

**UNIVERSIDAD MAYOR, REAL Y PONTIFICIA DE SAN FRANCISCO XAVIER DE  
CHUQUISACA**

**VICERRECTORADO**

**CENTRO DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN  
FACULTAD DE CIENCIAS Y TECNOLOGÍA**



**“DevOps en el Delivery de Comida a Domicilio en Bolivia: Un Modelo para la  
Eficiencia y la Satisfacción del Cliente”**

**TRABAJO EN OPCIÓN A DIPLOMADO EN DEVELOPEMENT**

**OPERATIONS “DEVOPS” V.1.**

**AUTOR: Huanca Colque Wilzor Tito**

**SUCRE-BOLIVIA**

**2024**

## **CESIÓN DE DERECHOS**

Al presentar este trabajo como requisito previo para la obtención del Diploma en Development Operations "DEVOPS" V.1. de la Universidad Mayor, Real y Pontificia de San Francisco Xavier de Chuquisaca, autorizo al Centro de Estudios de Posgrado e Investigación o a la Biblioteca de la Universidad, para que se haga de este trabajo un documento disponible para su lectura según normas de la Universidad

También cedo a la Universidad Mayor, Real y Pontificia de San Francisco Xavier de Chuquisaca, los derechos de publicación de este trabajo o parte de él, manteniendo mis derechos de autor hasta un periodo de 30 meses posterior a su aprobación.

Wilzor Tito Huanca Colque

.....

FIRMA:

## **DEDICATORIA Y AGRADECIMIENTOS**

A mis queridos padres, quienes han sido mi roca y mi faro a lo largo de esta travesía. Su generosidad, tanto en el apoyo emocional como en el respaldo económico, han sido el motor que impulsó cada uno de mis pasos. Sin su inquebrantable confianza y sacrificio, este logro no habría sido posible.

A mis hermanos, cuyo amor incondicional y complicidad han sido un regalo invaluable en mi vida. Sus palabras de aliento, sus consejos sinceros y su apoyo constante han sido un faro en los momentos de adversidad y una fuente inagotable de alegría en los momentos de celebración.

A mis amigos de la carrera, quienes se convirtieron en mi segunda familia. Su compañerismo, colaboración y solidaridad fueron un pilar fundamental en mi crecimiento académico y personal. Además, la competitividad sana que compartimos entre nosotros nos motivó a superarnos día a día, impulsándonos a alcanzar nuestros objetivos con determinación. Cada conversación, cada risa compartida, alimentaron mi espíritu y fortalecieron mi determinación, en un ambiente de sana competencia que nos desafió a ser mejores versiones de nosotros mismos.

A los docentes, cuya sabiduría, paciencia y dedicación fueron la luz que iluminó mi camino académico. Sus enseñanzas trascendieron las aulas y dejaron una huella imborrable en mi formación profesional.

## RESUMEN

Esta monografía se sumerge en el universo de las prácticas DevOps aplicadas específicamente a las empresas de entrega de comida a domicilio en Bolivia. Se aborda un estudio de caso detallado sobre PedidosYa, una empresa líder en América Latina, para comprender cómo la implementación de DevOps y el uso estratégico de tecnologías en la nube han contribuido a su éxito operativo. Se destaca el papel crucial de herramientas como Git, Jenkins, Docker y Kubernetes en diferentes etapas del ciclo de vida del desarrollo de software, desde la planificación hasta el despliegue y el monitoreo. La comparación con PatioService, una empresa emergente en Bolivia, arroja luz sobre las diferencias en la adopción de herramientas DevOps y sus efectos en la experiencia del usuario. Se profundiza en la situación socioeconómica de Bolivia, especialmente en el contexto de la pandemia de COVID-19, que ha impulsado el crecimiento del mercado de entregas de comida a domicilio. Se resalta la importancia de adaptar las estrategias de PedidosYa para enfrentar la competencia de empresas locales emergentes como PatioService. Además, se propone un diseño de implementación de DevOps adaptado al entorno boliviano, con énfasis en la selección de herramientas que optimicen la eficiencia operativa y minimicen los costos iniciales, con sugerencias para escalar a servicios en la nube a medida que la empresa crece. Este estudio proporciona una visión integral de cómo las prácticas DevOps pueden ser implementadas en el contexto específico de las empresas de delivery en Bolivia, con el objetivo de mejorar la eficiencia, la calidad del producto y la satisfacción del cliente.

