B. Boehm, D. Port, A. Egyed, Abi-Antoun, and M. T., “The MBASE Life Cycle Architecture Milestone Package,” in *Software Architecture: TC2 First Working IFIP*



- Conference on Software Architecture (WICSA1)*, 1999, pp. 511–528.
- [22] A. Hochstein, R. Zarnekow, and W. Brenner, “ITIL as common practice reference model for IT service management: formal assessment and implications for practice.,” *2005 IEEE Int. Conf. e-Technology, e-Commerce e-Service*, pp. 704-710., 2005.
- [23] J. Young, G. Ridley, and P. Carroll, “COBIT and Its Utilization: A Framework from the Literature,” *47th Hawaii Int. Conf. Syst. Sci.*, p. 80233., 2014.
- [24] ISO/IEC, “Calidad de los servicios TI,” 2019. <https://bit.ly/2HWudxi> (accessed Sep. 12, 2019).
- [25] M. A. McCarthy, L. M. Herger, S. M. Khan, and B. M. Belgodere, “Composable DevOps: Automated Ontology Based DevOps Maturity Analysis,” in *2015 IEEE International Conference on Services Computing*, Jun. 2015, pp. 600–607, doi: 10.1109/SCC.2015.87.
- [26] MediaOps Inc, “Where the world meets DevOps,” 2019. <https://devops.com> (accessed Aug. 05, 2019).
- [27] B. de França, H. Jeronimo, and G. Travassos, *Characterizing DevOps by Hearing Multiple Voices*. New York, NY: Association for Computing Machinery, 2016.
- [28] M. Senapathi, J. Buchan, and H. Osman, “DevOps Capabilities, Practices, and Challenges,” in *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 - EASE’18*, 2018, pp. 57–67.
- [29] L. Riungu, S. Mäkinen, L. Lwakatare, J. Tiihonen, and T. Männistö, “DevOps Adoption Benefits and Challenges in Practice: A Case Study,” M. T. (eds) Abrahamsson P., Jedlitschka A., Nguyen Duc A., Felderer M., Amasaki S., Ed. New York, New York, USA: Springer, Cham, 2016, pp. 590–597.
- [30] G. Ghantous and A. Gill, “DevOps: Concepts, Practices, Tools, Benefits and Challenges,” 2017, [Online]. Available: <http://bit.ly/3bIJdvk>.
- [31] C. VersionOne, “The 13th annual State of Agile Report,” pp. 1–15, 2019.
- [32] S. Jones, J. Noppen, and F. Lettice, “Management challenges for DevOps adoption within UK SMEs,” in *Proceedings of the 2nd International Workshop on Quality-Aware DevOps - 2016*, 2016, pp. 7–11.
- [33] L. Lwakatare, P. Kuvaja, and M. Oivo, “Dimensions of DevOps,” in *Agile Processes in Software Engineering and Extreme Programming*, & P. M. Lassenius C., Dingsøyr T., Ed. Helsinki, Finland: Springer, Cham, 2015, pp. 212–217.
- [34] M. Kamuto and J. Langerman, “Factors inhibiting the adoption of DevOps in large organisations: South African context,” in *2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2017, pp. 48–51.
- [35] I. Bucena and M. Kirikova, “Simplifying the DevOps Adoption Process,” 2017.
- [36] F. Erich, C. Amrit, and M. Daneva, “Report: DevOps Literature Review,” 2014. .
- [37] L. Chen, “Continuous Delivery: Overcoming adoption challenges,” *J. Syst. Softw.*, vol. 128, pp. 72–86, Jun. 2017.
- [38] J. Guerrero, K. Zuñiga, C. Certuche, and C. Pardo, “A systematic mapping study about DevOps,” *J. Cienc. e Ing.*, vol. 12, no. 1, pp. 48–62, 2020, doi: <https://doi.org/10.46571/JCI.2020.1.5>.

- [39] S. S. Samarawickrama and I. Perera, "Continuous scrum: A framework to enhance scrum with DevOps," in *2017 Seventeenth International Conference on Advances in ICT for Emerging Regions*, 2017, pp. 1–7.
- [40] S. Krusche, L. Alperowitz, B. Bruegge, and M. Wagner, "Rugby: an agile process model based on continuous delivery," in *RCoSE 2014 - Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering*, 2014, pp. 42–50.
- [41] S. Klepper, S. Krusche, S. Peters, B. Bruegge, and L. Alperowitz, "Introducing Continuous Delivery of Mobile Apps in a Corporate Environment: A Case Study," in *015 IEEE/ACM 2nd International Workshop on Rapid Continuous Software Engineering*, 2015, pp. 5–11.
- [42] J. Wettinger, U. Breitenbücher, and F. Leymann, "DevOpsSlang – Bridging the Gap between Development and Operations," & L. K. (eds) Villari, M., Zimmermann, W., Ed. Berlin, Heidelberg: Springer, 2014, pp. 108–122.
- [43] R. L. Baskerville, "Investigating Information Systems with Action Research," *Commun. Assoc. Inf. Syst.*, vol. 2, 1999, doi: 10.17705/1CAIS.00219.
- [44] M. Mendoza-Moreno, C. González-Serrano, and F. Pino, "Focus group como proceso en ingeniería de software: una experiencia desde la práctica," *DYNA, Universidad Nacional de Colombia*, pp. 51–60, 2013.
- [45] D. Stewart, P. Shamdasani, and D. Rook, *Focus Groups*. 2455 Teller Road, Thousand Oaks California 91320 United States of America: SAGE Publications, Ltd., 2007.
- [46] T. Hollweck, "Robert K. Yin. (2014). Case Study Research Design and Methods (5th ed.). Thousand Oaks, CA: Sage. 282 pages.," *Can. J. Progr. Eval.*, Mar. 2016, doi: 10.3138/cjpe.30.1.108.
- [47] B. Kitchenham, "Systematic review in software engineering," in *Proceedings of the 2nd international workshop on Evidential assessment of software technologies - EAST '12*, 2012, pp. 1–2.
- [48] R. Jabbari, N. Bin, K. Petersen, and B. Tanveer, "What is DevOps?," in *PXP '16 Workshops: Scientific Workshop Proceedings of XP2016*, 2016, pp. 1–11.
- [49] D. Pajk, M. Indihar Štemberger, and A. Kovačič, "Reference Model Design: An Approach and its Application," in *Information Technology Interfaces (ITI), ITI 2012 34th*, 2012, pp. 455–460, doi: 10.2498/iti.2012.0419.
- [50] H. A. Reijers, "Process Design and Redesign," in *Process-Aware Information Systems: Bridging People and Software Through Process Technology*, 2005, pp. 207–234.
- [51] OASIS®, "OASIS SOA Reference Model (SOA-RM) TC," 2020. <https://bit.ly/2Xi8DOA> (accessed Sep. 12, 2021).
- [52] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?," *Int. J. Hum. Comput. Stud.*, vol. 43, no. 5–6, pp. 907–928, Nov. 1995, doi: 10.1006/ijhc.1995.1081.
- [53] F. Ruiz and J. R. Hilera, "Using Ontologies in Software Engineering and Technology," in *Ontologies for Software Engineering and Software Technology*, Springer Berlin Heidelberg, pp. 49–102.
- [54] V. Basili, *Software modeling and measurement: the Goal/Question/Metric*

- paradigm*. Maryland, United States: University of Maryland at College Park, 1992.
- [55] J. Guerrero, C. Certuche, K. Zúñiga, and C. Pardo, "What is there about DevOps? Preliminary Findings from a Systematic Mapping Study," 2019.
  - [56] J. Paton-Romero and M. Piattini, "Green IT maturity models: A systematic mapping study," in *2017 12th Iberian Conference on Information Systems and Technologies (CISTI)*, 2017, pp. 1–6.
  - [57] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," in *EASE'08 Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering*, 2008, pp. 68–77.
  - [58] D. Budgen, M. Turner, P. Brereton, and B. Kitchenham, "Using mapping studies in software engineering.," in *PPIG 2008*, 2008, pp. 195–204.
  - [59] M. Genero, L. Cruz, and M. Piattini, *Métodos de investigación en ingeniería del software*. Bogotá, DC: Grupo Editorial ra-ma, 2014.
  - [60] M. Martins, "Devopsdays Organizing Guide," 2018. <http://bit.ly/3bEPpVr>.
  - [61] F. Erich, C. Amrit, and M. Daneva, "A qualitative study of DevOps usage in practice," *J. Softw. Evol. Process*, vol. 29, pp. 1–25, Jun. 2017.
  - [62] V. Gupta, P. Kapur, and D. Kumar, "Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling," *Inf. Softw. Technol.*, vol. 92, pp. 75–91, Dec. 2017.
  - [63] F. Elberzhager, T. Arif, M. Naab, I. Süß, and K. Sener, *From Agile Development to DevOps: Going Towards Faster Releases at High Quality – Experiences from an Industrial Context*. Vienna, Austria: Springer, Cham, 2017.
  - [64] G. Benguria, J. Alonso, I. Etxaniz, L. Orue-Echevarria, and M. Escalante, "Agile Development and Operation of Complex Systems in Multi-technology and Multi-company Environments: Following a DevOps Approach," in *European Conference on Software Process Improvement EuroSPI 2018*, 2018, pp. 15–27.
  - [65] A. Nagarajan and S. Overbeek, "A DevOps Implementation Framework for Large Agile-Based Financial Organizations," in *On the Move to Meaningful Internet Systems. OTM 2018 Conferences*, M. R. (eds) Panetto H., Debruyne C., Proper H., Ardagna C., Roman D., Ed. Utrecht, Netherlands: Springer, Cham, 2018, pp. 172–188.
  - [66] L. Lwakatare *et al.*, "DevOps in practice: A multiple case study of five companies," *Inf. Softw. Technol.*, vol. 114, pp. 217–230, Oct. 2019.
  - [67] J. Cusick, "A Survey of Maturity Models from Nolon to DevOps and Their Applications in Process Improvement.," 2019.
  - [68] A. Hemon, B. Lyonnet, F. Rowe, and B. Fitzgerald, "From Agile to DevOps: Smart Skills and Collaborations," *Inf. Syst. Front.*, pp. 1–19, Mar. 2019.
  - [69] D. Teixeira, R. Pereira, T. A. Henriques, M. Silva, and J. Faustino, "A Systematic Literature Review on DevOps Capabilities and Areas," *Int. J. Hum. Cap. Inf. Technol. Prof.*, vol. 11, no. 2, pp. 1–22, Apr. 2020, doi: 10.4018/IJHCITP.2020040101.
  - [70] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A Survey of DevOps Concepts and Challenges," *ACM Comput. Surv.*, vol. 52, no. 6, pp. 1–35, Jan. 2020, doi: 10.1145/3359981.
  - [71] M. Sánchez-Gordón and R. Colomo-Palacios, "Characterizing DevOps Culture: A

- Systematic Literature Review,” in *Software Process Improvement and Capability Determination - SPICE*, 2018, pp. 15–28, doi: <https://doi.org/10.1007/978-3-030-00623-5>.
- [72] C. Pardo, F. García, M. Piattini, F. Pino, and M. Baldassarre, “A Reference Ontology for Harmonizing Process-reference Models,” *Rev. Fac. Ing. Univ. Antioquia*, vol. 73, pp. 29–42, 2014.
- [73] D. Fensel, “Ontology-based knowledge management,” *Computer (Long. Beach. Calif.)*, vol. 35, no. 11, pp. 56–59, Nov. 2002, doi: 10.1109/MC.2002.1046975.
- [74] M. Fernandez-lopez, Gomez-Perez.A, and J. N, “METHONTOLOGY: from Ontological Art towards Ontological Engineering,” *Proceedings of the AAAI97 Spring Symposium*, Stanford, USA, pp. 33–40, 1997.
- [75] T. R. Gruber, “A translation approach to portable ontology specifications,” *Knowl. Acquis.*, vol. 5, no. 2, pp. 199–220, Jun. 1993, doi: 10.1006/knac.1993.1008.
- [76] T. Hikita and M. . Matsumoto, “Business process modelling based on the ontology and first-order logic,” *Proc. 3rd Int. Conf. Enterp. Inf. Syst. - ICEIS*, pp. 717–723, 2001.
- [77] C. Tautz and C. G. Wangenheim, “REFSENO: a representation formalism for software engineering ontologies,” 1998.
- [78] F. García *et al.*, “Towards a consistent terminology for software measurement,” *Inf. Softw. Technol.*, vol. 48, no. 8, pp. 631–644, Aug. 2006, doi: 10.1016/j.infsof.2005.07.001.
- [79] R. Almeida, I. Percheiro, C. Pardo, and M. M. da Silva, “An Ontology-Based Model for ITIL Process Assessment Using TIPA for ITIL,” 2018, pp. 104–118.
- [80] W3.org, “OWL 2 Web Ontology Language Document Overview (Second Edition),” 2020. <https://www.w3.org/TR/owl2-overview/> (accessed Nov. 30, 2020).
- [81] T. B. of T. of the L. S. J. University, “Protégé,” *Stanford university*, 2020. <https://protege.stanford.edu> (accessed Nov. 30, 2020).
- [82] D. and knowledge group-department of computer Science, “Hermit OWL Reasoner,” *University of Oxford*, 2020. <http://www.hermit-reasoner.com/>.
- [83] GitLab, “Introduction to CI/CD with GitLab,” 2020. <https://docs.gitlab.com/ee/ci/introduction/index.html> (accessed Nov. 30, 2020).
- [84] A. A. Khan and M. Shameem, “Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process,” *J. Softw. Evol. Process*, vol. 32, no. 10, Oct. 2020, doi: 10.1002/smr.2263.
- [85] N. Forsgren, M. A. Rothenberger, J. Humble, J. B. Thatcher, and D. Smith, “A Taxonomy of Software Delivery Performance Profiles. Investigating the Effects of DevOps Practices,” 2020.
- [86] S. Rafi, W. Yu, M. A. Akbar, A. Alsanad, and A. Gumaei, “Prioritization Based Taxonomy of DevOps Security Challenges Using PROMETHEE,” *IEEE Access*, vol. 8, pp. 105426–105446, 2020, doi: 10.1109/ACCESS.2020.2998819.
- [87] M. A. Akbar *et al.*, “Prioritization Based Taxonomy of DevOps Challenges Using Fuzzy AHP Analysis,” *IEEE Access*, vol. 8, pp. 202487–202507, 2020, doi: 10.1109/ACCESS.2020.3035880.
- [88] R. W. Macarthy and J. M. Bass, “An Empirical Taxonomy of DevOps in Practice,” in

- 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Aug. 2020, pp. 221–228, doi: 10.1109/SEAA51224.2020.00046.
- [89] C. VersionOne, “The 13th annual state of Agile Report,” pp. 1–15, 2019.
  - [90] K. Schwaber and J. Sutherland, “The scrum guide the definitive guide to scrum: The rules of the game.,” 2017. <https://bit.ly/2F7HKxd> (accessed Mar. 20, 2019).
  - [91] C. Ladas, “Scrumban-essays on kanban systems for lean software development,” 2009. <http://bit.ly/2lXQcLV> (accessed Mar. 20, 2019).
  - [92] K. Beck *et al.*, “Manifesto for Agile Software Development,” 2001. .
  - [93] J. Sandobalín, M. Zúñiga, E. Insfran, S. Abrahão, and C. Cano, “Una aproximación DevOps para el Desarrollo Dirigido por Modelos de Servicios Cloud,” 2016.
  - [94] Google, “DORA’s research program,” 2018. <https://www.devops-research.com/research.html> (accessed Feb. 03, 2021).
  - [95] F. J. Pino, C. Pardo, F. García, and M. Piattini, “Assessment methodology for software process improvement in small organizations,” *Inf. Softw. Technol.*, vol. 52, no. 10, pp. 1044–1061, Oct. 2010, doi: 10.1016/j.infsof.2010.04.004.
  - [96] V. R. Basili, G. Caldiera, and H. D. Rombach, “Software modeling and measurement: The Goal, metric, and question paradigm,” *NASA/GSFC AND AFOSR*, 1992.
  - [97] M. Mendoza, C. González, and F. Pino, “Focus Group como Proceso en la Ingeniería de Software: Una Experiencia desde la Práctica,” *Dyna*, vol. 80, no. 1, pp. 51–60, 2013.
  - [98] P. Runeson and M. Höst, “Guidelines for conducting and reporting case study research in software engineering,” *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, Apr. 2009, doi: 10.1007/s10664-008-9102-8.
  - [99] I. PayScale, “PayScale - Salary Comparison, Salary Survey, Search Wages,” 2021. <https://www.payscale.com> (accessed May 27, 2021).

