



Universidad  
Señor de Sipán

**FACULTAD DE INGENIERÍA, ARQUITECTURA Y  
URBANISMO**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**TESIS**

**Aplicación de buenas prácticas para entornos de  
desarrollo de software basados en DevOps para mejorar  
la integración y despliegue de proyectos en una empresa  
consultora de la ciudad de Lima**

**PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO DE SISTEMAS**

**Autor:**

**Bach. Aviles Odar Dennis Mauricio  
ORCID: <https://orcid.org/0000-0001-5456-9655>**

**Asesor:**

**Mg. Bravo Ruiz Jaime Arturo  
ORCID: <https://orcid.org/0000-0003-1929-3969>**

**Línea de Investigación:**

**Infraestructura, Tecnología y Medio Ambiente**

**Pimentel – Perú**

**2023**

**APLICACIÓN DE BUENAS PRÁCTICAS PARA ENTORNOS DE DESARROLLO  
DE SOFTWARE BASADOS EN DEVOPS PARA MEJORAR LA INTEGRACIÓN  
Y DESPLIEGUE DE PROYECTOS EN UNA EMPRESA CONSULTORA DE LA  
CIUDAD DE LIMA**

**Aprobación del jurado**

---

**DR. VÁSQUEZ LEYVA OLIVER**  
**Presidente del jurado de Tesis**

---

**MG. ATALAYA URRUTIA CARLOS WILLIAM**  
**Secretario del Jurado de Tesis**

---

**MG. BRAVO RUIZ JAIME ARTURO**  
**Vocal del Jurado de Tesis**



Universidad  
Señor de Sipán

### DECLARACIÓN JURADA DE ORIGINALIDAD

Quien suscribe la DECLARACIÓN JURADA, soy egresado (s) del Programa de Estudios de **Ingeniería de Sistemas** de la Universidad Señor de Sipán S.A.C, declaro bajo juramento que soy autor del trabajo titulado:

**APLICACIÓN DE BUENAS PRÁCTICAS PARA ENTORNOS DE DESARROLLO DE  
SOFTWARE BASADOS EN DEVOPS PARA MEJORAR LA INTEGRACIÓN Y  
DESPLIEGUE DE PROYECTOS EN UNA EMPRESA CONSULTORA DE LA CIUDAD DE  
LIMA**

El texto de mi trabajo de investigación responde y respeta lo indicado en el Código de Ética del Comité Institucional de Ética en Investigación de la Universidad Señor de Sipán, conforme a los principios y lineamientos detallados en dicho documento, en relación con las citas y referencias bibliográficas, respetando el derecho de propiedad intelectual, por lo cual informo que la investigación cumple con ser inédito, original y autentico.

En virtud de lo antes mencionado, firma:

Avilés Odar Dennis Mauricio	DNI: 71386344	
-----------------------------	---------------	---

## **Dedicatoria**

A mis padres, por haberme forjado y ayudado a culminar esta etapa de mi vida y por sentar los cimientos necesarios en mi formación personal y profesional.

A los docentes de la Universidad Señor de Sipán por compartir sus experiencias y conocimientos, los que me ayudaron a iniciar y culminar esta investigación.

## **Agradecimientos**

A la empresa en estudio por darme la posibilidad de realizar mi tesis en sus establecimientos y darme las facilidades y apoyo constante.

A la cooperación de todos los colaboradores del caso de estudio, en especial los que pertenecen al área de desarrollo y operaciones.

Al Mg. Ing. Jaime Arturo Bravo Ruiz por su asesoramiento constante durante la realización de mi trabajo de investigación.

## Índice

<i>Dedicatoria</i> .....	<i>iv</i>
<i>Agradecimientos</i> .....	<i>v</i>
<i>Índice</i> .....	<i>vi</i>
<i>Índice de Tablas y Figuras</i> .....	<i>viii</i>
<i>Resumen</i> .....	<i>xii</i>
<i>Abstract</i> .....	<i>xiii</i>
<i>I. INTRODUCCIÓN</i> .....	<i>14</i>
1.1. Realidad Problemática. ....	14
1.2. Formulación del Problema.....	26
1.3. Hipótesis. ....	26
1.4. Objetivos. ....	27
1.5. Teorías relacionadas al tema. ....	27
<i>II. MÉTODO</i> .....	<i>38</i>
2.1. Tipo y Diseño de Investigación.....	38
2.2. Variables, Operacionalización. ....	38
2.3. Población de estudio y muestra.....	41
2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad. ....	41
2.5. Procedimiento de análisis de datos. ....	44
2.6. Criterios éticos.....	45
<i>III. RESULTADOS Y DISCUSIÓN</i> .....	<i>47</i>

3.1.	Resultados .....	47
3.2.	Discusión de resultados. ....	84
3.3.	Aporte práctico. ....	86
IV.	<i>CONCLUSIONES Y RECOMENDACIONES</i> .....	109
4.1.	Conclusiones.....	109
4.2.	Recomendaciones.....	109
	<i>REFERENCIAS</i> . ....	111
	<i>ANEXOS</i> .....	115

## Índice de Tablas y Figuras

Tabla 1. <i>Operacionalización de variables</i> .....	39
Tabla 2 Validación del instrumento mediante juicio de expertos .....	42
Tabla 3. <i>Indicador Velocidad (Desarrollo - despliegue)</i> .....	47
Tabla 4. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	47
Tabla 5. <i>Proyecto 2 – Entidad financiera peruana</i> .....	48
Tabla 6. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	49
Tabla 7. <i>Indicador Velocidad (Frecuencia de liberación de código)</i> .....	50
Tabla 8. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	50
Tabla 9. <i>Proyecto 2 – Entidad financiera peruana</i> .....	51
Tabla 10. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	51
Tabla 11. <i>Indicador Calidad (Ratio de éxito)</i> .....	51
Tabla 12. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	52
Tabla 13. <i>Proyecto 2 – Entidad financiera peruana</i> .....	53
Tabla 14. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	53
Tabla 15. <i>Indicador Velocidad (Desarrollo - despliegue)</i> .....	54
Tabla 16. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	54
Tabla 17. <i>Proyecto 2 – Entidad financiera peruana</i> .....	55
Tabla 18. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	56
Tabla 19. <i>Indicador Velocidad (Frecuencia de liberación de código)</i> .....	56
Tabla 20. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	57
Tabla 21. <i>Proyecto 2 – Entidad financiera peruana</i> .....	57
Tabla 22. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	58
Tabla 23. <i>Indicador Calidad (Ratio de éxito)</i> .....	58
Tabla 24. <i>Proyecto 1 – Entidad financiera estadounidense</i> .....	59
Tabla 25. <i>Proyecto 2 – Entidad financiera peruana</i> .....	59
Tabla 26. <i>Proyecto 3 – Empresa de mejoramiento del hogar</i> .....	60

Tabla 27. <i>Contraste de resultados del pretest y postest</i> .....	60
Tabla 28. <i>Encuesta TAM – Pregunta N°01</i> .....	66
Tabla 29. <i>Encuesta TAM – Pregunta N°02</i> .....	67
Tabla 30. <i>Encuesta TAM – Pregunta N°03</i> .....	68
Tabla 31. <i>Encuesta TAM – Pregunta N°04</i> .....	69
Tabla 32. <i>Encuesta TAM – Pregunta N°05</i> .....	70
Tabla 33. <i>Encuesta TAM – Pregunta N°06</i> .....	71
Tabla 34. <i>Encuesta TAM – Pregunta N°07</i> .....	73
Tabla 35. <i>Encuesta TAM – Pregunta N°8</i> .....	74
Tabla 36. <i>Encuesta TAM – Pregunta N°9</i> .....	75
Tabla 37. <i>Encuesta TAM – Pregunta N°10</i> .....	76
Tabla 38. <i>Encuesta TAM – Pregunta N°11</i> .....	77
Tabla 39. <i>Encuesta TAM – Pregunta N°12</i> .....	78
Tabla 40. <i>Encuesta KANO – Pregunta N°01</i> .....	80
Tabla 41. <i>Encuesta KANO – Pregunta N°02</i> .....	81
Tabla 42. <i>Encuesta KANO – Pregunta N°03</i> .....	82
Tabla 43. <i>Encuesta KANO – Pregunta N°04</i> .....	83
Tabla 44. <i>Proyectos considerados en la investigación</i> .....	87
Tabla 45. <i>Diferentes entornos de desarrollo en fases del ciclo de software</i> . .....	87
Tabla 46. <i>Característica de los proyectos de muestra de estudio</i> . .....	89
Tabla 47. <i>Selección de buenas prácticas para el caso de estudio</i> . .....	91
Tabla 48. <i>Guías seleccionadas existentes para la implementación de DevOps</i> . .....	93
Tabla 49. <i>Formato de Evaluación, principales procesos y debilidades</i> .....	97
Tabla 50. <i>Formato para la fase de Planeación</i> .....	99
Tabla 51. <i>Formato de Identificación de requerimientos</i> .....	100
Tabla 52. <i>Formato de seguimiento o cambios de los requerimientos</i> .....	101
Tabla 53. <i>Formato de aprobación de los requerimientos</i> . .....	102

Tabla 54. <i>Formato de Identificación de requerimientos</i> .....	103
Tabla 55. <i>Costos de implementación</i> .....	130
Figura 1. Ciclo de vida del desarrollo de software.....	29
Figura 2. Indicador Velocidad (Desarrollo - Despliegue). ....	62
Figura 3. Mejoras en porcentajes del Indicador Velocidad (Desarrollo - Despliegue). ....	62
Figura 4. Indicador Velocidad (Frecuencia de liberación de código). ....	63
Figura 5. Mejoras en porcentajes del Indicador Velocidad (Frecuencia de liberación de código).....	63
Figura 6. Indicador Calidad (Ratio de éxito). ....	64
Figura 7. Mejoras en porcentajes del Indicador Calidad (Ratio de éxito).....	65
Figura 8. Pregunta N°01 – Encuesta TAM. ....	66
Figura 9. Pregunta N°02 – Encuesta TAM. ....	67
Figura 10. Pregunta N°03 – Encuesta TAM. ....	69
Figura 11. Pregunta N°04 – Encuesta TAM. ....	70
Figura 12. Pregunta N°05 – Encuesta TAM. ....	71
Figura 13. Pregunta N°06 – Encuesta TAM. ....	72
Figura 14. Pregunta N°07 – Encuesta TAM. ....	73
Figura 15. Pregunta N°08 – Encuesta TAM. ....	74
Figura 16. Pregunta N°09 – Encuesta TAM. ....	75
Figura 17. Pregunta N°10 – Encuesta TAM. ....	77
Figura 18. Pregunta N°11 – Encuesta TAM. ....	78
Figura 19. Pregunta N°12 – Encuesta TAM. ....	79
Figura 20. Pregunta N°01 – Encuesta KANO.....	80
Figura 21. Pregunta N°02 – Encuesta KANO.....	81
Figura 22. Pregunta N°03 – Encuesta KANO.....	82
Figura 23. Pregunta N°04 – Encuesta KANO.....	83

Figura 24. Aporte general del trabajo de investigación.....	86
Figura 25. Resumen de la guía de buenas prácticas basadas en DevOps. ....	95
<i>Figura 26.</i> Diagrama de Gantt por proyecto. ....	100
Figura 27. Prototipo de Diseño de Software.....	102
Figura 28. Docker image para los diferentes nodos. ....	104
Figura 29. Contenedor de Docker y los Servicios. ....	104
Figura 30. DevOps Repos.....	105
Figura 31. Azure test plants. ....	106
Figura 32. Interfaz de Azure Pipelines. ....	107
<i>Figura 33.</i> Interfaz de Azure Monitor.....	108
Figura 22. Esquema de procesos principales y fases de la empresa consultora.....	137
Figura 35 Ejemplo genérico de vista de casos de uso. ....	140
Figura 36 Ejemplo genérico de diagrama de componentes.....	141
Figura 37 Ejemplo genérico de diagrama de secuencia.....	141
Figura 38 Ejemplo genérico de diagrama de clases.....	142
Figura 39 Ejemplo genérico de diagrama de subsistemas. ....	142
Figura 40 Ejemplo genérico de vista de implementación. ....	143
Figura 41 Ejemplo genérico de digrama de procesos. ....	143
Figura 42 Ejemplo genérico de digrama de despliegue.....	144
Figura 43 Ejemplo genérico de digrama de modelo entidad relación. ....	144
Figura 44 Ejemplo genérico de matriz de trazabilidad con módulos ya funcionales. ....	145
Figura 45. Integración Continua para la empresa Consultora. ....	147
Figura 46. Despliegue continuo para la empresa Consultora. ....	148

## Resumen

Las organizaciones día a día enfrentan un nuevo y cambiante mecanismo tecnológico, es por ello que debe estar enfocada en resolver problemáticas específicas, generar ventaja competitiva, impactar positivamente mediante sus indicadores de desempeño y resultados y proporcionar mecanismos para enfrentar nuevos y propios desafíos, mediante la incorporación de las tecnologías de información. A pesar de ello, los desafíos a los que actualmente enfrenta una organización y en especial las empresas consultoras de Software se ven reflejados en la demora de entregas de software, la inexistencia de procesos para la gestión de cambios, la poca capacidad de procesamiento, procedimientos mal definidos para los pases entre ambientes, diversidad de errores en los entregables para el cliente y los bloqueos de los ambientes de pruebas por los programadores o desarrolladores del proyecto. El método utilizado fue la aplicación de buenas prácticas basadas en DevOps para las fases de integración y despliegue de software. Con base en la implementación, se concluyó que el impacto de la ejecución mediante una guía de buenas prácticas, encuestas de satisfacción para el cliente interno y externo y análisis de indicadores de velocidad y calidad, resultó de manera positiva logrando un mayor rendimiento y eficiencia en el caso de estudio. Para el indicador velocidad (desarrollo - despliegue), se obtuvo una mejora del 69%, 31%, 34%, indicador velocidad (frecuencia de liberación de código), 48%, 48% y 57% e indicador calidad (ratio de éxito), 34%, 46% y 19%, todos los indicadores en base a los 03 proyectos definidos en la muestra.

**Palabras Clave:** buenas prácticas, devops, integración, despliegue, empresa consultora de software, velocidad, calidad, satisfacción.

## Abstract

Every day organizations face a new and changing technological mechanism, which is why it must be focused on solving specific problems, generating competitive advantage, positively impacting through its performance indicators and results and providing mechanisms to face new and own challenges, through the incorporation of information technologies. Despite this, the challenges that an organization currently faces, and in particular Software consulting companies, are reflected in the delay in software deliveries, the non-existence of change management processes, the low processing capacity, procedures poorly defined for the passes between environments, diversity of errors in the deliverables for the client and the blocking of the test environments by the programmers or developers of the project. The method used was the application of good practices based on DevOps for the software integration and deployment phases. Based on this implementation, it was concluded that the impact of the execution through a guide of good practices, satisfaction surveys for internal and external clients and analysis of speed and quality indicators, resulted in a positive way, achieving greater performance and efficiency in the case study. For the speed indicator (development - deployment), an improvement of 69%, 31%, 34% was obtained, speed indicator (frequency of code release), 48%, 48% and 57% and quality indicator (success rate) , 34%, 46% and 19%, all indicators based on the 03 projects defined in the sample.

**Keywords:** good practices, devops, integration, deployment, software consulting company, speed, quality, satisfaction

## I. INTRODUCCIÓN

### 1.1. Realidad Problemática.

Las organizaciones día a día enfrentan un nuevo y cambiante mecanismo tecnológico. Cada día la tecnología es accesible a más personas y empresas en donde estas últimas tienen que estar a la altura de esta transformación digital [1]. En comparación con años anteriores, ahora la evolución tecnológica puede quedar obsoleta en solo meses y es necesario que las compañías tengan los parámetros e indicadores necesarios para adaptarse a todos los cambios de tal manera que pueda seguir cumpliendo con la calidad y eficiencia en el desarrollo de los servicios o productos [2].

Es importante que la organización defina y potencialice su esencia, su razón de ser. Debe estar enfocada en resolver problemáticas específicas, generar ventaja competitiva, impactar positivamente mediante sus indicadores de desempeño y proporcionar mecanismos para enfrentar nuevos y propios desafíos. Es necesario reconocer que el papel de Tecnologías de información es imprescindible para el logro de las metas estratégicas empresariales, más que todo funciona como un vehículo para establecer nuevas tácticas y estrategias [1].

A pesar de esto, los desafíos que enfrentan las empresas de consultoría de software pueden considerarse en los retrasos de las entregas de los desplegables, falta de procesos de gestión de cambios, bajo poder de procesamiento, flujos entre entornos mal definidos, bloqueos en el entorno de prueba por parte de los desarrolladores o programadores del proyecto [3].

La tecnología de los desarrollos de software es madura por lo que tienen un conjunto de buenas prácticas probadas. Las tecnologías con el pasar del tiempo, aparecen y desaparecen, se da en pocos años, y no existen guías necesarias de buenas prácticas. En el año 2019 en un informe de The Standish Group [4], se evaluó que el 68% de problemas de despliegue corresponde al desarrollo de proyectos fallidos, impactando de manera negativa en los entregables.

Perecen infinidad de motivos por la cual no existe un desarrollo de software de calidad, por ende, una incorrecta integración y despliegue del mismo; a partir de ello, solo se indicarán los que están vinculados directamente con el tema de estudio. En primer lugar, se encuentra la carencia de aplicación de buenas prácticas enfocadas en entornos de desarrollo, procesos mal definidos, que no están auditados y no incluyen una documentación, ausencia de roles, desarrolladores heterogéneos y uso de herramientas no estandarizadas [5].

A partir de todo lo descrito anteriormente, nació el marco ágil o agile, que es diferente de su predecesor (Método tradicional o en cascada). Este nuevo marco utiliza diferentes métodos y está centrado en responder al cambio, sin embargo, este se ve condicionado en sus etapas de desarrollo, implementación y posteriores pruebas del sistema. Frente a esas desventajas es que surge una necesidad de aplicar nuevas culturas en torno a la agilidad [6].

DevOps se centra en el uso de prácticas ágiles y por consiguiente en la entrega inmediata de servicios de Tecnologías de información. DevOps permite una estrecha colaboración entre los departamentos de operaciones y desarrollo que no afecten la calidad del software, sino al contrario, potencializar los procesos para hacerlos más eficientes [7].

El presente trabajo de investigación ayuda a subsanar las deficiencias presentadas en empresas consultoras, con la finalidad que no existan reprocesos, demoras en las entregas y en los cambios, poca estabilidad del software, baja seguridad y calidad. Es por ello, que DevOps responde a las mejoras de los tiempos, seguridad, estabilidad, calidad y, por ende, a generar valor a la organización.

### **Trabajos previos.**

Mishra & Otaiwi [8], realizaron la investigación, DevOps and software quality: A systematic mapping, en Noruega. En la actualidad, la capacidad y eficiencia de las organizaciones para proporcionar de forma recurrente y continua nuevas características de aplicaciones se ha convertido en un factor crítico en la industria del desarrollo de software.

El método empleado consiste en la realización de un mapeo sistemático del impacto de DevOps en la calidad del sistema a través de un conjunto de 11 preguntas. Los resultados obtenidos del trabajo de investigación revelan que las características (funciones) de uso compartido, automatización y medición de DevOps tienen una fuerte relación con el éxito y la calidad en el desarrollo de software. Se concluyó que DevOps aporta impactos positivos para garantizar la calidad del software. Este estudio se centró en la automatización, la cultura, la entrega continua y la retroalimentación rápida de DevOps. Además, que la calidad del software no es solo teoría, sino que también se puede identificar en la práctica para garantizar la calidad del software. Todavía hay estudios primarios limitados relacionados con el tema, aunque está creciendo y existe la necesidad de investigación empírica junto con investigación cualitativa basada en encuestas para comparar diferentes contextos en varias organizaciones y países.

Berinka, Foote, Burgo, & Prezzama [9], desarrollaron el artículo, *The Impact of Agile Methods and “DevOps” on Day 2+ Operations for Large Enterprises*, en Estados Unidos. Para las grandes empresas, las operaciones pueden ser muy complejas y costosas, por lo que es más importante que los usuarios operativos y los administradores estén preparados para el compromiso ágil con los desarrolladores. La investigación utiliza el método de comparación a través de una serie de entrevistas para analizar los hallazgos y desafíos clave que enfrentan las organizaciones de operaciones en la era de la ingeniería continua. Los resultados obtenidos se basan en recomendaciones. La primera recomendación es la creación de un arquitecto de software para representar el contexto de la misión importante para las operaciones de los desarrolladores, centrándose en los habilitadores transversales claves necesarios para respaldar esas operaciones. Una segunda recomendación es integrar DevOps en los sistemas y procesos de gestión de servicios de Tecnologías de Información existentes para los entornos operativos de destino. Se concluyó que las metodologías ágiles de desarrollo y operaciones (DevOps) están cambiando la forma en que se desarrolla y se entrega el software. En DevOps, el Día 1 representa cuándo se

entregan las capacidades y comienzan a existir las demandas operativas del software entregado. El día 2 (y los días siguientes), las organizaciones comienzan a hacer hincapié en la resiliencia, escala, agilidad, seguridad y más de las aplicaciones.

Yang, y otros [10], elaboraron la investigación, DevOps in Practice for Education Management Information System at ECNU, en China. Con el rápido avance del Internet, los sistemas de información educativa son más exigentes con la finalidad de ofrecer una mejor educación. Sin embargo, las limitaciones del desarrollo de sistemas educativos anteriores se exponen gradualmente, lo que ignora los requisitos cambiantes, los cuellos de botella de alta concurrencia y el desarrollo esbelto de los sistemas de información educativa. El método empleado fue: CI (Integración continua) y CD (Despliegue continuo) a través de Kubernetes. Las técnicas se plantearon en dos direcciones: gestión del código fuente de sistema de información y el proceso de integración continua durante el desarrollo. Mediante ello, se desarrolló y construyó un novedoso sistema de información educativa en ECNU basado en DevOps y técnicas relacionadas. Esta investigación revela la práctica de DevOps para el nuevo sistema de información educativa desde cuatro aspectos: CI (Integración continua), CD (Implementación continua), gestión de registros y calidad del código. Se concluyó que la práctica de DevOps para sistema de información establece que es muy conveniente para el desarrollo, prueba y lanzamiento de sistemas de información educativa, también mejora la confiabilidad, disponibilidad y escalabilidad de la plataforma de información especialmente considerando la garantía de eficiencia.

Pinheiro, Pinto, & Bonifácio [11], realizaron el artículo, Adopting DevOps in the real world: A theory, a model, and a case study, en Brasil. Debido a su creciente interés y definiciones imprecisas, los trabajos de investigación existentes han tratado de caracterizar DevOps y esto se debe a que se sabe poco sobre la adopción de DevOps en diversos profesionales y rubros. El método usado fue la teoría fundamental clásica para edificar una hipótesis a través de quince escenarios de adopción DevOps de manera exitosa en empresas de diferentes países y dominios. Se propuso un modelo (es decir, un flujo de

trabajo) basado en la teoría, utilizando definición GQM (objetivo, pregunta, métrica) y transcripciones de entrevistas. Se evaluó a través de un caso de estudio en un establecimiento perteneciente al gobierno brasileño. Se utilizó un grupo de enfoque para recopilar las percepciones de la empresa sobre la aplicación de DevOps. Los resultados contribuyen a: generar una comprensión adecuada de DevOps, desde la perspectiva de los profesionales; y ayudar a otras instituciones en el camino hacia la adopción de DevOps. Se evidenció que la colaboración es la preocupación central de DevOps, es decir, ocupa el 85% en el impacto de DevOps, en contraste con la sabiduría existente de que las herramientas y la automatización pueden ser suficientes para lograr el éxito con DevOps. Se concluyó que DevOps es un conjunto de prácticas y valores culturales que reducen las barreras entre los equipos de operaciones y desarrollo.

Stephen, Noppen, & Lettice [12], desarrollaron la investigación, Management challenges for DevOps adoption within UK SMEs, en Reino Unido. Un cambio organizacional sustancial es inherente a la adopción de DevOps, especialmente cuando existen métodos previos ya establecidos en donde está resultando difícil de adaptarse y adoptar DevOps por completo. Esta dificultad se debe a una combinación de la necesidad de mantener un sistema heredado, la falta de aceptación por parte de los altos ejecutivos, la estructura de gestión y la resistencia. Se aplicó el método Kanban para evaluar el estudio del progreso de los proyectos de software. Este estudio se realizó en una compañía de Reino Unido con más de 200 empleados. Se descubrió que, dentro de la organización de estudio del caso, se está adoptando el enfoque DevOps para el desarrollo de un nuevo sistema. Además, se observó que un 25% de proyectos aún no adoptan DevOps por completo y esto se debe a la falta de aceptación y resistencia a la transformación por parte de los colaboradores. Se concluye que DevOps es un tema interdisciplinario que se beneficiaría enormemente de una mayor atención de la investigación orientada a la gestión y potencialmente a la psicología. Todo el cambio va a depender de la organización, la estructura tradicional debe quedar en el pasado para poder aplicar la agilidad y las buenas

prácticas.

Giménez & Santos [13], desarrollaron la investigación, DevOps Maturity Diagnosis – A Case Study in Two Public Organizations, en Brasil. Muchos procesos de desarrollo de software abordan los problemas asociados con el mantenimiento de la estabilidad deseada en las operaciones de la organización, al tiempo que prestan cada vez más atención a la frecuencia y calidad de las entregas de software. Sin embargo, la brecha entre los objetivos de las unidades organizacionales y los equipos de desarrollo persiste debido a la inadecuación de las soluciones adoptadas para los escenarios únicos que enfrenta cada organización. El método empleado fue el modelo de matriz de madurez de DevOps y la de sus equipos para evaluar las habilidades y el conocimiento de las personas involucradas. Este modelo se examina desde tres puntos de vista: procesos, automatización y colaboración. Esta evaluación se aplicó a dos organismos públicos. El resultado obtenido fue que ambas organizaciones están evolucionando del nivel inicial al nivel consciente. Además, se notó que los participantes conocen la cultura DevOps, aunque las organizaciones aún no la adoptan por completo. Se concluyó que la exploración de la madurez de DevOps brinda una comprensión del grado de madurez de las empresas en términos de estandarización de procesos, herramientas de automatización y enfoques colaborativos, junto con información sobre sus oportunidades de mejora.

Gren, Torkar, & Feldt [14], hicieron el artículo, Group development and group maturity when building agile teams, en Suecia. Existe una superposición entre lo que se entiende por equipo ágil y grupo con madurez y esta confusión no se ha investigado mucho en relación con la construcción de un “equipo ágil”. El método utilizado fue Scrum y tuvo como propósito investigar cómo la construcción de equipos ágiles se conecta a un modelo de desarrollo grupal tomado de la psicología social. Se realizaron diez entrevistas semiestructuradas con coaches, Scrum Masters y gerentes responsables del proceso ágil de siete empresas diferentes, y recopiló datos de encuestas de 66 miembros del grupo de cuatro empresas (un total de ocho empresas diferentes). La encuesta incluyó una

herramienta de medición ágil y una parte del Cuestionario de desarrollo grupal. Los resultados muestran que los profesionales definen los aspectos del desarrollo del grupo como factores clave para una transición ágil exitosa. Además, la medición cuantitativa de la agilidad se correlacionó significativamente con la medición de la madurez del grupo. Se concluyó que agregar estos aspectos psicológicos a la descripción del "equipo ágil" podría aumentar la comprensión de la agilidad y, en parte, ayudar a definir un "equipo ágil". Se recomendó que en trabajos futuros se deberían desarrollar pautas específicas sobre cómo los desarrolladores de software en diferentes niveles de madurez pueden adoptar principios y prácticas ágiles de manera diferente.

Leite, Pinto, Kon, & Meirelles [15], desarrollaron la investigación, *The organization of software teams in the quest for continuous delivery: A grounded theory approach*, en Brasil. Para acelerar el tiempo de comercialización y mejorar la satisfacción del cliente, las organizaciones productoras de software han adoptado prácticas de entrega continua, lo que afecta las relaciones entre los profesionales de desarrollo e infraestructura. Sin embargo, ninguna literatura sustancial ha abordado cómo la industria del software estructura la organización de los equipos de desarrollo e infraestructura. En este estudio, el método utilizado es el de Grounded Theory, que involucra la construcción de hipótesis y teorías a través de la recolección de información y análisis de esta. Se investigó cómo las organizaciones productoras de software estructuran sus equipos de desarrollo e infraestructura, específicamente cómo es la división del trabajo entre estos grupos y cómo interactúan. Después de intercambiar ideas con 7 expertos de DevOps, se recopiló y analizó datos de 37 entrevistas semiestructuradas con profesionales de TI. Después de un análisis cuidadoso, se identificó cuatro estructuras organizativas comunes: (1) departamentos en silos, (2) DevOps clásico, (3) equipos multifuncionales y (4) equipos de plataforma. También se observó que algunas empresas están en transición entre estas estructuras. El principal aporte de este estudio es una teoría en forma de taxonomía que organiza las estructuras encontradas junto con sus propiedades. Esta teoría podría guiar a

los investigadores y profesionales a pensar en cómo estructurar mejor a los profesionales del desarrollo y la infraestructura en las organizaciones productoras de software.