## INDICE

1. Antecedentes y Justificación. ....	8
Antecedentes .....	8
Justificación .....	9
2. Situación Problemática .....	11
3. Formulación del Problema de Investigación o Pregunta Científica. ....	12
4. Objetivo General.....	12
5. Objetivos Específicos. ....	12
6. Diseño Metodológico. ....	12
6.1 Tipo de Investigación.....	12
6.1.1 Alcance de la Investigación .....	13
6.2 Métodos .....	13
6.2.1 Métodos teóricos.....	13
6.2.2 Método Análisis –Síntesis. ....	13
6.2.3 Métodos empíricos.....	13
6.3 Técnicas. ....	13
6.4 Procedimientos e instrumentos de investigación .....	14
CAPITULO I.....	15
MARCO TEORICO Y CONTEXTUAL .....	15
1.1 Marco teórico.....	15
1.1.1 Principales teorías que abordan la temática de DevOps .....	15
1.1.5 Infraestructura Tecnológica en Servicios de Entrega de Comida a Domicilio.....	19
1.2 Marco Contextual .....	20
CAPITULO II.....	23
DIAGNOSTICO.....	23
2.1 Introducción.....	23

2.1.1 Procesamiento y Análisis de Datos .....	23
Estudio de caso de éxito PedidosYa .....	23
2.1.2 Comparación de caso de éxito frente a otra boliviana .....	26
2.1.2 Tabulación y Codificación de datos.....	28
Comparación de herramientas de uso en cada etapa del ciclo de vida DevOps.....	28
2.1.3 Análisis y Discusión de Resultados.....	44
2.2 Conclusiones y Recomendaciones.....	46
2.2.1 Conclusiones .....	46
2.2.2 Recomendaciones .....	47
BIBLIOGRAFÍA .....	49

## ÍNDICE DE TABLAS

Tabla 1 Comparación de herramientas en la etapa de Planificación .....	28
Tabla 2 Comparación de herramientas en la etapa de Desarrollo .....	30
Tabla 3 Comparación de herramientas en la etapa de Construcción.....	32
Tabla 4 Comparación de herramientas en la etapa de Pruebas.....	35
Tabla 5 Comparación de herramientas en la etapa de Empaquetado .....	37
Tabla 6 Comparación de herramientas en la etapa de Monitoreo .....	42

## **1. Antecedentes y Justificación.**

### **Antecedentes**

En la actualidad, la industria de alimentaria, en particular las cadenas de restaurantes de comida, se enfrenta a una serie de desafíos operativos que afectan su eficiencia y capacidad para satisfacer las demandas de los clientes en un mercado altamente competitivo. Estos desafíos incluyen la necesidad de gestionar grandes volúmenes de pedidos, mantener la consistencia en la calidad de los productos y servicios, y adaptarse rápidamente a las cambiantes preferencias del consumidor.

La metodología DevOps ha surgido como una solución potencial para abordar estos desafíos al enfocarse en la automatización de procesos, la integración continua y la entrega continua. Originada en el ámbito del desarrollo de software, DevOps se centra en la colaboración estrecha entre los equipos de desarrollo (Dev) y operaciones (Ops) para agilizar los procesos de desarrollo, implementación y entrega de aplicaciones. Si bien DevOps ha sido ampliamente adoptado en la industria de la tecnología de la información, su aplicación en sectores como la restauración aún no se ha explorado completamente.

En la última década, el sector de restaurantes en Bolivia ha experimentado un crecimiento exponencial, como lo destaca un artículo de Micaela Villa publicado en el periódico "La Razón" el 6 de mayo de 2018. Según datos del Servicio de Impuestos Nacionales (SIN), procesados por la Unidad de Análisis y Estudios Fiscales del Ministerio de Economía y Finanzas Públicas, las ventas y los servicios facturados en restaurantes del país se han multiplicado por cuatro, pasando de \$us 174 millones en 2008 a \$us 696 millones el año pasado. Este incremento significativo del 300% refleja el dinamismo y la expansión del sector gastronómico boliviano en dicho período. Además, se observa una alta demanda de platos que incluyen carne de pollo y de res, así como opciones de comida rápida y parrillas, según lo reportado por los entrevistados (Villa, 2018).