# Anexo A. Mapeo sistemático

**A systematic mapping study about DevOps**  
**Un estudio de mapeo sistemático sobre DevOps**

**Jonathan Guerrero, Karen Zúñiga, Camilo Certuche and César Pardo Grupo de Investigación GTI, Programa de Ingeniería de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, Popayán, Colombia**

**Abstract.** DevOps is a very trendy term these days in the software development companies (SDC), term that emerges as a possible solution to finally reach an acceleration and a rise of productivity expected with the appearance of agile development approaches, but that until now had not materialized, through automation practices, continuous integration, continuous build and continuous deployment. This paper aims to show current knowledge about the process of adopting DevOps in SDC through a systematic mapping of the literature. However, the results obtained show that there is little detailed information regarding activities, tasks, roles and other important process elements for the adoption of DevOps. Similarly, it has been concluded that there is no a unified terminology, therefore, it is important to standardize it to simplify the understanding and application of DevOps. Furthermore, this paper shows the preview of the framework that it is being developed for the adoption of DevOps in the SDC.

**Keywords.** Software development; software operations, DevOps, mapping systematic study.

**Resumen.** DevOps es un término muy de moda en estos días en las empresas de desarrollo de software (EDS), un término que surge como una posible solución para finalmente alcanzar una aceleración y un aumento en la productividad, que se esperaban con la aparición de los enfoques de desarrollo ágiles, pero que hasta ahora no se había materializado, a través de prácticas de automatización, integración continua, compilación continua y despliegue continuo. Este documento tiene como objetivo mostrar el conocimiento actual sobre el proceso de adopción de DevOps en EDS a través de un mapeo sistema 'tico de la literatura. in embargo, los resultados obtenidos muestran que hay poca información detallada sobre actividades, tareas, roles y otros elementos de proceso importantes para la adopción de DevOps. Asimismo, se ha concluido que no hay una terminología unificada, por lo tanto, es importante homogeneizarla para facilitar la comprensión y la aplicación de DevOps. Adicionalmente, este documento muestra una vista previa del marco de trabajo que estamos desarrollando para la adopción de DevOps en EDS.

**Palabras Claves.** Desarrollo de software; operaciones de software, DevOps; estudio de mapeo sistemático.

**Cómo citar.** J. uerrero, K. Zúñiga, C. Certuche and C. Pardo, "A systematic mapping study about Devops", *Jou. Cie. Ing.*, vol. 12, no. 1, pp. 48-62, 2020. doi:10.46571/JCI.2020.1.5

**Received:** 28/02/2020 **Revised:** 02/04/2020 **Accepted:** 07/08/2020

## 1. Introduction

The different software development trends have been evolving since their inception, always trying to achieve better practices that guarantee the delivery of high-quality products, meeting the industry standards and the real needs of customers. Within this evolution we find constantly emerging traditional frameworks such as: Rational Unified Process (RUP) [1], Capability Maturity Model Integration (CMMI) [2], Iconix [3], Microsoft Solutions Framework (MSF) [4], among others; Agile approaches such as: Scrum [5], Lean Software Development (LSD) [6], eXtreme Programming (XP) [7], Crystal Clear [8], Adaptive Software Development (ASD) [9], among others; and even hybrid solutions that integrate practices of both traditional and agile solutions (Scrumban [10], Scrum and XP [11], CMMI and Scrum [12], among others), from the homogenization of their differences, comparison and integration of best practices [13].

Agile approaches aim to constantly deliver products of excellence and deliver value to their customers, guaranteeing the rapid return on investment, and the option to being open to changes in customer needs over time and permanent environment changes [1]. Similarly, from the perspective of the operations area are de facto frameworks, such as: Information Technology Infrastructure Library (ITIL) [14] and Control Objectives for Information and Related Technologies (COBIT) [15], and international standards such as ISO/IEC20000 [16], where elements, practices and/or activities related to the administration of information technology (IT) services and the administration of continuous improvement processes in IT services are proposed.

Currently, Software Development Companies (hereinafter SDC) seek to further increase their productivity due to the high degree of competitiveness of the software industry [17], [18], which tends to integrate and institutionalize agile and less complex processes that allow it to adapt to business changes in a timely manner [19]. Despite the progress in the design of solutions, models or frameworks to support different processes in software development, these alone do not achieve the best results in highly productive environments that require to continuously deliver services to their customers, [20]. That is when DevOps appear, the combination of the words Development (Dev) and Operations (Ops). DevOps is a term that, despite existing ten years ago, is currently becoming stronger [21], [22]. The main objective of DevOps is to unify the development and operation of the software in a single, integrated and continuous process, and try to break down the barriers between Dev and Ops[23]. From a DevOps approach, it is expected to take advantage of the experience and knowledge of people, processes and technology to stimulate collaboration and innovation throughout the software development and launch process [24], this done quickly, frequently and reliably [25], without sacrificing the quality and the value delivered to customers, distinctive characteristics of DNA in agile approaches but enhancing some of its benefits such as time to market [26].

In this regard, the objective of this systematic mapping focuses specifically on establishing a definition for DevOps, due to the different perceptions and definitions that can be found in the literature. Likewise, seeks to identify the solutions proposed by other authors about the conception of the implementation and / or operation of DevOps. After performing an analysis of the state of the art, some efforts are observed where solutions related to the adoption of DevOps in SDC have been proposed. Similarly, studies that identify aspects such as challenges and benefits that such adoption entails have been carried out. However, there are no studies with a sufficient level of detail to carry out the DevOps adopting process in a clear way and that describes a set of mechanisms that allow controlling what is related to DevOps [27]. In this regard, we consider that the systematic mapping carried out can be quite useful, since it synthesizes the current state of knowledge in DevOps, and it is also possible to establish the necessary attributes to develop a guide for the adoption, implementation and control of DevOps in the SDC. This paper is an extension of the conference paper

presented in [28], unlike the paper presented previously, here we present in detail the categorization of the process elements (identified and presented in section 3.4), which, in this document, was carried out following a reference ontology, seeking to lessen the ambiguity in certain terms. Also, the analysis of the answer to question Q6 (section 3.6), where the tools mentioned in the primary studies are categorized according to the process or area to which they support, has been considerably extended. In addition, the main observations section (section 4.1) has been complemented, commenting on what was observed after analyzing the process elements found. Likewise, it is also presented a preview of what our proposed solution to the reported inconveniences that SDC face when carrying out the DevOps adoption.

This study is organized as follows: Section 2 describes the research protocol carried out to perform the systematic mapping. Section 3 contains the results obtained from such mapping. Section 4 discusses the main observations of the results, as well as the limitations and implications of this field. In addition, Section 5 presents the framework to support the effort related to DevOps that is being developed, and finally Section 6 presents the conclusions and future work to be done regarding the adoption of DevOps in SDC.

## **2. Research protocol**

A systematic mapping is a method to collect and categorize existing information about a research topic. This systematic mapping was carried out following the guidelines presented in the following studies: [29, 30, 31, 32, 33]. The systematic mapping was carried out in three stages: Planning, Execution and Documentation.

### *2.1. Planning Stage*

In the planning stage the following activities were carried out:

- Creation of research questions.
- Definition of the search strategy.
- Definition of the selection criteria for primary studies.
- Definition of the quality assessment criteria.
- Definition of the data extraction strategy.
- Assortment of the synthesis methods.

### *2.2. Research questions*

In order to achieve the objective of systematic mapping and identify possible existing gaps to propose new areas of research, the research questions shown in Table 1 have been established.

Considering the related works, it is possible to observe that although there are different proposals that try to lessen some of the challenges of the SDC, there are no studies or proposals that consider the possible multi-model scenarios that can be generated in this type of environments.

### *2.3. Search strategy*

For the search, combinations of the “AND” and “OR” logical connectors were used on the identified keywords. The databases: Scopus, Google Scholar, SpringerLink and IEEE Xplore were used, in which the adapted search string for each of the search engines was introduced. Similarly, studies provided by experts, which have been classified as grey literature were used. (“agile” OR “agile



approach” OR ”agile process”) AND (DevOps OR ”continuous delivery” OR ”continuous release” OR ”continuous deployment”) AND (adopt OR integrate OR integration OR integrating) AND (”soft are development”). All information published from January 2009, date on which the term DevOps was first coined in the industry, [34] until June de 2020 was considered. This period of time has been validated during the execution of the systematic mapping, and most of the relevant publications about the adoption of DevOps are from 2009, especially in the last six years (2014 - 2020).

<b>Research questions</b>	<b>Motivation</b>
Q1. According to the analyzed literature, what is understood by DevOps?	An approach to a standard DevOps definition issued in the software environment.
Q2. What kind of instruments or research methods are used in the analyzed literature?	To determine research instruments or methods most commonly used in solving research questions regarding DevOps.
Q3. What kind of solutions have been proposed to simplify the adoption of DevOps in the SDC?	To determine the state of the art related to the adoption of DevOps, and establish the work to be done or to be improved.
Q4. What are the process elements that have been considered to apply DevOps?	To examine documented proposals, their approach, completeness and applicability.
Q5. What types of software development projects is DevOps used for?	To determine the types of software project development: standalone, web or mobile, on which the adoption of DevOps has been carried out.
Q6. What technological tools are used to simplify the adoption of DevOps?	To know the technological tools that support some type of activity to simplify the adoption of DevOps.
Q7. What benefits and challenges does the adoption of DevOps entail?	To determine the benefits and challenges companies that carry out the adoption of DevOps face.

*Table 1: Research questions*

#### 2.4. Selection criteria

The collected studies were evaluated considering the title, abstract and keywords, to determine whether or not they were included among the relevant studies. Then, they were analyzed in detail to select the primary studies. Those studies that met at least one of the following inclusion criteria were included: i) Studies in English that refer to the adoption of DevOps in software development companies and ii) Studies published between 2009 and 2020 in journals, conferences, prestigious congresses or workshops with peer review. On the other hand, studies that met any of the following exclusion criteria were not taken into account: i) Duplicate studies, ii) Studies whose main contribution is not related to the adoption of DevOps in SDC, iii) studies that contemplate the topic superficially iv) Types of debate studies, or available only in the form of abstracts or presentations.

#### 2.5. Quality Evaluation Criteria

To measure the quality of the selected studies and to determine the relevant studies about the adoption of DevOps in SDC, a questionnaire with a scoring system of three values was developed (-1, 0 and +1). The quality criteria established to evaluate the primary studies are explained in [28]. Table 2 presents the results of the evaluation of the studies according to the quality evaluation criteria. The sum of the score of each study will correspond to the final score (obtaining a value between -6 and +6). These scores will not be used to exclude a specific study from the systematic mapping, instead,

in the event of obtaining a poor score, it will be used to find more relevant studies, which will have more relevance in future researches.

Ref	Criteria						Score
	C1	C2	C3	C4	C5	C6	
[17]	1	1	1	0	-1	0	2
[21]	1	1	1	0	-1	1	3
[22]	0	0	1	-1	-1	1	0
[23]	1	-1	1	-1	-1	1	0
[27]	0	0	-1	-1	-1	0	-3
[35]	0	0	-1	-1	0	1	1
[36]	1	-1	-1	-1	-1	1	-2
[37]	-1	-1	-1	-1	0	1	-3
[38]	0	0	1	0	-1	1	1
[39]	0	-1	-1	-1	-1	1	-3
[40]	0	0	-1	-1	-1	0	-3
[41]	0	-1	0	-1	0	1	-1
[42]	1	0	-1	-1	-1	1	-1
[43]	1	-1	-1	-1	-1	1	-2
[44]	0	0	1	-1	-1	1	0
[45]	-1	1	1	0	-1	1	1
[46]	-1	0	1	-1	-1	1	-1
[47]	0	1	1	0	-1	-1	0
[48]	-1	0	0	-1	-1	-1	-4
[49]	0	-1	1	-1	-1	0	-2
[50]	1	0	-1	0	0	-1	-1
[51]	0	0	-1	-1	-1	-1	-4
[52]	0	1	1	0	0	0	2
[55]	1	1	-1	-1	0	1	1
[56]	1	1	-1	-1	0	1	1
[57]	0	-1	-1	-1	0	0	-3

Table 2: Evaluation of the studies according to the quality evaluation criteria.

## 2.6. Data Extraction Strategy

To ensure the application of the same data extraction criteria for all the selected studies and simplify their classification, a set of possible answers was used for each of the defined research questions, as shown in Table 3.

## 2.7. Synthesis method

The information of the selected primary studies was extracted and structured as follows: identification (title, publication, authors, reference), abstract, description (methodology, type of proposal, type of evaluation of the proposal) and highlighted aspects. Table 4 shows the relationship of the primary studies to the research questions that each one answered.

## 2.8. Execution Stage

Five iterations were carried out, a zero iteration where an initial review of grey literature was carried out which consisted of 10 articles, this review allowed us to know more in-depth the concepts which DevOps is usually associated with, in order to refine the final search string, then a new iteration was performed for each established search source. Table 5 shows the total number of studies: found, relevant, repeated and primary, acquired in the search sources of Scopus, Google Scholar, Springer e IEEE Xplore.