Di Nitto, Jamshidi, Guerriero, Spais, & Tamburri [16], elaboraron el artículo, A software architecture framework for quality-aware DevOps, en Italia. Algunas empresas siguen utilizando las metodologías en cascada o tradicionales sin emplear las mejores prácticas y esto no permite realizar un grupo de trabajo coherente para aumentar el desarrollo de software y la velocidad de las operaciones. El método empleado fue SQUID que significa Calidad de especificación en DevOps, es un marco de arquitectura de software que admite la documentación basada en modelos de arquitecturas de software y sus propiedades de calidad en escenarios de DevOps con el objetivo de proporcionar descripciones de arquitectura de software listas para DevOps. Para permitir que los profesionales de la arquitectura de software enriquezcan y elaboren adecuadamente sus especificaciones de arquitectura de una manera que sea coherente con DevOps, se encuestó a varias partes interesadas en el tema. Se estudió las preocupaciones y los desafíos que deben abordarse con respecto a la preparación de una arquitectura de software que esté lista para DevOps, es decir, descrita con todos los detalles necesarios para implementar escenarios de DevOps. El resultado obtenido fue que el 95% de las arquitecturas de software estaban alineadas al marco de DevOps y se enumeraron dos desafíos principales que aún se mantienen en la preparación de una arquitectura. Estos fueron: La gestión tradicional y el rechazo al cambio. Se concluye que DevOps es una estrategia de ingeniería de software emergente que implica los esfuerzos conjuntos de las personas de desarrollo y mejores prácticas, en donde SQUID es un marco actual y de mucho impacto en las arquitecturas de software.

Macarthy & Bajo [17], realizaron la investigación, An Empirical Taxonomy of DevOps in Practice, en Eslovenia. Un tema que hasta ahora sigue siendo controversial ha sido si es que DevOps es una cultura de toda la organización o de una descripción de trabajo. Como DevOps es un concepto emergente, sus definiciones y mejores prácticas

aún son confusas, lo que hace que su implementación en la práctica sea menos informada y algo arriesgada. El método informado utilizado fue el de Grounded Theory. a través de un estudio de caso exploratorio, basado en entrevistas con 11 profesionales de la industria en nueve organizaciones. Las transcripciones de las entrevistas se codificaron y analizaron. Este estudio proporcionó una taxonomía empírica de implementación, que describe la interacción de los desarrolladores con operaciones locales, operaciones subcontratadas, DevOps equipos, y DevOps equipos de puente. Se presentó un mapeo novedoso de los enfoques para implementaciones locales y basadas en la nube y se identificó los facilitadores en los diferentes modos. Además, se observó tres grupos distintos de actividades en el cuarto modo: el mantenimiento y aprovisionamiento de sistemas físicos, virtualización de funciones y creación de tuberías automatizadas, implementación y mantenimiento oportuno de aplicaciones, que pueden haber dado lugar a la implementación de DevOps como equipos de puente. Los entrevistados afirmaron que estas distinciones permitieron a los desarrolladores centrarse en ofrecer valor para el negocio. Se concluyó que DevOps se describe como una cultura y filosofía de ingeniería de software que utiliza equipos multifuncionales para construir, probar y lanzar software de manera más rápida y confiable a través de la automatización. Su adopción acelera el tiempo de entrega del software, mejora la calidad, la seguridad y la colaboración en el desarrollo de software.

Hart & Burke [18], elaboraron el artículo, *An exploratory study on the devops it alignment model*, en Estados Unidos. Este estudio aborda la práctica poco estudiada de DevOps. Aunque las organizaciones continúan implementando estas buenas prácticas, pocos estudios exploran conexiones con la teoría anterior. Los autores emplearon el método PLS-SEM, modelado de ecuaciones estructurales de mínimos cuadrados parciales y de esta manera evaluar este modelo de alineación de TI. El modelo propuesto sienta las bases para una mayor investigación sobre la influencia de la teoría en DevOps utilizando métodos de investigación cuantitativos. Este estudio contribuye a esta necesidad

desarrollando el modelo DevOps estratégico de alineación de TI. La muestra incluyó 57 empresas de las listas actuales de Forbes Global 2000 y Fortune 500. El resultado se enfoca en que la integración continua del software y el intercambio de conocimientos aumenta el nivel de alineación de las subunidades de TI en las grandes organizaciones que fomentan DevOps. Por otra parte, los gerentes deben identificar cómo desarrollar vínculos más estrechos entre subunidades con diferentes conjuntos de habilidades en sus organizaciones. Se concluye que es importante explorar cómo los diferentes tipos de conocimiento en diversas disciplinas requieren la formación de vínculos interdisciplinarios únicos y si estas relaciones tienen conexiones con la teoría de la contingencia y la gestión de la calidad.

Wettinger, Breitenbucher, Kopp, & Leymann [19], Streamlining DevOps automation for Cloud applications using TOSCA as standardized metamodel, en Alemania. DevOps como paradigma creciente tiene como finalidad integrar estrechamente a la parte de operaciones con los desarrolladores, pero esto aún sigue siendo un gran reto. Todos los enfoques de automatización de DevOps heterogéneos y propietarios difieren entre sí, es difícil integrarlos y combinarlos para desarrollar aplicaciones en la nube mediante un proceso automatizado. El método empleado fue un Modelado integrado y basado en estándares y marco de tiempo de ejecución basado en TOSCA. El resultado fue que la automatización es la clave para una colaboración eficiente y una estrecha integración entre el desarrollo y las operaciones y corresponde al 75% del éxito en un proyecto. Los autores concluyen que además de los cambios culturales y organizativos que son necesarios para aplicar DevOps en la práctica, se requieren herramientas para implementar la automatización de un extremo a otro de los procesos de implementación. La comunidad de DevOps impulsa constantemente nuevos artefactos de código, herramientas y enfoques para implementar dichos procesos automatizados.

Batra [20], en su artículo, Job-work fit as a determinant of the acceptance of large-scale agile methodology, en Estados Unidos. Define que inicialmente, el desarrollo ágil

generalmente se implementaba en proyectos con un solo equipo con menos de diez miembros y es por eso que esas empresas les cuesta aumentar su tamaño por miedo al fracaso o el exceso de colaboración, pero, recientemente, ha surgido un nuevo tipo de desarrollo ágil para proyectos más grandes que pueden denominarse escala (LSAM). El método utilizado es la taxonomía de LSAM (metodología ágil a gran escala) que hace hincapié en el Manifiesto Ágil. Basado en la literatura sobre desarrollo de software, el estudio investiga cinco antecedentes importantes de aceptación de la metodología: utilidad percibida, compatibilidad, norma subjetiva, obligatoriedad y apoyo externo. Como resultado del análisis PLS inicial, el estudio introduce la noción de ajuste trabajo-trabajo como un constructo de segundo orden compuesto por la utilidad y la compatibilidad percibidas. Basado en una encuesta de 123 personas encuestadas, el estudio encuentra que el constructo trabajo-trabajo encaja explica significativamente la aceptación de LSAM para proyectos grandes. La adecuación trabajo-trabajo también media las relaciones entre la norma subjetiva y el apoyo externo con la aceptación de LSAM. Se concluye que para implementar una metodología ágil es necesario tener una cultura o la aplicabilidad de buenas prácticas, como lo es DevOps.

Rahman, Mahdavi, & Williams [21], desarrollaron la investigación A systematic mapping study of infrastructure as code research, en Estados Unidos. Los investigadores no pueden identificar áreas de investigación potenciales relacionadas con IaC (infraestructura como código) que es la práctica para configurar automáticamente las dependencias del sistema y para aprovisionar instancias locales y remotas. Se han encontrado defectos y fallas de seguridad que ocurren en los scripts de IaC. El método empleado fue la aplicación sistemática de criterios de exclusión e inclusión que incluyen la eliminación de duplicados y la eliminación de publicaciones que no estén en inglés y no revisadas por pares, se identificó 32 publicaciones relacionadas con IaC. y se observaron los temas abordados en estas publicaciones aplicando análisis cualitativos. Los resultados arrojaron que se identificaron cuatro temas estudiados en publicaciones relacionadas con

laC mediante DevOps: (i) marco / herramienta para infraestructura como código; (ii) adopción de infraestructura como código; (iii) estudio empírico relacionado con la infraestructura como código; y (iv) pruebas en infraestructura como código. Según el análisis, el 50,0% de las 32 publicaciones estudiadas proponen un marco o herramienta para implementar la práctica de laC o ampliar la funcionalidad de una herramienta laC existente. Se concluyó que los hallazgos sugieren que el marco o las herramientas es un tema bien estudiado en la investigación de laC. Dado que los defectos y las fallas de seguridad pueden tener graves consecuencias para los entornos de implementación y desarrollo en DevOps, observamos la necesidad de realizar estudios de investigación que estudien los defectos y fallas de seguridad para laC.

Toh, Sahibuddin, & Abu [22], realizó la investigación, A Review on DevOps Adoption in Continuous Delivery Process, en Malasia. El manifiesto de DevOps se ha debatido ampliamente entre investigadores y profesionales de ingeniería de software. El problema radica en que DevOps ha estado rodeado por el fenómeno de la ambigüedad porque hasta ahora no existe una definición clara de las prácticas de implementación. Este artículo intenta identificar los problemas y cuestiones en practicar la adopción que puede mejorar el proceso de CD. Esto se logró mediante la realización de un estudio de literatura para identificar el DevOps y entrega continua. El método utilizado para realizar este trabajo fue filtrar el artefacto de Internet de la base de datos Scopus. Por lo tanto, se identificaron 96 artefactos de Internet para su posterior estudio y análisis. Como resultado, hay cuatro factores de adopción importantes que deben tenerse en cuenta para determinar el éxito de la cultura DevOps en el proceso de Entrega Continua. Estos son: Capacidad de los colaboradores para adaptarse a cualquier cambio de procesos o herramientas, capacidad de la institución para adaptarse al cambio estratégico, recursos disponibles para financiar el cambio y fechas del proyecto. Se concluyó que DevOps es reconocido como un conjunto de prácticas de desarrollo y operación de software que aboga por una estrecha integración entre el desarrollo y la operación en la entrega de software y la actividad de operación para

lograr un tiempo de entrega de software más rápido. DevOps produce una práctica ágil significativa para reducir el tiempo de ciclo del software en la entrega de productos de software. Mientras tanto, Continuous Delivery (CD) es uno de los procesos de DevOps que permite al equipo de producción de software lanzar nuevas funciones y nuevos productos rápidamente.

### **Justificación e importancia del estudio.**

La presente investigación pertenece al área prioritaria de ciencias de la computación de la línea de investigación infraestructura, tecnología y medio ambiente de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Señor de Sipán.

La investigación es pertinente porque contribuye de manera positiva al avance del conocimiento, y este punto surge a partir de la poca investigación o poca cultura que se tiene sobre la aplicación de nuevas prácticas basados en DevOps ya que es un cambio cultural que puede mejorar la comunicación y la colaboración a través de la optimización en el flujo de trabajo entre diferentes campos como desarrollo, operaciones de TI y otras partes de la organización para garantizar la entrega. Su importancia radica en el valor y calidad de lo que se le entrega al cliente, de esta manera, se logrará una alta satisfacción del mismo, mejor calidad, entregas rápidas y de menor costo.

Para el país, en especial, para la ciudad de Lima, esta aplicación de buenas prácticas es viable porque ayudaría e incrementaría la calidad y productividad en las empresas consultoras, pues han adherido esta nueva cultura a todos sus colaboradores. Por consiguiente, el tema de estudio permite conocer los grandes beneficios que trae este nuevo avance para obtener éxito y crear valor en las organizaciones.

### **1.2. Formulación del Problema.**

¿De qué manera se podría aplicar las buenas prácticas para mejorar la integración y despliegue de proyectos en una empresa consultora de la Ciudad de Lima?

### **1.3. Hipótesis.**

Mediante la aplicación de buenas prácticas en los entornos de desarrollo basado

en DevOps se podrá mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima.

#### **1.4. Objetivos.**

##### **Objetivo general.**

Aplicar buenas prácticas para entornos de desarrollo de software basado en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima.

##### **Objetivos específicos.**

- Identificar el entorno de desarrollo de software en el que se aplicará las buenas prácticas.
- Seleccionar las buenas prácticas para el entorno de desarrollo elegido.
- Elaborar una guía de buenas prácticas basadas en otras existentes.
- Implementar las buenas prácticas en el entorno de desarrollo de software basado en DevOps.
- Evaluar el impacto de la ejecución de las buenas prácticas en entornos de desarrollo basados en DevOps.

#### **1.5. Teorías relacionadas al tema.**

##### **1.5.1. DevOps**

###### **1.5.1.1. Definición**

De acuerdo con Castañeda [23] “DevOps son un conjunto de prácticas orientadas a reducir el tiempo que transcurre entre el momento de la adición de un cambio en las líneas de código al código fuente y el instante en el cual dichas líneas de código han sido implementadas en producción, asegurando alta calidad”.

DevOps simboliza una transformación en la cultura de Tecnología de Información, que permite una entrega rápida de servicios o productos de TI mediante la adopción de buenas prácticas Lean y Agiles. Su cambio también se centra en los individuos (y cultura), busca mejorar la participación y el apoyo entre los equipos de desarrollo y operaciones

[24].

Willis [25], menciona que DevOps representa una gran cantidad de Movimientos de filosofía y gestión documentados en el marco de referencia del sistema de conocimiento, como la teoría de restricciones, lean, entre otros. Esta convergencia ha permitido obtener una serie de prácticas técnicas, arquitectónicas y culturales enfocados en resolver los desafíos correspondientes a la calidad, velocidad y eficiencia en la entrega de aplicaciones o servicios de la velocidad, la calidad y la eficiencia de la entrega de servicios y aplicaciones basados en software.

#### **1.5.1.2. Historia**

De acuerdo con Willis [25], los antecedentes de DevOps se remontan al año de 1956 cuando aparece el modelo de estandarización en el desarrollo de software, llamado modelo cascada. Este trae consigo las fases correspondientes a: análisis, diseño, implementación, pruebas y mantenimiento. Este modelo se usó por mucho tiempo debido a las limitaciones tecnológicas por las que se atravesaba. Tiempo después empezaron a aparecer dificultades en torno a las contradicciones e indecisiones del cliente y desde allí surge la necesidad de encontrar otra solución. A partir de ese momento aparece el término Ágil.

Una teoría rápida para comprender sería imaginarse que el término Ágil o Agile es un modelo en cascada en una dimensión más corta, con más repeticiones constantes y corto para la parte de entrega y despliegue del producto. Estos cambios derivaron a una mayor presión para realizar muchos más despliegues a los equipos de desarrollo.

El término DevOps, empezó a evidenciarse por los años 2007 y 2008 cuando las comunidades de desarrollo de software y de operaciones de TI encontraban deficiencias críticas de disfunción. Encontraron que el modelo tradicional no les favorecía y su inconformidad partía desde que el modelo solo exigía que los programadores se mantuvieran al margen de los asuntos organizacionales y operativos. Los objetivos, indicadores de rendimiento y desempeño de desarrolladores y operaciones no estaban

alineados. Lo que empezó en blogs de internet o algunas reuniones informales es ahora uno de los principales aspectos del ámbito del sistema actual, y un tema que hasta ahora sigue generando ventaja competitiva en las organizaciones.

### 1.5.1.3. Ventajas de aplicar DevOps en una organización

Según la compañía Aplyca [26], las principales ventajas de aplicar DevOps son las siguientes:

- Entregar o liberar código de forma rápida, sin esperar a acumular muchos requerimientos o implementaciones independientes.
- Facilita el trabajo y la coordinación en equipo y la comunicación entre los desarrolladores de software.
- Antes de la publicación del código, se pueden identificar errores.
- Mayor colaboración y evolución de la empresa de desarrollo.
- Menor pérdida en costos.
- Mayor desarrollo de proyectos.
- Menor trabajo y más confianza al momento de replegar y desplegar el código a la etapa de producción.
- Mayor claridad del código que se está implementando.

### 1.5.1.4. Ciclo de vida



Figura 1. Ciclo de vida del desarrollo de software.

Fuente [27]

El software es un producto muy complejo desde la forma de construir, elaborar, mantener mejorar y evolucionar, por esto se cuenta con métodos estructurados para su formación y creación. Existen métodos que complementan y se le conocen como modelos de ciclo de vida de desarrollo de Software, en inglés, The Software Development Life Cycle (SDLC).

Son muchos los modelos importantes, los más conocidos son: Cascada, Prototipado, Espiral, Modelo V, Iterativo, Ágil e Incremental [27].

Todos los modelos tienen etapas de planificación, análisis, diseño, desarrollo, despliegue y mantenimiento, se sabe que la transformación digital necesita modelos de ingeniería del Software que permite poner a disposición las aplicaciones de software a las necesidades de los negocios que resultan ser cambiantes, es por eso que esta investigación propone fases o etapas que concuerden con el ciclo de software el ágil: [28].

- Concepción.
- Incepción.
- Construcción / Iteraciones.
- Transición.
- Producción.
- Retiro

En ese orden de fases, las prácticas DevOps para la entrega continua del sistema se basan en:

- Construcción / Iteraciones.
- Transición
- A partir de ello, se comprenden las siguientes etapas: [29].
- Requerimientos de software: Son definidos en un documento de requerimientos del producto a desarrollarse.
- Análisis: Los resultados son necesariamente los modelos, reglas de negocio o algún esquema.

- Diseño: Los resultados obtenidos son parte de la arquitectura de sistema.
- Codificación: Incluye el desarrollo, las pruebas y la integración misma de los componentes del sistema.
- Pruebas: Se descubre sistemáticamente los componentes del software.
- Operaciones: Incluye procesos como la instalación, migración. Soporte y mantenimiento del sistema completo.
- Las prácticas tienen diversas etapas, como:
  - Codificación.
  - Pruebas.
  - Operaciones.

#### **1.5.1.5. Herramientas de DevOps**

Existe una afinidad entre desarrolladores y profesionales de Operaciones de TI y el DevOps facilita su interacción, de esta manera se busca una mejora y también la entrega de valor del negocio. Posee un ciclo de vida que guarda un vínculo con la automatización de los procedimientos del desarrollo del software, desde su inicio o creación hasta la operación en sí, consta de ocho etapas que pueden variar debido a que algunas etapas se pueden agrupar con otras o cada etapa presente diferentes componentes [17].

A continuación, algunos grupos de herramientas que necesita el DevOps [6]:

##### **a) Manejo del código fuente:**

Es también conocido como el control de versiones, y consiste en el seguimiento que se realiza en los cambios de un código de software, esto puede servir para desarrollar una aplicación sencilla donde una persona sea la protagonista, como también para desarrollar un software de un rango más amplio donde se necesite de más personas y un trabajo en conjunto.

Estas herramientas son muy importantes porque ayudan a optimizar el desarrollo del software, para esto, es necesario de una buena comunicación entre los miembros que se encuentran involucrados realizando diferentes funciones que

son fundamentales.

**b) Virtualización:**

Para el desarrollo en la cultura DevOps se requiere de un procedimiento diario que consiste en un testeo regular del código, de esta manera la conversión de los servicios a la virtualización resulta ser fundamental, como resultado de esto se proporciona un entorno que simula ambientes de manera ininterrumpida; estos ambientes deben tener los mismos requisitos y las mismas características para poder ser instalado en el software.

**c) Cloud:**

El Cloud Computing ayuda a alcanzar dos aspectos fundamentales de DevOps, la agilidad y la comodidad, considerando esto se gestiona una buena infraestructura. En un inicio era conocido por algunas organizaciones como una forma de ajuste de presupuesto porque se escogía un proveedor de servicios en lugar de poder invertir en una infraestructura física, con el Cloud se realiza un pago por uso, sin contratos previos o pagos por adelantado. No solo se caracteriza por favorecer al ahorro económico, sino también por la rapidez y facilidad con que la nube mantiene la infraestructura de TI haciendo que la operación tenga continuidad, esta agilidad y eficacia se da para el equipo de Operaciones de TUI y a sus programadores también.

**1.5.2. Automatización de Procesos**

**1.5.2.1. Integración continua**

Con esta fase se integran frecuentemente los códigos programados a la parte principal de un repositorio común. En vez de desarrollar partes del código de manera aislada las partes del código y unirlas al final del ciclo, un programador o desarrollador podrá contribuir alguna modificación a una parte del repositorio n veces al día. El motivo de esta automatización es disminuir costos de integración, permitiendo que los desarrolladores puedan realizar las integraciones de manera rápida y con mayor frecuencia

[5].

- **Requisitos**

1. El grupo de trabajo debe desarrollar pruebas automatizadas con la finalidad de encontrar inconsistencia en nuevos proyectos.
2. Será necesario contar con un servidor de integración continuo que ayude a la ejecución y monitoreo de las pruebas de forma automática para cada confirmación que se llegue a realizar.
3. Los programadores deben fusionar los cambios realizados de manera continua y rápida al menos una vez en todo el día.

- **Ventajas**

1. La solución es mucho más rápida frente algún problema que surja en la integración.
2. Los errores son mínimos y se evita que estos se envíen a producción.
3. Las regresiones son captadas muy rápido a través de las pruebas automáticas.
4. Las pruebas no equivalen un gran costo porque se pueden realizar en cuestión de segundos.
5. El control de calidad es más óptimo.

**1.5.2.2. Entrega continua**

La entrega continua corresponde a una extensión de la fase anterior, en donde la entrega del software se automatiza con la finalidad de que las implementaciones sean confiables en producción y fáciles de manejar. Para ello, se necesita una base de código que se pueda desplegar de manera automática. Esta etapa es atractiva porque automatiza todas las actividades desde la integración hasta la liberación de cambios ya aprobados y correctamente adecuados [5].

- **Requisitos**

1. El entorno de pruebas debe establecer una gran cantidad del código fuente, es por eso que debe tener una base sólida de integración continua.
2. Las implementaciones que se necesiten deben ser automatizadas.

- **Ventajas**

1. Existe una menor complejidad en la fase de desarrollo y una mayor velocidad.
2. La realimentación es inmediata porque los lanzamientos son mas seguidos.

### 1.5.2.3. Despliegue continuo

Es un proceso más avanzado que el anterior porque no se requiere de una revisión humana, sino que se encuentran totalmente automatizados. La implementación es una forma de apresurar la realimentación, permitiendo que el trabajo en equipo sea óptimo. Debido a este tipo de despliegue, los programadores pueden visualizar su trabajo realizado al poco tiempo de la realización, pero requiere el mayor esfuerzo en lo que corresponde a procesos automatizados y buenas prácticas [5].

- **Requisitos**

1. La calidad de cada uno de los lanzamientos será óptima mientras mejor sea la calidad de pruebas.
2. En lo que respecta la documentación, esta debe ser actualizada y comunicada con otros departamentos como soporte o marketing.

- **Ventajas**

1. Cada cambio se activa de manera automática.
2. Los riesgos son mínimos por cada nueva versión, debido a que se identifican los problemas en cada cambio y se pueden encontrar soluciones de manera rápida.
3. El flujo es de alta calidad y continuo. Este es el beneficio mas trascendente para los clientes.

### **1.5.3. Valor del Negocio**

Philip Kotler conceptúa el valor como la diferencia que existe entre los beneficios existentes que espera el satisfactor y el costo total, por lo que se puede decir que el valor es el resultado de la diferencia entre los beneficios que posee un producto y el costo que tiene. [30].

El impacto del TI se basa en tres aspectos importantes:

#### **1.5.3.1. Calidad reducida**

*Defectos no detectados*, cumplen un proceso de desarrollo a operaciones y luego producción sin la necesidad de ser identificados.

*Mala disponibilidad del servicio*, los complejos sistemas hacen que los cambios sean difíciles de implementar.

*Incapacidad de integrar datos*: los sistemas de Tecnología de información que tienen su estructura propia no presentan una buena operabilidad con otros softwares de negocio.

#### **1.5.3.2. Velocidad reducida**

*Baja velocidad para entrega de productos*: los procesos y sistemas complejos generan defectos y hace que se entreguen los trabajos de manera más lenta.

*Falta de prioridades*: son débiles, incluso inexistentes los procesos para establecer acuerdos necesarios.

*Creciente y constante demanda de la limitación de recursos*, hace que los trabajos se sigan postergando y no sean atendidos.

#### **1.5.3.3. Incremento en costos**

*Aumento de costos de mantenimiento*, el no priorizar la eliminación de procesos y herramientas produce costos elevados de mantenimiento.

*Estancamiento del crecimiento*, cuando se eleva el importe económico para que la operación siga funcionando, disminuye la capacidad de financiamiento en crecimiento e innovación.

*Incapacidad de mapear los costos al valor*, la no visualización de los verdaderos costos

limita la inversión.

#### **1.5.4. DevOps y las mejores prácticas**

DevOps tiene como visión general el integrar las áreas de Desarrollo y Operaciones de TI, de esta manera logra que fluya una mejor comunicación entre los equipos de ambas áreas, sin embargo, no reemplaza los demás marcos de trabajo. Con el pasar del tiempo las empresas han mejorado sus prácticas para gestionar operaciones teniendo éxito, la utilización de DevOPS no reemplaza lo que funcionaba con anterioridad, sino que ahora puede complementarse con otros marcos como el Agile, ITIL, Lean, entre otras, haciendo que se potencie la capacidad de las empresas, especialmente de los bancos [31].

Tomando como base las prácticas de la industria (Lean, ITIL, Agile) el DevOps Institute genera 15 prácticas esenciales:

- Alinearse con el valor que tiene el negocio definiendo a su vez la voz que tiene el cliente.
- Se da el aumento de la colaboración y destruye los silos con la gestión de relaciones.
- Se identifican las restricciones y los cuellos de botella que existen con el mapeo de las cadenas de valor.
- El desperdicio es eliminado de donde sea posible haciendo que se reduzca las entregas con la optimización Lean de Procesos.
- Se comparte el conocimiento y se crea la cultura de aprendizaje con la Gestión del conocimiento.
- Crea transparencia en la cadena de valor y realiza el trabajo visible con la Gestión Visual.
- Se trabaja en lotes más pequeños y a su vez se crean ciclos de retroalimentación con Scrum Agile.
- Se detectan problemas conforme vayan ocurriendo y falla más rápido al darle movimiento a las pruebas a la izquierda.
- Logra el equilibrio de manera correcta estableciendo flexibilidad y estabilidad con el control de cambios.

- Se incorpora información de donde se necesite, especialmente de la Gestión de la configuración del servicio.
- Apoya a la automatización y crea flujo con la Gestión de despliegues y liberaciones.
- Ayuda a resolver los incidentes con la Gestión de Incidentes.
- Convierte y transforma el conocimiento para ser mejor y que se acerque a la calidad de la fuente con la Gestión de Problemas y Kaizen Lean.
- Institucionaliza la mejora diaria con modelos de la Mejora continua.
- El fracaso se acepta como una oportunidad de mejora con un enfoque en la antifragilidad.

## II. MÉTODO

### 2.1. Tipo y Diseño de Investigación.

El tipo de investigación del presente informe es tecnológica aplicada porque permite una búsqueda de posibles soluciones del entorno tecnológico que permitan dar respuestas a una situación en concreto. Esta se basa principalmente en los descubrimientos que se encuentren a nivel tecnológico del trabajo de investigación, permitiendo alinear el producto y la teoría [18]. Además, permite aplicar un marco de buenas prácticas basado en DevOps para la mejor integración y despliegue en un entorno de desarrollo. Por otra parte, es cuantitativa, debido a que es necesario la recopilación y el análisis pertinente de la información estadística sobre las variables de estudio.

El diseño de investigación será cuasiexperimental. Esta fue propuesta por Campbell y Stanley [32], por primera vez. Se está empleando este tipo de diseño porque se utilizarán más de una variable de estudio y se podrá estudiar el impacto que tiene en el caso de estudio, es decir, como la variable independiente actúa frente a la dependiente.

### 2.2. Variables, Operacionalización.

#### 2.2.1. Variables:

##### a. Variable independiente

Aplicación de buenas prácticas en entorno de desarrollo basado en DevOps.

##### b. Variable dependiente

Integración y despliegue de proyectos.

Tabla 1. Operacionalización de variables

Variables de estudio	Dimensiones	Indicadores	Ítems	Técnica e instrumentos de recolección de datos
<b>Variable dependiente</b> Integración y despliegue de proyectos	Velocidad	Tiempo desde Desarrollo - Despliegue	$V = \sum DD + DT + DP$ <p>V = Tiempo promedio en días                      DD = Días de desarrollo                      DT = Días de testing                      DP = Días de despliegue a producción</p>	<p><b>Técnica:</b> Observación</p> <p><b>Instrumento:</b> Ficha de observación</p>
		Frecuencia de liberación del código	$F = \sum CD/n$ <p>F = Frecuencia de liberación de código                      CD = Cantidad de despliegues por Sprint                      n = Cantidad de Sprint</p>	<p><b>Técnica:</b> Observación</p> <p><b>Instrumento:</b> Ficha de observación</p>

<b>Variable independiente</b>	Calidad	Ratio de éxito	$RE = \sum \frac{DE}{TD} * 100$	<b>Técnica:</b>
		Aplicación de buenas prácticas en entorno de desarrollo basado en DevOps	RE = Ratio de éxito TD = DE + DF DE = Despliegues exitosos por Sprint DF = Despliegues fallidos por Sprint	Observación <b>Instrumento:</b> Ficha de Observación
	Calidad	Cantidad de clientes satisfechos con la calidad de sus productos	$NC = \sum T/D * 100$	<b>Técnica:</b>
			NC = Nivel de Clientes satisfechos T= Total de Clientes satisfechos D = Cantidad de clientes encuestados	Encuestas <b>Instrumento:</b> Modelo de encuesta De satisfacción de clientes

*Nota:* Los indicadores se han considerado en base a las necesidades de la organización. Fuente: Elaboración propia.