Además, según PedidosYa, una de las empresas líderes en tecnología de quick-commerce en Latinoamérica, cerró el año 2022 con un crecimiento de más del 45%, con el quick-commerce casi duplicando su facturación respecto al año anterior. Este crecimiento evidencia la consolidación de la tendencia de hacer compras de necesidad inmediata para recibirlas en minutos, que se ha fortalecido significativamente en 2022. Estas cifras subrayan

la importancia y el impacto creciente del quick-commerce en el panorama del consumo en Bolivia (Salas, 2022).

Según un informe de Kantar, se ha observado que el consumo de alimentos fuera de casa ha vuelto a los niveles prepandémicos en todo el mundo, a pesar de la inflación disparada. Este informe destaca que los consumidores eligen salir a comer en un restaurante la mitad del tiempo que no preparan comida en casa. Este cambio de hábitos se ha extendido a todos los continentes, impulsado por la comodidad, los estilos de vida y el aumento de los costos. Contra todo pronóstico, mientras el sector del comercio de alimentos obtiene el 10% del crecimiento de su cuota de valor de la subida de precios, el sector Foodservice ha demostrado una agilidad notable al ofrecer comodidad y precios asequibles. Estos hallazgos sugieren la importancia de explorar nuevas estrategias, como la implementación de modelos DevOps, para abordar los desafíos operativos en la industria de la restauración y adaptarse a las cambiantes demandas del mercado (Kantar, 2023).

## **Justificación**

La adopción de prácticas DevOps en el sector de entrega de comida a domicilio en Bolivia es una iniciativa estratégica que busca mejorar la eficiencia operativa y la satisfacción del cliente en un mercado altamente competitivo y en constante evolución. La implementación de un modelo de DevOps en este contexto ofrece una serie de beneficios significativos que pueden abordar los desafíos específicos que enfrentan las empresas de entrega de comida a domicilio en Bolivia.

- **Eficiencia operativa mejorada:** DevOps permite la automatización de procesos clave en la cadena de entrega de comida a domicilio, desde la gestión de pedidos hasta la preparación y entrega de alimentos. Al optimizar estos procesos, las empresas pueden reducir los tiempos de espera, minimizar los errores operativos y mejorar la eficiencia en general.
- **Mayor agilidad y adaptabilidad:** En un mercado dinámico como el de la entrega de comida a domicilio en Bolivia, la capacidad de adaptarse rápidamente a las demandas cambiantes del cliente es esencial. DevOps facilita la entrega continua de nuevas características y mejoras, lo que permite a las empresas responder ágilmente a las tendencias del mercado y a las preferencias del consumidor.

- **Consistencia en la experiencia del cliente:** La implementación de prácticas DevOps garantiza una mayor consistencia en la experiencia del cliente, tanto en términos de calidad de los alimentos como en la velocidad y precisión del servicio de entrega. Esto contribuye a fortalecer la lealtad del cliente y a mejorar la reputación de la marca en un mercado altamente competitivo.
- **Optimización de recursos:** DevOps facilita una gestión más eficiente de los recursos humanos y tecnológicos, lo que conduce a una reducción de costos operativos y a una maximización del retorno de inversión para las empresas de entrega de comida a domicilio en Bolivia. Al automatizar tareas repetitivas y optimizar flujos de trabajo, las empresas pueden asignar recursos de manera más efectiva y centrarse en áreas estratégicas de crecimiento.

Además, de acuerdo con la página de estadísticas "Statista", los principales gigantes del sector de pedidos online de comida incluyen a Delivery Hero, Glovo, Just Eat TakeAway, Uber Eats, Rappi, Meituan y Ele.me. Estas empresas, reconocidas a nivel mundial por su presencia y dominio en diferentes regiones, han marcado el rumbo de la industria y continúan siendo referentes en innovación y calidad de servicio. Su influencia y alcance han contribuido significativamente a la transformación del mercado de entrega de comida a domicilio, consolidándose como actores clave en el ámbito del comercio electrónico B2C (Fernández, 2024).