Questions	Answers
Q1. According to the analyzed literature, what is understood by DevOps?	a. Gap between development and operations, b. Culture of collaboration and communication, c. Integration, delivery and continuous deployment, d. Quality assurance, e. Automation, f. Others
Q2. What kind of instruments or research methods are used in the analyzed literature?	a. Systematic mapping / Literature review, b. Case study, c. Survey / interview, d. Proposals, e. Others
Q3. What kind of solutions have been proposed to simplify the adoption of DevOps in the SDC?	a. Approaches, b. Model, c. Methodology, d. Technique, e. Framework, f. Others, g. No concrete solution is proposed
Q4. What are the process elements that have been considered to apply DevOps?	a. Roles, b. Practices, c. Activities, d. Templates, e. Artifacts, f. Others
Q5. What types of software development projects is DevOps used for?	a. Standalone, b. Web, c. Mobile, d. The type of project is not clarified
Q6. What technological tools are used to simplify the adoption of DevOps?	a. Tools for integration, delivery and continuous deployment, b. Tools to promote communication and collaboration, c. Others
Q7. What benefits and challenges does the adoption of DevOps entail?	a. Economic, b. Improves processes, c. Improves collaboration and communication, d. Quality assurance, e. Customer satisfaction, f. Productivity improvement, g. Resistance to change, h. Others

Table 3: Classification scheme.

### 3. Results

Below are the results obtained for each of the research questions defined in the creation of this systematic mapping which are duly referenced to enable further in-depth study by the reader.

#### 3.1. Question Q1: According to the analyzed literature, what is understood by DevOps?

Different definitions for DevOps were found in the analyzed literature, it seems that a definition or a common understanding of what DevOps implies has not yet been reached [43], so some definitions may be characterizing DevOps from a very specific perspective [43]. When analyzing each definition, the common characteristics and / or aspects (hereinafter concepts) that the authors have considered to define DevOps were identified. The common concepts that have been identified in the definitions are: (i) Collaboration and Communication, (ii) Development and Operations, (iii) Cover the gap, (iv) Quality Assurance, (v) Automation, (vi) Integration, delivery and continuous deployment and (vii) Set of practices.

According to the analyzed studies, it was also observed that DevOps represents: an approach [35, 52, 55], a cultural movement [17, 57], a set of practices [23, 27], an emerging paradigm [38, 56], a mixture

of two words [39], a phenomenon [40], an interdisciplinary theme [22], a set of principles [41], a conceptual framework [42], a development methodology [51], an artificial word [44], a philosophy [47] and a neologism [36]. From the above, it can be evidenced that DevOps has been interpreted in several ways because it is a relatively new subject.

Ref	Question						
	Q1	Q2	Q3	Q4	Q5	Q6	Q7
[17]	X	X	X				
[21]	X	X	X	X		X	X
[22]	X		X	X		X	X
[23]	X	X	X				X
[27]	X	X	X	X	X	X	X
[35]	X	X	X	X			X
[36]	X	X	X				X
[37]		X	X				X
[38]	X		X				
[39]	X		X				X
[40]	X	X	X	X			X
[41]	X		X				
[42]	X	X	X	X		X	X
[43]	X	X	X		X		
[44]	X	X	X	X	X	X	
[45]		X	X	X	X	X	
[46]		X	X	X	X	X	
[47]	X	X	X	X		X	
[48]		X	X				
[49]	X	X	X				X
[50]	X	X	X	X	X	X	X
[51]	X	X	X	X		X	
[52]	X	X	X	X		X	
[55]	X	X	X	X	X		
[56]	X	X	X	X		X	
[57]	X	X	X				

Table 4: Contributions of the primary studies to each research question.

No.	Data source	Found	Relevant	Relevant repeated (Previous iterations)	Primary selected
0	Grey Literature	10	9	0	7
1	Scopus	522	10	7	3
2	Google Scholar	219	7	4	6
3	Springer	487	8	4	4
4	IEEE Xplore	42	3	3	3
Total		1280	37	18	23

Table 5: Classification scheme

### *3.2. Question Q2: What type of research is carried out in the analyzed literature?*

The studies selected in this research can be divided into two categories: (i) studies with concrete proposals on the adoption of DevOps [17, 23, 40, 44, 45, 46, 47, 49, 50, 56] and (ii) systematic reviews of literature [21, 27, 35, 36, 42, 48, 55, 57]. Regarding the first category, it was found that 90% of the studies [17, 23, 40, 44, 45, 46, 47, 49, 50] registered on the adoption of DevOps use case studies to validate their proposals, these Case studies have been carried out in a range of one to five years in their application depending on the size of the project. On the other hand, from these studies 77% [17, 23, 40, 44, 45, 46, 49] complement the findings with surveys and / or interviews to determine the impact at the level of culture and organization. Regarding the second category, only 38% [21, 36, 48] of them use their findings to propose, even if in a conceptual way, some solution for the implementation of DevOps with some agile framework.

### *3.3. Question Q3: What kind of solutions have been proposed for the adoption of DevOps?*

20% [17, 21, 38, 45, 56] of the analyzed literature propose concrete solutions to guide the SDC in the adoption of DevOps. On the other hand, 80% [22, 23, 27, 35, 36, 37, 39, 40, 41, 42, 43, 44, 46, 47, 48, 49, 50, 51, 52, 55, 57] of the analyzed studies, do not propose a concrete solution for the adoption of DevOps in SDC, but rather describe their experience in adopting DevOps. It is possible to observe that 56% [17, 21, 22, 27, 35, 36, 37, 39, 40, 41, 42, 49, 50, 56] identify benefits and challenges, in addition, 60% [17, 21, 22, 23, 35, 40, 41, 44, 46, 47, 48, 51, 52, 55, 56] propose a set of process elements that could be considered for the adoption of DevOps. Similarly, 40% [21, 27, 42, 44, 45, 46, 47, 50, 52, 56] mention the technological tools that could support practices to adopt DevOps. It is important to mention that detailed information on the benefits and challenges that were identified in the analyzed literature is available in [28].

### *3.4. Question Q4: What are the process elements that have been considered to apply DevOps?*

After analyzing the primary studies, it was possible to observe that 60% (15 studies out of 25 in total) of them propose or mention process elements, which were categorized according to the process elements proposed in the Ontology of Process-reference Models - PrMO [53]. PrMO defines concepts and their own definitions and relationships for process elements such as: process, activity, work products, role and tool. The ontology also defines other concepts for process elements such as process category and tasks, however, since no such process elements were found, they are not mentioned in this work. Due to space limitations in Table 6 only some process elements identified in each of the selected studies are presented, in this sense, the column Activity and Tool just show the amount of activities and tools, respectively, the complete table with all activities identified is found in: <https://bit.ly/2VCTdyK>. As regards to the tools, these are listed in the Table 7. From the total of selected primary studies, it was identified that 46% of them propose processes that support DevOps, from the processes that support studies are those related to: continuous integration, continuous delivery, continuous supervision, continuous deployment, quality assurance, configuration process, DevOps Framework Approach. Likewise, 38% of the studies propose activities, 8% work products, 38% roles, 69% technological tools and 31% propose other process elements, for example: areas or dimensions. It should be considered that a study can propose several process elements and that these can support different processes. Table 6 shows only the number of tools proposed by each author because the technological tools that support DevOps are addressed in depth in the research question Q6.

### *3.5. Question Q5: What types of software development projects (standalone, web or mobile) is DevOps used for?*

100% of the selected studies describing software development projects with evidence of progress or proposals in the adoption of DevOps make use of Cloud services, for example: continuous integration services, monitoring, among others [17, 23, 40, 44, 45, 46, 47, 49, 50] and there are no studies where the creation of services to support practices related to the adoption of DevOps over local networks is proposed. Only in [46] describes artifacts and specific practices for the adoption of DevOps in mobile development environments, the other proposals are focused on Web environments.

Ref	Process Element				
	Process	Activity	Product	Rol	Tool
[21]	NF	NF	List of impediments, DevOps maturity model, DevOps tool list, DevOps tool list used in the company	NF	45
[22]	NF	NF	NF	Software Development Manager, Developer and Test Analyst with DevOps approach, Systems Manager	4
[27]	CD, DC, QA	11 activities	NF	NF	19
[35]	NF	NF	NF	Engineer, Team and DevOps Department, Developer with DevOps approach, Systems Manager	NF
[40]	NF	24 activities	NF	NF	NF
[42]	Measurement, Automation	19 activities	NF	NF	2
[44]	DC	2 activities	NF	NF	2
[45]	NF	NF	NF	Developer with DevOps approach, Product owner, Project Leader, Team leader, Program Manager, Release Manager	1
[46]	CI, CD, CS	NF	NF	DevOps Team	1
[47]	PC, DFA	10 activities	NF	NF	4
[50]	NF	NF	NF	NF	14
[51]	CI, CD, DC	3 activities	NF	NF	5
[52]	NF	NF	NF	Product Owner, Architect, Production Engineer, Launch Manager	NF
[56]	CI, CS, DC	NF	NF	NF	29

Ref: reference; NF: information not found; CI: continuous integration; CD: continuous delivery; CS: continuous supervision; DC: continuous deployment; QA: quality assurance; PC: configuration process; DFA: DevOps Framework Approach

Table 6: Process elements related to DevOps identified by study.

No.	Area/Process	%	Tool
1	Source control management (Repository)	12	Github [21, 27, 47, 50], Bitbucket [21, 22, 27], GitLab [21, 56], Mercurial [22], Apache Subversion [21,56], SonarQube [50, 56], Maven [47, 56]
2	Continuous integration and orchestration	13	Codship [21, 27], Travis CI [21, 27, 56], Jenkins [21, 22, 27, 44, 47, 50, 56], Atlassian bamboo [21], Circle CI [21], TeamCity [21], Rancher [51], Gitflow [45]
3	IaaS/PaaS	2	Heroku [27, 56]
4	Infrastructure as code	2	Amazon cloudformation [50, 56]
5	Monitoring	20	Nagios [21,27,42,56], New Relic [27,50], AWS CloudWatch [21,27], Apache Kafka [46], Zabbix [21,56], Splunk [21,51], AppDynamics [21], Bugzilla [21], Track & TesTtrack [21], MantisBT [21], Assembla [21], Kinesis [21]
6	Managing and Database management	7	MongoDB [27], DBMaestro [21], LiquiBase [21], RedGate [21]
7	Record / Security	3	Loggly [27], Papertrail [27]
8	Testing	8	Cucumber [21, 27], Junit [27, 56], Selenium [21, 50], TestComplete [21], Jmeter [21]
9	Collaboration	5	Slack [21,27], HipChat [21,22,27], PagerDuty [21]
10	Containerization	8	Docker [21, 25, 47, 50], Mercurial [21], Bitbucket [50], Rocker [21], Vagrant [21]
11	Configuration Management	10	Docker [21, 47, 50, 56], Puppet [21, 27, 42, 50, 51], Ansible [21,27,50], Cheff [21,50,51,56], Vagrant [21], SaltStack [21]
12	Knowledge exchange	5	Crowdbase [21], Nuclion [21], Confluence [21], Trello [56]
13	Planning	5	Clarizen [21], Confluence [21], Asana [21]
Total		100	60 tools

Table 7: Technological tools that support DevOps processes / areas.

3.6. *Question Q6: What technological tools are used to simplify the adoption of DevOps?* A total of 60 technological tools that support the processes involved in the adoption of DevOps were identified; to classify them, the catalog defined in [27]. Table 7 shows the areas or processes in which the tools were categorized, the percentage of tools found in each area, the names of the tools along with their reference. It was observed that the most used tools in the industry are: (i) Github, (ii) Jenkins, (iii) Puppet and (iv) Docker.

3.7. *Question Q7: What benefits and challenges does the adoption of DevOps entail?* In the analyzed literature it is observed that the adoption of DevOps in the SDC entails certain benefits, including: (i) response time (time to market) that a product takes from the moment it is conceived until it is for sale: [22, 23, 27, 36, 37, 49], (ii) to close the gap between Dev and Ops: [22, 23, 27, 36, 49], (iii) improved product quality: [22, 37, 42, 50] and (iv) improved customer satisfaction: [36, 37, 49]. In addition to the benefits that the DevOps adoption entails, certain challenges that hinder this adoption process were identified, these include: (i) resistance to change from the people: [21, 22, 23, 27, 40, 49], (ii) lack of understanding of the meaning of DevOps: [21, 23, 39, 40], (iii) lack of collaboration between development and operations: [21, 23, 35, 39] and (iv) to change the culture of the company: [21, 22, 23, 40]. The detailed list of benefits and challenges that were identified in the analyzed literature is found in [28].

## 4. Discussion

This section presents an analysis of the results obtained from the systematic mapping performed, in order to identify the improvements that can be made on the proposals found.

### 4.1. Main observations

The objective of this systematic mapping is to know the current status of the adoption of DevOps in SDC. In this regard, after analyzing the results, it can be deduced that according to the analysis of what DevOps represents in the analyzed studies, “neologism” is the term that would characterize and classify most of the definitions. In addition, it is observed that this neologism refers to the cultural movement to which a company must adapt to adopt DevOps. Also, it is clear that DevOps is not only a set of practices, but it is related to all suggestion at the level of tasks, activities, artifacts, templates and roles. Therefore, it is considered more appropriate to use the term “process elements”. Since there are definitions that do not take into account some of the common concepts identified, the following definition of DevOps is proposed: DevOps is a neologism that arises from the combination of the words Development and Operations. It is a cultural movement that allows to bridge the gap between developers and operations personnel. DevOps is related to a set of process elements that foster collaboration and communication, automation, quality assurance and integration, delivery and continuous deployment.

A large part of the proposals for the adoption of DevOps is focused on the practices of continuous integration, continuous deployment and process automation. However, the defined proposals are not very elucidated, so there is still a long way to go in the definition of standards for the contextualized appropriation of this cultural movement that impacts software projects from different perspectives such as architecture, culture, people, processes and tools.

As for process elements to support DevOps, it has been observed that these are not clearly defined, some of them are only identified, but not thoroughly described, even, the definitions of the process elements are ambiguous, and in most cases, incomplete. The authors propose activities and / or tasks that involve the adoption of DevOps, but do not explain carefully how to implement them nor what roles are responsible for executing them. As for the roles, it is only mentioned that it is necessary to have advanced knowledge about DevOps, however, they do not deepen in their assigned responsibilities, the stage in which they participate, among others. It was also possible to observe that the authors define the process elements according to their experience and not to a standard or model. In addition, it was not evidenced in the analyzed studies a detailed analysis of the specific activities that should be performed in each process supported by DevOps. Neither in the conducted case studies, was seen proof of a notable improvement in the productivity achieved by applying DevOps in the case studies companies.