### **2.3. Población de estudio y muestra.**

*Población de estudio*, en la presente investigación la población está compuesta por 15 proyectos, desarrollados por distintos grupos de trabajo y que serán evaluados por la empresa para seleccionar una muestra.

*Muestra*, ha sido determinada por el método de muestreo no probabilista por conveniencia ya que por temas de privacidad la empresa delimitó la investigación a 3 proyectos.

### **2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad.**

#### **2.4.1. Técnicas de recolección de datos, validez y confiabilidad**

*Observación*: Es una técnica indispensable en el proceso de investigación, en la cual se obtiene la mayor cantidad de información relevante. Gran parte de los conocimientos que constituye la ciencia ha sido conseguida mediante la observación [33] .

*Encuestas*: Es la sucesión de diferentes preguntas que pueden ser abiertas o cerradas y que se cumple con lograr el objetivo [22]. En este caso se emplearán encuestas para medir el grado de satisfacción tanto de los colaboradores como los clientes del negocio. Por otra parte, se evaluará la validez que mide el proceso de integración y despliegue de software.

*Validez*: Para el presente proyecto de investigación la validez se ha realizado a través de un juicio de experto, con la finalidad de obtener un punto de vista acertado respecto a la guía de buenas prácticas.

#### **2.4.2. Instrumentos de recolección de datos, validez y confiabilidad**

*Ficha de observación*: Este instrumento es muy comúnmente empleado para conocer la realidad de los procesos, las dificultades y las oportunidades de mejora. Resultados (Pretest - Postest).

*Modelo de encuesta de satisfacción (cliente):* Este modelo de encuesta permitirá conocer el grado de conformidad respecto del producto o servicio brindado. Anexo N°03 Encuesta de satisfacción del cliente.

*Modelo de encuesta de satisfacción (colaborador):* este modelo de encuesta ayudará a conocer el grado de facilidad y factibilidad de uso mediante la implementación de una guía de prácticas orientadas a DevOps. Anexo N°03 Encuesta de satisfacción del colaborador.

*Instrumento de Validación No Experimental por Juicio de expertos:* este instrumento de validación permitirá medir la validez de la guía a través del método de juicio de experto (Anexo N° 04 Instrumento de Validación No Experimental por Juicio de expertos) El resumen de la validación de la guía de buenas prácticas se muestra en la siguiente tabla y lo que evalúa es que si está de acuerdo (sí) o en desacuerdo (no) con la pregunta indicada:

Tabla 2 Validación del instrumento mediante juicio de expertos

		¿Está de	¿Está de	¿Cree que
	¿La Cultura	acuerdo con	acuerdo	las
	DevOps	el marco de	con cada	empresas
	expuesta en la	aplicabilidad	una de las	consultoras
	guía es lo	indicado en	fases	de software
	suficientemente	la guía para	indicadas	tomen
N°	Experto	los	en los	como guía
	entendible al	procesos de	procesos de	el
	usuario para su	desarrollo	desarrollo	documento
	fácil	(Dev) y los	(Dev) y los	elaborado y
	comprensión y	procesos de	procesos de	lo apliquen
	aplicabilidad?	operaciones	operaciones	en su día a
		(Ops)?	(Ops)?	día?

<i>Ing. Ricardo</i>					
<b>1</b>	<i>Huachos</i>	Sí	Sí	Sí	Sí
<i>Huamanchahua</i>					
<i>Bach. Alan</i>					
<b>2</b>	<i>Antonio</i>	Sí	Sí	Sí	Sí
<i>Hernández</i>					
<i>López</i>					
<i>Ing. Marco</i>					
<b>3</b>	<i>Antonio Risco</i>	Sí	Sí	Sí	Sí
<i>Mejía</i>					
<b>Total</b>		Sí	Sí	Sí	Sí

Fuente: Elaboración propia

La validación se realizó presentando la matriz a los jueces expertos, dando por conclusión que todos están de acuerdo con que la cultura DevOps expuesta en la guía es lo suficientemente entendible el usuario para su fácil comprensión y aplicabilidad, todos están de acuerdos con el marco de aplicabilidad indicado en la guía para los procesos de desarrollo (Dev) y los procesos de operaciones (Ops), todos están de acuerdo con cada una de las fases indicadas en los procesos de desarrollo (Dev) y los procesos de operaciones (Ops) y todos están de acuerdo con que las empresas consultoras de software tomen como guía el documento elaborado y lo apliquen en su día a día, además evidencian a través de los comentarios que la guía está implementada en la empresa de estudio.

#### **2.4.3. Costo de implementación:**

Para este caso de estudio no se incluyó ningún costo adicional ya que la empresa consultora en mención cuenta con las licencias de los programas que se utilizarán para la implementación de las buenas prácticas seleccionadas. Sin embargo, en el Anexo N° 05 se detalla un aproximado de los valores monetarios de los recursos que han sido utilizados durante el proyecto para que sea usado como referencia en próximas implementaciones.

## 2.5. Procedimiento de análisis de datos.

### 2.5.1. Velocidad:

Para medir la velocidad por cada indicador de método planteado, se realiza la recopilación de datos mediante una ficha de observación.

*Indicador N°01 - Tiempo desde Desarrollo – Despliegue*, es el tiempo promedio que se calcula desde la etapa de desarrollo – despliegues definidos a través de las historias de usuario y los Sprints. Su fórmula es la siguiente:

$$V = \sum DD + DT + DP$$

Donde:

V = Tiempo promedio en días

DD = Días de desarrollo

DT = Días de testing

DP = Días de despliegue a producción

*Indicador N°02 – Frecuencia de liberación de código*, es el promedio de la cantidad de despliegues correspondientes a cada Sprint. Su fórmula es la siguiente:

$$F = \sum CD/n$$

Donde:

F = Frecuencia de liberación de código

CD = Cantidad de despliegues por Sprint

n = Cantidad de Sprints

### 2.5.2. Calidad:

*Indicador N°03 – Ratio de éxito*, es el porcentaje promedio de éxito sobre el proyecto que se calcula en base a los despliegues exitosos y fallidos por sprint. Su fórmula es la siguiente:

$$VRE = \sum \frac{DE}{TD} * 100$$

Donde:

RE = Ratio de éxito

TD = DE + DF

DE = Despliegues exitosos por Sprint

DF = Despliegues fallidos por Sprint

Para medir la calidad por cada indicador de método planteado, se realiza la recolección de datos mediante los modelos de satisfacción.

*Indicador N°04 – Nivel de usuarios satisfechos con la calidad de los proyectos de desarrollo*, este indicador evalúa la satisfacción del usuario, su fórmula es la siguiente:

$$NU = \sum T/D * 100$$

Donde:

NU: Nivel de clientes satisfechos

T: Total de clientes satisfechos

D: Cantidad de clientes atendidos

## **2.6. Criterios éticos.**

Las decisiones en torno a esta investigación se han tomado por cada etapa del desarrollo y en base a los criterios establecidos por el caso de estudio garantizando la seguridad, bienestar de los colaboradores involucrados y la calidad del trabajo.

*Confidencialidad de los participantes en la investigación*, no se brindará los datos personales de los colaboradores y la alta dirección de la empresa en estudio. Solo se han considerado información concisa como ubicación geográfica, tamaño, giro de la empresa.

*Confidencialidad de la información brindada*, todos los datos proporcionados por la organización no podrán ser modificados, divulgados ni proporcionada sin una autorización

correspondiente. Se respetará lo dispuesto por la organización.

*Plagio*, esta acción no será utilizada en esta investigación, porque atenta contra los derechos de autor y de propiedad intelectual y, por ende, está prohibido. El texto ha sido compuesto por el estudiante a cargo del trabajo de investigación.

### **2.6.1. Criterios de Rigor Científico.**

*Confirmabilidad*, describe cómo se puede conocer el escenario o ruta de lo que pudo hacer cualquier otro autor [34]. Este concepto enmarca que los datos o registros levantados deben estar documentados con la finalidad de examinar mucha información y estudiar conclusiones que se asemejen a lo realizado.

*Transferibilidad*, permite ampliar los resultados de la investigación a otro tipo de poblaciones [34]. Cabe la posibilidad que lo expuesto pueda adaptarse a otros contextos, eso dependerá de las necesidades de las empresas sí coinciden con las especificadas en este proyecto.

### III. RESULTADOS Y DISCUSIÓN.

#### 3.1. Resultados

El análisis de los resultados alcanzados se fundamenta en la ejecución de la guía de buenas prácticas basadas en DevOps principalmente en las etapas de integración y despliegue. Para eso se tomaron Sprint por proyecto, divididos en Historias de usuario. A continuación, se muestran los resultados conseguidos por los tres indicadores antes de la aplicación de buenas prácticas basadas en DevOps.

Tabla 3. *Indicador Velocidad (Desarrollo - despliegue)*

Ficha de observación – Pretest					
Indicador	Descripción	Técnica	Unidad de Medida	Instrumento	Fórmula
Velocidad Desarrollo – Despliegue	Se evalúa el tiempo de desarrollo de las historias desde desarrollo hasta su entrega a producción	Observación	Días	Ficha de observación	$V = \frac{DD+DT+DP}{\text{Días de desarrollo} + \text{Días de testing} + \text{Días de despliegue a producción}}$

Tabla 4. *Proyecto 1 – Entidad financiera estadounidense*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint catorce	16	5	1	22
Sprint catorce	15	5	1	22
Sprint quince	8	5	1	14
Sprint dieciséis	19	2	1	22
Sprint dieciséis	19	2	1	22
Sprint dieciséis	14	5	0	19
Sprint diecisiete	2	2	0	4
Sprint diecisiete	10	7	1	18
Sprint dieciocho	4	3	5	12
Sprint dieciocho	8	0	0	8
Sprint diecinueve	5	2	1	8
Sprint diecinueve	8	2	0	10
Sprint veinte	1	1	1	3
Sprint veintiuno	9	20	0	29
Sprint veintiuno	6	20	0	26
Sprint veintiuno	8	18	3	29
Sprint veintiuno	8	18	3	29
Sprint veintiuno	9	20	0	29
<b>Total</b>	<b>170</b>	<b>137</b>	<b>19</b>	<b>18.11</b>

Tabla 5. *Proyecto 2 – Entidad financiera peruana*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint nueve	23	6	1	30
Sprint nueve	18	6	1	25
Sprint nueve	11	6	1	18

Sprint diez	17	3	2	22
Sprint diez	19	3	2	24
Sprint once	11	4	1	16
Sprint doce	5	2	0	7
Sprint doce	8	4	1	13
Sprint trece	4	3	5	12
Sprint trece	8	0	0	8
Sprint trece	5	2	1	8
Sprint catorce	8	2	0	10
Sprint quince	1	1	0	2
Sprint quince	9	14	2	25
<b>Total</b>	<b>147</b>	<b>56</b>	<b>24</b>	<b>15.71</b>

Tabla 6. *Proyecto 3 – Empresa de mejoramiento del hogar*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint dos	10	7	1	18
Sprint dos	7	5	1	13
Sprint tres	4	2	1	7
Sprint tres	17	7	1	25
Sprint cuatro	18	2	1	21
Sprint cinco	16	5	0	21
Sprint cinco	14	2	0	16
Sprint seis	20	7	1	28
Sprint seis	19	3	5	27

Sprint seis	19	0	0	19
Sprint siete	7	2	1	10
Sprint siete	5	2	0	7
<b>Total</b>	<b>156</b>	<b>44</b>	<b>12</b>	<b>17.7</b>

Tabla 7. *Indicador Velocidad (Frecuencia de liberación de código)*

<b>Ficha de observación - Pretest</b>					
<b>Indicador</b>	<b>Descripción</b>	<b>Técnica</b>	<b>Unidad de Medida</b>	<b>Instrumento</b>	<b>Fórmula</b>
Velocidad (Frecuencia de liberación de código)	Se evalúa las veces que se libera historias de usuario a producción por sprint	Observación	Número de despliegues	Ficha de observación	CD= Cantidad de despliegues por sprint

Tabla 8. *Proyecto 1 – Entidad financiera estadounidense*

<b>Ítem</b>	<b>SPRINT</b>	<b>CD</b>
1	Sprint catorce	3
2	Sprint quince	2
3	Sprint dieciséis	4
4	Sprint diecisiete	5
5	Sprint dieciocho	3
6	Sprint diecinueve	3
7	Sprint veinte	2

8	Sprint veintiuno	3
<b>TOTAL</b>		<b>3.13</b>

Tabla 9. *Proyecto 2 – Entidad financiera peruana*

Ítem	SPRINT	CD
1	Sprint nueve	2
2	Sprint diez	3
3	Sprint once	5
4	Sprint doce	4
5	Sprint trece	4
6	Sprint catorce	2
7	Sprint quince	3
<b>TOTAL</b>		<b>3.29</b>

Tabla 10. *Proyecto 3 – Empresa de mejoramiento del hogar*

Ítem	SPRINT	CD
1	Sprint dos	4
2	Sprint tres	5
3	Sprint cuatro	2
4	Sprint cinco	2
5	Sprint seis	5
6	Sprint siete	3
<b>TOTAL</b>		<b>3.5</b>

Tabla 11. *Indicador Calidad (Ratio de éxito)*

<b>Ficha de observación - Pretest</b>
---------------------------------------

Indicador	Descripción	Técnica	Unidad de Medida	Instrumento	Fórmula
Calidad (Ratio de éxito)	Se evalúa el ratio código una vez desplegado de manera exitosa, no requiere retroalimentación por Sprint	Observación	%	Ficha de observación	$RE = \frac{DE}{TD} * 100\%$ $TD = DE + DF$ DE = Despliegues correctos por Sprint DF = Despliegues incorrectos por Sprint TD = Total de Sprint RE = Ratio de éxito

Tabla 12. Proyecto 1 – Entidad financiera estadounidense

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint catorce	1	1	2	50%
2	Sprint quince	1	0	1	100%
3	Sprint dieciséis	1	0	1	100%
4	Sprint diecisiete	1	1	2	50%
5	Sprint dieciocho	1	1	2	50%
6	Sprint diecinueve	1	4	5	20%

7	Sprint veinte	0	1	1	0%
8	Sprint veintiuno	2	1	3	67%
<b>TOTAL</b>		<b>8</b>	<b>9</b>	<b>17</b>	<b>54.63%</b>

Tabla 13. *Proyecto 2 – Entidad financiera peruana*

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint nueve	1	1	2	50%
2	Sprint diez	0	2	2	0%
3	Sprint once	1	0	1	100%
4	Sprint doce	1	2	3	33%
5	Sprint trece	1	1	2	50%
6	Sprint catorce	1	2	3	33%
7	Sprint quince	0	1	1	0%
<b>TOTAL</b>		<b>5</b>	<b>9</b>	<b>14</b>	<b>38%</b>

Tabla 14. *Proyecto 3 – Empresa de mejoramiento del hogar*

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint dos	1	2	3	33%
2	Sprint tres	1	1	2	50%
3	Sprint cuatro	1	0	1	100%
4	Sprint cinco	1	0	1	100%
5	Sprint seis	1	2	3	33%
6	Sprint siete	1	4	5	20%
<b>TOTAL</b>		<b>8</b>	<b>9</b>	<b>17</b>	<b>56%</b>

Después de la aplicación de buenas prácticas DevOps, se tomaron nuevamente Sprint e historias de usuario y los resultados fueron los siguientes:

Tabla 15. *Indicador Velocidad (Desarrollo - despliegue)*

<b>Ficha de observación - PostTest</b>					
<b>Indicador</b>	<b>Descripción</b>	<b>Técnica</b>	<b>Unidad de Medida</b>	<b>Instrumento</b>	<b>Fórmula</b>
Velocidad Desarrollo – Despliegue	Se evalúa el tiempo de desarrollo de las historias desde desarrollo hasta su entrega a producción	Observación	Días	Ficha de observación	$V = \frac{1}{DD+DT+DP}$ DD= Días de desarrollo DT= Días de testing DP= Días de despliegue a producción

Tabla 16. *Proyecto 1 – Entidad financiera estadounidense*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint veintidós	12	4	0	16
Sprint veintitrés	12	2	0	14
Sprint veintitrés	4	2	0	6
Sprint veintitrés	12	2	1	15
Sprint veinticuatro	11	3	1	15
Sprint veinticuatro	8	3	0	11
Sprint veinticinco	5	1	0	6
Sprint veinticinco	7	1	1	9
Sprint veintiséis	6	3	2	11

Sprint veintiséis	9	3	0	12
Sprint veintisiete	2	1	1	4
Sprint veintiocho	3	1	0	4
Sprint veintiocho	3	0	1	4
Sprint veintiocho	4	0	0	4
Sprint veintiocho	7	4	0	11
Sprint veintinueve	10	5	2	17
Sprint veintinueve	12	6	2	20
Sprint veintinueve	9	5	0	14
<b>Total</b>	<b>136</b>	<b>46</b>	<b>11</b>	<b>10.72</b>

Tabla 17. *Proyecto 2 – Entidad financiera peruana*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint dieciséis	19	3	1	23
Sprint dieciséis	18	4	1	23
Sprint diecisiete	10	5	1	16
Sprint diecisiete	7	5	2	14
Sprint diecisiete	7	5	2	14
Sprint dieciocho	9	3	1	13
Sprint dieciocho	8	3	0	11
Sprint diecinueve	10	5	1	16
Sprint diecinueve	5	2	5	12
Sprint veinte	3	2	0	5
Sprint veintiuno	2	1	1	4
Sprint veintidós	4	1	0	5
Sprint veintidós	4	1	0	5
Sprint veintidós	3	2	2	7

<b>Total</b>	<b>109</b>	<b>42</b>	<b>24</b>	<b>12</b>
--------------	------------	-----------	-----------	-----------

Tabla 18. *Proyecto 3 – Empresa de mejoramiento del hogar*

<b>SPRINT</b>	<b>DD</b>	<b>DT</b>	<b>DP</b>	<b>CT</b>
Sprint ocho	8	7	0	15
Sprint ocho	6	5	1	12
Sprint nueve	2	2	0	4
Sprint nueve	3	7	1	11
Sprint diez	12	2	1	15
Sprint once	14	5	0	19
Sprint once	14	2	0	16
Sprint once	12	7	1	20
Sprint doce	11	3	3	18
Sprint trece	9	0	0	9
Sprint trece	7	2	2	11
Sprint trece	5	2	1	8
<b>Total</b>	<b>103</b>	<b>44</b>	<b>10</b>	<b>13.17</b>

Tabla 19. *Indicador Velocidad (Frecuencia de liberación de código)*

<b>Ficha de observación - PostTest</b>					
<b>Indicador</b>	<b>Descripción</b>	<b>Técnica</b>	<b>Unidad de Medida</b>	<b>Instrumento</b>	<b>Fórmula</b>
Velocidad (Frecuencia de liberación de historias)	Se evalúa las veces que se libera historias	Observación	Número de	Ficha de observación	CD= Cantidad de despliegues por sprint

liberación de código)	de usuario a producción por sprint	despligu es
-----------------------	------------------------------------	-------------

Tabla 20. *Proyecto 1 – Entidad financiera estadounidense*

Ítem	SPRINT	CD
1	Sprint veintidós	1
2	Sprint veintitrés	1
3	Sprint veinticuatro	1
4	Sprint veinticinco	2
5	Sprint veintiséis	2
6	Sprint veintisiete	2
7	Sprint veintiocho	3
8	Sprint veintinueve	1
<b>TOTAL</b>		<b>1.63</b>

Tabla 21. *Proyecto 2 – Entidad financiera peruana*

Ítem	SPRINT	CD
1	Sprint dieciséis	1
2	Sprint diecisiete	2
3	Sprint dieciocho	2
4	Sprint diecinueve	2
5	Sprint veinte	3
6	Sprint veintiuno	1
7	Sprint veintidós	1

<b>TOTAL</b>	<b>1.71</b>
--------------	-------------

Tabla 22. Proyecto 3 – Empresa de mejoramiento del hogar

<b>Ítem</b>	<b>SPRINT</b>	<b>CD</b>
1	Sprint ocho	2
2	Sprint nueve	1
3	Sprint diez	1
4	Sprint once	1
5	Sprint doce	3
6	Sprint trece	1
<b>TOTAL</b>		<b>1.5</b>

Tabla 23. Indicador Calidad (Ratio de éxito)

<b>Ficha de observación - PosTest</b>					
<b>Indicador</b>	<b>Descripción</b>	<b>Técnica</b>	<b>Unidad de Medida</b>	<b>Instrumento</b>	<b>Fórmula</b>
Calidad (Ratio de éxito)	Se evalúa el ratio código de desplegado de manera exitosa, no requiere retroalimentación por Sprint	Observación	%	Ficha de observación	$RE = \frac{DE}{TD} * 100\%$ $TD = DE + DF$ $DE =$ Despliegues correctos por Sprint $DF =$ Despliegues

incorrectos  
 por Sprint  
 TD = Total de  
 Sprint  
 RE = Ratio  
 de éxito

Tabla 24. *Proyecto 1 – Entidad financiera estadounidense*

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint veintidós	2	0	2	100%
2	Sprint veintitrés	2	1	3	100%
3	Sprint veinticuatro	1	0	1	67%
4	Sprint veinticinco	2	0	2	100%
5	Sprint veintiséis	3	1	4	75%
6	Sprint veintisiete	4	1	5	80%
7	Sprint veintiocho	3	1	4	75%
8	Sprint veintinueve	2	1	3	67%
<b>TOTAL</b>		<b>8</b>	<b>9</b>	<b>17</b>	<b>83%</b>

Tabla 25. *Proyecto 2 – Entidad financiera peruana*

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint dieciséis	1	1	2	50%
2	Sprint diecisiete	1	1	2	50%
3	Sprint dieciocho	2	0	2	100%
4	Sprint diecinueve	3	1	4	75%
5	Sprint veinte	1	1	2	50%

6	Sprint veintiuno	5	2	7	71%
7	Sprint veintidós	2	0	2	100%
<b>TOTAL</b>		<b>15</b>	<b>6</b>	<b>21</b>	<b>70.86%</b>

Tabla 26. *Proyecto 3 – Empresa de mejoramiento del hogar*

Ítem	SPRINT	DE	DF	TD	RE
1	Sprint ocho	2	2	4	50%
2	Sprint nueve	1	0	1	100%
3	Sprint diez	4	2	6	67%
4	Sprint once	3	0	3	100%
5	Sprint doce	1	2	3	33%
6	Sprint trece	2	1	3	67%
<b>TOTAL</b>		<b>8</b>	<b>9</b>	<b>17</b>	<b>69.5%</b>

En base a los datos obtenidos, se contrastaron los resultados del pretest y postest. Los resultados obtenidos fueron los siguientes:

Tabla 27. *Contraste de resultados del pretest y postest*

Proyecto	Dimensión	Indicador	Test		Diferencia
			PreTest	PosTest	
Proyecto 1	Velocidad	Velocidad			
		(Desarrollo -	18.11	10.72	7.39
		Despliegue)			
		Velocidad			
		(Frecuencia de	3.13	1.63	1.5
		liberación de código)			

Proyecto 2	Calidad	Calidad (Ratio de éxito)	54.63%	83%	28.37%
	Velocidad	Velocidad (Desarrollo - Despliegue)	15.71	12	3.71
		Velocidad (Frecuencia de liberación de código)	3.29	1.71	1.58
		Calidad	Calidad (Ratio de éxito)	38%	70.86%
Proyecto 3	Velocidad	Velocidad (Desarrollo - Despliegue)	17.7	13.17	4.53
		Velocidad (Frecuencia de liberación de código)	3.5	1.5	2
		Calidad	Calidad (Ratio de éxito)	56%	69.5%

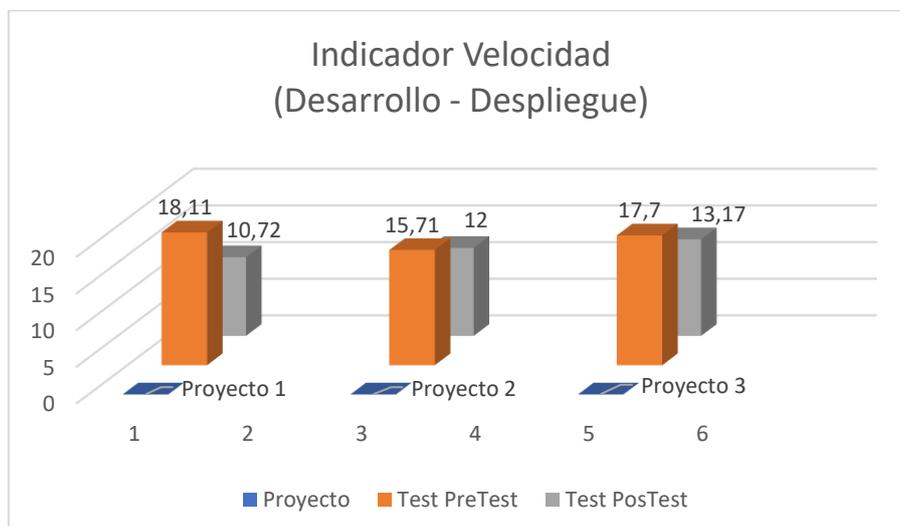


Figura 2. Indicador Velocidad (Desarrollo - Despliegue).

Fuente: Elaboración propia.

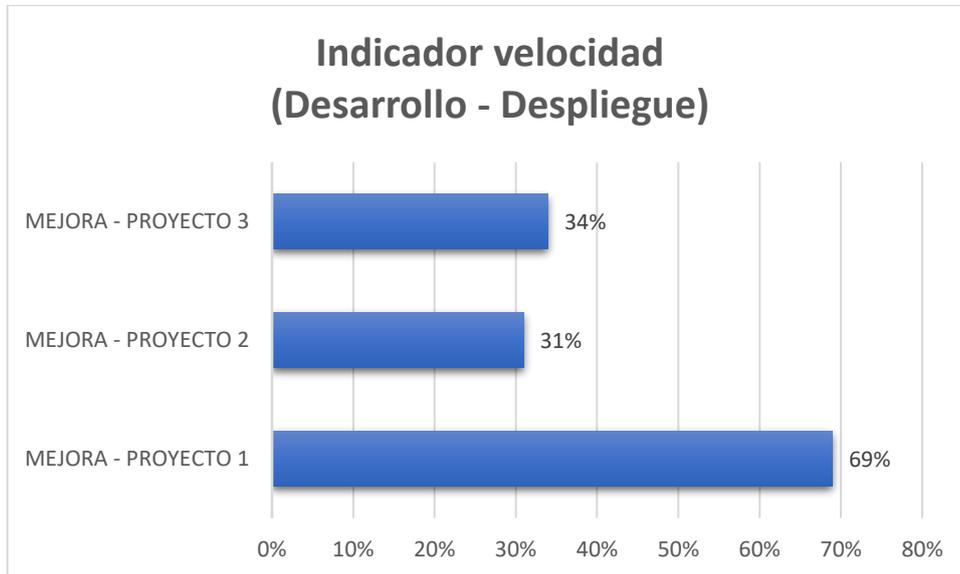


Figura 3. Mejoras en porcentajes del Indicador Velocidad (Desarrollo - Despliegue).

Fuente: Elaboración propia.

Para el proyecto 1, en el pretest dio un valor promedio de 18.11 días y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 10.72 días; infiriendo en base a lo obtenido que se redujo en 7.39 días, lo que equivale al 69% de mejora.

Para el proyecto 2, en el pretest dio un valor promedio de 15.71 días y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 12 días; infiriendo en base a lo obtenido que se redujo en 3.71 días, lo que equivale al 31% de mejora.

Para el proyecto 3, en el pretest dio un valor promedio de 17.7 días y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 13.17 días; infiriendo en base a lo obtenido que se redujo en 4.53 días, lo que equivale al 34% de mejora.

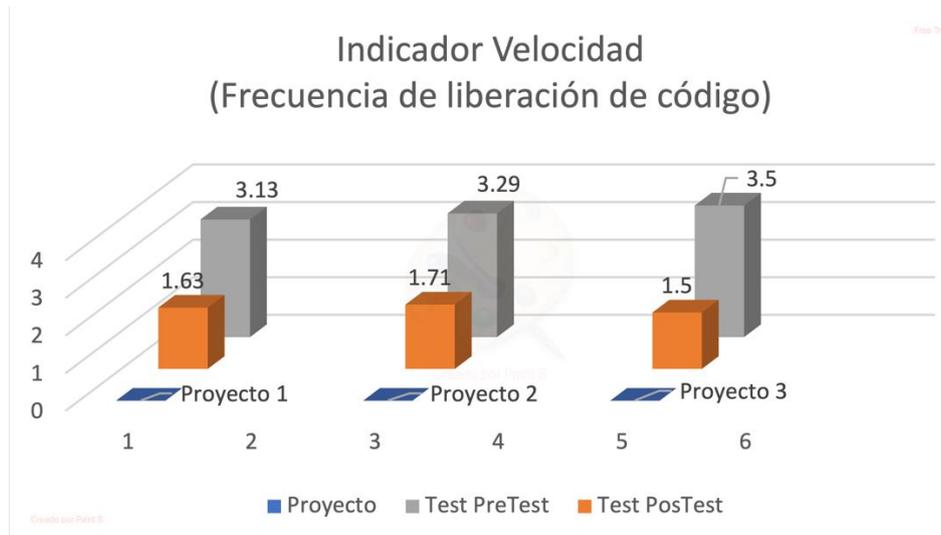


Figura 4. Indicador Velocidad (Frecuencia de liberación de código).