Un ejemplo destacado de la implementación exitosa de tecnologías como DevOps en el sector de delivery de comida a domicilio es el caso de Rappi. Desde su inicio en 2015 como un servicio de ventas al por menor, Rappi ha experimentado un crecimiento significativo al expandir sus operaciones para incluir entregas de restaurantes, supermercados y compras realizadas por personal shoppers a domicilio. Con presencia en cinco países de América Latina y un equipo de aproximadamente 200 desarrolladores, Rappi se ha convertido en un líder en la industria.

La complejidad de las necesidades técnicas de Rappi ha sido abordada mediante la implementación de tecnologías innovadoras, incluyendo soluciones de Amazon Web Services (AWS). Herramientas como Amazon Elastic Compute Cloud, Amazon Relational Database Service y Amazon Elastic Container Service han permitido a Rappi operar su negocio principal de manera eficiente, escalable y flexible. Gracias a estas tecnologías,

Rappi ha podido impulsar su crecimiento, expandir sus operaciones, lanzar nuevos servicios y consolidar su éxito en toda Latinoamérica.

Los testimonios de Felipe Villamarin, Co-Fundador, y Alessandro Comissario, VP de DevOps e Infraestructura, destacan el papel fundamental que las tecnologías de AWS han desempeñado en el éxito continuo de Rappi. Este caso de éxito ilustra cómo la implementación efectiva de herramientas tecnológicas puede impulsar la innovación y el crecimiento en el sector de delivery de comida a domicilio, proporcionando un ejemplo inspirador para nuestra investigación sobre la aplicación de DevOps en empresas similares en Bolivia (AWS, s.f.).

## **2. Situación Problemática**

Las empresas dedicadas al desarrollo de plataformas de delivery de comida a domicilio en Bolivia se enfrentan a una serie de desafíos operativos que afectan su capacidad para garantizar la eficiencia y la satisfacción del cliente. Estos desafíos abarcan diversos aspectos de su operación y pueden influir en la calidad del servicio ofrecido, así como en la experiencia general del cliente. Algunos de los desafíos más destacados incluyen:

### **Complejidad en la gestión de plataformas tecnológicas:**

Las empresas de delivery deben lidiar con la complejidad inherente a la gestión de plataformas tecnológicas que abarcan desde la toma de pedidos hasta la entrega final.

La diversidad de sistemas y tecnologías utilizadas para la gestión de pedidos, seguimiento de entregas y experiencia del usuario puede resultar en dificultades para integrar y coordinar eficientemente estos componentes.

### **Rápida evolución de las preferencias del consumidor:**

Las preferencias del consumidor en cuanto a métodos de pedido, opciones de pago y tiempos de entrega pueden cambiar rápidamente, lo que requiere que las empresas de delivery sean ágiles y adaptables.

La capacidad de implementar rápidamente nuevas funcionalidades y mejoras en las plataformas de entrega es esencial para mantenerse al día con las demandas del mercado y satisfacer las expectativas del cliente.

### **Seguridad de datos y cumplimiento normativo:**

La protección de la información del cliente y el cumplimiento de las regulaciones de privacidad de datos son aspectos críticos para las empresas de delivery.

El manejo seguro de datos personales y financieros de los clientes, así como el cumplimiento de normativas legales y regulatorias, son requisitos indispensables para garantizar la confianza y la satisfacción del cliente.

### **3. Formulación del Problema de Investigación o Pregunta Científica.**

¿Cómo diseñar e implementar un modelo efectivo de DevOps adaptado a las necesidades y características específicas de las empresas de desarrollo de plataformas de delivery de comida a domicilio en Bolivia?

### **4. Objetivo General.**

Diseñar un modelo de implementación de DevOps adaptado a las necesidades y características específicas de las empresas de desarrollo de delivery de comida a domicilio en Bolivia, con el fin de mejorar la eficiencia operativa y la experiencia del cliente.

### **5. Objetivos Específicos.**

- 1 Investigar** las prácticas actuales de gestión de tecnología y procesos en empresas de desarrollo de plataformas de delivery de comida a domicilio en Bolivia, identificando áreas de mejora en eficiencia operativa y satisfacción del cliente.
- 2 Analizar** la infraestructura tecnológica existente en estas empresas para evaluar la viabilidad de integrar prácticas DevOps en el contexto del delivery de comida a domicilio.
- 3 Identificación de herramientas y tecnologías relevantes**, para evaluar el impacto de la implementación en la eficiencia operativa teniendo en cuenta diferentes puntos en el contexto de Bolivia.