### 4.2. Limitations of the systematic mapping

The limitation to academic search engines represents the state of the art of the research on the adoption of DevOps in SDC. The inclusion of studies only in English may mean that relevant studies in other languages are not considered, nonetheless English is the language most used to publish studies on this subject. In addition, although this systematic mapping has yielded relevant results, they serve as a starting point for a later version of the systematic mapping presented here.



### 4.3. Importance for research and practice

The observations of this systematic mapping are of great importance to those researchers who are planning to research DevOps, and more specifically in the issue of the adoption of DevOps in SDC. For researchers it is an area of great interest, since, as it has been noted, it is a new field, in which there is little work done. Researchers, thanks to the development in this field, will be able to create guides that allow the adoption, implementation and control of DevOps correctly, or improve the existing practices that companies are currently adopting to implement DevOps, so that good practices of DevOps in SDC can be consolidated. Finally, organizations will be greatly benefited by the progress of this field, because so far, some organizations have developed and implemented DevOps according to their own criteria which they have considered appropriate, however, this has been carried out without a detailed evaluation of the results achieved in terms of the benefits obtained.

## 5. A Framework to support the effort related to DevOps

In order to address the main issues identified in this systematic mapping, a framework to support the effort related to DevOps is being developed, Figure 1 shows the structure of the framework which is composed of the following elements:

- DevOps Ontology allows understanding and organizing the knowledge related to the terms and relationships involved in the DevOps adoption process. This ontology uses as a basis the PrMO (Ontology of Process-reference Models) [53] and SMO (Software Measurement Ontology) [54].
- The reference model (DevOps Model) defines a set of fundamental process elements to consider (activities, roles, products, tools) for the adoption of DevOps. These elements will be categorized by the dimensions: (i) processes, (ii) technology, (iii) people and (iv) culture
- DevOpsProcess:softwaredevelopmentprocessthatincludestheanalysis,designand implementation phases, through the incorporation of the best practices of Agile and DevOps approaches, to enhance the development, integration and continuous deployment.
- DevOpsEvaluationModel:solutionthatallowstheevaluationoftheadoptionofDevOps from the reference model DevOps Model and DevOps Process.
- Tools: includes a set of electronic guides that simplifies the understanding of the DevOps reference model and the DevOps adoption process with agile methodologies or frameworks proposed in this work. It also, includes a prototype that facilitates the evaluation of the adoption of DevOps in the software development processes of the software development companies based on the indicators defined in the reference model.

## 6. Conclusions and future work

In the last ten years DevOps has become a topic of interest within the software industry. As it has been observed, although it is a relatively new concept and there is no common definition of what DevOps means, some SDC have carried out the implementation and adoption of certain practices related to DevOps. However, although there is coincidence in some of the terms used regarding the adoption of DevOps, for example, in the names of the proposed activities, there is still heterogeneity and therefore ambiguity in the terms. This is because each author covers DevOps from a particular perspective. So far, there is no evidence of a guide / standard to carry out the process of adoption and evaluation of DevOps. In this regard, the results obtained in the systematic mapping presented in this study demonstrate the novelty of this field and the need to develop a guide that simplifies the understanding of DevOps and allows its adoption, implementation and / or control in the SDC. Bearing in mind the shortcomings found in this current research stream, we have presented a

detailed summary of our research proposal, which defines a set of elements with which to facilitate the activities related to DevOps.

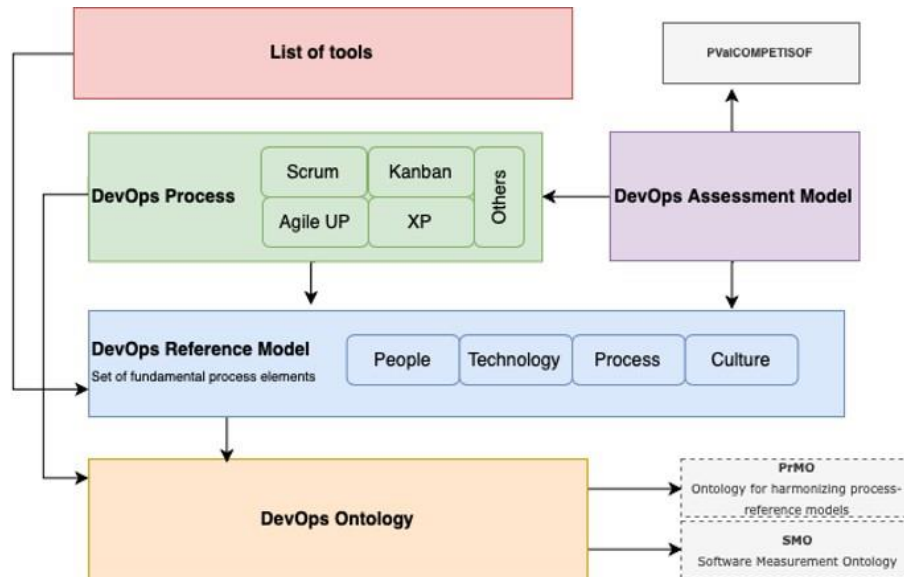


Figure 1: A summary of the proposed framework.

## References

- [1] P. Kruchten, *The Rational Unified Process - An Introduction*. Boston, MA, USA: Addison- Wesley-Longman, 2000.
- [2] CMMI Institute, "Capability maturity model integration for development (cmmi-dev, v2.0)," 2018. [Online]. Available: <https://bit.ly/2J2mGLB>. [Accessed: 27-Mar-2020].
- [3] ICONIX oft are Engineering, "Iconix," 2019. [Online]. Available: <https://bit.ly/3aAQLQ9> [Accessed: 27Mar-2020].
- [4] G. Lory, D. Campbell and A. Robin, "Microsoft olutions Frame ork v3 Overvie ," 2003.
- [5] J. ch aberandK. urtherland, "Thescrumguidethedefinitiveguidetoscrum:Therulesof the game," 2017. [Online]. Available: <https://bit.ly/2F7HKxd>. [Accessed: 27-Mar-2020].
- [6] M. Poppendieck and T.Poppendieck, *Lean Software Development: An Agile Toolkit*. Crawfordsville, IN, USA: Addison-Wesley Professional, 2003.
- [7] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change*. Boston, MA, USA: AddisonWesley-Longman, 2004.
- [8] A. Cockburn, "Crystal Clear: A Human-Po ered Methodology for mall Teams.," Pearson Education, 2004.
- [9] J. A. Highsmith, "Adaptive oft are Development: A Collaborative Approach to Managing Complex systems," vol. 12, 2000.
- [10] C. Ladas, *Scrumban-essays on kanban systems for lean software development*. Seattle, WA, USA: Modus Cooperandi Press, 2009.
- [11] H. Kniberg, *Scrum and XP from the Trenches*. Morrisville, NC, USA: Lulu.com, 2007.

- [12] J. Utherland, C. Jakobsen and K. Johnson, "Crum and CMMI level 5: The magic potion for code warriors," in 41st Annual Hawaii International Conference on System Sciences (HIC 2008), 2008, pp. 466–466.
- [13] C. Pardo, P. Chilito, D. Viveros and F. Pino, "Crum+: A scaled crum for the agile global software development project management with multiple models," *Rev. Fac. Ing.*, 2019.
- [14] A. Hochstein, R. Zarneko and W. Brenner, "ITIL as common practice reference model for IT service management: formal assessment and implications for practice," 2005 IEEE Int. Conf. e-Technology, eCommerce e-Service, pp. 704-710., 2005.
- [15] J. Young, G. Ridley and P. Carroll, "COBIT and Its Utilization: A Framework from the Literature," 47th Hawaii Int. Conf. Syst. Sci., p. 80233., 2014.
- [16] IO/IEC, "Calidad de los servicios TI," 2019. [Online]. Available: <https://bit.ly/2HWudxi>. [Accessed: 27Mar-2020].
- [17] S. Samarawickrama and I. Perera, "Continuous scrum: A framework to enhance scrum with DevOps," in 2017 seventeenth International Conference on Advances in ICT for Emerging Regions, 2017, pp. 1–7.
- [18] C. Pardo and J. Masso, "Hacia una Ontología para el Gobierno de Desarrollo de Software en Pymes," *Publicaciones e Investig.*, vol. 9, pp. 99–112, 2015.
- [19] C. Pardo and J. Maso, "Revisión sistemática de la integración de modelos de desarrollo de software dirigido por modelos y metodologías ágiles," *Inf. Técnico*, vol. 8, pp. 87–99, 2016.
- [20] M. Virmani, "Understanding Devops & Bridging The Gap From Continuous Integration To Continuous Delivery," in Fifth International Conference on the Innovative Computing Technology (INTECH 2015). IEEE, 2015, pp. 78–82.
- [21] I. Bucena and M. Kirikova, "Simplifying the DevOps Adoption Process," in BIR Workshops, 2017.
- [22] J. Jones, J. Noppen, and F. Lettice, "Management challenges for DevOps adoption within UK SMEs," in Proceedings of the 2nd International Workshop on Quality-Aware DevOps - 2016, 2016, pp. 7–11.
- [23] L. Riungu-Kalliosaari, J. Mäkinen, L. Lakatare, J. Tiihonen and T. Mannisto, "DevOps Adoption Benefits and Challenges in Practice: A Case Study," 590-597. 10.1007/978-3-319-49094-6\_44, 2016.
- [24] J. Michelsen, "Dysfunction Junction: A Pragmatic Guide to Getting Started with DevOps," pp. 1–26, 2014.
- [25] K. Lees, J. Gardner and P. Eaton, "DevOps and Agile Development - A VMware Field Perspective," VMware, Inc., Palo Alto, CA, USA, 2020. [Online]. Available: <https://bit.ly/3eVxcDc>
- [26] C. Prado, F. Pino and W. Ortega, "Mapeo sistemático sobre la evaluación de la agilidad en organizaciones de desarrollo de software," *ITECKNE Innovación e Investig. en Ing.*, vol. 16, pp. 64–76, 2019.
- [27] G. Ghantous and A. Hill, "DevOps: Concepts, Practices, Tools, Benefits and Challenges," in PACIS 2017, 2017.
- [28] J. Guerrero, C. Certuche, K. Zúñiga and C. Pardo, "What is there about DevOps? Preliminary Findings from a Systematic Mapping Study," in JIII IC 2019, 2019.
- [29] J. Paton-Romero and M. Piattini, "Re-evaluating IT maturity models: A systematic mapping study," in 2017 12th Iberian Conference on Information Systems and Technologies (CISTI), 2017, pp. 1–6.
- [30] K. Petersen, R. Feldt, J. Mujtaba and M. Mattsson, "Systematic mapping studies in software engineering," in EA '08 Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering, 2008, pp. 68–77.
- [31] B. Kitchenham, "Systematic review in software engineering," in Proceedings of the 2nd international workshop on Evidential assessment of software technologies - EA T '12, 2012, pp. 1–2.

- [32] D. Budgen, M. Turner, P. Brereton and B. Kitchenham, "Using mapping studies in soft are engineering.," in PPI 2008, 2008, pp. 195–204.
- [33] M. Genero , L. Cruz, and M. Piattini, "Métodos de investigación en ingeniería del soft are,," Bogotá, DC: Grupo Editorial ra-ma, 2014.
- [34] M. Martins, "Devopsdays Organizing uide," 2018. [Online]. Available: <http://bit.ly/3bEPpVr>. [Accessed: 27-Mar-2020].
- [35] F. Erich, C. Amrit, and M. Daneva, "A qualitative study of DevOps usage in practice," *J. oft . Evol. Process*, vol. 29, pp. 1–25, Jun. 2017.
- [36] B. de Franc a, H. Jeronimo and . Travassos, "Characterizing DevOps by Hearing Multiple Voices,," Ne York, NY: Association for Computing Machinery, 2016.
- [37] L.Chen,"ContinuousDelivery:Overcomingadoptionchallenges,"*J. yst. oft .*,vol.128,pp. 72–86, Jun. 2017.
- [38] J. Wettinger, U. Breitenbu 'cher and F. Leymann, "DevOp lang – Bridging the Gap between Development and Operations," & L. K. (eds) Villari, M., Zimmermann, W., Ed. Berlin, Heidelberg: Springer, 2014, pp. 108–122.
- [39] L.Lwakatare,P.KuvajaandM.Oivo,"DimensionsofDevOps,"in*AgileProcessesin oft are Engineering and Extreme Programming*, & P. M. Lassenius C., Dingsøy T., Ed. Helsinki, Finland: Springer, Cham, 2015, pp. 212–217.
- [40] M. Kamuto, and J. Langerman, "Factors inhibiting the adoption of DevOps in large organisations: outh African context," in 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 48–51.
- [41] V. upta, P. Kapur and D. Kumar, "Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling," *Inf. oft . Technol.*, vol. 92, pp. 75–91, Dec. 2017.
- [42] F. Erich, C. Amrit and M. Daneva, "Report: DevOps Literature Revie ,," 2014.
- [43] R. Jabbari,N. Bin, K. Petersen and B. Tanveer, "What is DevOps?," in PXP '16 Workshops: Scientific Workshop Proceedings of XP2016, 2016, pp. 1–11.
- [44] F. Elberzhager, T. Arif, M. Naab, I. uß, I and . Koban, "From Agile Development to DevOps: Going Towards Faster Releases at High Quality – Experiences from an Industrial Context,," Vienna, Austria: Springer, Cham, 2017.
- [45] .Krusche,L.Alpero itz,B.Bruegge,andM.Wagner,"Rugby:anagileprocessmodelbased on continuous delivery," in RCo E 2014 - Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering, 2014, pp. 42–50.
- [46] . Klepper, . Krusche, . Peters, B. Bruegge and L. Alpero itz, "Introducing Continuous Delivery of Mobile Apps in a Corporate Environment: A Case tudy," in 015 IEEE/ACM 2nd International Workshop on Rapid Continuous Software Engineering, 2015, pp. 5–11.
- [47] G. Benguria, J. Alonso, I. Etxaniz,L. Orue-Echevarria and M. Escalante, "Agile Development and Operation of Complex Systems in Multi-technology and Multi-company Environments: Follo ing a DevOps Approach," in European Conference on oft are Process Improvement EuroSPI 2018, 2018, pp. 15–27.
- [48] A. Nagarajan and . Overbeek, "A DevOps Implementation Frame ork for Large Agile-Based Financial Organizations," in *On the Move to Meaningful Internet Systems. OTM 2018 Conferences*, M. R. (eds) Panetto H., Debruyne C., Proper H., Ardagna C., Roman D., Ed. Utrecht, Netherlands: Springer, Cham, 2018, pp. 172–188.
- [49] M. enapathi, J. Buchan, and H. Osman, "DevOps Capabilities, Practices, and Challenges," in *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 - EA E'18*, 2018, pp. 57–67.