Fuente: Elaboración propia.

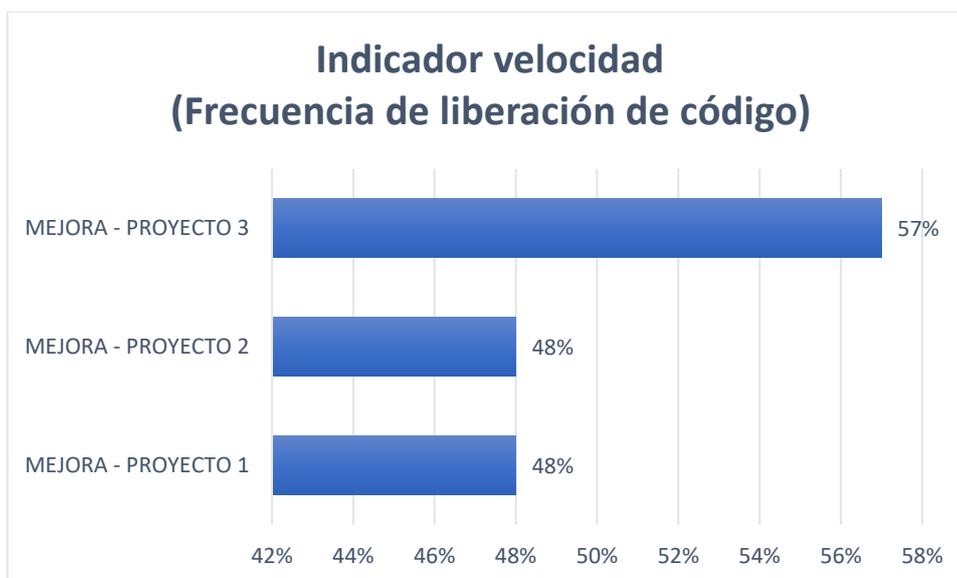


Figura 5. Mejoras en porcentajes del Indicador Velocidad (Frecuencia de liberación de código).

Fuente: Elaboración propia.

Para el proyecto 1, en el pretest dio un valor de 3.13 de despliegues por sprint y posterior a la implementación de buenas prácticas basadas en DevOps para el posttest el resultado promedio fue de 1.63 despliegues por sprint; infiriendo en base a lo obtenido que aumentó en 1.5 despliegues por sprint, lo que equivale al 48% de mejora.

Para el proyecto 2, en el pretest dio un valor de 3.29 de despliegues por sprint y posterior a la implementación de buenas prácticas basadas en DevOps para el posttest el resultado promedio fue de 1.71 despliegues por sprint; infiriendo en base a lo obtenido que aumentó en 1,58 de despliegues por sprint, lo que equivale al 48% de mejora.

Para el proyecto 3, en el pretest dio un valor de 3,5 de despliegues por sprint y posterior a la implementación de buenas prácticas basadas en DevOps para el posttest el resultado promedio fue de 1,5 de despliegues por sprint; infiriendo en base a lo obtenido que aumentó en 2 de despliegues por sprint, lo que equivale al 52% de mejora 57%.

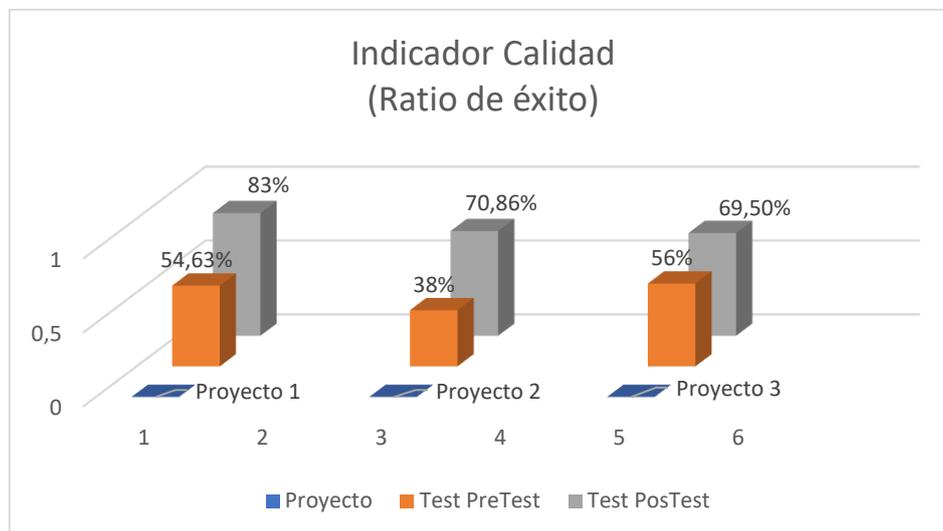


Figura 6. Indicador Calidad (Ratio de éxito).

Fuente: Elaboración propia.

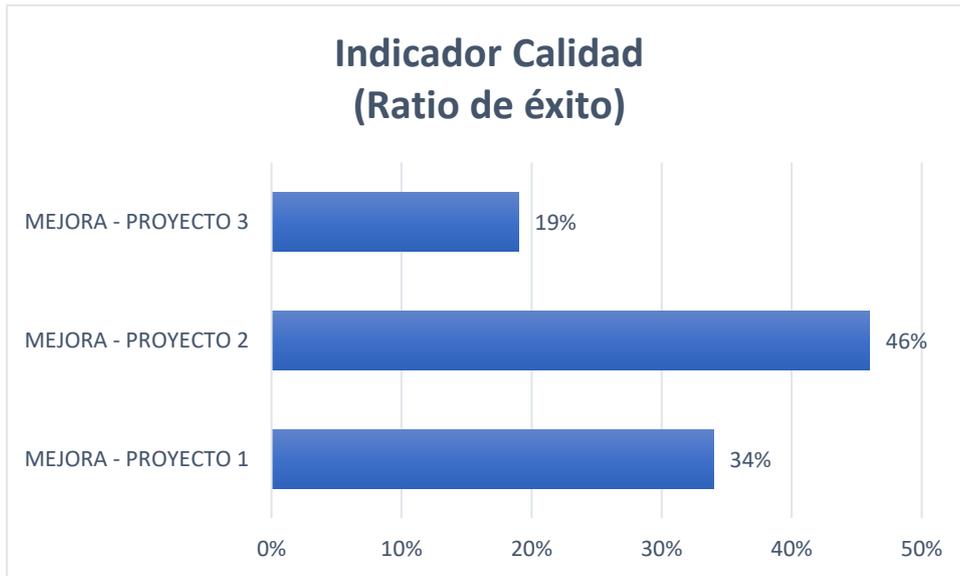


Figura 7. Mejoras en porcentajes del Indicador Calidad (Ratio de éxito).

Fuente: Elaboración propia.

Para el proyecto 1, en el pretest dio un valor de 54.63% de despliegues exitosos y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 83% de despliegues exitosos; infiriendo en base a lo obtenido que aumentó en 28.37% en despliegues exitosos, lo que equivale al 34% de mejora.

Para el proyecto 2, en el pretest dio un valor de 38% de despliegues exitosos y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 70.86% de despliegues exitosos; infiriendo en base a lo obtenido que aumentó en 32.86% de despliegues exitosos, lo que equivale al 46% de mejora.

Para el proyecto 3, en el pretest dio un valor de 56% de despliegues exitosos y posterior a la implementación de buenas prácticas basadas en DevOps para el postest el resultado promedio fue de 69.50% de despliegues exitosos; infiriendo en base a lo obtenido que aumentó en 13.5% de despliegues exitosos, lo que equivale al 19% de mejora.

Se realizaron dos encuestas para el cliente y el colaborador. Para cliente interno fue el modelo TAM y para el externo el modelo KANO. Los resultados fueron los siguientes:

Resultados de encuesta: TAM

**1. Una guía de buenas prácticas basadas en DevOps ayudaría a hacer sus tareas más rápido**

Tabla 28. Encuesta TAM – Pregunta N°01

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	1	5%
Más probable que improbable	2	10%
Muy probable	13	65%
Extremadamente probable	4	20%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>



Figura 8. Pregunta N°01 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 4 que pertenecen al 20% precisan que es extremadamente probable que una guía de buenas prácticas ayudaría a hacer sus

tareas más rápido, 13 que pertenecen al 65% afirman que es muy probable, 2 trabajadores que pertenecen al 10% respondieron que es más probable que improbable, 1 trabajador que pertenece al 5% opino de manera neutral, dejando a los demás indicadores con valor 0.

## 2. Usar una guía de buenas prácticas basadas en DevOps mejoraría el desempeño de su trabajo

Tabla 29. Encuesta TAM – Pregunta N°02

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	2	10%
Más probable que improbable	1	5%
Muy probable	15	75%
Extremadamente probable	2	20%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

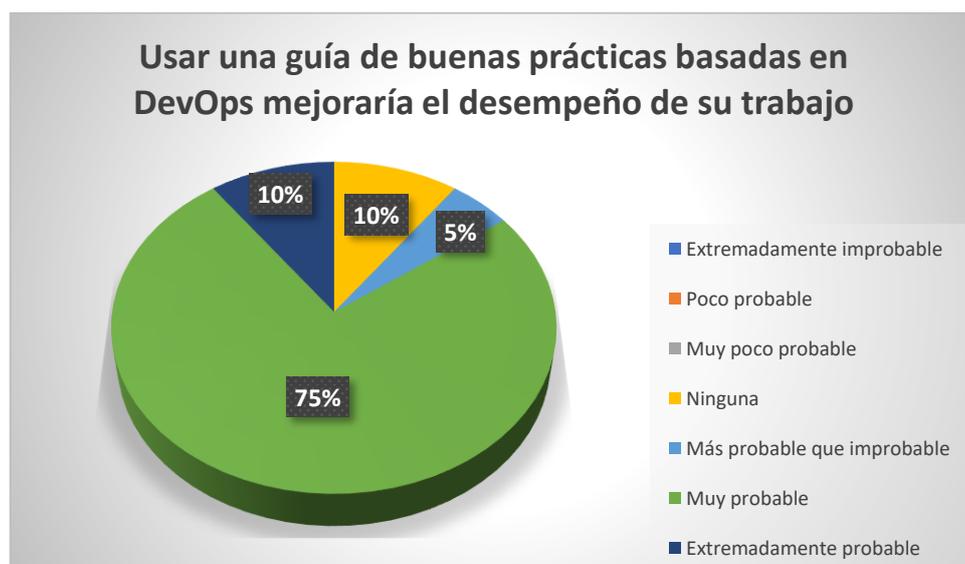


Figura 9. Pregunta N°02 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 2 que pertenecen al 10% precisan que es extremadamente probable que usar una guía de buenas prácticas basada en DevOps mejoraría el desempeño de su trabajo, 15 que pertenecen al 75% afirman que es muy probable, 1 que pertenecen al 5% respondieron que es más probable que improbable, 2 que pertenecen al 10% opinó de manera neutral, dejando a los demás indicadores con valor 0.

### 3. Usar una guía de buenas prácticas basadas en DevOps incrementaría su productividad

Tabla 30. Encuesta TAM – Pregunta N°03

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	0	0%
Más probable que improbable	2	10%
Muy probable	17	85%
Extremadamente probable	1	5%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>



Figura 10. Pregunta N°03 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 1 que corresponde al 5% precisan que es extremadamente probable que usar una guía de buenas prácticas basadas en DevOps incrementaría su productividad, 17 que pertenecen al 85% afirman que es muy probable, 2 que pertenecen al 10% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

#### 4. Usar una guía de buenas prácticas basadas en DevOps aumentaría la efectividad en su trabajo

Tabla 31. Encuesta TAM – Pregunta N°04

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	0	0%
Más probable que improbable	4	20%
Muy probable	11	55%
Extremadamente probable	5	25%

<b>TOTAL</b>	20	100%
--------------	----	------

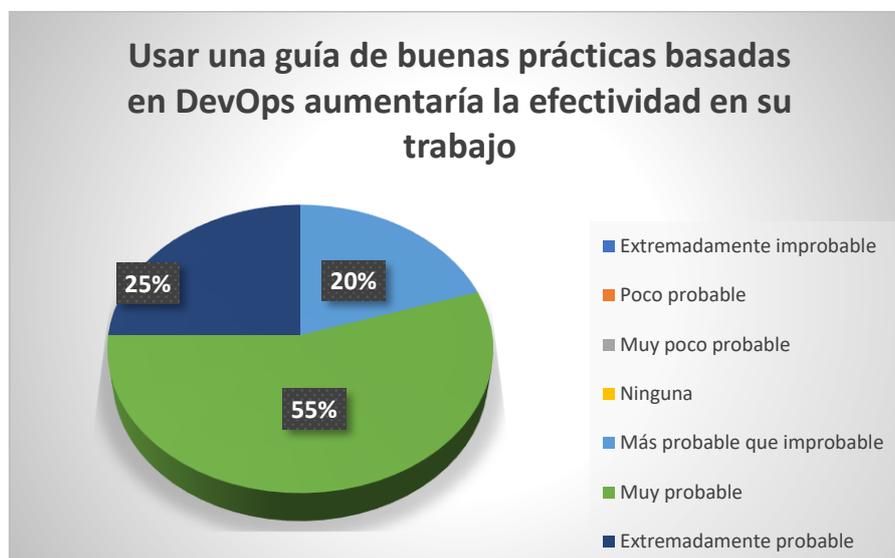


Figura 11. Pregunta N°04 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 5 que corresponde al 25% precisan que es extremadamente probable que usar una guía de buenas prácticas basadas en DevOps aumentaría la efectividad en su trabajo, 11 que pertenecen al 55% afirman que es muy probable, 4 que pertenecen al 20% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

#### 5. Usar una guía de buenas prácticas basadas en DevOps facilitaría la realización de su trabajo

Tabla 32. Encuesta TAM – Pregunta N°05

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	0	0%

Más probable que improbable	5	25%
Muy probable	9	45%
Extremadamente probable	6	30%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>



Figura 12. Pregunta N°05 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 6 que corresponde al 30% precisan que es extremadamente probable que usar una guía de buenas prácticas basadas en DevOps facilitaría la realización de su trabajo, 9 que pertenecen al 45% afirman que es muy probable, 5 que pertenecen al 25% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

#### 6. Encontraría una guía de buenas prácticas basadas en DevOps útil en su trabajo

Tabla 33. Encuesta TAM – Pregunta N°06

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%

Muy poco probable	0	0%
Ninguna	0	0%
Más probable que improbable	1	5%
Muy probable	18	90%
Extremadamente probable	1	5%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

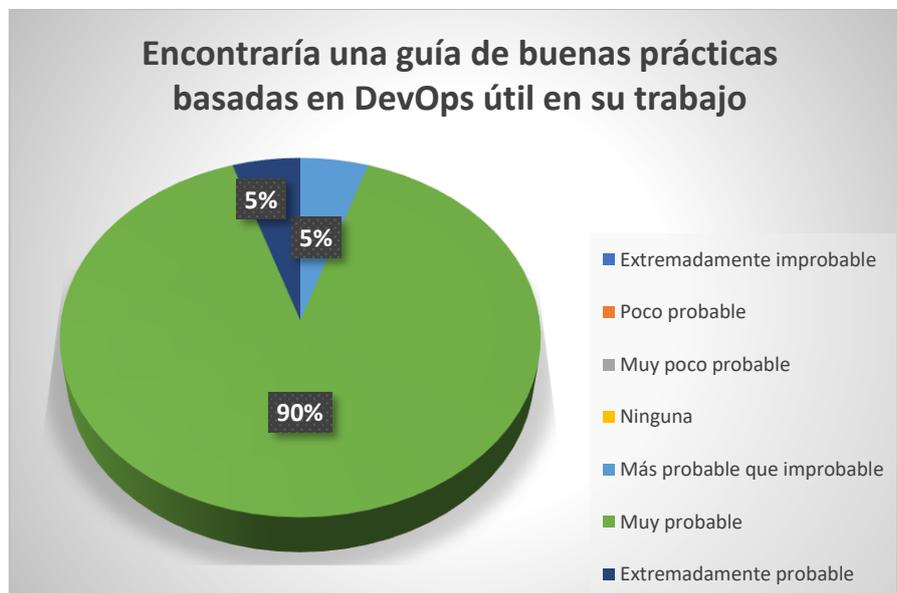


Figura 13. Pregunta N°06 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 1 que corresponde al 5% precisan que es extremadamente probable que encontraría una guía de buenas prácticas basadas en DevOps útil en su trabajo, 18 que pertenecen al 90% afirman que es muy probable, 1 que pertenecen al 5% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

**7. Aprender a utilizar una guía de buenas prácticas basadas en DevOps sería fácil para usted**

Tabla 34. Encuesta TAM – Pregunta N°07

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	0	0%
Más probable que improbable	8	40%
Muy probable	12	60%
Extremadamente probable	0	0%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

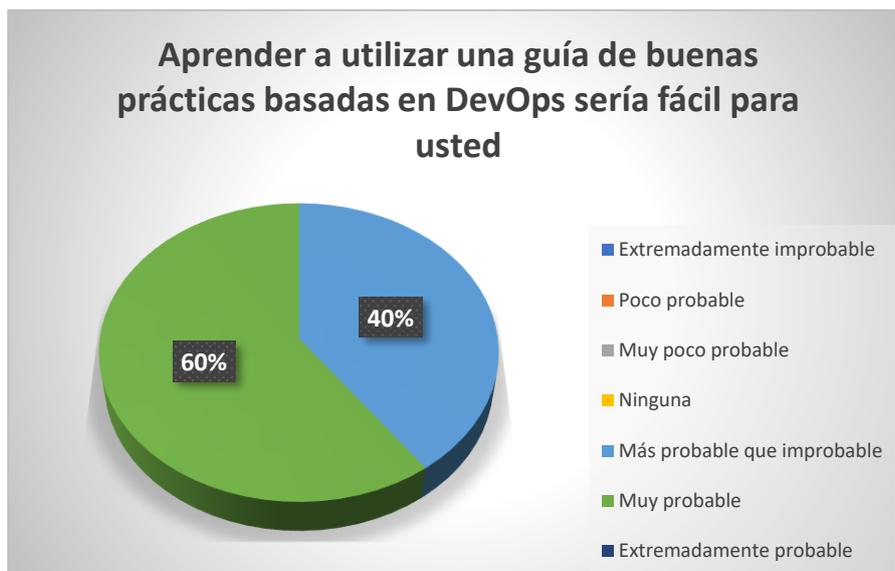


Figura 14. Pregunta N°07 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 12 que pertenecen al 60% afirman que es muy probable que aprender a utilizar una guía de buenas prácticas basadas en DevOps sería fácil para usted, 8 que pertenecen al 40% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

**8. Le resultaría fácil utilizar una guía de buenas prácticas basadas en DevOps para hacer lo que usted quiere que haga**

Tabla 35. Encuesta TAM – Pregunta N°8

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	0	0%
Más probable que improbable	7	35%
Muy probable	12	60%
Extremadamente probable	1	5%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>



Figura 15. Pregunta N°08 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 1 que corresponde al 5% precisan que es extremadamente probable que le resultaría fácil utilizar una guía de buenas prácticas basadas en DevOps para hacer lo que usted quiere que haga, 12 que pertenecen

al 60% afirman que es muy probable, 7 que pertenecen al 35% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

**9. Le resultaría fácil utilizar una guía de buenas prácticas basadas en DevOps para hacer lo que usted quiere que haga**

Tabla 36. Encuesta TAM – Pregunta N°9

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	2	10%
Más probable que improbable	6	30%
Muy probable	10	50%
Extremadamente probable	2	10%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>



Figura 16. Pregunta N°09 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 2 que pertenecen al 10% precisan que es extremadamente probable que su interacción con una guía de buenas prácticas basadas en DevOps sería claro y entendible, 10 que pertenecen al 50% afirman que es muy probable, 6 trabajadores que pertenecen al 30% respondieron que es más probable que improbable, 2 que pertenecen al 10% opinaron de manera neutral, dejando a los demás indicadores con valor 0.

**10. Encuentra una guía de buenas prácticas basadas en DevOps flexible para interactuar con él**

Tabla 37. Encuesta TAM – Pregunta N°10

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	2	10%
Más probable que improbable	6	30%
Muy probable	10	50%
Extremadamente probable	2	10%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

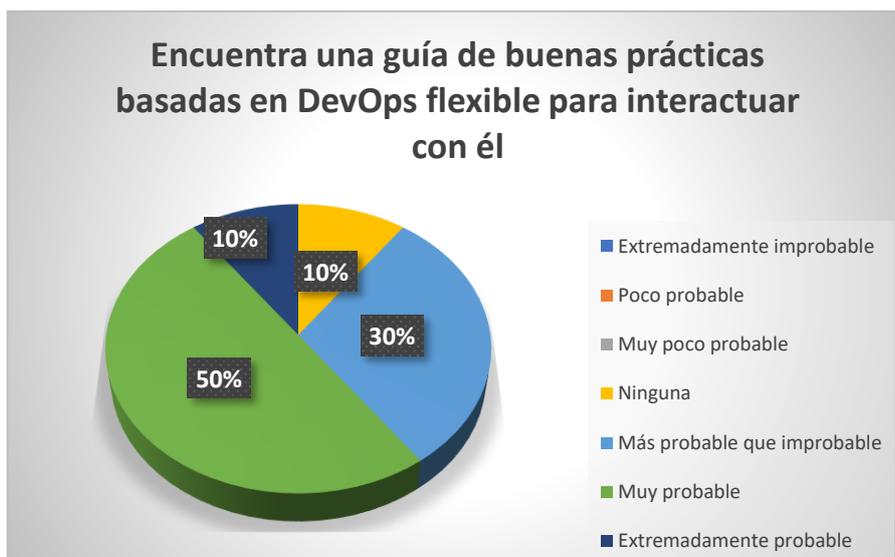


Figura 17. Pregunta N°10 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 2 que pertenecen al 10% precisan que es extremadamente probable que encuentra una guía de buenas prácticas basadas en DevOps flexible para interactuar con él, 10 que pertenecen al 50% afirman que es muy probable, 6 trabajadores que pertenecen al 30% respondieron que es más probable que improbable, 2 que pertenecen al 10% opinaron de manera neutral, dejando a los demás indicadores con valor 0.

#### 11. Sería fácil para usted llegar a ser un experto en el uso de la guía de buenas prácticas basadas en DevOps

Tabla 38. Encuesta TAM – Pregunta N°11

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%
Muy poco probable	0	0%
Ninguna	3	15%
Más probable que improbable	4	20%

Muy probable	13	65%
Extremadamente probable	0	0%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

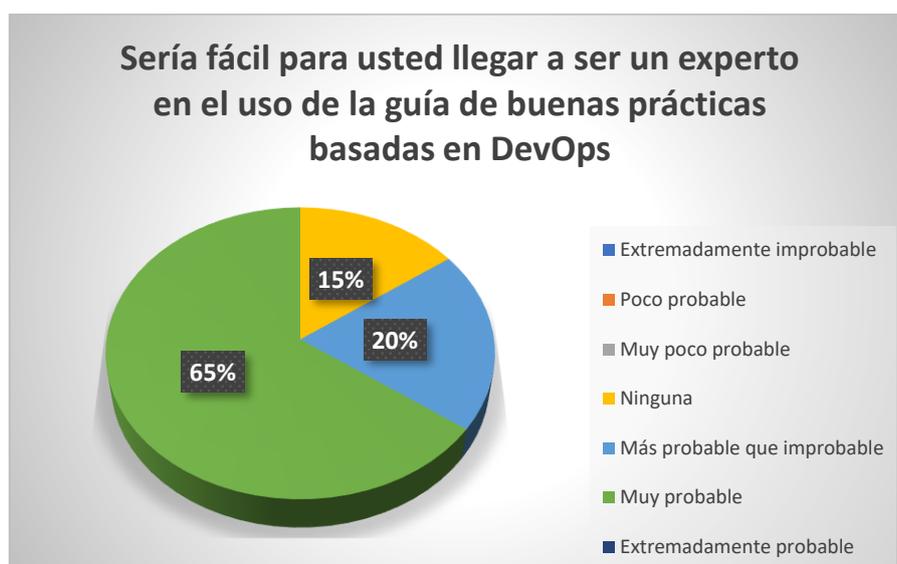


Figura 18. Pregunta N°11 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 13 que pertenecen al 65% afirman que es muy probable que sería fácil para usted llegar a ser un experto en el uso de la guía de buenas prácticas basadas en DevOps, 4 trabajadores que pertenecen al 20% respondieron que es más probable que improbable, 3 que pertenecen al 15% opinaron de manera neutral, dejando a los demás indicadores con valor 0.

## 12. Encuentra una guía de buenas prácticas basadas en DevOps fácil de utilizar

Tabla 39. Encuesta TAM – Pregunta N°12

ITEMS	F	%
Extremadamente improbable	0	0%
Poco probable	0	0%

Muy poco probable	0	0%
Ninguna	0	15%
Más probable que improbable	5	25%
Muy probable	15	75%
Extremadamente probable	0	0%
<b>TOTAL</b>	<b>20</b>	<b>100%</b>

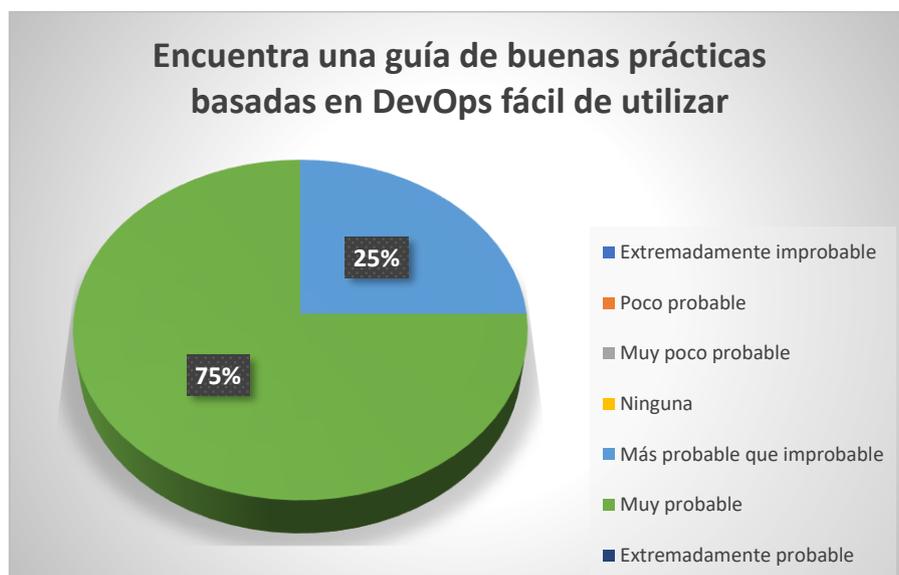


Figura 19. Pregunta N°12 – Encuesta TAM.

Fuente: Elaboración propia.

Análisis: De los 20 trabajadores encuestados, 15 que pertenecen al 75% afirman que es muy probable que encuentra una guía de buenas prácticas basadas en DevOps fácil de utilizar, 5 que pertenecen al 25% respondieron que es más probable que improbable, dejando a los demás indicadores con valor 0.

Resultados de encuesta: KANO

1. **¿Cómo se siente si la empresa incorpora una guía de buenas prácticas para la entrega de su producto (software)?**

Tabla 40. Encuesta KANO – Pregunta N°01

ITEMS	F	%
Me satisface	7	47%
Debería integrarla	5	33%
Normal	2	13%
Puedo admitirlo	1	7%
No me agrada	0	0%
<b>TOTAL</b>	<b>15</b>	<b>100%</b>



Figura 20. Pregunta N°01 – Encuesta KANO.

Fuente: Elaboración propia.

Análisis: De los 15 clientes encuestados, 7 clientes que pertenecen al 47% de encuestados que afirman que les va a satisfacer que la empresa incorporara una guía de buenas prácticas para la entrega de su producto (software), 5 que corresponde al 33% de clientes encuestados respondieron que debería integrarla, 2 que representan el 13% de clientes encuestados respondieron que sería normal hacerlo, 1 que corresponde al 7% de

clientes encuestados afirmó que puede admitirlo, dejando a los demás indicadores con valor 0.

**2. ¿Cómo se siente si con la aplicación de la guía de buenas prácticas se redujeran tiempos en la gestión de cambios o la entrega de su producto?**

Tabla 41. Encuesta KANO – Pregunta N°02

ITEMS	F	%
Me satisface	12	80%
Debería integrarla	2	13%
Normal	1	7%
Puedo admitirlo	0	0%
No me agrada	0	0%
<b>TOTAL</b>	<b>15</b>	<b>100%</b>



Figura 21. Pregunta N°02 – Encuesta KANO.

Fuente: Elaboración propia.

Análisis: De los 15 clientes encuestados, 12 que pertenecen al 80% afirman que les va a satisfacer que la aplicación de la guía de buenas prácticas redujera tiempos en la

gestión de cambios o la entrega de su producto, 2 que corresponde al 13% respondieron que debería integrarla, 1 que representan el 7% que sería normal hacerlo, dejando a los demás indicadores con valor 0.