### **6. Diseño Metodológico.**

#### **6.1 Tipo de Investigación.**

El enfoque de investigación utilizado en este estudio es principalmente exploratorio y descriptivo. El enfoque exploratorio permitirá profundizar en un tema relativamente nuevo como es la aplicación de DevOps en empresas de delivery de comida a domicilio en Bolivia.

Por otro lado, el enfoque descriptivo busca comprender y describir los desafíos operativos específicos que enfrentan estas empresas y cómo la implementación de DevOps puede abordarlos.

### **6.1.1 Alcance de la Investigación**

La investigación se centra en el estudio de las empresas de delivery de comida a domicilio en Bolivia, con especial énfasis en sus desafíos operativos y la viabilidad de implementar la metodología DevOps para mejorar su eficiencia y la satisfacción del cliente. Se considerarán aspectos como la infraestructura tecnológica existente, los procesos operativos y las necesidades y preferencias del consumidor en el contexto boliviano. Además, se llevará a cabo una investigación exhaustiva de los servicios de delivery disponibles en la región, incluyendo aquellos que no tengan una presencia pública, con el objetivo de obtener una comprensión completa del panorama de delivery en Bolivia.

## **6.2 Métodos**

### **6.2.1 Métodos teóricos.**

Se emplearon métodos teóricos, como el análisis documental y la revisión bibliográfica, para investigar y comprender los fundamentos teóricos de la metodología DevOps, así como los desafíos operativos específicos en el sector de delivery de comida a domicilio en Bolivia. Esto proporcionó una base sólida para el desarrollo de la monografía.

### **6.2.2 Método Análisis –Síntesis.**

El método de análisis-síntesis fue fundamental en diversas etapas del proceso de investigación. Se utilizó para descomponer y clasificar la información recopilada, identificar patrones y tendencias en los desafíos operativos de las empresas de delivery, así como para sintetizar y organizar los hallazgos en la redacción de la monografía.

### **6.2.3 Métodos empíricos.**

Dado el carácter teórico y descriptivo de la investigación, no se utilizaron métodos empíricos.

## **6.3 Técnicas.**

Las técnicas utilizadas en esta investigación incluyen:

**Revisión sistemática de la literatura:** Se llevará a cabo una revisión exhaustiva de la literatura académica y técnica relacionada con DevOps en el contexto de la industria de

delivery de comida a domicilio en Bolivia. Esta revisión proporcionó una base sólida de conocimientos teóricos y prácticos sobre el tema.

**Investigación en diferentes sitios de información:** Se explorará diversos recursos en línea, como sitios web de empresas, foros de la industria y redes sociales, para recopilar datos relevantes sobre las prácticas y desafíos en el sector de delivery de comida a domicilio en Bolivia.

**Investigación en Google Play y App Store:** Se realizará una investigación detallada en las plataformas de distribución de aplicaciones móviles, Google Play y App Store, para analizar las aplicaciones de delivery de comida a domicilio disponibles en Bolivia. Esto incluyó el análisis de características, calificaciones y comentarios de usuarios para comprender mejor el panorama de las aplicaciones de delivery en el país e identificar tendencias y preferencias de los usuarios.

#### **6.4 Procedimientos e instrumentos de investigación**

Los procedimientos e instrumentos de investigación a utilizar son los siguientes:

**Selección de fuentes bibliográficas y sitios de información relevantes:** Se identificaron y seleccionaron cuidadosamente fuentes bibliográficas, sitios web de empresas, foros de la industria y redes sociales pertinentes para la investigación.

**Extracción y análisis de datos:** Se recopilaron datos de diversas fuentes, incluyendo literatura académica, sitios web y plataformas de distribución de aplicaciones móviles. Estos datos fueron analizados en detalle para identificar patrones, tendencias y áreas de interés.

# CAPITULO I

## MARCO TEORICO Y CONTEXTUAL

### 1.1 Marco teórico

La industria de la hostelería, que abarca restaurantes, cafeterías y otros establecimientos de servicio de alimentos, enfrenta desafíos constantes para optimizar la experiencia del cliente y la eficiencia operativa. La implementación de prácticas DevOps en los servicios de delivery, una metodología de desarrollo y operación de software que promueve la colaboración, la automatización y la entrega continua, emerge como una solución prometedora para abordar estos retos.