- [50] L. Lwakatare, T. Kilamo, T. Karvonen, T. Sauvola, V. Heikkilä, J. Itkonen, and C. Lassenius, "DevOps in practice: A multiple case study of five companies," *Inf. Syst. Technol.*, vol. 114, pp. 217–230, Oct. 2019.
- [51] J. Cusick, "A Survey of Maturity Models from Nolon to DevOps and Their Applications in Process Improvement," 2019.
- [52] A. Hemon, B. Lyonnet, F. Rowe, and B. Fitzgerald, "From Agile to DevOps: Smart Skills and Collaborations," *Inf. Syst. Front.*, pp. 1–19, Mar. 2019.
- [53] C. Pardo, F. García, M. Piattini, F. Pino and M. Baldassarre, "A Reference Ontology for Harmonizing Process-reference Models," *Rev. Fac. Ing. Univ. Antioquia*, vol. 73, pp. 29–42, 2014.
- [54] M. Barcellos, R. Falbo and R. Moro, "A Well-Founded Software Measurement Ontology," in *FOIS 2010.*, vol. 209, pp. 213–226.
- [55] D. Teixeira, R. Pereira, T. A. Henriques, M. Silva, and J. Faustino, "A Systematic Literature Review on DevOps Capabilities and Areas," *International Journal of Human Capital and Information Technology Professionals*, vol. 11, no. 2, pp. 1–22, 2020.
- [56] L. Leite, C. Rocha, F. Kon, D. Milojicic, and P. Meirelles, "A Survey of DevOps Concepts and Challenges," *ACM Comput. Surv.* 52, 6, Article 127, 35 pages. 2019.
- [57] M. Sánchez-Gordón and R. Colomo-Palacios, "Characterizing DevOps Culture: A Systematic Literature Review," in *Proceedings of the Software Process Improvement and Capability Determination. SPICE 2018*, pp. 3–15, 2018.



# Anexo B. Resultados cuestionario grupo focal

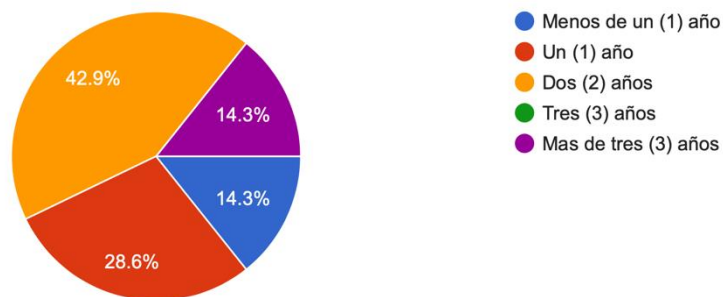
¿Cuál es el rol que desempeña en la empresa donde labora actualmente?

7 respuestas

VP de Ingeniería
Docente
Analista de Calidad de Software
Cloud Architect
Ingeniera de Requisitos
Arquitecto/Desarrollador
Analista de pruebas

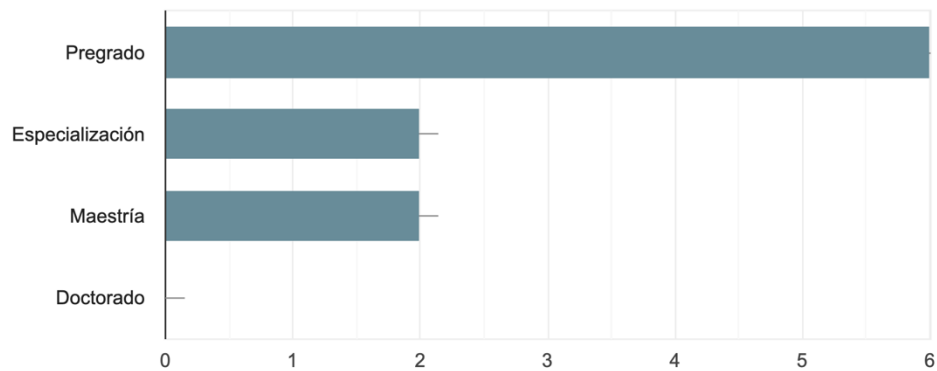
¿Cuántos años de experiencia tiene investigando y/o aplicando el tema de DevOps?

7 respuestas



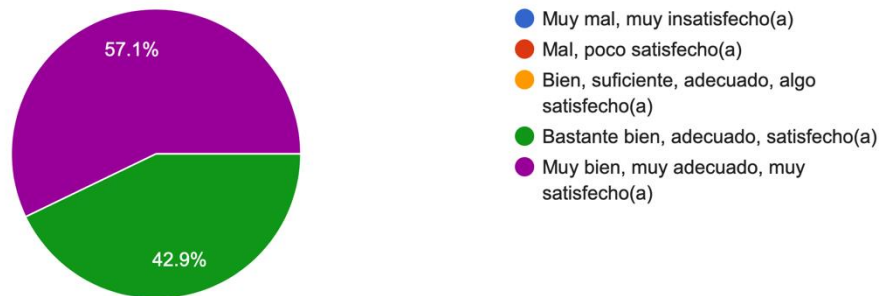
### ¿Cuál o cuales son sus estudios realizados?

7 respuestas



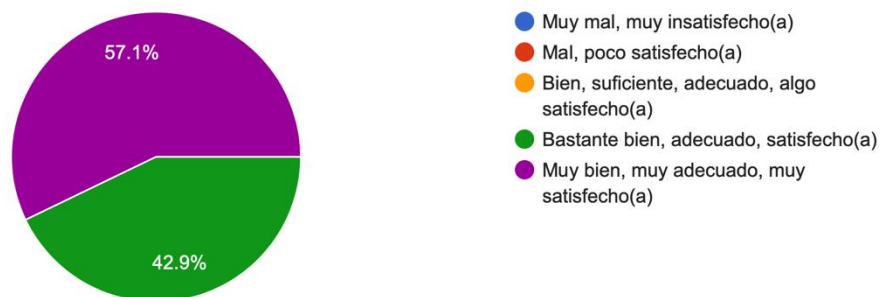
### ¿Considera que los valores de DevOps propuestos son de fácil comprensión?

7 respuestas



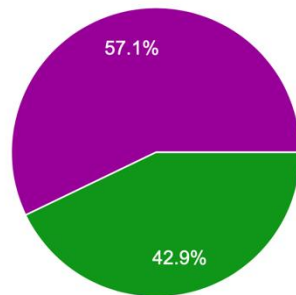
### ¿Considera que los principios de DevOps propuestos son de fácil comprensión?

7 respuestas



¿Considera que las dimensiones propuestas son de fácil comprensión?

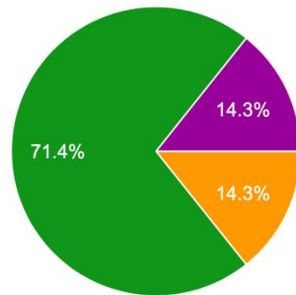
7 respuestas



- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)

¿Considera que las prácticas propuestas son de fácil comprensión?

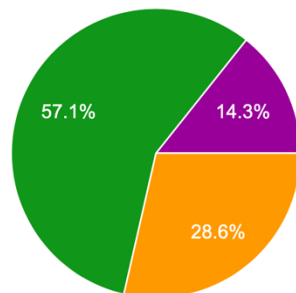
7 respuestas



- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)

De acuerdo a su experiencia: ¿Considera que los elementos definidos en el modelo de referencia son apropiados y pueden aplicarse con éxito en un proyecto de desarrollo de software?

7 respuestas

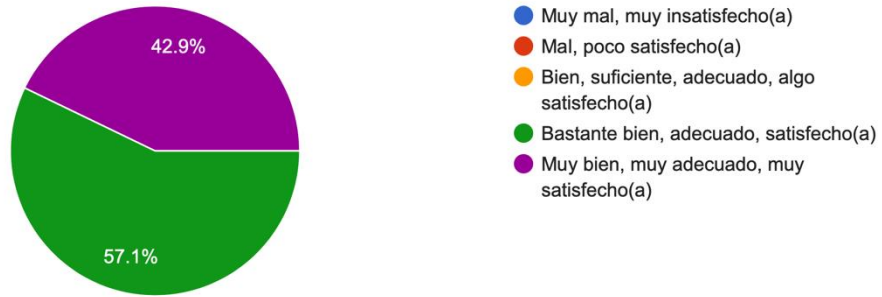


- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)



¿Considera que los valores propuestos capturan la esencia principal de DevOps?

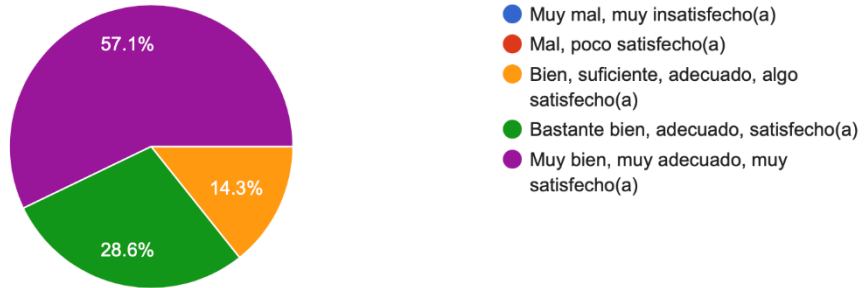
7 respuestas



De acuerdo a su experiencia: ¿Considera que los principios definidos para DevOps son apropiados e interpretan correctamente las dinámicas actuales de la industria del Software?

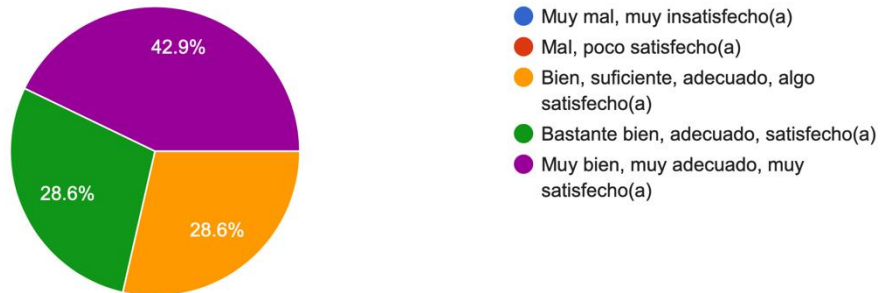


7 respuestas



De acuerdo a su experiencia: ¿Considera que los principios definidos para DevOps son suficientes?

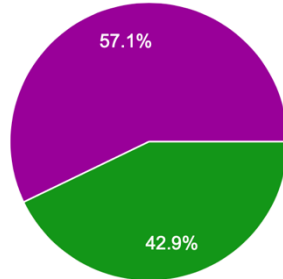
7 respuestas



¿Considera que las dimensiones definidas para DevOps agrupan correctamente los principios y practicas definidas?



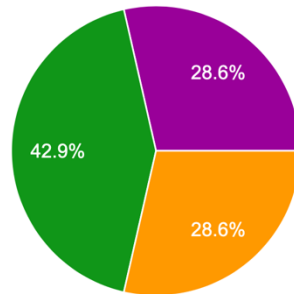
7 respuestas



- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)

¿Considera que las prácticas fundamentales propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?

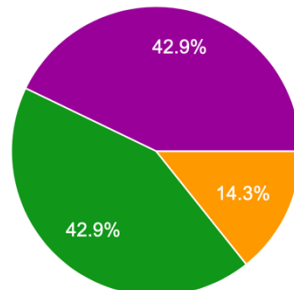
7 respuestas



- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)

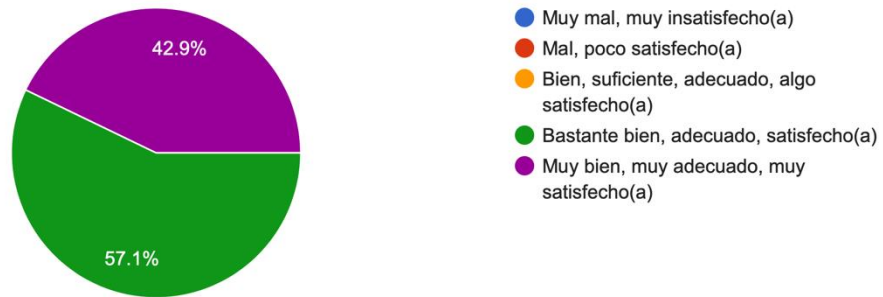
¿Considera que las prácticas complementarias propuestas son apropiadas y pueden aplicarse con éxito en un proceso de desarrollo de software?

7 respuestas

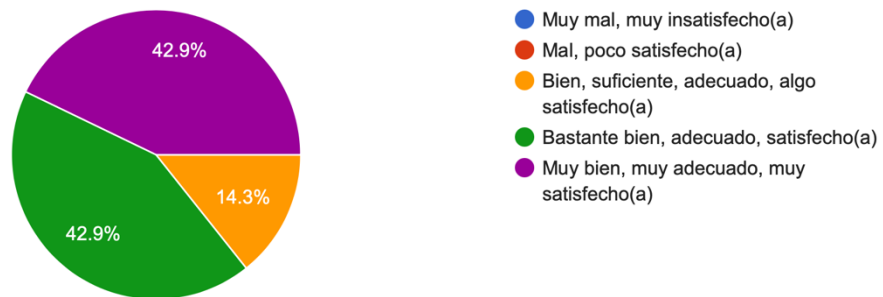


- Muy mal, muy insatisfecho(a)
- Mal, poco satisfecho(a)
- Bien, suficiente, adecuado, algo satisfecho(a)
- Bastante bien, adecuado, satisfecho(a)
- Muy bien, muy adecuado, muy satisfecho(a)

¿Considera que las prácticas propuestas son suficientes para un proceso de adopción de DevOps?  
7 respuestas



¿Considera que el modelo propuesto propuesto sirve de referencia para proyectos de mejora que tienen por objetivo la definición de procesos relacionados con DevOps?  
7 respuestas



¿Considera que se deben agregar, eliminar o modificar elementos (valores, principios, dimensiones o practicas) de la propuesta?

7 respuestas

Ninguno

Revisar si el despliegue continuo al menos en un nivel básico debe ser una práctica fundamental. Sería interesante establecer los requisitos previos que debe cumplir la organización para iniciar un proceso de implementación de DevOps.

Yo le agregaría para la parte de integración continua alguna referencia de como incorporar el proceso de pruebas para que quede listo para pasar a producción, mirar si hay elementos de riesgos se puedan presentar.

Tal vez hablar del shift left

Como sugerencia se podría dejar la práctica "Supervisión de la seguridad" como complementaria, porque no bloquea la implementación de la propuesta pero sí permite mejorarla.

No, todos los elementos presentados son claros

No, en mi opinión es un trabajo bastante completo

¿Tiene algún comentario adicional acerca del modelo de referencia propuesto?

7 respuestas

Ninguno

Excelente propuesta y bien fundamentada.

Excelente muy bien, felicitaciones. Muchos exitos.!

:)

Muy buen trabajo. Un aporte acerca de la gestión de requisitos: es ideal que sean cortos, completos, claros, realistas y acordados para evitar cuellos de botella desde el área de análisis.