### 3. ¿Cómo se siente si hubieran demoras en tiempos de entrega de proyectos?

Tabla 42. Encuesta KANO – Pregunta N°03

ITEMS	F	%
Me satisface	0	0%
Debería integrarla	0	0%
Normal	0	0%
Puedo admitirlo	1	7%
No me agrada	14	93%
<b>TOTAL</b>	<b>15</b>	<b>100%</b>

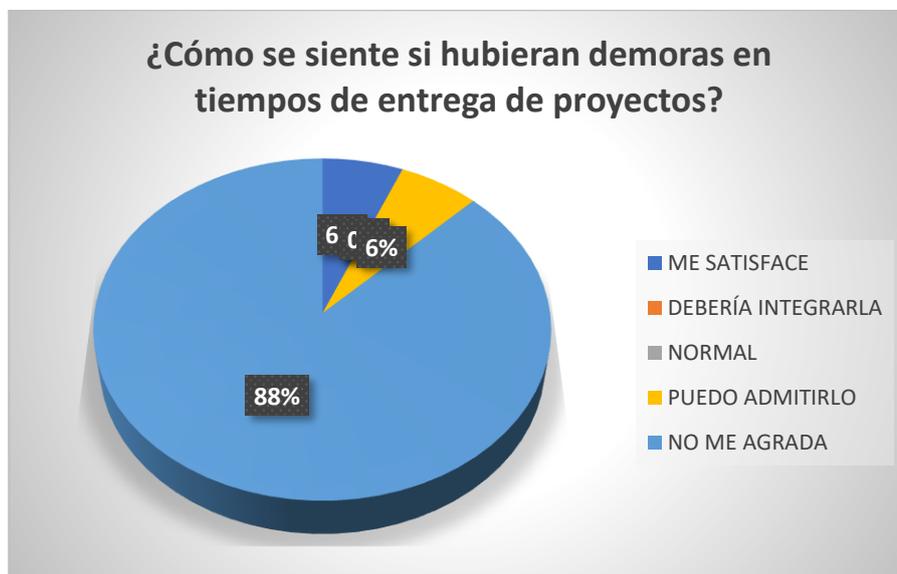


Figura 22. Pregunta N°03 – Encuesta KANO.

Fuente: Elaboración propia.

Análisis: De los 15 clientes encuestados, 14 que pertenecen al 93% afirman que no les agradaría demoras en tiempos de entrega de sus proyectos, 1 que corresponde al 7% respondió que pueden admitirlo, dejando a los demás indicadores con valor 0.

**4. ¿Cómo se siente si una guía de buenas prácticas le diera soluciones rápidas?**

Tabla 43. Encuesta KANO – Pregunta N°04

ITEMS	F	%
Me satisface	15	100%
Debería integrarla	0	0%
Normal	0	0%
Puedo admitirlo	0	0%
No me agrada	0	0%
<b>TOTAL</b>	<b>15</b>	<b>100%</b>

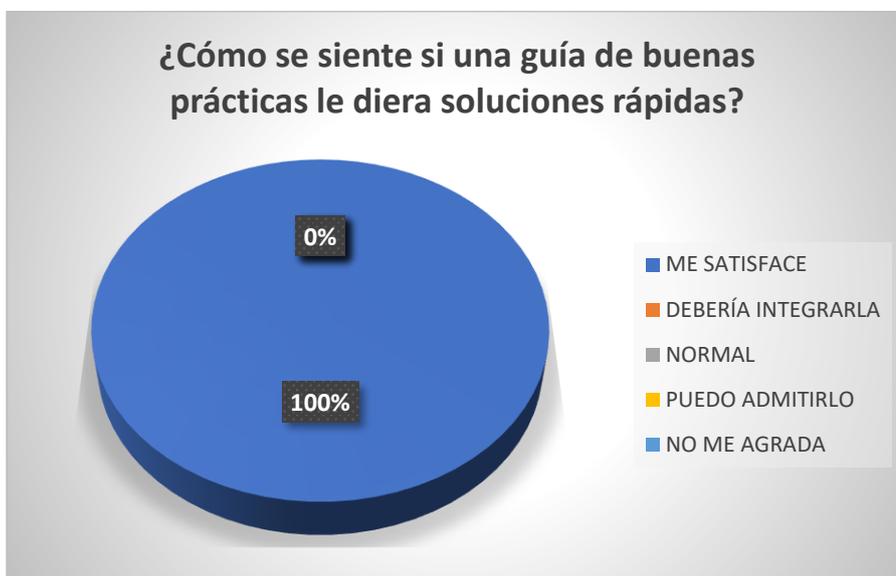


Figura 23. Pregunta N°04 – Encuesta KANO.

Fuente: Elaboración propia.

Análisis: De los 15 clientes encuestados, 15 que pertenecen al 100% afirman que les va a satisfacer que una guía de buenas prácticas le diera soluciones rápidas, dejando a los demás indicadores con valor 0.

### **3.2. Discusión de resultados.**

En referencia a los indicadores presentados en los resultados, se puede aseverar que la implementación de buenas prácticas basadas en DevOps, mejora el proceso de desarrollo del software, en especial en las etapas de integración y despliegue, tal como se muestra en las Figuras 3,4,5, en la cual se resumen los tres indicadores evaluados para la población identificada previamente (Proyecto 1 - Entidad financiera estadounidense, Proyecto 2 - entidad financiera peruana y Proyecto 3 - empresa de mejoramiento del hogar).

El indicador correspondiente a velocidad (Desarrollo – Despliegue), se obtuvo según las figuras citadas, una mejora del 69%, 31% y 34% que simbolizan una velocidad de hasta 7.39, 3.71 y 4.53 días menos respectivamente para los tres proyectos indicados, es decir, desde que comienza el proyecto hasta la entrega en producción. Además se encontró una disminución en 1.5, 1.53 y 2 (respectivamente entre los 3 proyectos de muestra) adicional correspondiente a la velocidad (frecuencia de liberación de código) por cada uno de los sprint detallados; mediante esto mejoró la calidad, incrementando hasta en un 28.37%, 32.86% y 13.5% (respectivamente entre los 3 proyectos de muestra) la calidad (ratio de éxito) para el proyecto una vez desplegado, concluyendo que la historia de usuario está correcta y no requiere alguna retroalimentación de ningún tipo. De la misma forma, Hart & Burke [18], en su artículo, evidenciaron disminución en los tiempos de resolución de incidentes y disminución en tiempos de despliegue en un 53% y 37.5% respectivamente, tras la aplicación de DevOps.

DevOps permite reducir la velocidad (desarrollo - despliegue), desde que inicia un desarrollo correspondiente a una historia de usuario, el testeo, hasta su respecto

despliegue en producción. Mishra & Otaiwi [8], consideran a DevOps como estrategia que permite obtener calidad en los diferentes proyectos de desarrollo de software en distintos ámbitos, en especial en MYPES. Aseguró que, gracias a su aplicación, obtuvo una mejora de la velocidad (desarrollo – despliegue) de hasta un 70% para un ámbito MYPE, además afirmó que mediante su aplicación, las empresas que utilicen esta cultura se beneficiarán al entregar valor de manera rápida por cada sprint.

Por otra parte, las técnicas para la automatización del despliegue a través de herramientas de pipelines, ayudó a potenciar el indicador de velocidad (frecuencia de liberación de código) de un 3.13 a 1.63, de 3.29 a 1.21 y de 3.5 a 1.5 liberaciones por cada sprint posterior a DevOps respecto a la población en estudio. Macarthy & Bajo [17], en su investigación, demuestran que el desarrollo de DevOps en la organización influye en indicadores como tiempos de los despliegues y la liberación del código a través de contenedores y microservicios utilizando pipelines. Al igual que, Pinheiro & Bonifacio [11], resaltan que aplicar ese tipo de técnicas de automatización, incrementa la calidad de los entregables y las diferentes fases del ciclo de vida del software hasta su posterior pase a producción.

Asimismo, la adopción de las buenas prácticas DevOps para el despliegue a través de la automatización de pruebas mediante herramientas anteriormente planteadas, generan una mejora continua en indicadores de dimensiones de calidad y de tiempo, tal como se ha demostrado en este trabajo de investigación. Por consiguiente, esto permite que aumente la satisfacción por parte de los colaboradores y el usuario final, así como se muestran los resultados de las encuestas aplicadas al cliente interno y externo del caso de estudio. Lo mismo demuestra Gren, Torkar & Feldt [14], ellos manifiestan que la rapidez en el marco Scrum es uno de los puntos fundamentales para aplicar DevOps, es pieza indispensable para optimizar el flujo de trabajo y contribuir a la satisfacción de las necesidades y los requerimientos del usuario final.

Finalmente, la presente investigación demostró la ventaja y un resultado positivo de la implementación de buenas prácticas basadas en DevOps sobre las etapas de integración y despliegue de software en una empresa, dicha optimización se vio reflejada en los indicadores de calidad y velocidad. Estos significantes resultados, dichos anteriormente, se consiguieron en tan solo en 3 meses implementando la cultura DevOps, aun quedando un largo camino para seguir aprendiendo y que el caso de estudio llegue a una alta eficiencia como se espera. Stephen, Noppen & Lettice [12], califican a las empresas con alta eficiencia en DevOps como empresas que alcanzan logros e indicadores muy elevados. Además, el autor señala que el tiempo para llegar a convertirse en organizaciones con eficiencia puede variar, pero todo el que lo logre verá el gran impacto y los beneficios que trae consigo específicamente para los procesos de integración y despliegue de software.

### 3.3. Aporte práctico.



Figura 24. Aporte general del trabajo de investigación.

Fuente: Elaboración propia.

**Obj. 1. Identificar el entorno de desarrollo de software en el que se aplicará las buenas prácticas.**

En base a la muestra identificada en apartados precedentes, se determinó para este trabajo de investigación que se utilizarán 03 proyectos:

Tabla 44. *Proyectos considerados en la investigación.*

<b>Proyecto</b>	<b>Giro de la empresa</b>
<b>1</b>	Entidad financiera estadounidense
<b>2</b>	Entidad financiera peruana
<b>3</b>	Empresa de mejoramiento del hogar

Nota: Estos proyectos son unos de los más destacados en la organización.

Fuente: Elaboración propia.

A partir de la identificación de los proyectos, se seguirá utilizando el entorno de desarrollo e integración ya definido por la empresa, sin embargo, ahora se utilizará siguiendo la guía de buenas prácticas mencionada en el Anexo N° 06, de tal manera que se lleve un mejor control y automatización de procesos, haciendo uso de los recursos brindados por la empresa consultora sin generar un costo adicional. Para ello se menciona el entorno de desarrollo e integración de manera detallada a continuación.

Tabla 45. *Diferentes entornos de desarrollo en fases del ciclo de software.*

<b>Niveles del entorno de desarrollo</b>	<b>Programa</b>	<b>Comentarios</b>
Entorno de desarrollo	Visual Studio	Se trabaja con el Visual Studio 2019 Community. En su mayoría, se emplean librerías gratuitas, solo para casos excepcionales

---

		como entidades financieras, se utiliza software de paga para algunos desarrollos.
	Azure repos	Es un software pagado que permite realizar subidas ilimitadas
Entorno de integración	SFTP	Este protocolo permite una conexión entre el servidor de internet particular y de esta forma se transfieren los archivos de manera local o a algún servidor.
		Solo emplea el documento de definición del sistema para realizar las pruebas según requerimientos funcionales establecidos.
Entorno de pruebas o testing	No utiliza	En caso encuentre alguna observación, envía los bugs de cómo se muestra los errores en el navegador para la revisión y modificación respectiva.

---

Nota: Esta información brinda la realidad de los entornos de desarrollo en la institución.

Fuente: Elaboración propia.

La empresa consultora en estudio utiliza Visual Studio Community porque es una versión gratuita que ofrece una amplia gama de características y funcionalidades que ayudan al desarrollador a programar de manera fácil al hacer uso de librerías gratuitas, además cuenta con integración de servicios en la nube, puesto que se integra fácilmente con Azure DevOps, permitiendo aprovechar las capacidades del desarrollo colaborativo, integración y entrega continua sin generar costos adicionales a la empresa.

La selección de Azure repos es porque nos proporciona una plataforma completa, colaborativa y escalable para el desarrollo de software. Su integración, automatización y capacidades de gestión de versiones hacen que sea una opción sólida para equipos de desarrollo que buscan eficiencia y calidad en todo el ciclo de vida del software.

El protocolo SFTP ofrece una solución segura, eficiente y ampliamente aceptada para la transferencia de archivos, lo que lo convierte en una opción sólida para organizaciones que buscan proteger la confidencialidad e integridad de sus datos durante las operaciones de transferencia de archivos.

Para la infraestructura tecnológica, se hace mención que los equipos de la empresa consultora cuentan con un único requisito mínimo que vendría a ser pc's o laptops con un mínimo de 16GB de memoria RAM.

**Obj. 2. Seleccionar las buenas prácticas para el entorno de desarrollo elegido.**

Antes de seleccionar las buenas prácticas para el desarrollo del objetivo es necesario establecer lo siguiente:

**a. Información del personal de trabajo**

La empresa consultora tiene aproximadamente 15 colaboradores, pero la cantidad exacta asignada a los tres proyectos es la siguiente:

Tabla 46. *Característica de los proyectos de muestra de estudio.*

Proyecto	Participantes		
	Líder técnico	Desarrolladores	Calidad
Proyecto uno	1	1	1
Proyecto dos	1	3	1
Proyecto tres	1	3	1

Fuente: Elaboración propia.

## **b. Información del proyecto en desarrollo**

- Proyecto 1: La entidad financiera estadounidense ha solicitado dos requerimientos a gran escala. El primero representa un estudio de mercado en base al aplicativo existente para incorporarle un sistema de encriptación; de acuerdo a eso es necesario verificar si es apto. El segundo requerimiento corresponde a la implementación de intercambio de divisas (control de entrada de efectivo – salida de efectivo a otra cuenta bancaria).
- Proyecto 2: La entidad financiera peruana requiere de un sistema que controle el uso de sus aplicaciones tanto móvil como web.
- Proyecto 3: La empresa del mejoramiento del hogar necesita un sistema que administrativo que gestione las áreas de soporte y además registre el desempeño del colaborador para su evaluación y retroalimentación.

## **c. Información de la metodología de desarrollo**

Actualmente la metodología ágil que utiliza la empresa consultora es SCRUM, pero de manera informal. No existe alguna guía o estructura correctamente definida que permita uniformizar el desarrollo de cada uno de los proyectos. En la mayoría de los proyectos, se observa que la documentación no se encuentra unificada y no mantiene un mismo modelo.

Después de describir la información básica necesaria para hacer la selección de las buenas prácticas, se identificó que la empresa consultora utiliza entornos de desarrollo que se interrelacionan entre sí para poder implementar buenas prácticas basadas en DevOps.

Las buenas prácticas para los entornos de desarrollo (Azure Repos) definido en este trabajo de investigación son las que otorga Azure DevOps. Este producto ofrece un conjunto de servicios y herramientas que apoyan en la administración del ciclo de vida para el desarrollo del software. Para ello, se usó el criterio de selección tomando en cuenta los aspectos generales de los 03 proyectos mencionados, cultura Devops y el marco de aplicabilidad de este para los procesos de desarrollo (Dev) y los procesos de operaciones

(Ops). También se ha tomado en cuenta las buenas prácticas que se utilizarán en esta investigación en base a las necesidades que presenta cada proyecto, además mediante casos de estudio previos que hayan tenido un giro del negocio parecido y enfocados en las limitaciones de la empresa.

Tabla 47. Selección de buenas prácticas para el caso de estudio.

Buenas prácticas	Descripción	Casos de Éxito
Entender las herramientas de colaboración entre los equipos de desarrollo y los equipos de QA (Aseguramiento de la calidad).	Tanto desarrolladores como los de operaciones deben usar el control de versiones para todo. Y deberían compartir la misma fuente única de verdad.	Guía de implementación y buenas prácticas de integración continua para el desarrollo de aplicaciones web en empresas del área metropolitana de San Salvador (De La Fuente, & Flores, 2017)
Implementar automatización de las pruebas	El utilizar herramientas de automatización para las pruebas correspondientes tiene la capacidad de seleccionar el código fuente y permitir ejecutar recursos estándar para garantizar y verificar el código correcto.	Guía DevOps autentia (Autentia, 2016)
Obtener herramientas que permitan reconocer cualquier tipo de solicitud.	El incorporar herramientas que permitan capturar alguna petición del software es de	DevOps como estrategia de aporte a la calidad de productos de software en

	<p>gran importancia. Ninguna implementación o alguna modificación solicitada debe hacer fuera de la aplicación de Devops</p>	<p>MIPYMES desarrolladoras en el contexto colombiano</p>
<p>Desarrollar pruebas de aceptación para cada despliegue realizado.</p>	<p>Dentro de los procedimientos de pruebas de DevOps deben establecer pruebas de aceptación y asegurarse que se logren los objetivos de los criterios seleccionados.</p>	<p>Lineamientos para la implementación del modelo CALMS de DevOps en mipymes desarrolladoras de software en el contexto surcolombiano. (Bejarano &amp; Heredia, 2019)</p>
<p>Hallazgo de problemas a través de la retroalimentación entre los equipos</p>	<p>La realimentación es elemental entre los equipos, debido a que permite detectar los problemas y para eso se debe utilizar herramientas de pruebas para solucionarlos.</p>	<p>Guía Metodológica DevOps – Minvivienda (Minvivienda, 2022)</p>
<p>Soporte para producción</p>	<p>Se debe dar soporte en todo momento y para eso es necesario tomar como base casos existentes para otorgar soluciones inmediatas.</p>	<p>DevOps, la nueva tendencia en el desarrollo de sistemas TI – Empresa Everis (Jiménez, 2016)</p>

Despliegue continuo.	El despliegue continuo disminuye los tiempos de los desarrolladores para identificar nuevos requerimientos o errores en el ambiente de producción.	Prácticas Devops de entrega continua de software para la Transformación digital de los negocios [1].
----------------------	--	--

---

Fuente: Elaboración propia.

### Obj. 3. Elaborar una guía de buenas prácticas basadas en otras existentes.

Para iniciar con una guía de buenas prácticas, en primer lugar, es necesario incorporar la Cultura DevOps a la organización. Después de ello, fue necesario identificar buenas prácticas existentes para analizar su estructura y cómo se acomodan a las necesidades del caso de estudio. Las que se tomaron como base fueron las siguientes:

Tabla 48. *Guías seleccionadas existentes para la implementación de DevOps.*

Guías de buenas prácticas	Descripción
<p>Guía de implementación y buenas prácticas de integración continua para el desarrollo de aplicaciones web en empresas del área metropolitana de San Salvador. (De La Fuente, &amp; Flores, 2017)</p>	<p>Aborda la integración continua para la optimización del proceso de desarrollo de software. Considera puntos clave la retroalimentación temprana, despliegue en menor tiempo y pruebas automatizadas.</p>
<p>Guía DevOps autentia. (Autentia, 2016)</p>	<p>Especifica que trabajar bajo el concepto de DevOps, permite agilizar la recuperación de escenarios que presenten contingencia, debido a que se encuentran identificados</p>

activos de hardware y software y además las tareas rutinarias se encuentran automatizadas.

Indica que la cultura basada en DevOps brinda las fases, actores fundamentales y los lineamientos necesarios para la automatización y optimización de los procesos del ciclo de vida del desarrollo del software.

Guía Metodológica DevOps – Minvivienda. (Minvivienda, 2022)

Esta guía muestra como DevOps, no solo mejora la comunicación entre las áreas, sino también la gestión de los incidentes a través de la infinidad de herramienta de almacenamiento, filtrado y monitorización de mensajes.

DevOps, la nueva tendencia en el desarrollo de sistemas TI – Empresa Everis. (Jiménez, 2016)

---

Fuente: Elaboración propia.

En base a la selección realizada, se elaboró la guía de buenas prácticas basadas en DevOps dirigido a la empresa consultora de software mencionado en el Anexo N°06. Se justifica esta elección porque contribuye positivamente a los objetivos y necesidades de la empresa respecto a los proyectos seleccionados, además que fomenta la colaboración y comunicación entre el equipo de desarrollo. Por otro lado, contribuye a dar entregas más rápidas y confiables acelerando el ciclo de vida entre el desarrollo y despliegue, lo que resulta dar entregas más rápidas y confiables, ya que también interviene la automatización de pruebas y la infraestructura como código. A la vez, hay un porcentaje de reducción de errores y tiempo de inactividad, optimización de recursos, retroalimentación continua, adaptabilidad, escalabilidad y seguridad.

# GUÍA DE BUENAS PRÁCTICAS BASADA EN DEVOPS

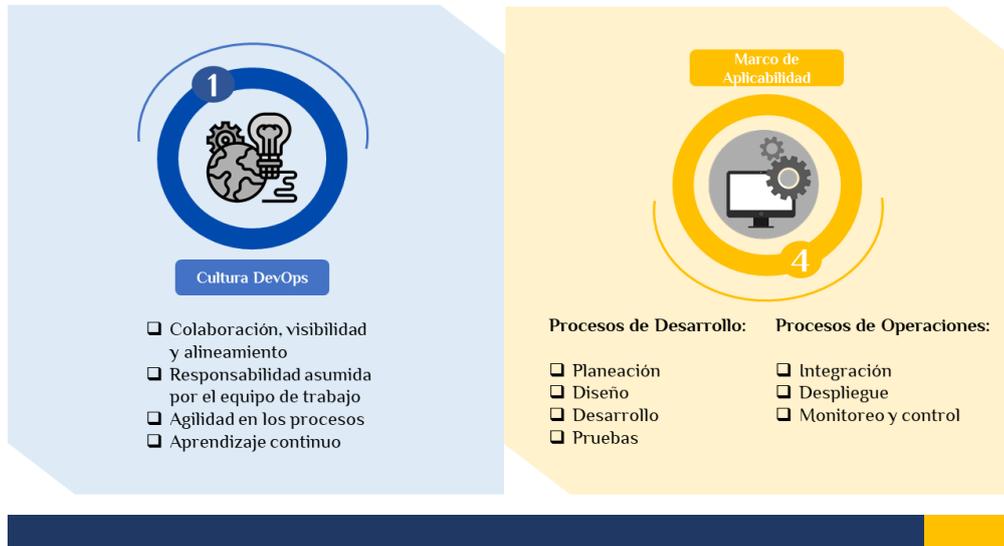


Figura 25. Resumen de la guía de buenas prácticas basadas en DevOps.

Fuente: Elaboración propia.

**Obj. 4. Implementar las buenas prácticas en el entorno de desarrollo de software basado en DevOps.**

**Cultura DevOps:**

## 1. Colaboración, visibilidad y alineamiento

Para iniciar con la incorporación de la cultura DevOps, fue necesario una reunión con todos los colaboradores y la alta dirección, para analizar la misión, visión, procesos, valores y FODA con la finalidad de buscar mejoras involucrando a todo el personal de la empresa consultora.

Se mejoró la misión, visión y los valores personales y corporativos

**Misión:**

Desarrollar herramientas que potencien los negocios de los clientes y tengan mayor impacto y mejores oportunidades en el mercado, siempre apoyados en

nuestros valores corporativos.

**Visión:**

Ser una de las mejores empresas consultora de software a nivel nacional en implementar tecnologías que se adapten a los cambios y condiciones del mercado y a las diferentes necesidades y demandas de nuestra cartera de clientes.

**Valores personales:**

Compromiso, búsqueda de la mejora constante, trabajo en equipo, facilidad para los cambios, mentalidad de liderazgo y honestidad.

**Valores corporativos:**

- Disfrutar y compartir el trabajo, sentir el desarrollo de Software parte de nuestro ADN.
- Buscar la excelencia e innovación mediante la constante capacitación a nuestros colaboradores, el liderazgo y la mejora continua.
- Generar rentabilidad para el beneficio de todos los que forman parte de la empresa consultora.
- Demostrar transparencia, confianza, eficiencia en el ambiente laboral y con nuestros clientes.

Con la finalidad de dar seguimiento al cumplimiento de los objetivos, de estrategias y del alineamiento se concretó que todos los viernes, se realizará una reunión con el personal para dar seguimiento a lo acordado. Se debe tener presente que incorporar una nueva cultura y forma de trabajo lleva su tiempo y tiene que ser constante la supervisión, y de esta forma proponer oportunidades de mejora.

**2. Responsabilidad asumida por el equipo de trabajo**

Para esta actividad, los jefes de área se reunieron con la finalidad de conocer los procesos que manejan cada uno. Es necesario el compromiso tanto del área de desarrollo como el de operaciones para que trabajen de la mano. Para eso, se realizó un formato para evaluar los principales procesos y las debilidades y actuar frente a eso.

Tabla 49. *Formato de Evaluación, principales procesos y debilidades.*

<b>Formato de evaluación</b>	<b>Descripción</b>
Área	<ul style="list-style-type: none"> <li>- Área de gestión de proyectos</li> <li>- Área de desarrollo</li> <li>- Área de calidad</li> <li>- Área de operaciones</li> </ul>
Principales procesos	<p><b>Procesos estratégicos</b></p> <ul style="list-style-type: none"> <li>- Planificación estratégica</li> <li>- Gestión de proyectos</li> </ul>
	<p><b>Procesos del negocio</b></p> <ul style="list-style-type: none"> <li>- Gestión de análisis y diseño</li> <li>- Gestión de desarrollo</li> <li>- Gestión de calidad del producto</li> <li>- Gestión de operaciones</li> </ul>
	<p><b>Procesos de soporte</b></p> <ul style="list-style-type: none"> <li>- Recursos Humanos</li> <li>- Finanzas y tesorería</li> <li>- Tecnologías de Información</li> <li>- Logística</li> </ul>
Debilidades	<ul style="list-style-type: none"> <li>- No tiene una estructura directiva definida</li> </ul>

- Poca cultura del trabajo en equipo
- Poca información comercial
- No innovación en productos / tecnologías
- Sistemas de seguimiento de proyectos no adecuados
- Falta de implementación de buenas prácticas
- Poca comunicación e interacción entre las áreas

---

Fuente: Elaboración propia.

### **3. Agilidad en los procesos**

Cabe mencionar que DevOps, permite la colaboración de metodologías como parte de su aplicación, debido a que esto ayudará a potenciar de manera más fácil la mejora continua en el ciclo de desarrollo de Software. A partir de eso, la metodología SCRUM, realizaron los siguientes tipos de reuniones con la finalidad de agilizar los procesos.

- **Planificación del Sprint**

Los encargados del proyecto o Scrum Máster junto con el equipo de trabajo se dividieron las etapas y tareas faltantes de los 03 proyectos que se tomaron como muestra. Se seleccionaron los responsables por cada una de las tareas.

- **Sprint diarios**

Los equipos de trabajo se reúnen todos los días (Sprint diarios) de 15 - 30 minutos para conversar el avance, limitaciones o algunos obstáculos que se han presentado, con la finalidad de que el proceso se realice de manera rápida pero eficiente y se logren obtener los resultados esperados.

### **4. Aprendizaje continuo**

Como se especifica en la guía, los equipos que emplean la cultura DevOps deben tener una mentalidad de crecimiento. Es por eso, que los líderes de los equipos, de manera trimestral realizarán un plan de capacitación en temas que requieran un conocimiento a profundidad o una práctica más avanzada. El éxito de una organización también se refleja en base a las oportunidades de crecimiento que se les brinda a los colaboradores.

## Marco de aplicabilidad de DevOps:

### 1. Planeación

Cómo se detalló en la planificación del Sprint, el equipo se reúne para tener en claro estos puntos, mediante la siguiente matriz y los diagramas de Gantt para especificar los tiempos y las tareas.

Tabla 50. *Formato para la fase de Planeación*

<b>Fase de Planeación</b>	
- <b>Equipo humano que participa en el proyecto</b>	Se debe detallar el equipo que participa en el proyecto
- <b>Licenciamientos</b>	Es necesario establecer si será necesario adquirir alguna licencia adicional
- <b>Base de datos</b>	Detallar la base de datos que se utilizará en el proyecto
- <b>Lenguaje de programación</b>	Definir el lenguaje de programación
- <b>Componentes propios (en caso se necesite)</b>	Establecer los componentes del proyecto.
- <b>Seguridad e infraestructura</b>	Definir los lineamientos que se deberán tomar en cuenta como parte de la seguridad e infraestructura.

- **Fortalezas y limitaciones** Se deben tener en cuenta fortalezas y limitaciones
- **Cantidad de entregables al cliente final** Establecer cantidad de entregables que se realizarán a lo largo de todo el proyecto

Fuente: Elaboración propia.

Los diagramas de Gantt deben especificar las tareas (completas, incompletas y no iniciadas) y los tiempos, como a continuación se muestra:



Figura 26. Diagrama de Gantt por proyecto.

Fuente: Elaboración propia.

## 2. Gestión de requerimientos y diseño técnico

Es necesario que los cambios, nuevos requerimientos y aprobación de estos queden custodiados en un solo repositorio, para eso se ha elaborado las siguientes tablas:

- **Identificación de requerimientos**

Tabla 51. *Formato de Identificación de requerimientos*

<b>Identificación de requerimientos</b>	
<b>Proyecto:</b>	Nombre del proyecto
<b>Requerimientos funcionales</b>	Identificación de R. funcionales
<b>Requerimientos no funcionales</b>	Identificación de R. no funcionales
<b>Participantes en reunión</b>	
Participante uno	Participante dos
Participante tres	Participante cuatro

Fuente: Elaboración propia.