#### 1.1.1 Principales teorías que abordan la temática de DevOps

##### **Teoría de sistemas complejos:**

DevOps se fundamenta en la noción de que los sistemas de software son entidades complejas compuestas por múltiples componentes interrelacionados. Esta teoría sostiene que cambios en una parte del sistema pueden afectar otras áreas, subrayando la importancia de un enfoque holístico y colaborativo para el desarrollo, despliegue y operación del software.

##### **Teoría de restricciones (TOC):**

La Teoría de Restricciones postula que en cualquier sistema existe una restricción que limita la capacidad del sistema para lograr sus objetivos. En el contexto de DevOps, estas restricciones pueden surgir en cualquier etapa del ciclo de vida del desarrollo de software, y la metodología DevOps busca identificar y eliminar estas restricciones para mejorar el flujo de trabajo y la eficiencia global del sistema (Xeridia, 2018).

##### **Los cinco pasos de la TOC:**

- **Identificar el cuello de botella:** El primer paso es identificar el punto del proceso que está limitando el flujo general. Esto se puede hacer mediante la medición de métricas clave, como el tiempo de ciclo y el rendimiento.

- **Explotar el cuello de botella:** Una vez identificado el cuello de botella, el siguiente paso es optimizar su rendimiento. Esto puede implicar mejorar los procesos, eliminar cuellos de botella o agregar recursos.
- **Subordinar todo al cuello de botella:** Todos los demás pasos en el proceso deben ajustarse para adaptarse al rendimiento del cuello de botella. Esto significa que no tiene sentido optimizar otras partes del proceso si el cuello de botella las está reteniendo.
- **Elevar el cuello de botella:** Si el cuello de botella aún está limitando el rendimiento general del sistema, entonces es necesario aumentar su capacidad. Esto puede implicar agregar recursos, invertir en nuevas tecnologías o cambiar los procesos.
- **Volver al paso 1:** La TOC es un proceso continuo de mejora. Una vez que se ha elevado el cuello de botella, es importante volver al paso 1 y volver a identificar el nuevo cuello de botella.

### **Beneficios de aplicar la TOC en DevOps:**

- Mejora el flujo de trabajo y reduce el tiempo de ciclo.
- Aumenta la previsibilidad de las entregas.
- Mejora la calidad del software.
- Reduce los costos.
- Aumenta la satisfacción del cliente.

### **Teoría de la comunicación y colaboración:**

DevOps promueve una cultura de comunicación abierta y colaboración estrecha entre los equipos de desarrollo y operaciones. Esta teoría destaca la importancia de superar las barreras organizativas y fomentar la colaboración entre diferentes funciones para alcanzar objetivos comunes de manera más efectiva.

### **Principios ágiles y Lean:**

DevOps se inspira en los principios ágiles y Lean, que enfatizan la entrega continua, la retroalimentación rápida y la mejora continua. Estas teorías sugieren que las organizaciones pueden lograr una mayor eficiencia y calidad al adoptar prácticas que fomenten la adaptabilidad, la innovación y la eliminación de desperdicios en los procesos de desarrollo y operación de software.

### **Teoría de la automatización:**

La automatización es un pilar central de DevOps, basada en la idea de que muchas tareas manuales y repetitivas pueden y deben ser automatizadas para mejorar la eficiencia, reducir errores y acelerar el tiempo de entrega. Esta teoría sostiene que la automatización de procesos y tareas rutinarias libera tiempo y recursos para actividades más creativas y de mayor valor agregado.

#### **1.1.3 Conceptos básicos de DevOps.**

DevOps (Development and Operations) surge como una fusión de las áreas de desarrollo y operaciones de software, buscando romper las barreras tradicionales entre ambos departamentos y fomentar una cultura de colaboración y trabajo en equipo (Microsoft Azure, s.f.).

### **Principios clave de DevOps:**

- **Colaboración:** Fomentar la comunicación y el trabajo conjunto entre equipos de desarrollo, operaciones, seguridad y otras áreas para lograr un enfoque holístico en el ciclo de vida del software.