La propuesta es robusta y muy bien fundamentada

Buen trabajo

# Anexo C. Documentación estudio de caso

## *Aceptación de participación en el estudio de caso*

A continuación, se presenta el documento de aceptación utilizado de participación de la empresa en el estudio de caso. Los datos personales del representante de la empresa se mantienen ocultos por razones de confidencialidad.

## **Estudio de Caso DevOps-Model**

Este formulario sirve de instrumento para realizar uno de los procesos de evaluación de la propuesta DevOps Model - Modelo de referencia para la adopción de DevOps en empresas de desarrollo de software.

Con el fin de direccionar el estudio de caso, se definió la siguiente pregunta de investigación principal:

¿DevOps Model permite de manera comprensible, útil y práctica la definición de un modelo de valoración de adopción de DevOps de acuerdo con las características de cada organización?

A partir de esta pregunta, se definieron las siguientes preguntas específicas:

- ¿Los componentes de DevOps Model (Valores, principios, dimensiones y prácticas) son de fácil comprensión?
- ¿Los resultados obtenidos mediante la aplicación del modelo de valoración generado a partir de DevOps Model proporciona información útil a las organizaciones para mejorar sus procesos de adopción de DevOps?
- ¿La aplicación de DevOps Model requiere de un nivel de esfuerzo aceptable para las organizaciones?

Este instrumento presenta los siguientes elementos:

1. Aceptación de participación en estudio de caso
2. Cuestionario de preguntas del método de valoración, donde el evaluado asignara un valor de SI o NO a cada una de las preguntas establecidas de acuerdo con la evidencia del cumplimiento en el proceso de desarrollo.

Para obtener una valoración de “SI” en el instrumento de valoración en una de las preguntas definidas, se debe tener evidencia de su cumplimiento total, es decir, en caso de que el cuestionamiento se cumpla parcialmente deberá asignar un valor de “NO” y usar el campo “Comentarios” del instrumento para describir el estado actual de la implementación de los elementos relacionados a la pregunta.

Una vez se diligencia el formulario la información registrada será usada para el análisis y establecimiento del grado de cumplimiento de prácticas, principios, valores y dimensiones de DevOps.

## **Aceptación de la participación en el estudio de caso**

**Nombre de la propuesta a evaluar en el estudio de caso:** DevOps Model y el método de valoración de la adopción de DevOps en empresas de desarrollo de software

**Nombre del proyecto:** DevOps Model - Modelo de referencia para la adopción de DevOps en empresas de desarrollo de software

### **Investigadores:**

Investigador principal: Esp. Ing. Jonathan Guerrero Astaiza

Director: PhD. César Jesús Pardo Calvache

### **Objetivo del estudio de caso:**

- Aplicación del modelo de valoración basado en DevOps Model en la organización con el fin de evaluar su utilidad y practicidad para llevar a cabo la valoración de la adopción de DevOps en procesos de desarrollo de software en las organizaciones.

**Beneficios del estudio:** El modelo de referencia propuesto permitirá a la industria de software disponer de una solución para llevar a cabo la valoración del proceso de adopción por medio de las prácticas, principios, dimensiones y valores de DevOps que sirva como punto de partida para diseñar planes de mejora de sus procesos.

**Riesgos asociados con el estudio:** La investigación no presenta riesgos para los participantes en el estudio. Los datos recolectados se usarán con fines académicos y se conservará el anonimato del nombre de la organización y de los participantes.

**Compensación:** No habrá ningún tipo de retribución económica por participar en la investigación y los resultados de esta serán utilizados para el mejoramiento de la propuesta.

**Voluntariedad:** Usted esta siendo invitado a participar en este estudio de investigación. Antes de decidir si participa o no, debe conocer y comprender cada uno de los siguientes apartados. Este proceso se conoce como consentimiento informado, siéntase con absoluta libertad para preguntar sobre cualquier aspecto que le ayude a resolver sus dudas al respecto. Una vez que haya comprendido el estudio y si usted desea participar, entonces se le pedirá que firme esta forma de consentimiento, de la cual se le entregara una copia firmada y fechada.

- No habrá ninguna consecuencia desfavorable en caso de no aceptar la invitación.
- La participación es libre y voluntaria; si decide participar en el estudio, puede retirarse en el momento que lo desee, aun cuando el investigador responsable no se lo solicite, informando las razones de su decisión, la cual será respetada en su integridad.
- En el transcurso del estudio usted podrá solicitar información actualizada sobre el mismo, al investigador responsable.
- Si considera que no hay dudas ni preguntas acerca de su participación, puede, si así lo desea, firmar el consentimiento informado que se anexa a este documento.
- La investigación obtenida solamente será utilizada para la investigación mencionada en el presente documento y ante cualquier inquietud por favor comunicarse con los investigadores: Departamento de Sistemas, Universidad del Cauca, teléfono: +57 8209800 ext 2145.

**Aceptación de participación:**

He leído y comprendo la información anterior y mis preguntas han sido respondidas de manera satisfactoria. He sido informado y entiendo que los datos obtenidos en el estudio pueden ser publicados o difundidos con fines científicos. Convengo en participar en este estudio de investigación.

**Nombre:**

**Documento de identificación:**

---

**Firma**

Recibiré una copia firmada y fechada de esta forma de consentimiento.

Esta parte debe ser completada por el investigador (o su representante):

He explicado al Sr(a)\_\_\_\_\_ la naturaleza y los propósitos de la investigación; le he explicado acerca de los riesgos y beneficios que implica su participación. He contestado a las preguntas en la medida de lo posible y he preguntado si tiene alguna duda. Acepto que he leído y conozco la normatividad correspondiente para realizar investigación con seres humanos y me apego a ella.

Una vez concluida la sesión de preguntas y respuestas, se procederá a firmar el presente documento.

---

**Firma del Investigador**

---

**Fecha**

### *Instrumento de Evaluación Diligenciado*

A continuación, se presentan algunas de las respuestas diligenciadas del instrumento de evaluación usado en el estudio de caso. Los datos personales del representante de la empresa y sus participantes se mantienen ocultos por razones de confidencialidad.

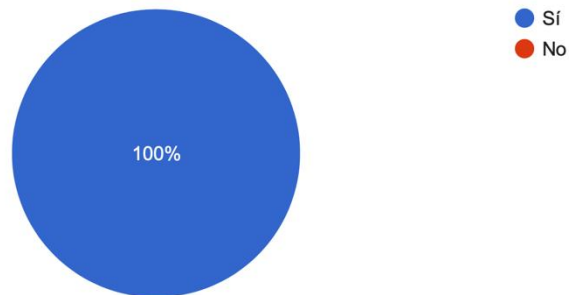
#### Cargo ocupado en la empresa

6 respuestas

TechLead
SRE
SRE Tech Lead
Tech Lead de Arquitectura
QA Automation
DevOps

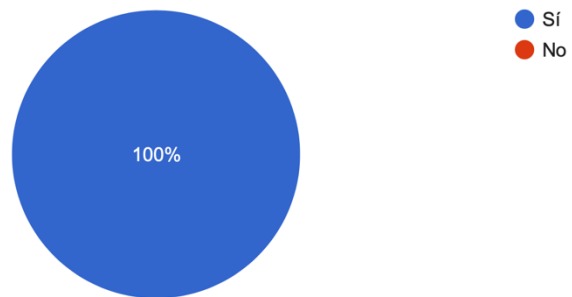
#### ¿Se tiene un sistema de control de versiones?

6 respuestas

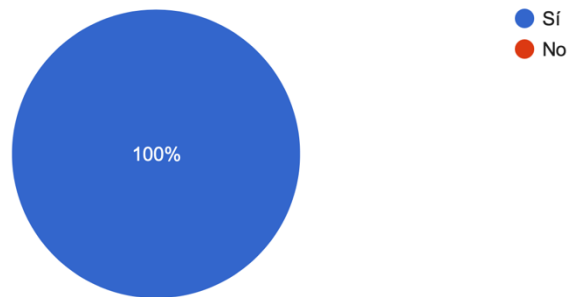




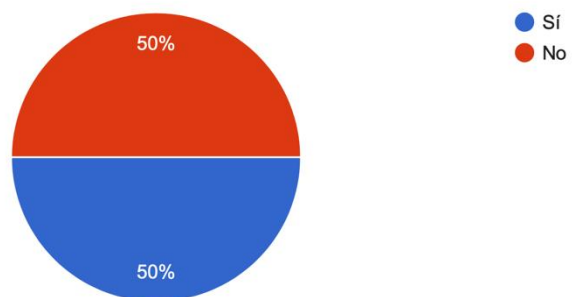
¿Equipos de desarrollo, operaciones y calidad tienen acceso al sistema de control de versiones?  
6 respuestas



¿Equipos de desarrollo, operaciones y calidad tienen acceso al sistema de control de versiones?  
6 respuestas

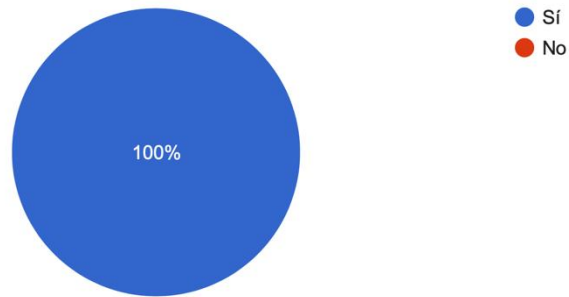


¿Se ejecutan pruebas unitarias antes de la integración?  
6 respuestas



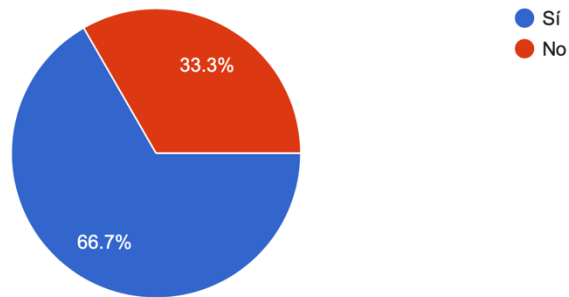
¿La compilación de código está automatizada?

6 respuestas



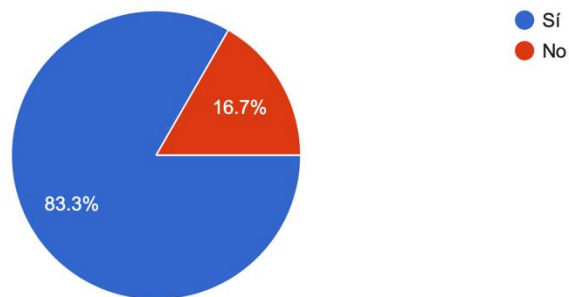
¿Hay un protocolo, procedimiento, proceso o similar para el acceso a datos?

6 respuestas



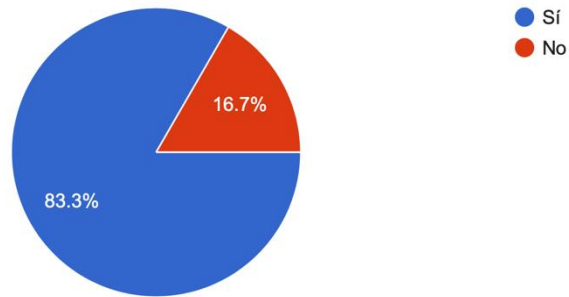
¿Las prácticas de seguridad están alineadas con el flujo de trabajo del equipo?

6 respuestas



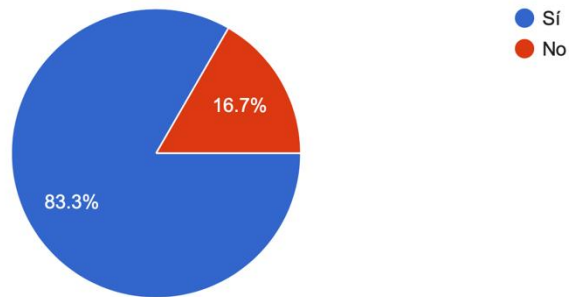
¿Existen políticas claras sobre el uso de herramientas o software de terceros?

6 respuestas



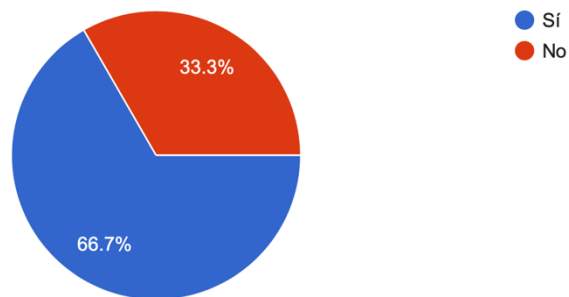
¿Se gestionan y evalúan los resultados obtenidos?

6 respuestas



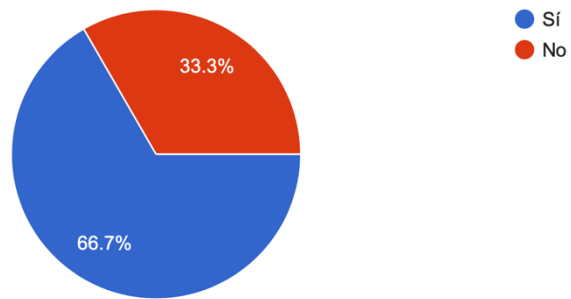
¿Existe una base de datos donde se identifican todos los activos de TI?

6 respuestas



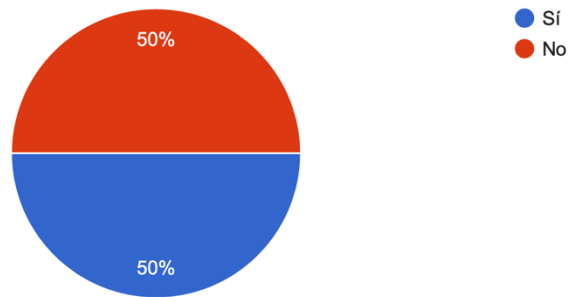
¿Se gestiona la información correspondiente a las configuraciones?

6 respuestas



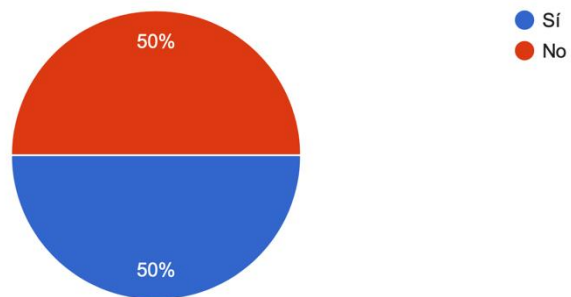
¿Existe un proceso definido para hacer seguimiento a los logs de la aplicación?