- **Seguimiento o cambios de los requerimientos**

Tabla 52. *Formato de seguimiento o cambios de los requerimientos*

<b>Seguimiento o cambios de los requerimientos</b>	
<b>Proyecto:</b>	Nombre del proyecto
<b>Requerimientos aprobados</b>	Detalle de los requerimientos aprobados
<b>Requerimientos incompletos</b>	Detalle de los requerimientos incompletos
<b>Requerimientos no iniciados</b>	Detalle de los requerimientos no iniciados
<b>Modificación de requerimientos</b>	Detalle de la modificación de los requerimientos
<b>Participantes en reunión</b>	
Participante uno	Participante dos
Participante tres	Participante cuatro

Fuente: Elaboración propia.

- **Aprobación de los requerimientos**

Tabla 53. Formato de aprobación de los requerimientos.

<b>Aprobación de los requerimientos</b>	
<b>Proyecto:</b>	Nombre del proyecto
<b>Requerimientos aprobados</b>	Detalle de los requerimientos aprobados
<b>Revisión de los requerimientos</b>	Detalle de la revisión de requerimientos
<b>Conformidad de requerimientos</b>	Definir si existe alguna inconformidad
<b>Aprobación de participantes</b>	
Participante uno	Participante dos
Participante tres	Participante cuatro

Fuente: Elaboración propia.

Después de tener establecido cómo se realizará el seguimiento de los requerimientos, se deberá realizar el prototipo de diseño.



Figura 27. Prototipo de Diseño de Software.

Fuente: Elaboración propia.

Como parte de la metodología ágil Scrum, se elaborarán las historias de usuario correspondientes:

Tabla 54. *Formato de Identificación de requerimientos*

<b>Historia de usuario</b>	
Número	Usuario
Nombre de Historia	
Prioridad	Riesgo de Desarrollo
Puntos asignados	Interacción asignada
Programador responsable	
Descripción	
Validación	

Fuente: Elaboración propia.

### **3. Etapas de desarrollo y operaciones**

**Docker** es una tecnología de virtualización que sirve para empaquetar las aplicaciones y hacer que sea portables y mucho más sencillo la distribución de estas. Es una tendencia que utilizan grandes empresas.

Antes de la instalación de Docker, se deben cumplir con estas características:

1. Tipo de sistema operativo: 64 bit
2. Edición: Windows 10 en adelante
3. Número de compilaciones (OS build): mayor a 19041
4. Memoria RAM: Mínimo 4GB
5. Procesador: Mínimo CoreI3

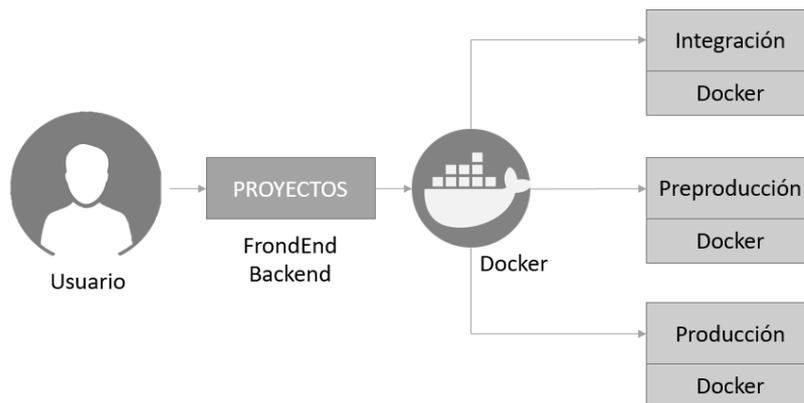


Figura 28. Docker image para los diferentes nodos.

Fuente: Elaboración propia.

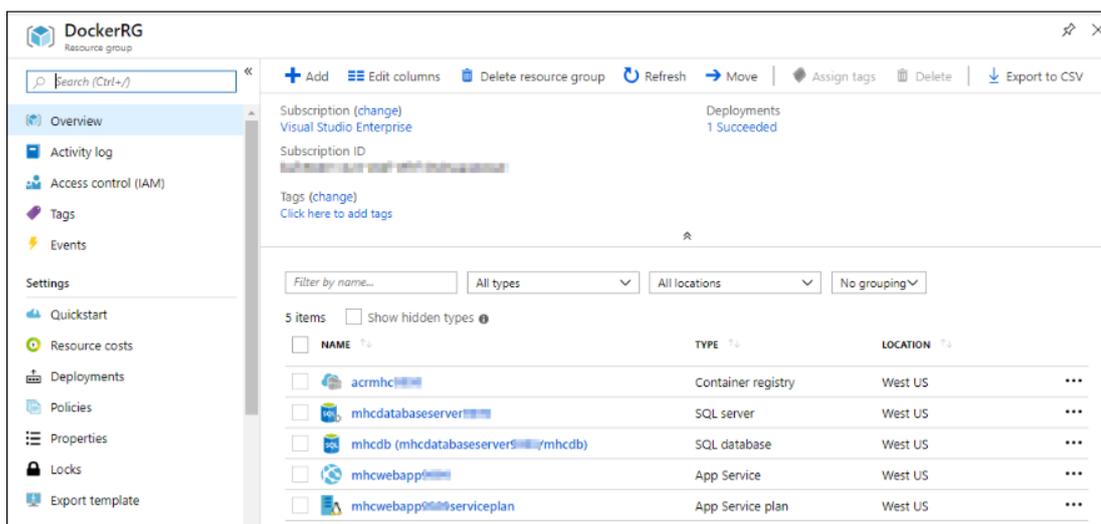


Figura 29. Contenedor de Docker y los Servicios.

Fuente: Elaboración propia.

Es preciso señalar que a pesar de que el control de código fuente (SCC) y su administración es algo inherente en los desarrolladores, al crear aplicaciones de Docker como parte de DevOps, es importante señalar que no se deben enviar las imágenes de Docker a una aplicación de registro (Azure Container Registry). Estas, deben crearse únicamente en el código fuente que se está integrando en la canalización de compilación, en función del repositorio de código fuente (por ejemplo, GIT).

Después del ID de entorno de desarrollo, para almacenar el código fuente en un solo repositorio, se utilizará **Azure Repos**

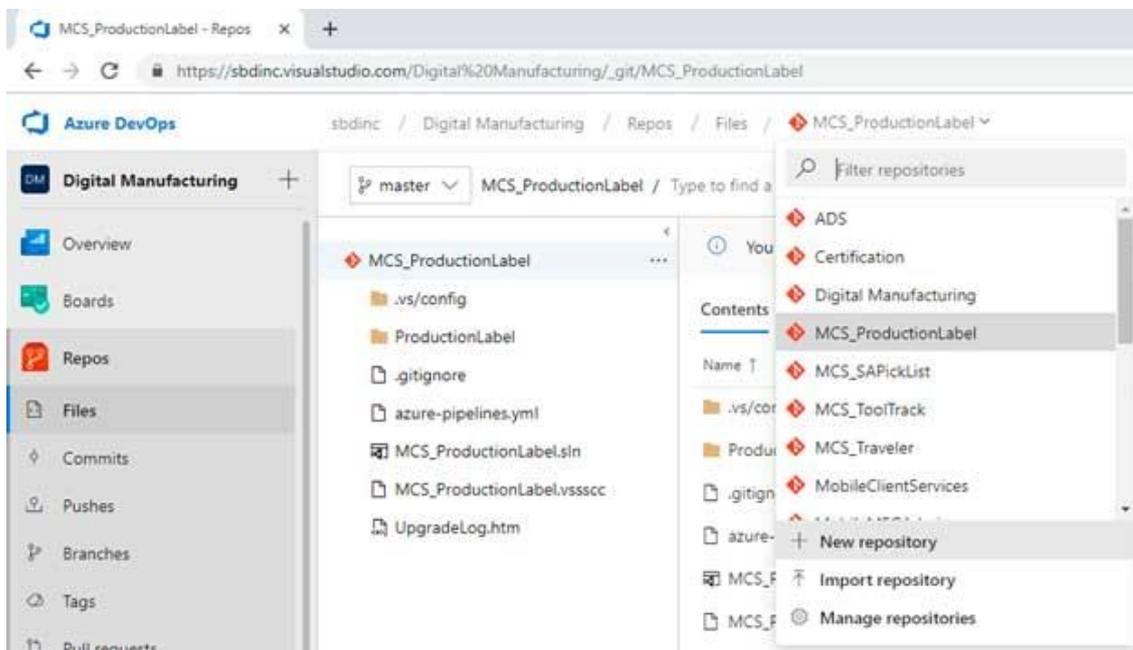


Figura 30. DevOps Repos.

Fuente: Elaboración propia.

Para las pruebas se hizo uso de la **herramienta Azure test plants**. Esta permite planear y ejecutar pruebas con scripts y mantener un seguimiento de estas y de esta forma detectar los errores para tomar las medidas correctivas pertinentes. Además, ayuda a evaluar la calidad a lo largo del desarrollo probando las aplicaciones web o de escritorio.

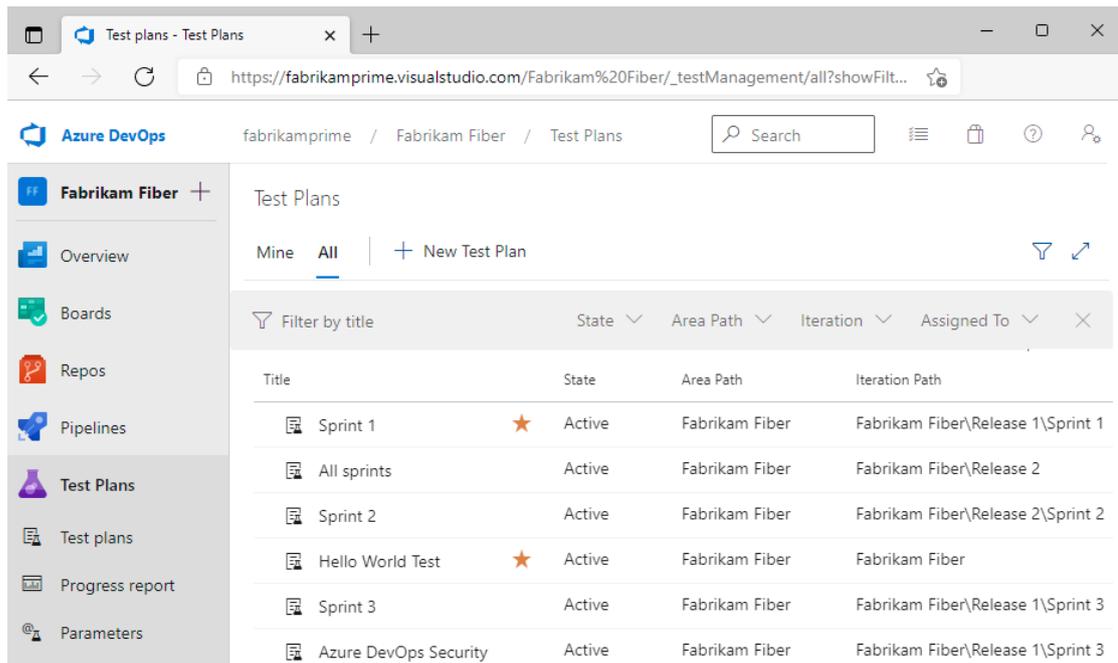


Figura 31. Azure test plants.

Fuente: Elaboración propia.

La integración continua es aquel que permite automatizar las pruebas y compilaciones que se ejecutan a medida que se desarrolla el código y se hace “commit en el sistema”. Una vez efectuado, continúa el proceso de validación. La integración continua es la encargada de automatizar este flujo, permitiendo incrementar la eficiencia. Los posibles errores se pueden determinar con prontitud, antes de fusionar el código desarrollado a través de la combinación con la rama maestra

**Azure Pipelines** crea y prueba automáticamente proyectos de código para ponerlos a disposición de otros. Funciona con prácticamente cualquier tipo de proyecto o lenguaje. Azure Pipelines acopla tanto la integración continua (CI) y entrega continua (CD) con la finalidad de probar y construir su código y enviarlo a cualquier destino.

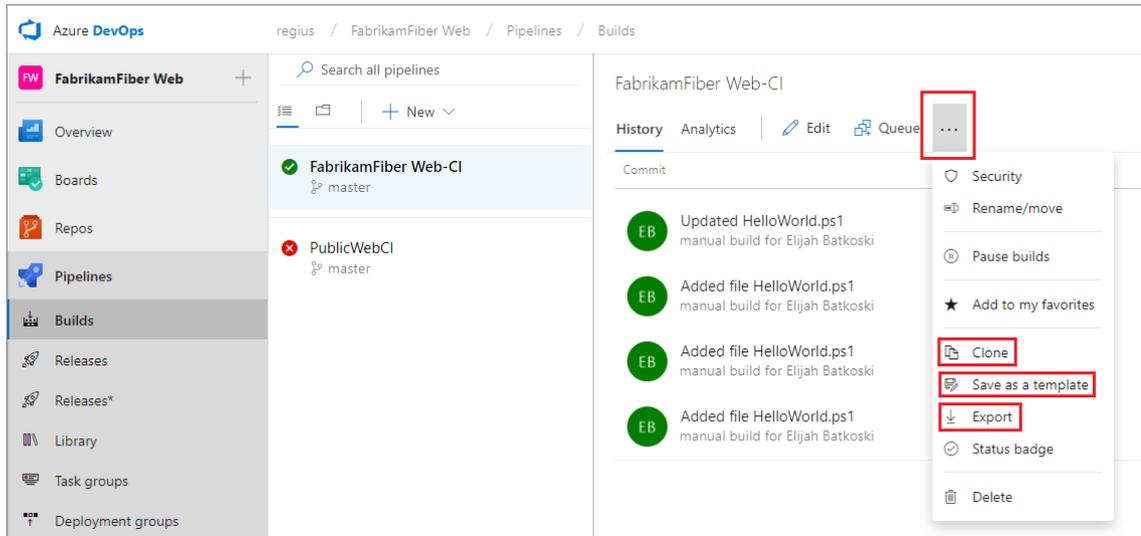


Figura 32. Interfaz de Azure Pipelines.

Fuente: Elaboración propia.

Para esta investigación se utilizó **Hashicorp Terraform**, que es una herramienta iaC de código abierto (infraestructura como código) para aprovisionar y administrar la infraestructura en la nube. Codifica la infraestructura en los archivos de configuración que describen el estado deseado para la topología. Terraform permite la administración de cualquier infraestructura (como nubes públicas, nubes privadas y servicios SaaS) mediante proveedores de Terraform.

La cultura DevOps va mucho más allá de la implementación; también se debe asumir la responsabilidad de la entrega de software así como su rendimiento. Todo el proceso DevOps crea un bucle de comentarios, que a la larga puede ayudar a mejorar el proyecto y validar la decisión de implementar el software. El registro y la supervisión son componentes indispensables que sustentan el aprendizaje validado y que van de la mano con la correcta entrega del software y la eficiencia de los procesos.

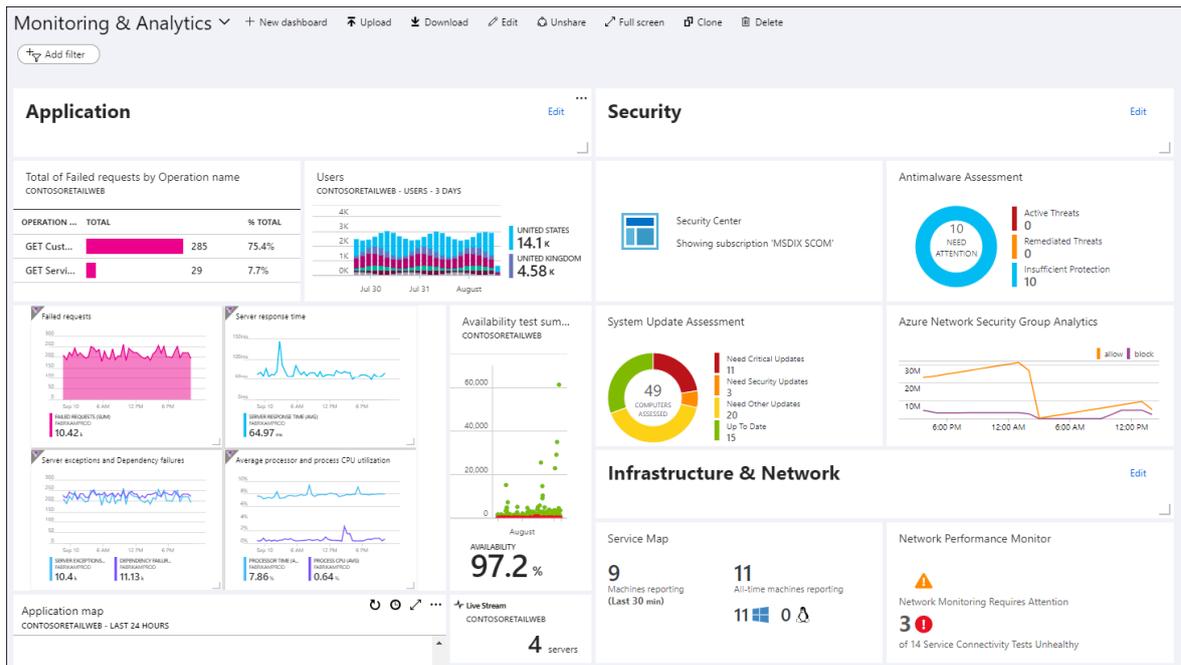


Figura 33. Interfaz de Azure Monitor.

Fuente: Elaboración propia.

**Obj. 5. Evaluar el impacto de la ejecución de las buenas prácticas en entornos de desarrollo basados en DevOps para la mejora de integración continua en proyectos de software.**

Para la evaluación del impacto, se desarrollaron dos encuestas existentes validas (Modelo Kano, modelo TAM) para el cliente y el colaborador respectivamente, para conocer el grado de satisfacción o de aceptación sobre la aplicación de buenas prácticas basadas en DevOps. Ambos instrumentos se encuentran detalladas en el Anexo N°03.

## IV. CONCLUSIONES Y RECOMENDACIONES

### 4.1. Conclusiones.

- El entorno de desarrollo de las fases del ciclo de software identificado del caso de estudio fue Visual Studio, para el entorno de integración, azure repost y el protocolo SFTP y en el entorno de pruebas se observó que no se utiliza alguna herramienta para ello.
- Se seleccionaron 07 buenas prácticas mediante casos de estudio previos que hayan tenido un giro del negocio parecido y enfocados en las limitaciones de la empresa.
- Se elaboró una guía de buenas prácticas a través de unas guías existentes, incorporando aspectos generales, cultura Devops y el marco de aplicabilidad del mismo para los procesos de desarrollo (Dev) y los procesos de operaciones (Ops).
- Se implementó las buenas prácticas en el entorno de desarrollo de software basado en DevOps en los tres proyectos que se seleccionaron como muestra, iniciando con inculcar una cultura en DevOps y aplicando el marco de DevOps, con las diferentes herramientas que se deben emplear para cada etapa del desarrollo de software, en especial, para los procesos de integración y despliegue.
- Se concluyó que el impacto de la ejecución, mediante encuestas de satisfacción para el cliente interno y externo y análisis de indicadores de velocidad y calidad, resultó de manera positiva logrando un mayor rendimiento y eficiencia en el caso de estudio. Para el indicador velocidad (desarrollo - despliegue), se obtuvo una mejora del 69%, 31%, 34%, indicador velocidad (frecuencia de liberación de código), 48%, 48% y 57% e indicador calidad (ratio de éxito), 34%, 46% y 19%, todos los indicadores en base a los 03 proyectos definidos en la muestra.

### 4.2. Recomendaciones.

- En base a los descubrimientos de esta investigación, se recomienda que, las empresas

del rubro de consultorías de software, implementen buenas prácticas, y una gran opción es DevOs, que permiten mejorar la velocidad y calidad en el proceso de integración y despliegue, para conseguir ser más competitivos en el mercado.

- Se recomienda que las organizaciones que implementen las buenas prácticas en DevOps, no impongan el uso de herramientas antes de haber inculcado una cultura de DevOps en su empresa. Es importante reconocer que una gestión del cambio es una iniciativa de mejora y que es progresivo y necesita ser capacitado e incentivado a lograrlo.
- Para la implementación de DevOps es necesario alinearlos a una metodología ágil, con la finalidad de ser menos complejo para la aplicación y por innovación en los procesos.

## REFERENCIAS.

- [1] P. Castañeda, «Prácticas DevOps de entrega continua de software para la transformación digital de los negocios,» Universidad EAFIT, Medellín, 2019.
- [2] K. Terán, «Análisis y formulación de estrategia general para preparar la adopción de la cultura DevOps en matrices bancarias de la ciudad de Guayaquil,» Universidad Católica de Santiago de Guayaquil, Guayaquil, 2020.
- [3] F. Casas, «Implementación de DevOps para mejorar la integración y despliegue de software en el sector de seguros,» Universidad Cesar Vallejo, Lima, 2020.
- [4] The Standish Group, « Resultados de proyectos en el sector de las tecnologías de información,» 2019. [En línea].
- [5] E. Acosta, «Documentación y adopción de buenas prácticas para el desarrollo de software basado en los fundamentos DevOps,» Universidad de Antioquia, Medellín, 2019.
- [6] T. Brown, «DevOps vs Agile: What's the difference?,» 2020. [En línea]. Available: <https://opensource.com/article/20/2/devops-vs-agile>.
- [7] E. Novoseltseva, «Los beneficios y herramientas DevOps,» 3 Abril 2018. [En línea]. Available: <https://apiumhub.com/es/tech-blog-barcelona/beneficios-herramientas-devops/>.
- [8] A. Mishra y Z. Otaiwi, «DevOps and software quality: A systematic mapping,» *Scimedirect*, 2020.
- [9] Berinka, Foote, Burgo y Prezzama, «The Impact of Agile Methods and “DevOps” on Day 2+ Operations for Large Enterprises,» *Scopus*, vol. 283, 15 Julio 2021.
- [10] D. Yang, D. Wang, D. Yang, Q. Dong, Y. Wang, H. Zhou y D. Hong, «DevOps in Practice for Education Management Information System at ECNU,» *ScienceDirect*, vol. 176, pp. 1382-1391, 2020.

- [11] W. Pinheiro, G. Pinto y R. Bonifácio, «Adopting DevOps in the real world: A theory, a model, and a case study,» *ScienceDirect*, vol. 157, 2019.
- [12] J. Stephen, J. Noppen y F. Lettice, «Management challenges for DevOps adoptin within UK SMEs,» *ACM*, pp. 7-11, 2016.
- [13] P. Giménez y G. Santos, «DevOps Maturity Diagnosis - A case Study in Two Public Organizations,» *ACM*, nº 11, pp. 1-8, 2020.
- [14] L. Gren, R. Torkar y R. Feldt, «Group development and group maturity when building agile teams,» *ACM*, vol. 124, pp. 104-119, 2017.
- [15] L. Leite, G. Pinto, F. Kon y P. Meirelles, «The organization of software teams in the quest for continuous delivery: A grounded theory approach,» *ScienceDirect*, vol. 139, 2021.
- [16] E. Di Nitto, P. Jamshidi, M. Guerriero, I. Spais y D. Tamburri, «A software architecture framework for quality-aware,» *ACM*, pp. 12-17, 2016.
- [17] R. Macarthy y J. Bajo, «An Empirical Taxonomy of DevOps in Practice,» *Scopus*, pp. 221-228, 2020.
- [18] M. Hart y J. Burke, «An exploratory study on thr DevOps it alignment model,» *Scopus*, vol. 15, pp. 127-154, 2020.
- [19] J. Wettinger, U. Breitenbucher, O. Kopp y F. Leymann, «Streamlining DevOps automation for Cloud applications using TOSCA as standardized metamodel,» *ScienceDirect*, vol. 56, pp. 317-332, 2016.
- [20] D. Batra, «Job-work fit as a determinant of the acceptance of large-scale agile methodology,» *SciencieDirect*, vol. 168, 2020.
- [21] A. Rahman, R. Mahdavi y L. Williams, «A systematic mapping study of infraestructure as code research,» *ScienceDirect*, vol. 108, pp. 65-77, 2019.

- [22] Z. Toh, S. Sahibuddin y R. Abu, «A Review on DevOps Adoption in Continuous Delivery Process,» *IEEE*, 2021.
- [23] L. Bass, I. Weber y Z. Liming, *DEVOPS A Software Architect's Perspective*. Westford, Massachusetts: Pearson Education Inc., 2015.
- [24] Gartner, *Market Guide for Software Change and Configuration Management Software* Stamford, Gartner, 2015.
- [25] J. Willis, «The Convergence Of DevOps,» 2012. [En línea]. Available: <https://itrevolution.com/theconvergence-of-devops/>.
- [26] Aplyca, «Integración continua y entrega continua,» 2018. [En línea]. Available: <https://www.aplyca.com/es/blog/integracion-entrega-continua-ci-cd>.
- [27] Cleverism, «7 Basic Software Development Life Cycle (SDLC) Methodologies: Which One is Best?,» 2018. [En línea]. Available: <https://www.cleverism.com/software-development-life-cycle-sdlcmethodologies/>.
- [28] S. Ambler y M. Lines, *Disciplined agile delivery*, Boston: IBM Press, 2012.
- [29] R. Winston, *Managing*, IEEE, 1970.
- [30] ESAN Graduate School of Business, «Marketing relacional: Entre el valor y el precio,» 2017. [En línea]. Available: <https://www.esan.edu.pe/apuntes-empresariales/2017/03/marketingrelacional-entre-el-valor-y-el-precio/>.
- [31] L. Andersson, «ITIL and DevOps - better together,» 2017. [En línea]. Available: <https://www.axelos.com/news/blogs/september-2017/itil-and-devops-bettertogether>.
- [32] D. Campbell y J. Stanley, «Diseños experimentales y cuasiexperimentales en la investigación social,» 1976. [En línea]. Available: <https://knowledgesociety.usal.es/sites/default/files/campbell-stanley-252-disec3b1os-experimentales-y-cuasiexperimentales-en-lainvestigacion3b3n-social.pdf>.

- [33] L. Díaz, «La Observación,» 2011. [En línea]. Available:  
[http://www.psicologia.unam.mx/documentos/pdf/publicaciones/La\\_observacion\\_Lidia\\_Diaz\\_Sanjuan\\_Texto\\_Apoyo\\_Didactico\\_Metodo\\_Clinico\\_3\\_Sem.pdf](http://www.psicologia.unam.mx/documentos/pdf/publicaciones/La_observacion_Lidia_Diaz_Sanjuan_Texto_Apoyo_Didactico_Metodo_Clinico_3_Sem.pdf).
- [34] Y. Lincoln y E. Guba, *Effective evaluation: improving the usefulness of evaluation results through responsive and naturalistic approaches*, San Francisco, 1981.
- [35] I. Mejia, R. Ramirez, H. Jimenez y J. Rosas, «A new method a architecture entreprise,» *Conference IEEE bussines*, pp. 200-215, 2019.
- [36] H. Jimenez, R. Rodriguez y J. Tiparra, *Diagnóstico de TEA*, Madrid: Latinoamérica SA, 1978.
- [37] P. SZNAJDLEDER, *Java a fondo - estudio del lenguaje y desarrollo de aplicaciones - 2a ed.*, México: Alfaomega, 2012.
- [38] I. Mejia, M. Tuesta y M. Forero, «A new method of enterprise archicture small organizations,» *Computer Science Techology*, pp. 150-170, 2020.
- [39] J. Palma y R. Marín, *Inteligencia Artificial*, Madrid: McGrawHill, 2008, pp. 3-8.
- [40] K. Rojas, «Identificación de efectos negativos de la TEA en el aprendizaje,» *IEEE conference Techology children especial*, pp. 200-215, 2018.
- [41] K. Rojas, «Identificación de efectos negativos de la TEA en el aprendizaje,» *IEEE conference xyz*, pp. 200-215, 2018.

## ANEXOS

### Anexo 1. Resolución de aprobación del proyecto de investigación



UNIVERSIDAD  
SEÑOR DE SIPÁN

FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO  
RESOLUCIÓN N°0457-2022/ FIAU-USS

Pimentel, 15 de Julio de 2022

VISTO:

El Acta de reunión N° 1507 - 2022 del Comité de investigación de la Escuela profesional de INGENIERÍA DE SISTEMAS remitida mediante Oficio N°0166-2022/FIAU-IS-USS de fecha 15 de julio de 2022, y;

CONSIDERANDO:

Que, de conformidad con la Ley Universitaria N° 30220 en su artículo 48° que a letra dice: "La investigación constituye una función esencial y obligatoria de la universidad, que la fomenta y realiza, respondiendo a través de la producción de conocimiento y desarrollo de tecnologías a las necesidades de la sociedad, con especial énfasis en la realidad nacional. Los docentes, estudiantes y graduados participan en la actividad investigadora en su propia institución o en redes de investigación nacional o internacional, creadas por las instituciones universitarias públicas o privadas.";

Que, de conformidad con el Reglamento de grados y títulos en su artículo 21° señala: "Los temas de trabajo de investigación, trabajo académico y tesis son aprobados por el Comité de Investigación y derivados a la facultad o Escuela de Posgrado, según corresponda, para la emisión de la resolución respectiva. El periodo de vigencia de los mismos será de dos años, a partir de su aprobación. En caso un tema perdiera vigencia, el Comité de Investigación evaluará la ampliación de la misma.