6 respuestas



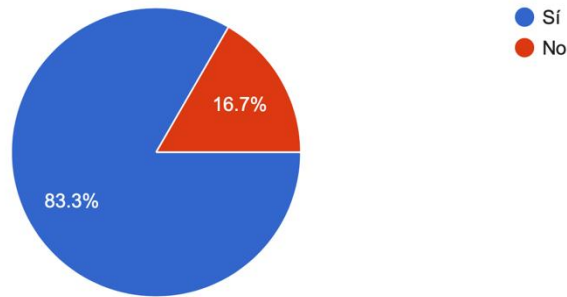
¿Existe un plan de capacitación relacionado a temas de DevOps?

6 respuestas



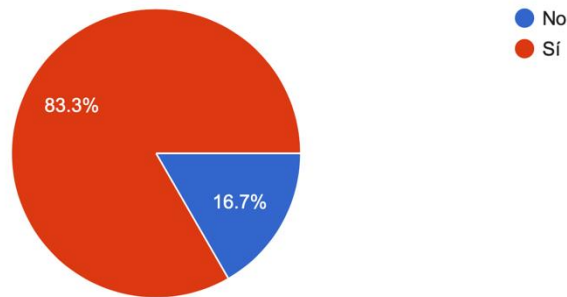
¿Existen procesos que permitan medir la cultura organizacional?

6 respuestas



¿Hay formas de reconocimiento y/o estímulo personal?

6 respuestas



*Fragmento del Informe de resultados entregado a la organización*

**Resultados generales**

En la Tabla 3.3 se muestra la distribución y porcentaje del grado de cumplimiento individual y general de cada una de las doce prácticas fundamentales propuestas en DevOps Model, así como el valor del grado de cumplimiento correspondiente de acuerdo con la escala definida en base a la norma ISO/IEC 15504.

Tabla 3.3 Resumen del grado de cumplimiento de prácticas fundamentales.

IdPrac	ValC1	ValC2	ValC3	ValC4	ValC5	ValC6	Total Val	Grado Cumplimiento
IC	100%	100%	86%	86%	71%	86%	88%	Completamente alcanzado
EC	100%	100%	100%	100%	100%	100%	100%	Completamente alcanzado
PC	88%	88%	100%	63%	13%	50%	67%	Ampliamente alcanzado
GR	100%	100%	100%	100%	50%	75%	88%	Completamente alcanzado
GD	88%	75%	75%	75%	75%	75%	77%	Ampliamente alcanzado
SS	83%	100%	83%	100%	100%	67%	89%	Completamente alcanzado
DE	100%	75%	100%	100%	50%	75%	83%	Ampliamente alcanzado

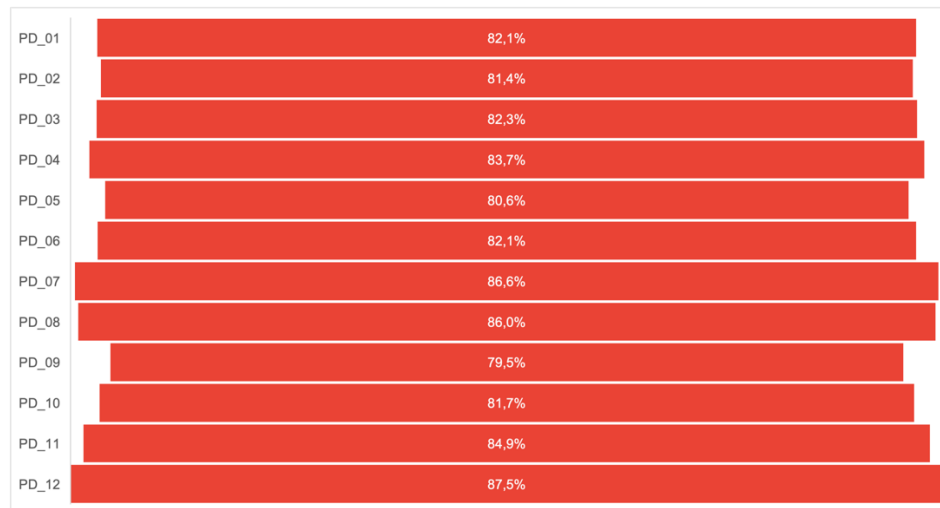
GC	100%	100%	50%	75%	50%	50%	71%	Ampliamente alcanzado
MC	67%	100%	100%	83%	83%	67%	83%	Ampliamente alcanzado
ED	100%	100%	100%	0%	50%	50%	67%	Ampliamente alcanzado
RC	100%	100%	100%	100%	100%	50%	92%	Completamente alcanzado
Mcu	100%	100%	100%	67%	100%	33%	83%	Ampliamente alcanzado

Acrónimos utilizados: **IdPrac**: Identificador de la práctica; **ValCX**: Valoración registrada por colaborador;  
**Total Val**: Valoración promedio de la empresa por práctica

Como se puede observar en la valoración realizada todas las prácticas definidas en DevOps Model tienen un nivel alto de cumplimiento, siendo las prácticas de pruebas continuas y la de educación en torno a DevOps la de menor grado de cumplimiento con un 67% y la práctica de entrega continua con un grado de completamente alcanzada con un valor de 100% lo correspondiente a la práctica de entrega continua.

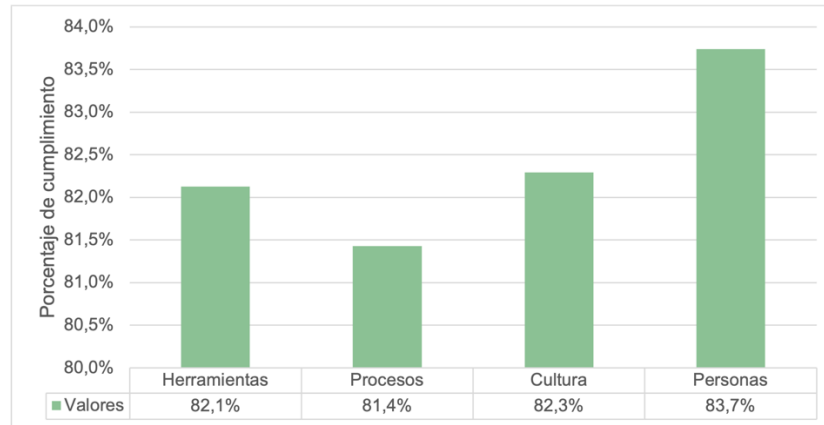
En la Figura 3-2 se puede observar el grado de cumplimiento general de cada uno de los principios definidos en DevOps Model, se puede observar que cada uno de ellos esta por encima del 80% lo que define un grado de cumplimiento *ampliamente alcanzado* según la escala definida y entre estos los principios PD\_07, PD\_08 y PD\_12 alcanzaron grado de cumplimiento *completamente alcanzado* por superar el umbral de 86%.

Figura 3.2 Grado de cumplimiento de principios de DevOps.



En la figura 3.3 se puede observar el grado de cumplimiento general de cada uno de los valores dimensiones Definidas en DevOps Model, se puede observar que las cuatro dimensione están en un nivel *ampliamente alcanzado*, siendo la dimensión de *Personas* la del porcentaje de cumplimiento mas alto con el 84% y la dimensión de *procesos* con el mas bajo porcentaje de cumplimiento con el 81%.

Figura 3.3 Grado de cumplimiento de dimensiones de DevOps



En la Tabla 3.4 se muestran cada uno de los valores definidos en DevOps Model y el grado de cumplimiento alcanzado por la empresa según los datos registrados por los seis colaboradores incluidos en el proceso por el evaluador. Se puede observar que todos los valores tienen un grado de cumplimiento *Ampliamente alcanzado* con el valor mas alto en el valor de la medición con un 85% y los valores mas bajos con un 83% los valores de *Automatización y Colaboración*.

Tabla 3.4 Grado de cumplimiento de valores de DevOps.

Valor DevOps	Procentaje obtenido	Grado de cumplimiento
Automatización	83%	Ampliamente alcanzado
Colaboración	83%	Ampliamente alcanzado
Medición	85%	Ampliamente alcanzado
Comunicación	84%	Ampliamente alcanzado

Finalmente en base a los datos obtenidos y aplicando las métricas respectivas se obtiene un porcentaje de 84% en el grado de implementación de DevOps de acuerdo con los elementos definidos en DevOps Model, lo cual categoriza a la empresa en un grado *Ampliamente alcanzado*, además, como parte del producto de trabajo denominado informe de evaluación se identificaron 8 oportunidades de mejora respecto a las prácticas fundamentales definidas en DevOps Model, las cuales se listan en la Tabla 3.5. Estos resultados obtenidos permiten a la empresa llevar a cabo la definición de un plan general de mejora que permita mejorar todo un conjunto de prácticas y actividades que le permitan lograr plenamente la implementación de DevOps.

Tabla 3.5 Oportunidades de mejora encontradas en el estudio de caso.

IdPrac	gPF	Oportunidad de mejora
IC	88%	La práctica de integración continua es la base para el cumplimiento de otras prácticas de DevOps como entrega, pruebas y despliegue continuo por lo cual se recomienda estandarizar en todos los proyectos los procesos de pruebas unitarias locales antes de enviar integración y adicionalmente incorporar pruebas de unidad automáticas en los cambios realizados repositorio de código compartido.
PC	67%	Se recomienda formalizar e informar el proceso de pruebas de la organización incluyendo las pruebas sobre las API's (unidad, funcionalidad, carga, confiabilidad, documentación) y pruebas de rendimiento.

		Como la dinámica de la industria es muy variable y es necesario la adaptación se recomienda abandonar planes de pruebas generales e incorporarse en planes de pruebas de acuerdo con cada sprint y formalizar la documentación o registros sobre incidentes en la ejecución de pruebas.
GD	77%	Estandarizar y formalizar el protocolo, procedimiento, proceso o similar para el acceso a datos. Informar en los proyectos que aplique información referente a los modelos de datos sobre los que se trabaja y especialmente donde existen datos heredados.
SS	89%	Implementar procesos que permitan la detección temprana de vulnerabilidades y actualizar políticas de seguridad de acuerdo con los elementos detectados.
DE	83%	Socializar el plan estratégico con los colaboradores.
GC	71%	A medida que se avanza en la construcción de procesos que permiten llevar mas rápido a producción cualquier producto es indispensable contar con todos los elementos de backup que permitan la recuperación rápida de los sistemas no solo a nivel de código fuente sino a nivel de infraestructura tecnológica donde se tenga caracterizado los activos de TI y sus configuraciones. Adicionalmente formalizar el proceso gestión y divulgación al interior de los equipos sobre los incidentes, problemas y cambios.
MC	83%	Se recomienda hacer especial seguimiento a los productos ya desplegados en producción, implementar actividades que permitan recoger información del funcionamiento del sistema desde logs de la aplicación y sistemas de recepción de feedback sobre la experiencia de los usuarios de la aplicación desplegada.
ED	67%	Incluir temas relacionados a DevOps y sus prácticas en los planes de capacitación de los colaboradores.

Acrónimos utilizados: **IdPrac**: Identificador de la práctica fundamental; **gPF**: grado de cumplimiento de la práctica fundamental.

## Resultados individuales

A nivel individual se encontraron los siguientes datos por colaborador:

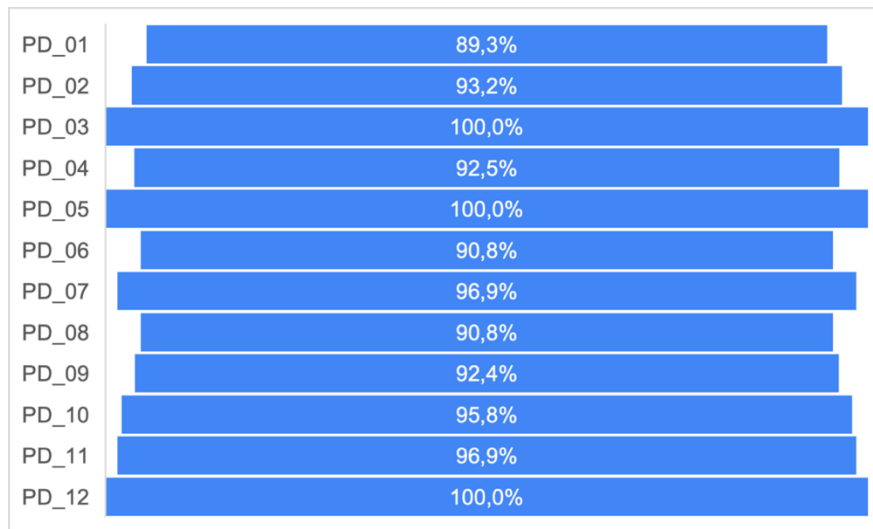
*Colaborador 1.*

### Grado de cumplimiento de prácticas fundamentales

<b>IC</b>	<i>Integración continua</i>	100%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	88%
<b>GR</b>	<i>Gestión de requisitos</i>	100%
<b>GD</b>	<i>Gestión de datos</i>	88%
<b>SS</b>	<i>Supervisión de la seguridad</i>	83%
<b>DE</b>	<i>Dirección estratégica</i>	100%
<b>GC</b>	<i>Gestión de la configuración</i>	100%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	67%
<b>ED</b>	<i>Educación en torno a DevOps</i>	100%
<b>RC</b>	<i>Realimentación continua e innovación</i>	100%
<b>Mcu</b>	<i>Medición de la cultura</i>	100%



Figura 3.4. Grado de cumplimiento de principios de DevOps Colaborador1.



**Grado de cumplimiento de dimensiones de DevOps**

DD_01	Herramientas	89%
DD_02	Procesos	93%
DD_03	Cultura	100%
DD_04	Personas	93%

**Grado de cumplimiento de valores de DevOps**

Automatización	95%
Colaboración	95%
Medición	97%
Comunicación	93%

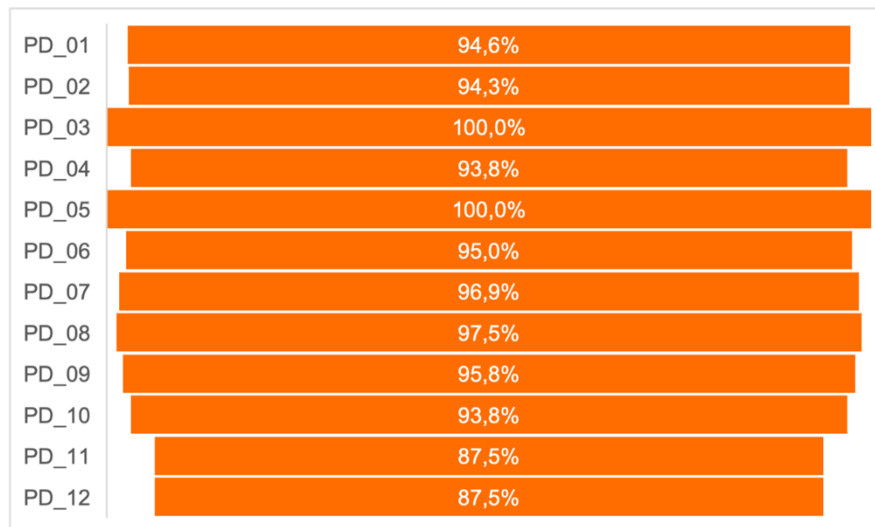
**Grado de cumplimiento implementación de DevOps** 95%

Colaborador 2.