Que, de conformidad con el Reglamento de grados y títulos en su artículo 24° señala: La tesis es un estudio que debe denotar rigurosidad metodológica, originalidad, relevancia social, utilidad teórica y/o práctica en el ámbito de la escuela profesional. Para el grado de doctor se requiere una tesis de máxima rigurosidad académica y de carácter original. Es individual para la obtención de un grado; es individual o en pares para obtener un título profesional. Asimismo, en su artículo 25° señala: "El tema debe responder a alguna de las líneas de investigación institucionales de la USS S.A.C.".

Que, mediante documentos de vistos, el Comité de investigación de la referida Escuela profesional acordó aprobar el tema de las Tesis perteneciente, ampliación de la vigencia de trabajo de investigación, y modificar el tema de las Tesis perteneciente que se detallan en el Acta de reunión N° 1507 - 2022, de la línea de investigación de INFRAESTRUCTURA, TECNOLOGÍA Y AMBIENTE, a cargo del egresado del Programa de estudios INGENIERÍA DE SISTEMAS, hasta la fecha que indica en el anexo de la presente resolución.

Estando a lo expuesto, y en uso de las atribuciones conferidas y de conformidad con las normas y reglamentos vigentes;

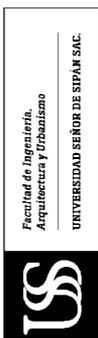
SE RESUELVE:

**ARTÍCULO 1°:** APROBAR, el tema de las Tesis perteneciente a la línea de investigación de INFRAESTRUCTURA, TECNOLOGÍA Y MEDIO AMBIENTE, a cargo de los egresados del Programa de estudios de INGENIERÍA DE SISTEMAS según se detalla en el anexo de la presente Resolución.

**ARTÍCULO 2°:** AMPLIAR VIGENCIA, del trabajo de investigación a cargo de los estudiantes y /o egresados del Programa de estudios de INGENIERÍA DE SISTEMAS que se detallan en la presente Resolución.

**ARTÍCULO 3°:** MODIFICAR, el tema de las Tesis perteneciente a la línea de investigación de INFRAESTRUCTURA, TECNOLOGÍA Y MEDIO AMBIENTE, a cargo de los estudiantes y /o egresados del Programa de estudios de INGENIERÍA DE SISTEMAS según se detalla en el anexo de la presente Resolución.

**ARTÍCULO 2°:** DEJAR SIN EFECTO, toda Resolución emitida por la Facultad que se oponga a la presente Resolución.



Anexo 2. Carta de aceptación de la institución para la recolección de datos.



Lima, 13 de julio de 2022

Quien suscribe:

**Sr. Juan de Dios García Bravo**

**Representante Legal – ITSight Consulting SAC**

**AUTORIZA:**

**Permiso para recojo de información pertinente en función del proyecto de investigación, denominado: Aplicación de buenas prácticas para entornos de desarrollo de software basados en DEVOPS para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima.**

Por el presente, el que suscribe, Juan de Dios García Bravo, representante legal de la empresa ITSIGHT CONSULTING SAC, AUTORIZO al alumno Dennis Mauricio Avilés Odar, identificado con DNI N°71386344, estudiante de la Escuela Profesional de Ingeniería de Sistemas, y autor del trabajo de investigación denominado: Aplicación de buenas prácticas para entornos de desarrollo de software basados en DEVOPS para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima, al uso de dicha información que conforma el expediente técnico así como hojas de memorias, cálculos entre otros como planos para efectos exclusivamente académicos de la elaboración de tesis de pregrado enunciada líneas arriba de quien solicita se garantice la absoluta confidencialidad de la información solicitada.

Atentamente.



JUAN DE DIOS GARCÍA BRAVO  
GERENTE GENERAL  
ITSIGHT CONSULTING S.A.C

Juan de Dios Garcia Bravo – DNI 42287810

Cargo de la empresa: Gerente General

Anexo 3. Instrumentos de recolección de datos, con su respectiva validación de los instrumentos.

## **ENCUESTA DE SATISFACCIÓN AL CLIENTE (MODELO KANO)**

Es un modelo desarrollado por Noriaki Kano, que evalúa ideas nuevas y el desarrollo de productos o servicios mediante las respuestas de los clientes. Este modelo tiene seis categorías, pero las tres primeras tienen mayor influencia sobre la satisfacción del cliente. Estas son:

1. Factores básicos: son aquellos requerimientos que causarán un disgusto frente algunos puntos que no lo satisfagan. El cliente mira a estos requisitos como algo previo para entrar a los siguientes.
2. Factores de entusiasmo: son aquellos que generan satisfacción al cliente si son realizados como se debe, pero no causan descontento si no se entregan. La organización que use estos factores se distinguirá de su competencia, porque es algo que el cliente siempre desea.
3. Factores de desempeño: estos factores si son trascendentales, están conectados estrechamente a las necesidades de los clientes que medirán el desempeño de la organización.

Valores para su evaluación:

- Me gusta: Es el aspecto máximo con un valor de 5 positivo.
- Debería incorporarla: Se considera un valor de 4 positivo.
- Normal: Aspecto neutro, con un valor de 3.
- Puedo tolerarlo: Se considera un valor de 2 negativo.
- No me gusta: Es el aspecto mínimo con un valor de 1 negativo.

Objetivo: Conocer su grado de satisfacción frente a los servicios que presta la empresa consultora.

Indicaciones: Evalúe los siguientes aspectos y asigne una variable marcando con una X en las casillas, teniendo en cuenta que ME GUSTA es el más alto (satisfecho) y NO ME GUSTA el más bajo (insatisfecho). Gracias por su colaboración.

Factores básicos: GUÍA DE BUENAS PRÁCTICAS		
¿Cómo se siente si la empresa incorpora una guía de buenas prácticas para la entrega de su producto (software)?	ME GUSTA	<input type="checkbox"/>
	DEBERÍA INCORPORARLA	<input type="checkbox"/>
	NORMAL	<input type="checkbox"/>
	PUEDO TOLERARLO	<input type="checkbox"/>
	NO ME GUSTA	<input type="checkbox"/>
Factores de entusiasmo: REDUCIR TIEMPOS DE GESTIÓN DE CAMBIOS Y ENTREGA DEL PRODUCTO		
¿Cómo se siente si con la aplicación de la guía de buenas prácticas se redujeran tiempos en la gestión de cambios o la entrega de su producto?	ME GUSTA	<input type="checkbox"/>
	DEBERÍA HACERLO	<input type="checkbox"/>
	NORMAL	<input type="checkbox"/>
	PUEDO TOLERARLO	<input type="checkbox"/>
	NO ME GUSTA	<input type="checkbox"/>
¿Cómo se siente si hubiera demora en tiempos de entrega de proyecto?	ME GUSTA	<input type="checkbox"/>
	DEBERÍA HACERLO	<input type="checkbox"/>
	NORMAL	<input type="checkbox"/>
	PUEDO TOLERARLO	<input type="checkbox"/>
	NO ME GUSTA	<input type="checkbox"/>
Factores de desempeño: SOLUCIONES RÁPIDAS Y DE CALIDAD		
	ME GUSTA	<input type="checkbox"/>

¿Cómo se siente si una guía de buenas prácticas le diera soluciones rápidas y de calidad como consumidor?	DEBERÍA HACERLO	
	NORMAL	
	PUEDO TOLERARLO	
	NO ME GUSTA	

### **ENCUESTA DE SATISFACCIÓN AL COLABORADOR (MODELO TAM)**

Es un modelo de aceptación tecnológica que fue creado para explicar el uso de las diferentes tecnologías o marcos de desarrollo en diferentes ambientes, modelando cómo los usuarios aceptan y lo utilizan.

TAM posee dos variables importantes:

1. Utilidad percibida: Grado en que la persona que utilizará el aporte tecnológico, estima que su uso propuesto mejoraría su rendimiento en el trabajo.
2. Facilidad de uso percibida: Grado en que la persona que utilizará el aporte tecnológico cree que su uso es de manera fácil y no genera mucho esfuerzo.

Valores para su evaluación:

- Extremadamente improbable: Se considera un valor de 0 negativo.
- Bastante improbable: Se considera un valor de 1 negativo.
- Ligeramente improbable: Se considera un valor de 2 negativo.
- Ninguno: Es un aspecto neutro, se considera un valor de 3.
- Ligeramente probable: Se considera un valor de 4 positivo.
- Bastante probable: Se considera un valor de 5 positivo.
- Extremadamente probable: Se considera un valor de 6 positivo

Objetivo: Conocer el grado de satisfacción al aplicar las buenas prácticas basadas en DevOps..

Indicación: Evalué los siguientes aspectos y asigne una variable marcando con una X en las casillas. Gracias por su colaboración.

	<b>Extremadamente improbable</b>	<b>Bastante improbable</b>	<b>Ligeramente improbable</b>	<b>Ninguno</b>	<b>Ligeramente probable</b>	<b>Bastante probable</b>	<b>Extremadamente probable</b>
Aplicar una guía de buenas prácticas basada en DevOps ayudaría a realizar sus tareas en menor tiempo posible.							
Aplicar una guía de buenas prácticas basada en DevOps mejoraría el desempeño de su trabajo.							
Aplicar una guía de buenas prácticas basada en DevOps incrementaría su							

productividad y rendimiento en el trabajo.							
Aplicar una guía de buenas prácticas basada en DevOps aumentaría la efectividad en su trabajo.							
Aplicar una guía de buenas prácticas basada en DevOps facilitaría la realización de su trabajo.							

	<b>Extremadamente improbable</b>	<b>Bastante improbable</b>	<b>Ligeramente improbable</b>	<b>Ninguno</b>	<b>Ligeramente probable</b>	<b>Bastante probable</b>	<b>Extremadamente probable</b>
Aprender a aplicar una guía de buenas prácticas basada en DevOps, sería sencillo para usted.							
Le resultaría fácil aplicar una guía de buenas prácticas basada en DevOps para hacer lo que usted necesite.							
Su interacción con una guía de buenas prácticas basado en DevOps sería clara y entendible							
Encuentra a una guía de buenas prácticas basada en							

DevOps flexible para interactuar con él.							
Sería fácil para usted llegar a aplicar una guía de buenas prácticas basado en DevOps							

Anexo 4. Instrumento de Validación No Experimental por Juicio de expertos.

La presente matriz fue usada como un instrumento de validación por los jueces expertos, lo que permite, mediante sus calificaciones y observaciones poder evaluar y retroalimentar la guía de buenas prácticas.



**1. VALIDACIÓN DEL INSTRUMENTO**

**1.1. Instrumento de Validación No Experimental por Juicio de expertos**

<b>1. NOMBRE DEL JUEZ</b>		RICARDO HUACHOS HUAMANCHAHA
<b>2.</b>	<b>PROFESIÓN</b>	ING. DE SISTEMAS
	<b>ESPECIALIDAD</b>	DESARROLLO DE SOFTWARE
	<b>GRADO ACADÉMICO</b>	TITULADO
	<b>EXPERIENCIA PROFESIONAL (AÑOS)</b>	25
	<b>CARGO</b>	JEFE DE SISTEMAS
<b>TÍTULO DE LA INVESTIGACIÓN:</b> Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima		
<b>3. DATOS DEL TESISISTA</b>		
<b>3.1</b>	<b>NOMBRES Y APELLIDOS</b>	Dennis Mauricio Avilés Odar
<b>3.2</b>	<b>PROGRAMA DE PREGRADO</b>	Ingeniería de Sistemas
<b>4. INSTRUMENTO EVALUADO</b>	Guía de buenas prácticas basadas den DevOps para una empresa consultora de Software	
<b>5. OBJETIVOS DEL INSTRUMENTO</b>	<b>GENERAL</b> Validar el contenido de la Guia de buenas prácticas basadas den DevOps para una empresa consulta de Software.	
	<b>ESPECÍFICOS</b> Usar la validación de la guia para dar valor técnico a las conclusiones de uno de los objetivos específicos del proyecto de tesis titulado "Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima"	
A continuación, se le presentan los indicadores en forma de preguntas o propuestas para que Ud. los evalúe marcando con un aspa (x) en "A" si está de ACUERDO o en "D" si está en DESACUERDO. SI ESTÁ EN DESACUERDO POR FAVOR ESPECIFIQUE SUS SUGERENCIAS.		
<b>No</b>	<b>DETALLE DE LOS ITEMS DEL INSTRUMENTO</b>	
01	¿La Cultura DevOps expuesta en la guía es lo suficientemente entendible al usuario para su fácil comprensión y aplicabilidad?	A ( <input checked="" type="checkbox"/> )      D (    ) SUGERENCIAS:

02	¿Está de acuerdo con el marco de aplicabilidad indicado en la guía para los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A ( X ) SUGERENCIAS:	D ( )
03	¿Está de acuerdo con cada una de las fases indicadas en los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A ( X ) SUGERENCIAS:	D ( )
04	¿Cree que las empresas consultoras de software tomen como guía el documento elaborado y lo apliquen en su día a día?	A ( X ) SUGERENCIAS:	D ( )
<b>PROMEDIO OBTENIDO:</b>		A ( X )	D ( ):
<b>6. COMENTARIOS GENERALES</b>			
<p>La guía cumple con los lineamientos expuestos en el presente instrumento de validación, además de ello considero que será un buen ejemplo de aplicabilidad en cualquier empresa que lo necesite debido al detalle expuesto en cada una de las fases y procesos indicados en el documento.</p>			
<b>7. OBSERVACIONES</b>			



Juez Experto

**1. VALIDACIÓN DEL INSTRUMENTO**

**1.1. Instrumento de Validación No Experimental por Juicio de expertos**

<b>1. NOMBRE DEL JUEZ</b>		Alan Antonio Hernández López
<b>2.</b>	<b>PROFESIÓN</b>	Ing. Computación y Sistemas
	<b>ESPECIALIDAD</b>	Gestión de Proyectos
	<b>GRADO ACADÉMICO</b>	Bachiller
	<b>EXPERIENCIA PROFESIONAL (AÑOS)</b>	25
	<b>CARGO</b>	Gerente Proyectos
<b>TÍTULO DE LA INVESTIGACION:</b> Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima		
<b>3. DATOS DEL TESISISTA</b>		
<b>3.1</b>	<b>NOMBRES Y APELLIDOS</b>	Dennis Mauricio Avilés Odar
<b>3.2</b>	<b>PROGRAMA DE PREGRADO</b>	Ingeniería de Sistemas
<b>4. INSTRUMENTO EVALUADO</b>		Guía de buenas prácticas basadas en DevOps para una empresa consultora de Software
<b>5. OBJETIVOS DEL INSTRUMENTO</b>		<p><b>GENERAL</b> Validar el contenido de la Guía de buenas prácticas basadas den DevOps para una empresa consulta de Software.</p> <p><b>ESPECÍFICOS</b> Usar la validación de la guía para dar valor técnico a las conclusiones de uno de los objetivos específicos del proyecto de tesis titulado "Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima"</p>
A continuación, se le presentan los indicadores en forma de preguntas o propuestas para que Ud. los evalúe marcando con un aspa (x) en "A" si está de ACUERDO o en "D" si está en DESACUERDO, SI ESTÁ EN DESACUERDO POR FAVOR ESPECIFIQUE SUS SUGERENCIAS.		
<b>No</b>	<b>DETALLE DE LOS ITEMS DEL INSTRUMENTO</b>	
01	¿La Cultura DevOps expuesta en la guía es lo suficientemente entendible al usuario para su fácil comprensión y aplicabilidad?	A (5/ ) D ( ) SUGERENCIAS:

02	¿Está de acuerdo con el marco de aplicabilidad indicado en la guía para los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A (5/ ) SUGERENCIAS:	D ( )
03	¿Está de acuerdo con cada una de las fases indicadas en los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A (5/ ) SUGERENCIAS:	D ( )
04	¿Cree que las empresas consultoras de software tomen como guía el documento elaborado y lo apliquen en su día a día?	A (5/ ) SUGERENCIAS:	D ( )
<b>PROMEDIO OBTENIDO:</b>		A (5/ )	D ( ):
<b>6. COMENTARIOS GENERALES</b>			
<p>Actualmente la guía metodológica indicada, se encuentra implementada en la empresa consultora de Software</p>			
<b>7. OBSERVACIONES</b>			



Juez Experto

Alan Antonio Hernández López

DNI: 10280692

Innovación Tecnológica y Empresarial S.A.C

RUC: 20536312316

Gerente de Proyectos

Teléfono: 9918-95114

Email: a.hernandez@innovate.pe

## 1. VALIDACIÓN DEL INSTRUMENTO

### 1.1. Instrumento de Validación No Experimental por Juicio de expertos

<b>1. NOMBRE DEL JUEZ</b>		Marco Antonio Risco Mejía
<b>2.</b>	<b>PROFESIÓN</b>	Especialista en Arquitectura de las soluciones
	<b>ESPECIALIDAD</b>	Arquitectura y Tecnología de la Información
	<b>GRADO ACADÉMICO</b>	Ing. Electrónico
	<b>EXPERIENCIA PROFESIONAL (AÑOS)</b>	24
	<b>CARGO</b>	Gerente General
<b>TÍTULO DE LA INVESTIGACIÓN:</b> Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima		
<b>3. DATOS DEL TESISISTA</b>		
<b>3.1</b>	<b>NOMBRES Y APELLIDOS</b>	Dennis Mauricio Avilés Odar
<b>3.2</b>	<b>PROGRAMA DE PREGRADO</b>	Ingeniería de Sistemas
<b>4. INSTRUMENTO EVALUADO</b>		Guía de buenas prácticas basadas en DevOps para una empresa consultora de Software
<b>5. OBJETIVOS DEL INSTRUMENTO</b>		<p><b>GENERAL</b> Validar el contenido de la Guía de buenas prácticas basadas en DevOps para una empresa consultora de Software.</p> <p><b>ESPECÍFICOS</b> Usar la validación de la guía para dar valor técnico a las conclusiones de uno de los objetivos específicos del proyecto de tesis titulado "Aplicación de buenas prácticas para entornos de desarrollo de software basados en DevOps para mejorar la integración y despliegue de proyectos en una empresa consultora de la ciudad de Lima"</p>
A continuación, se le presentan los indicadores en forma de preguntas o propuestas para que Ud. los evalúe marcando con un aspa (x) en "A" si está de ACUERDO o en "D" si está en DESACUERDO, SI ESTÁ EN DESACUERDO POR FAVOR ESPECIFIQUE SUS SUGERENCIAS.		
<b>No</b>	<b>DETALLE DE LOS ITEMS DEL INSTRUMENTO</b>	
01	¿La Cultura DevOps expuesta en la guía es lo suficientemente entendible al usuario para su fácil comprensión y aplicabilidad?	A ( X )      D (   ) SUGERENCIAS:

02	¿Está de acuerdo con el marco de aplicabilidad indicado en la guía para los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A ( X ) SUGERENCIAS:	D ( )
03	¿Está de acuerdo con cada una de las fases indicadas en los procesos de desarrollo (Dev) y los procesos de operaciones (Ops)?	A ( X ) SUGERENCIAS:	D ( )
04	¿Cree que las empresas consultoras de software tomen como guía el documento elaborado y lo apliquen en su día a día?	A ( X ) SUGERENCIAS:	D ( )
<b>PROMEDIO OBTENIDO:</b>		A ( X )	D ( )
<b>6. COMENTARIOS GENERALES</b>			
<p>La Guía de Buenas prácticas elaboradas, basada en DevOps, nos permite como empresa tener un enfoque de trabajo enfatizando la entrega rápida, de manera incremental y continua de los productos de un proyecto de desarrollo de software. También permite que, los equipos de desarrollo compile, pruebe y publique software de forma más rápida y fiable, ya que nos permite aplicar principios y prácticas ágiles.</p>			
<b>7. OBSERVACIONES</b>			
Sin observaciones.			



**Marco A. Risco Mejía**  
**Gerente General**

**Juez Experto**

Anexo 5. Costos de implementación.

El presente cuadro muestra un aproximado de los costos de implementación utilizados en el presente proyecto de investigación.

Tabla 55. *Costos de implementación*

<b>Tipo</b>	<b>Giro de la empresa</b>	<b>Cantidad</b>	<b>Tiempo</b>	<b>Precio</b>
	Laptop Lenovo Intel i5 10 Generacion, 16 de ram, 512 SSD	1 unidad	3 años	S/.2400.00
<b>Hardware</b>	Mouse inalámbrico Logitech M535	1 unidad	3 años	S/.110.00
	Teclado inalámbrico Logitech K380	1 unidad	3 años	S/.250.00
	Visual Studio 2019 Community	8 usuarios	Indefinido	Gratuito
<b>Software</b>	Azure DevOps - Plan Básico	8 licencias	Indefinido	\$8.00
	Azure Blob - Bucket	1 licencia	Gb por uso	\$ 0.018 x GB
	Docker	1 instalador	Indefinido	Gratuito
<b>Personal</b>	Desarrollador	8 personas	1 mes	S/.3000.00

Fuente: Elaboración propia.

Anexo 6

Guía de Buenas prácticas basadas DevOps para una empresa consultora de  
Software

**GUÍA DE BUENAS PRÁCTICAS  
BASADA EN DEVOPS PARA UNA  
EMPRESA CONSULTORA DE  
SOFTWARE**



## INDICE

<i>Capítulo I. Aspectos Generales</i> .....	133
1. Introducción.....	133
2. Objetivo.....	133
3. Alcance. ....	134
4. Responsables. ....	134
5. Definiciones.....	134
<i>Capítulo II. Cultura DevOps</i> .....	135
1. Colaboración, visibilidad y alineamiento. ....	135
2. Responsabilidad asumida por el equipo de trabajo. ....	135
3. Agilidad en los procesos.....	136
4. Aprendizaje continuo.....	136
5. Integración y distribución continua (CI/CD) .....	136
<i>Capítulo III. Marco de aplicabilidad de DevOps</i> .....	137
1. Procesos de Desarrollo (Dev) .....	137
2. Procesos de Operaciones (Ops) .....	146

## **Capítulo I.**

### **Aspectos Generales**

#### **1. Introducción.**

La empresa consultora, determina la necesidad e importancia que tiene eliminar las barreras existentes en el desarrollo de software y estandarizar sus procesos mediante la integración, visibilidad, colaboración y la transparencia tanto entre equipos desarrolladores (Dev) y los de infraestructura (Ops). De esta forma, se reducirán los tiempos en el desarrollo y puesta en marcha de cualquier proyecto y permitirá un mayor flujo de productividad empresarial, garantizando seguridad en los procesos y mejor calidad en los entregables.

La guía de buenas prácticas DevOps determina los lineamientos, fases y los componentes que se deben tener en cuenta para la aplicabilidad de la metodología en los proyectos que desarrolla la empresa consultora, basado en un marco de trabajo que abarca la automatización de los procesos que comprende el ciclo de vida de desarrollo de Software (desarrollo, integración, despliegue), la cual debe ser atendida a través de una cultura organizativa y no únicamente por procesos o herramientas.

Dentro de este marco de trabajo se da mayor relevancia a distintos aspectos en el entorno de trabajo: buena intercomunicación y colaboración entre los diferentes equipos, mejora en la documentación de los procesos y aprender de errores anteriores para estar abiertos a nuevos pensamientos e ideas y de esta forma entregar un producto de calidad.

#### **2. Objetivo.**

Establecer una cultura organizativa a través de una guía de buenas prácticas basada en DevOps que abarque los lineamientos, las etapas y los principales actores fundamentales en la automatización de los proyectos que desarrolle la empresa

consultora a diferentes clientes externos, pensando en la resolución de problemas en periodos de tiempo más cortos, con menor dificultad, mejor integración, escalabilidad y disponibilidad y de esta forma obtener satisfacción y fidelización del cliente.

### **3. Alcance.**

La guía de buenas prácticas basadas en DevOps, se aplica a todos los procesos de desarrollo de software, en especial a la integración y despliegue continuo, mediante el establecimiento de esta cultura organizativa que facilite la implementación y la puesta en marcha de los proyectos que se desarrollan en la empresa consultora, mejorando la comunicación en los equipos, optimizando las correcciones y centralizando el proyecto, de esta forma el despliegue será exitoso y por ende el usuario final recibirá la calidad que él requiere.

### **4. Responsables.**

Equipo de trabajo por cada proyecto que se realice (Líder técnico, desarrolladores, personal de calidad)

### **5. Definiciones.**

- **Software:** Es un término que hace referencia a los distintos tipos de programas, aplicaciones y sistemas de información.
- **Hardware:** Son todos los componentes físicos de la computadora y sus periféricos.
- **Tecnología:** Conjunto ordenado de procesos y conocimientos que permite producir bienes y servicios, teniendo en cuenta la ciencia, aspectos sociales, culturales y económicos involucrados y que permiten satisfacer las necesidades de la sociedad.

- **DevOps:** Es una cultura organizativa, unión de procesos, personas y tecnología que ofrece valor y calidad a los usuarios finales de manera constante.
- **Entrega continua:** enfoque que tiene la ingeniería de software, en la cual los equipos de trabajo producen software en cortos ciclos y en periodos de tiempo más cortos.
- **La Integración Continua:** consiste en que los desarrolladores versionan sus cambios de forma periódica en un repositorio compartido con un sistema de control de versiones.
- **Usuario:** Es aquella persona que utiliza una computadora, servicio o cualquier sistema informático.

## Capítulo II.

### Cultura DevOps

Para garantizar la adopción de manera exitosa de DevOps en la empresa consultora es necesario seguir los siguientes principios fundamentales que deben contribuir los equipos de trabajo.

#### 1. Colaboración, visibilidad y alineamiento.

Cuando existe la colaboración entre los equipos de desarrollo y de operaciones podemos hablar de una buena cultura DevOps. El trabajo no se puede realizar de manera aislada, estos deben compartir entre sí sus procesos de integración y despliegue, sus preocupaciones o sus prioridades. Estos equipos deben planear juntos el trabajo y de esta forma se alinearán sus objetivos e indicadores de éxito en función con el negocio.

#### 2. Responsabilidad asumida por el equipo de trabajo.

A medida que los equipos de trabajo planean y alinean su forma de trabajo; participan y asumen responsabilidad en más etapas correspondientes al ciclo de vida, no solo las que

son principales y fundamentales en sus roles. Es decir, los desarrolladores asumen responsabilidad no solo por la calidad e innovación establecida en esa fase, sino también por la estabilidad de todos los cambios que se realicen y por el rendimiento que este genere. De igual forma, los ingenieros de confiabilidad de sitio (Site Reliability Engineering - SRE) deben asegurar de incluir la seguridad, cumplimiento normativo y la gobernanza en las fases correspondientes al planeamiento y desarrollo.

### **3. Agilidad en los procesos.**

Los equipos de trabajo que utilizan la cultura DevOps mantienen y priorizan la agilidad porque lanzan versiones de software de los diferentes proyectos en periodos cortos. Al tener despliegues en tiempos más cortos, se facilita la planeación y administración de riesgos, debido a que el progreso es incremental, lo que reduce que el impacto de estabilidad en el sistema sea negativo.

### **4. Aprendizaje continuo.**

El aprendizaje avanza a lo largo del tiempo, es por eso por lo que los equipos de desarrollo basados en la cultura Devops deben tener una mentalidad constante de crecimiento, que permita responder de manera rápida a los errores e incorporan lo que aprenden a sus diversos procesos, de modo que continuamente mejoran y de esta forma aumenten la satisfacción del cliente y agilizar la capacidad de adaptación y agilizar la innovación. Siempre habrá espacio para crecer y evaluar nuevas y mejores prácticas.

### **5. Integración y distribución continua (CI/CD)**

Utilizando la automatización durante las diversas etapas del desarrollo de aplicaciones, el método CI/CD distribuirá las aplicaciones a los clientes con frecuencia. Los tres conceptos principales que se le asignan son integración, distribución e implementación continua.

Puesto que se refiere a una solución para los problemas que se pueden generar en la integración del nuevo código para los equipos de desarrollo y de operaciones.

En particular, este proceso de integración y distribución continua incorpora automatización y supervisión continua durante todo el ciclo de vida de las aplicaciones, desde las fases de integración y prueba hasta las fases de distribución e implementación.

### Capítulo III.

#### Marco de aplicabilidad de DevOps

DevOps ofrece un marco que reúne a dos principales procesos: Desarrollo (Dev) u Operaciones (Ops), cada uno tiene etapas que deberá desarrollar. A continuación, se muestra una imagen de los procesos principales y su detalle:

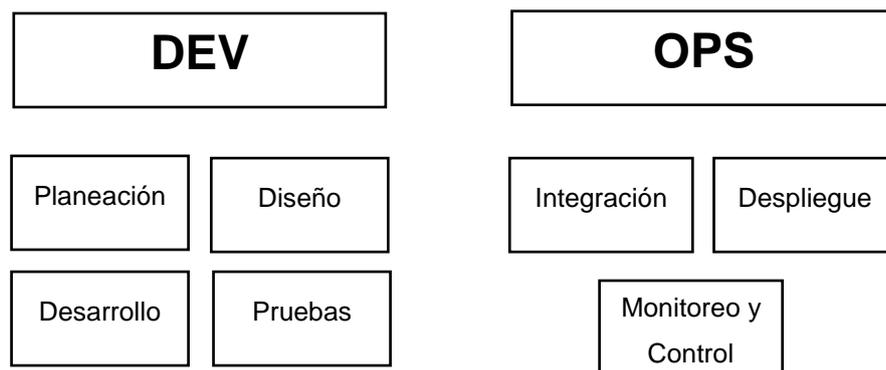


Figura 34. Esquema de procesos principales y fases de la empresa consultora.