**Grado de cumplimiento de prácticas fundamentales**

<b>IC</b>	<i>Integración continua</i>	100%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	88%
<b>GR</b>	<i>Gestión de requisitos</i>	100%
<b>GD</b>	<i>Gestión de datos</i>	75%
<b>SS</b>	<i>Supervisión de la seguridad</i>	100%
<b>DE</b>	<i>Dirección estratégica</i>	75%
<b>GC</b>	<i>Gestión de la configuración</i>	100%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	100%
<b>ED</b>	<i>Educación en torno a DevOps</i>	100%
<b>RC</b>	<i>Realimentación continua e innovación</i>	100%
<b>Mcu</b>	<i>Medición de la cultura</i>	100%

Figura 3.5. Grado de cumplimiento de principios de DevOps Colaborador2.



**Grado de cumplimiento de dimensiones de DevOps**

DD_01	Herramientas	95%
DD_02	Procesos	94%
DD_03	Cultura	100%
DD_04	Personas	94%

**Grado de cumplimiento de valores de DevOps**

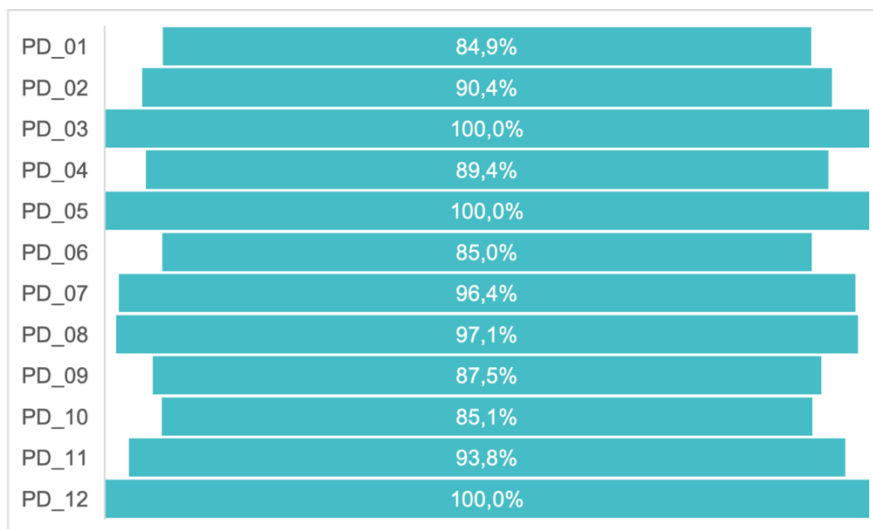
Automatización	97%
Colaboración	95%
Medición	93%
Comunicación	93%

**Grado de cumplimiento implementación de DevOps** 94%

**Grado de cumplimiento de prácticas fundamentales**

<b>IC</b>	<i>Integración continua</i>	86%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	100%
<b>GR</b>	<i>Gestión de requisitos</i>	100%
<b>GD</b>	<i>Gestión de datos</i>	75%
<b>SS</b>	<i>Supervisión de la seguridad</i>	83%
<b>DE</b>	<i>Dirección estratégica</i>	100%
<b>GC</b>	<i>Gestión de la configuración</i>	50%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	100%
<b>ED</b>	<i>Educación en torno a DevOps</i>	100%
<b>RC</b>	<i>Realimentación continua e innovación</i>	100%
<b>Mcu</b>	<i>Medición de la cultura</i>	100%

Figura 3.6 Grado de cumplimiento de principios de DevOps Colaborador3.



**Grado de cumplimiento de dimensiones de DevOps**

DD_01	Herramientas	85%
DD_02	Procesos	90%
DD_03	Cultura	100%
DD_04	Personas	89%

**Grado de cumplimiento de valores de DevOps**

Automatización	93%
Colaboración	94%
Medición	96%
Comunicación	92%

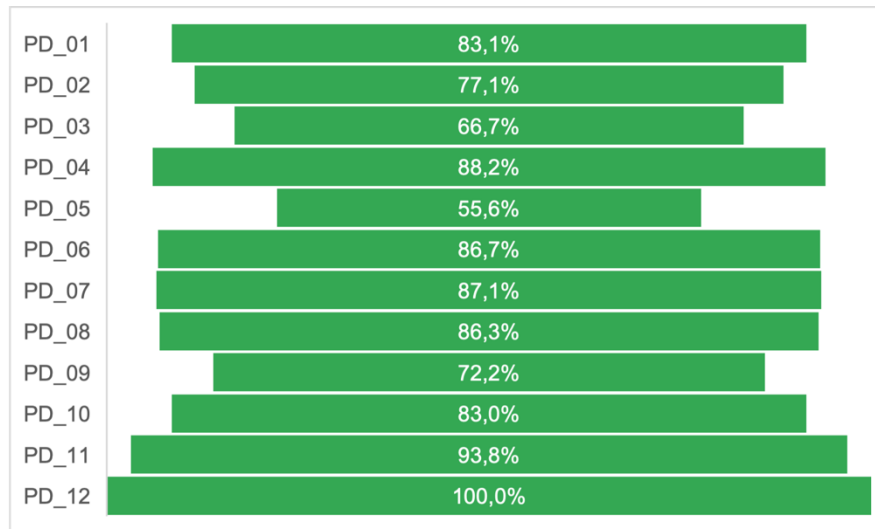
**Grado de cumplimiento implementación de DevOps** 94%

Colaborador 4

**Grado de cumplimiento de prácticas fundamentales**

<b>IC</b>	<i>Integración continua</i>	86%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	63%
<b>GR</b>	<i>Gestión de requisitos</i>	100%
<b>GD</b>	<i>Gestión de datos</i>	75%
<b>SS</b>	<i>Supervisión de la seguridad</i>	100%
<b>DE</b>	<i>Dirección estratégica</i>	100%
<b>GC</b>	<i>Gestión de la configuración</i>	75%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	83%
<b>ED</b>	<i>Educación en torno a DevOps</i>	0%
<b>RC</b>	<i>Realimentación continua e innovación</i>	100%
<b>Mcu</b>	<i>Medición de la cultura</i>	67%

Figura 3.7. Grado de cumplimiento de principios de DevOps Colaborador4.



**Grado de cumplimiento de dimensiones de DevOps**

DD_01	Herramientas	83%
DD_02	Procesos	77%
DD_03	Cultura	67%
DD_04	Personas	88%

**Grado de cumplimiento de valores de DevOps**

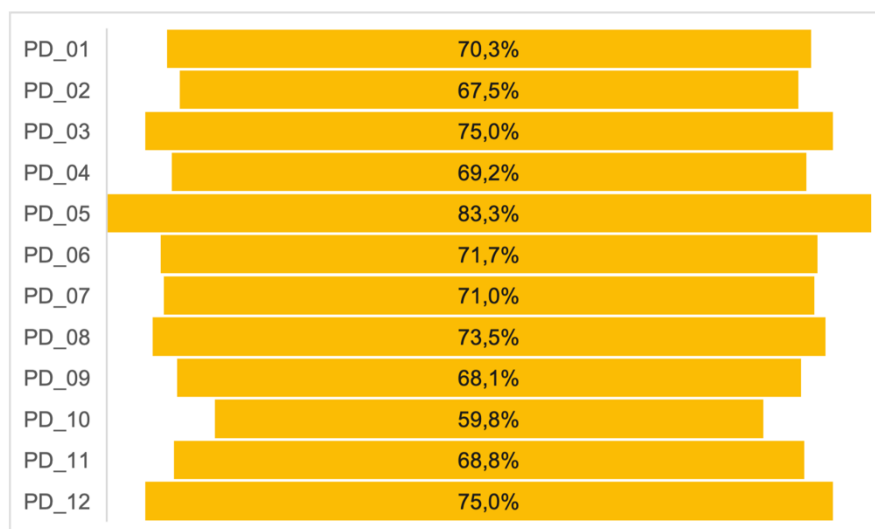
Automatización	75%
Colaboración	79%
Medición	88%
Comunicación	88%

**Grado de cumplimiento implementación de DevOps** 83%  
*Colaborador 5*

**Grado de cumplimiento de prácticas fundamentales**

<b>IC</b>	<i>Integración continua</i>	71%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	13%
<b>GR</b>	<i>Gestión de requisitos</i>	50%
<b>GD</b>	<i>Gestión de datos</i>	75%
<b>SS</b>	<i>Supervisión de la seguridad</i>	100%
<b>DE</b>	<i>Dirección estratégica</i>	50%
<b>GC</b>	<i>Gestión de la configuración</i>	50%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	83%
<b>ED</b>	<i>Educación en torno a DevOps</i>	50%
<b>RC</b>	<i>Realimentación continua e innovación</i>	100%
<b>Mcu</b>	<i>Medición de la cultura</i>	100%

Figura 3.8. Grado de cumplimiento de principios de DevOps Colaborador5.



**Grado de cumplimiento de dimensiones de DevOps**

<b>DD_01</b>	Herramientas	70%
<b>DD_02</b>	Procesos	67%
<b>DD_03</b>	Cultura	75%
<b>DD_04</b>	Personas	69%

**Grado de cumplimiento de valores de DevOps**

Automatización	72%
Colaboración	73%
Medición	71%

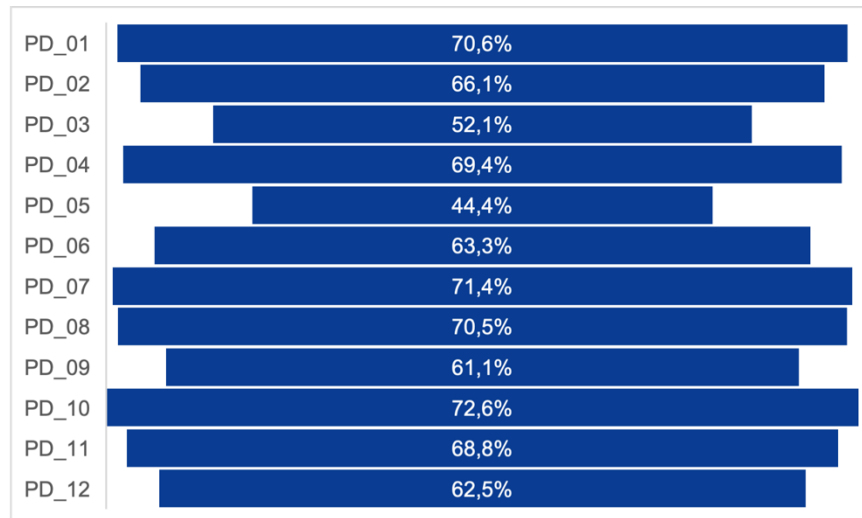
Comunicación 71%

**Grado de cumplimiento implementación de DevOps** 72%  
*Colaborador 6*

**Grado de cumplimiento de prácticas fundamentales**

<b>IC</b>	<i>Integración continua</i>	86%
<b>EC</b>	<i>Entrega continua</i>	100%
<b>PC</b>	<i>Pruebas continuas</i>	50%
<b>GR</b>	<i>Gestión de requisitos</i>	75%
<b>GD</b>	<i>Gestión de datos</i>	75%
<b>SS</b>	<i>Supervisión de la seguridad</i>	67%
<b>DE</b>	<i>Dirección estratégica</i>	75%
<b>GC</b>	<i>Gestión de la configuración</i>	50%
<b>MC</b>	<i>Monitoreo y observabilidad continua</i>	67%
<b>ED</b>	<i>Educación en torno a DevOps</i>	50%
<b>RC</b>	<i>Realimentación continua e innovación</i>	50%
<b>Mcu</b>	<i>Medición de la cultura</i>	33%

Figura 0-1. Grado de cumplimiento de principios de DevOps Colaborador6.



**Grado de cumplimiento de dimensiones de DevOps**

<b>DD_01</b>	Herramientas	71%
<b>DD_02</b>	Procesos	66%
<b>DD_03</b>	Cultura	52%
<b>DD_04</b>	Personas	69%

**Grado de cumplimiento de valores de DevOps**

Automatización	62%
Colaboración	63%
Medición	67%

**Análisis de resultados del estudio de caso**

En la Tabla 3.6 y la Tabla 3.7 se presenta el esfuerzo y los costos asociados al estudio de caso tomando en cuenta los roles implicados en el proceso y el tiempo dedicado por cada uno de estos roles para realizar el proceso de valoración de las prácticas fundamentales, principios, valores y dimensiones de acuerdo con el modelo de referencia DevOps Model. Las actividades tomadas en cuenta para realizar el cálculo del esfuerzo fueron las siguientes: (1) planificación de la valoración, (2) ejecución de la valoración, (3) actividad de aclaración de dudas y (4) actividad de generación y socialización de resultados.

Respecto a los costos, se usó como referencia los valores promedio por hora en dólares americanos de los roles implicados de acuerdo con las tarifas definidas en <https://www.payscale.com> y según cada actividad desarrollada. De acuerdo con los valores obtenidos de esfuerzo y costo se puede pensar el proceso de valoración realizado es aplicable a pequeñas y medianas empresas que quieran iniciar una vigilancia para construir procesos de mejora ya el esfuerzo y costo es relativamente bajo, esto en parte a que el instrumento de valoración se puede diligenciar rápidamente si se tiene el conocimiento y la información requerida, eliminando cualquier efecto negativo o sobrecarga considerable en las actividades diarias de las personas involucradas. Además, los resultados de las métricas se calculan rápidamente desde la Hoja de cálculo generada.

Tabla 3.6. Relación de esfuerzo del estudio de caso.

Roles	Tiempo invertido por actividades programadas medido en minutos				
	Planificación de la valoración	Ejecución de la valoración	Aclaración de dudas	Generación y socialización de resultados	Esfuerzo total
VP of Engineering	55	10	2	20	82
TechLead	0	7	0	0	7
SRE	0	7	0	0	7
SRE Tech Lead	0	20	0	0	20
Tech Lead de Arquitectura	0	7	0	0	7
QA Automation	0	90	0	0	90
Ingeniero DevOps	55	120	0	0	175
Total del esfuerzo del estudio de caso					388 minutos

Tabla 3.7. Relación de costos del estudio de caso.

Roles	Costo hora promedio por rol en dólares americanos según [99]	Total costo por rol en dólares americanos
VP of Engineering	72.13	98,57
TechLead	19.74	2,30
SRE	40.15	4,68
SRE Tech Lead	40.15	13,38
Tech Lead de Arquitectura	72.36	8,44
QA Automation	25.94	38,91
Ingeniero DevOps	47.69	139,1
Costo total del estudio de caso		305,38 dolares