Fuente: Elaboración propia.

Después de esa clasificación, se especifican los lineamientos necesarios que deberán considerar en cada etapa de los procesos de desarrollo y operaciones:

#### 1. Procesos de Desarrollo (Dev)

Estos procedimientos se deben ejecutar de la mano con la metodología ágil SCRUM. Los involucrados principales son: Product Owner, Scrum Máster, Equipo de Desarrollo, Equipo

de Operaciones. Cada uno de ellos tiene responsabilidades distintas y específicas de acuerdo con lo que se identifique en la etapa de planeación de cada sprint.

Estas actividades una vez identificadas y definidas de ser consideradas en el backlog de desarrollo, en las cuales están definidos los tiempos por cada historia de usuario o caso de uso, horas estimadas de desarrollo por cada actividad y los responsables consignados.

### ***Fase de Planeación***

En primera instancia se deberán reunir los equipos de trabajo (desarrollo y operaciones) con el product owner y de esta forma evaluar las tecnologías que se usarán en los procesos principales y sus respectivas etapas. Los elementos básicos e indispensables que se deberán analizar son los siguientes:

- Capacidades de integración
- Licenciamientos
- Tiempos de implementación y complejidad
- Fortalezas y limitaciones
- Equipo humano que participará por proyecto
- Compatibilidades con el software base de ser necesario: bases de datos, el lenguaje de programación y componentes propios en caso lo requiera el proyecto.
- Compatibilidad con los recursos de la infraestructura
- Seguridad y disponibilidad

Una vez analizados esos aspectos, será necesario crear matrices de evaluación que permitan analizar y contrastar de manera definida cada una de las posibles tecnologías y a partir de ello definir el stack tecnológico que se deberá utilizar en el proyecto.

Luego de definido el stack tecnológico, serán seleccionados los actores o responsables en cada etapa y a partir de ello se elaborará el cronograma de implementación con todos los aspectos analizados.

### ***Gestión de requerimientos***

Esta etapa se enfoca en identificar, controlar y seguir los requerimientos y sus cambios en cualquier situación. Para realizar esta etapa, se deben considerar los siguientes aspectos:

- Gestionar el análisis de los requisitos
- Obtener la aprobación de todos los participantes del proyecto
- Gestionar los cambios, identificar los tiempos y priorizar. La gestión de los requerimientos es un proceso que se desarrollará en todo el ciclo de vida del software.
- Construcción del prototipo con el modelo visual del sistema que se implementará, con la finalidad de observar gráficamente una visión total del proyecto a ejecutar.

La construcción de ese prototipo la debe realizar el equipo conformado por el Product Owner, analistas de los requerimientos y los encargados del diseño de las interfaces.

De forma paralela, el grupo de analistas funcionales deberán recopilar los requisitos y proceder a elaborar los casos de uso. El resultado de esto es la entrega de las historias de usuario o casos de uso de acuerdo con el modelo estandarizado que utiliza la empresa consultora. Cada caso de uso debe contener el siguiente detalle:

- Diagrama de caso de uso UML
- Requerimientos funcionales
- Requerimientos no funcionales
- Rol al que pertenece el caso
- Tabla de variables de entrada indicando tipos de datos
- Tabla de variables de salida indicando tipos de datos
- Validaciones
- Especificación del proceso principal
- Especificación de procesos alternativos

Esta fase se hace de acuerdo con las buenas prácticas DevOps y la metodología ágil Scrum, en donde se definen los Sprint que varían de acuerdo con la cantidad de casos y la complejidad de estos. En el backlog de gestión de requisitos se deben considerar los

tiempos aproximados y responsables dependiendo de la experiencia del equipo y cantidad de desarrolladores.

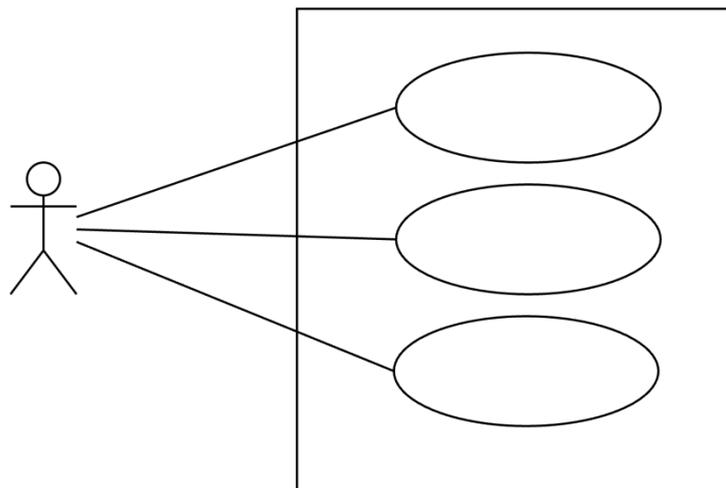
Las historias o casos de usuario deben cumplir con las siguientes características:

- Definiciones claras y precisas para que el equipo de desarrollo pueda estimar de manera precisa el tiempo que va a tomar el desarrollo de cada caso o historia.
- Contar con todas las especificaciones funcionales a implementar.
- Permitir elaborar casos de prueba con casos de uso como base .
- Permitir al equipo de operaciones entender el alcance de las funcionalidades.
- Permitir estimar la infraestructura necesaria para soportar la solución desarrollada.
- Permitir establecer equipos de mantenimiento que conozcan funcionalidades.

### ***Fase de Diseño técnico***

En esta parte se elaborará el documento de arquitectura de software y de esta forma se culminará el detalle de aspectos técnicos. Tal documento debe tener mínimamente lo siguiente:

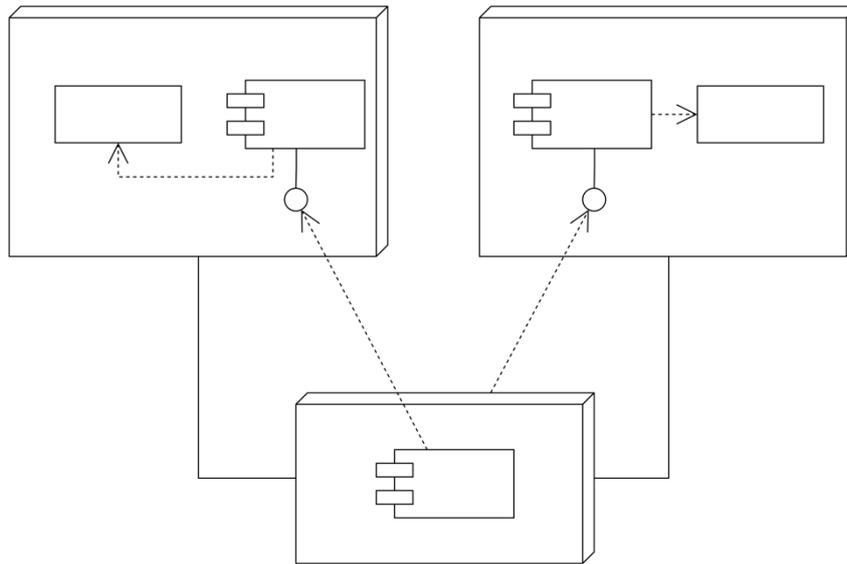
- Vistas de casos de uso.



*Figura 35* Ejemplo genérico de vista de casos de uso.

Fuente: Elaboración propia

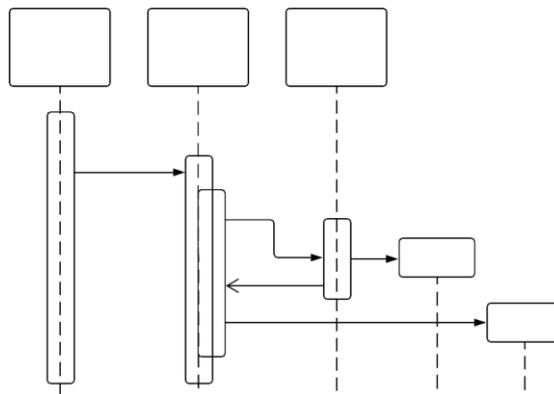
- Diagrama de componentes.



*Figura 36* Ejemplo genérico de diagrama de componentes.

Fuente: Elaboración propia

- Diagramas de secuencia.



*Figura 37* Ejemplo genérico de diagrama de secuencia.

Fuente: Elaboración propia

- Diagrama de clases.

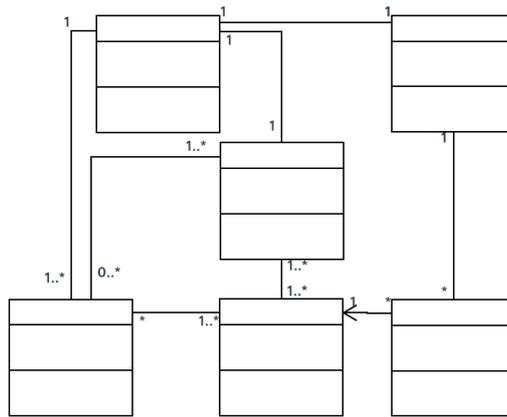


Figura 38 Ejemplo genérico de diagrama de clases.

Fuente: Elaboración propia

- Diagrama de subsistemas.

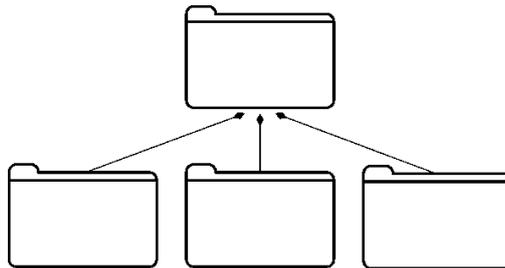


Figura 39 Ejemplo genérico de diagrama de subsistemas.

Fuente: Elaboración propia

- Vista de implementación

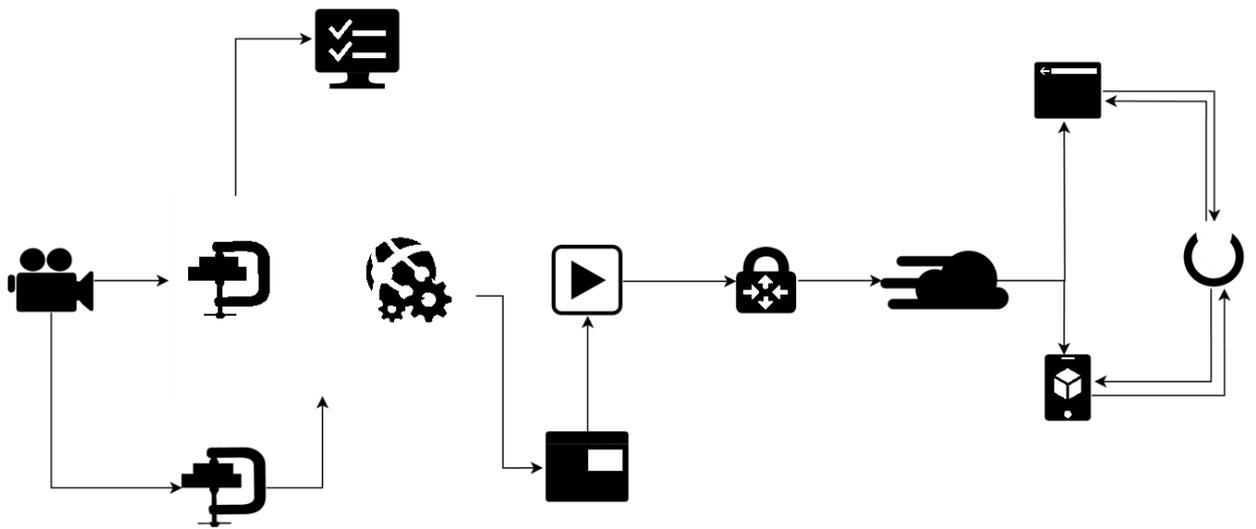


Figura 40 Ejemplo genérico de vista de implementación.

Fuente: Elaboración propia

- Diagrama de procesos

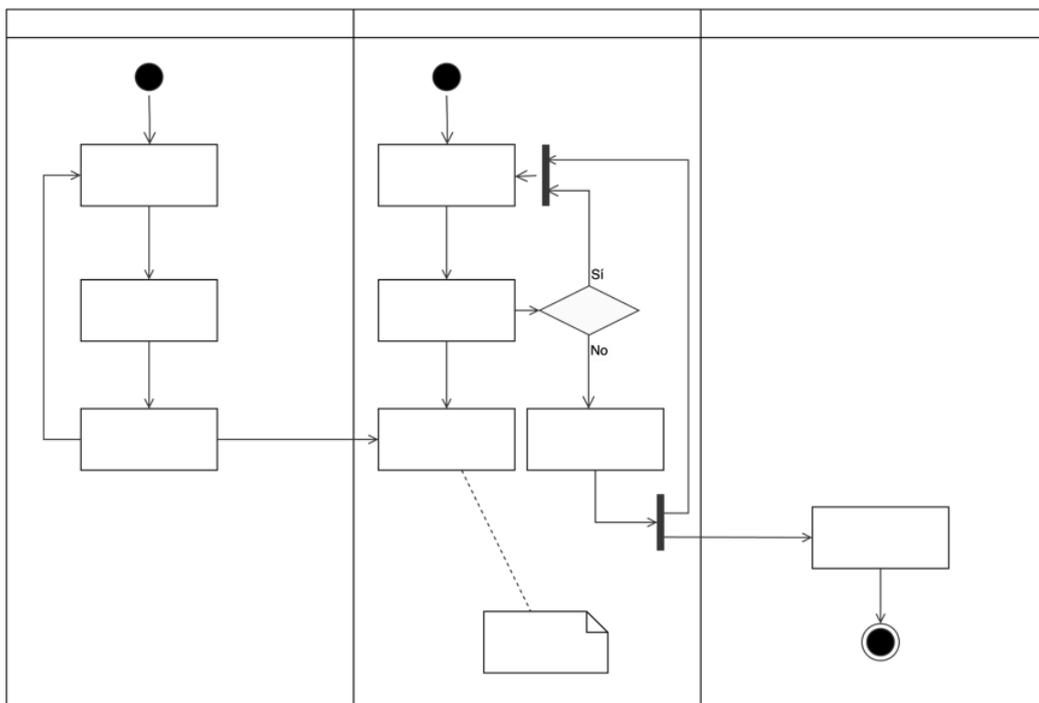


Figura 41 Ejemplo genérico de digrama de procesos.

Fuente: Elaboración propia

- Diagrama de despliegue

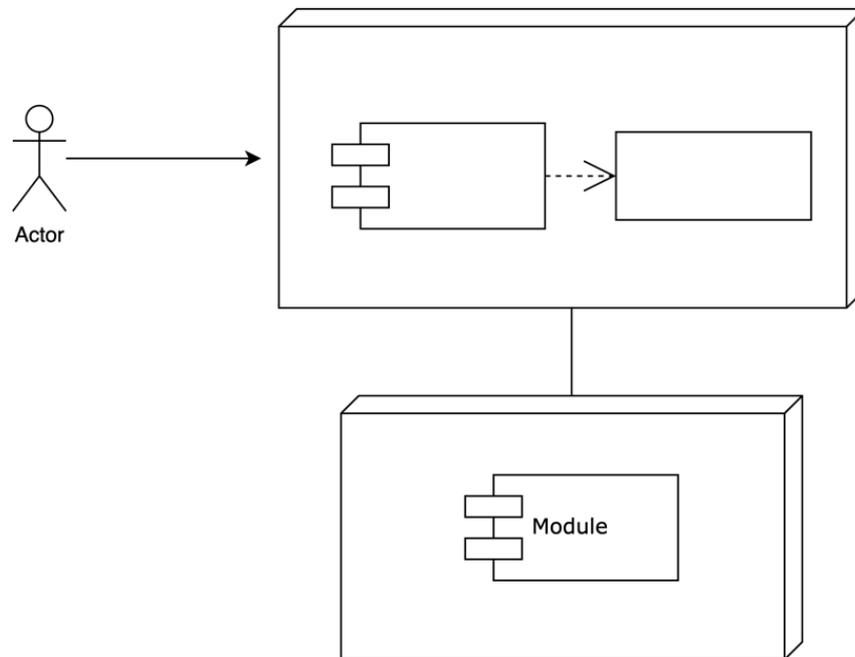


Figura 42 Ejemplo genérico de digrama de despliegue.

Fuente: Elaboración propia

- Diagrama de modelo entidad relación

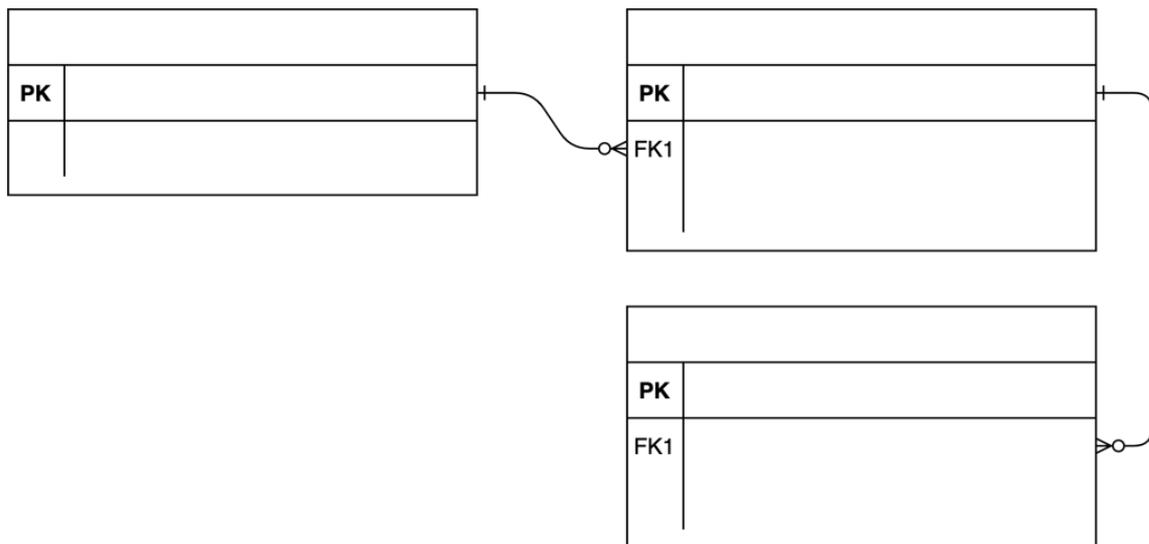


Figura 43 Ejemplo genérico de digrama de modelo entidad relación.

Fuente: Elaboración propia

- Matriz de trazabilidad con módulos y funcionales.

Matriz de trazabilidad							
Identificación	Identificación asociada	Descripción del requisito	Objetivos	Entregables del EDT	Diseño del producto	Desarrollo	Casos de prueba
001	1.0						
	1.1						
002	2.0						
	2.1						

Figura 44 Ejemplo genérico de matriz de trazabilidad con módulos y funcionales.

Fuente: Elaboración propia

### **Fase de Desarrollo**

En esta etapa se ejecutará la funcionalidad escrita en las historias de usuario o casos de uso. En base a ello, se deberá seguir los siguientes aspectos:

- Para codificar y programar de forma ágil y organizada se debe tener un IDE Entorno de desarrollo y el respectivo manual de cómo se instala y configura.
- Para almacenar y versionar todo el código fuente producido, se deberá contar con un repositorio de código fuente, como GIT o Azure Repos. Con esto se garantiza poder desarrollar actualizaciones y mantenimientos sobre los componentes de software desarrollados.
- Cumplir con los estándares de codificación según el lenguaje de programación seleccionado.
- Cumplir con los estándares de programación de bases de datos.
- Validar la integración de los componentes

Luego de ello se deberá compilar y construir el código fuente y para eso se deberá:

- Compilar los componentes de software en caso de ser un lenguaje que lo requiera.
- Ejecutar técnicas de perfilado y afinamiento o por su referencia en inglés, profiling and performance tuning, que permitan garantizar que los componentes de software

desarrollados en la fase de codificación se encuentren afinados para soportar el rendimiento esperado en producción, se deben afinar los sistemas operativos, servidores de aplicación, bases de datos y conexiones de red.

- Construcción de pruebas unitarias y de integración, para poder usarlas a modo de regresión siguiendo las buenas prácticas de Integración continua.

### ***Fase de Pruebas***

Se deberán desarrollar los casos de pruebas de los componentes de lo implementado, están deberán ser:

- Pruebas funcionales, integración, seguridad y carga a través de herramientas automatizadas.
- Se deben ejecutar pruebas de APIs, integraciones y servicios.
- Se debe crear pruebas de integración continua, para resolver pequeños bugs antes de la entrada a producción, hace una verificación de los próximos cambios y se encarga de la automatización de los tests creados.
- Uno de los aspectos importantes en DevOps es garantizar que el despliegue de nuevas automatizaciones se ejecute de forma continua, a partir de ello, surge la necesidad de hacer pruebas frecuentemente.

## **2. Procesos de Operaciones (Ops)**

### ***Fase de integración continua***

Una vez finalizado el desarrollo de un conjunto de requerimientos o funcionalidades que deben ser incluidas en el release del producto y se acerca la fecha de su liberación, será necesario e indispensable fusionar todas las funcionalidades en la rama principal o rama máster del proyecto. Cuando es momento del merge (fusión), muchas veces es necesario seleccionar un día o un rango de días para que todos los involucrados en la automatización realicen el commit, debido a que es una tarea ardua y complicada y que puede ocasionar varios errores en la rama principal.

A partir de esa necesidad surge en concepto de integración continua (CI), que interviene para ayudar en esta parte del proceso. CI es una práctica del desarrollo del software en la que los equipos de trabajo integran sus avances con frecuencia. Cada colaborador debe integrar sus automatizaciones al menos 1 vez al día, lo que equivale a que al término de ese día se encuentren múltiples integraciones. Es necesario que esta compilación se realice de manera automatizada y que incluya pruebas o test y de esta se identifiquen errores de manera más rápida y eficiente.

Como consecuencia, el utilizar esta práctica, permitirá reducir el número de problemas de integración, mejorar la productividad del equipo de desarrollo, entregar valor al cliente de manera rápida y detectar oportunamente los errores y fallos. Como punto fundamental está la transparencia del proceso, porque como todos los integrantes del equipo utilizan el mismo repositorio de código, se conocerá realmente el estado en el que se encuentra el proyecto. No hay necesidad de esperar que transcurra un gran rango de tiempo para conocer el avance y además las reuniones se acortarán y solo se hablarán de temas precisos.

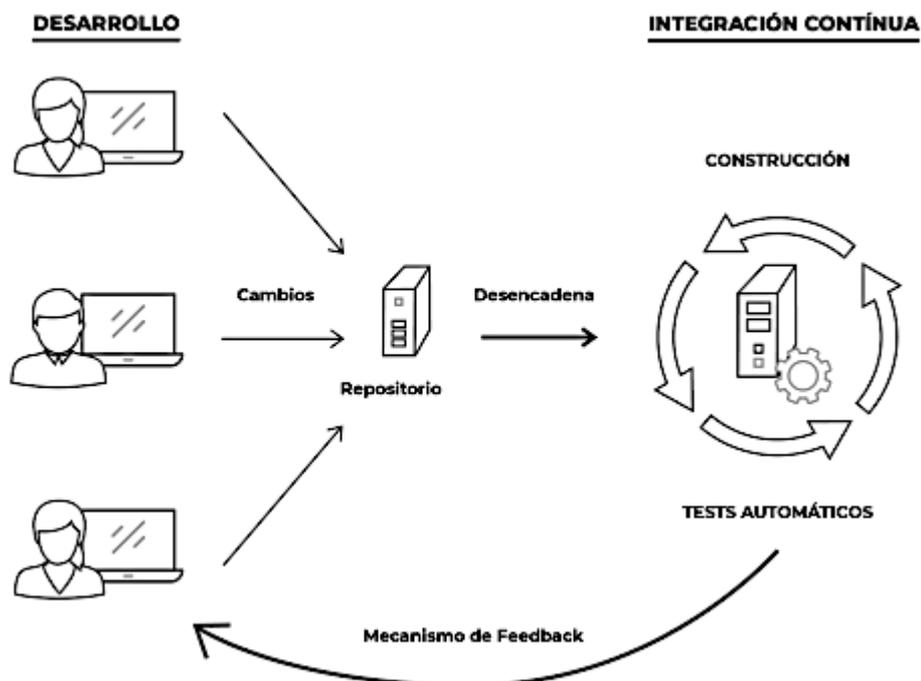


Figura 45. Integración Continua para la empresa Consultora.

Fuente: Elaboración propia.

### ***Fase de despliegue continuo***

La entrega o despliegue continuo es la capacidad de trasladar los cambios que incluyan los de configuración, nuevas funcionalidades o algunas correcciones de errores al entorno de producción para que llegue a usuarios finales de manera segura, rápida, continua y lo más automática posible.

Se deberán hacer despliegues rutinarios, predecibles y automáticos en cualquier ocasión en un sistema a gran escala o distribuido. De esta forma, se asegurará que a pesar de que los equipos de trabajo se encuentren trabajando, el código estará siempre en un estado desplegable. Es decir, cuando se suba algún cambio, no romperá la funcionalidad ya existente y no se podrá filtrar algún error que pueda causar problemas a gran escala al usuario final.

Siempre se debe tener presente que no se pueden realizar despliegues dejando de lado la calidad y estabilidad de los proyectos. Al automatizar tales procesos, el proyecto funcionará igual y seguirá brindando la calidad deseada.

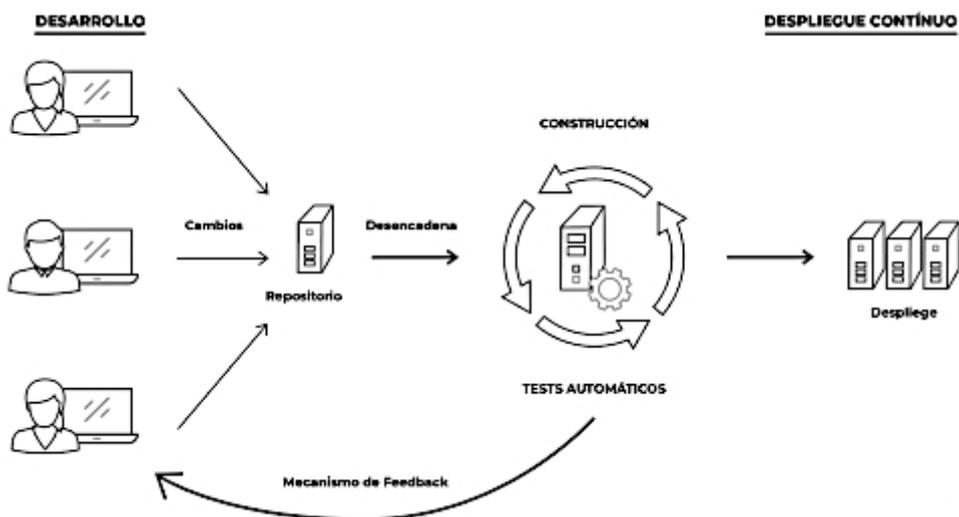


Figura 46. Despliegue continuo para la empresa Consultora.

Fuente: Elaboración propia.

### ***Fase de monitoreo y control***

En esta fase se deberán controlar todos los componentes tanto hardware como software y para ello se deberá seguir con lo siguiente:

- Establecer herramientas de monitoreo sobre componentes del sistema a implementar.
- Implementar plataformas de trazabilidad para conocer el estado actual del sistema.
- Se debe contar con sistemas de monitorización frecuente que permitan identificar por medio de alertas, algunas cuestiones de calidad de operaciones que puedan afectar en entorno de producción.

### ***Gestión de la configuración***

Este componente debe estar unida con las configuraciones de la Base de Datos de Gestión de la Configuración (CMDB), como el repositorio de información. Esa base de datos será la única fuente para la información relativa a la configuración de los componentes del sistema de la información.

### ***Aprendizaje validado***

A lo largo del ciclo de vida, se recopilan datos de telemetría y comentarios con el fin de tomar decisiones informadas para el siguiente ciclo. Esta información se conoce como aprendizaje validado. El aprendizaje validado ayuda a proporcionar conclusiones sobre nuevas formas de reducir el tiempo en cada ciclo. Los datos recopilados se usan para buscar formas de aumentar la automatización, mejorar los procesos y preparar una implementación más rápida y eficiente.

NOMBRE DEL TRABAJO	AUTOR
<b>AVILES_ODAR_DENNIS_MAURO-urni tin.docx</b>	<b>Dennis Aviles odor</b>

RECuento DE PALABRAS	RECuento DE CARACTERES
<b>17859 Words</b>	<b>94097 Characters</b>

RECuento DE PÁGINAS	TAMAÑO DEL ARCHIVO
<b>83 Pages</b>	<b>1.4MB</b>

FECHA DE ENTREGA	FECHA DEL INFORME
<b>Mar 18, 2024 9:27 AM GMT-5</b>	<b>Mar 18, 2024 9:28 AM GMT-5</b>

● **13% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 11% Base de datos de Internet
- Base de datos de Crossref
- 5% Base de datos de trabajos entregados
- 2% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● **Excluir del Reporte de Similitud**

- Material bibliográfico
- Coincidencia baja (menos de 8 palabras)
- Material citado