- **Automatización:** Implementar herramientas y procesos automatizados para tareas repetitivas, como pruebas, despliegue y configuración, liberando tiempo para actividades de mayor valor.
- **Entrega continua:** Adoptar un enfoque incremental de desarrollo y entrega de software, lanzando nuevas versiones con frecuencia y en ciclos cortos, permitiendo una retroalimentación rápida y una mejora constante.
- **Monitoreo y observabilidad:** Implementar sistemas de monitoreo y análisis para recopilar datos sobre el rendimiento del software, identificar problemas de manera proactiva y tomar decisiones informadas.
- **Cultura de aprendizaje:** Fomentar una cultura de aprendizaje continuo y mejora constante, donde los errores se consideran oportunidades para aprender y mejorar.

#### 1.1.4 Beneficios de DevOps en la hostelería:

La implementación de DevOps en la industria de la hostelería puede generar múltiples beneficios, incluyendo (SES.EUS, 2023):

- **Mejora de la experiencia del cliente:** Reducción de tiempos de espera, mayor precisión en los pedidos, menor incidencia de errores y una comunicación más efectiva con los clientes.
- **Aumento de la eficiencia operativa:** Automatización de tareas manuales, optimización de procesos, mejor gestión de inventario y reducción de costos operativos.
- **Mayor agilidad y adaptación:** Capacidad para responder rápidamente a cambios en las demandas del mercado, implementar nuevas funcionalidades y adaptar los servicios a las necesidades de los clientes.
- **Mejora de la calidad del software:** Implementación de pruebas automatizadas, detección temprana de errores y un enfoque continuo en la mejora del rendimiento del software.
- **Mayor satisfacción del personal:** Fomento de la colaboración, reducción del estrés laboral y un ambiente de trabajo más positivo.

#### 1.1.5. Desafíos y limitaciones de la implementación de DevOps:

A pesar de sus beneficios, la implementación de DevOps en la hostelería presenta algunos desafíos:

- **Cultura organizacional:** La resistencia al cambio y la falta de comprensión de los principios de DevOps pueden obstaculizar la adopción de la metodología.
- **Falta de recursos y habilidades:** La implementación de DevOps requiere inversión en herramientas, capacitación del personal y desarrollo de nuevas habilidades.
- **Integración con sistemas existentes:** La integración de DevOps con sistemas heredados o complejos puede ser un desafío técnico.
- **Medición del impacto:** La cuantificación precisa del impacto de DevOps en la experiencia del cliente y la eficiencia operativa puede ser difícil.

### **1.1.5 Infraestructura Tecnológica en Servicios de Entrega de Comida a Domicilio**

#### **Sistemas de Gestión de Pedidos**

En la industria de entrega de comida a domicilio, los sistemas de gestión de pedidos son esenciales para recibir, procesar y gestionar eficientemente los pedidos de los clientes. Estos sistemas suelen incluir interfaces para usuarios finales, restaurantes asociados y repartidores. Ejemplos comunes de plataformas de gestión de pedidos incluyen Uber Eats, Rappi y Glovo.

#### **Tecnologías de Rastreo y Logística**

Las tecnologías de rastreo y logística son cruciales para asegurar una entrega precisa y oportuna de los pedidos. Estas tecnologías permiten a los clientes rastrear el progreso de sus pedidos en tiempo real y proporcionan a los repartidores rutas optimizadas para la entrega. Sistemas de GPS y aplicaciones de seguimiento en tiempo real son componentes clave en esta área.

#### **Infraestructura en la Nube**

La computación en la nube desempeña un papel fundamental en la infraestructura tecnológica de los servicios de entrega de comida a domicilio. Las empresas de este sector

suelen utilizar servicios en la nube para alojar sus aplicaciones, bases de datos y almacenamiento de datos. Proveedores como Amazon Web Services (AWS), Google Cloud Platform y Microsoft Azure son opciones populares para esta infraestructura.

### **Seguridad de la Información y Cumplimiento Normativo**

Dada la sensibilidad de los datos personales y financieros de los clientes, la seguridad de la información es una preocupación importante en la industria de entrega de comida a domicilio. Las empresas deben implementar medidas de seguridad robustas para proteger los datos del cliente y cumplir con las regulaciones de privacidad, como el Reglamento General de Protección de Datos (GDPR) en Europa.

### **Integraciones y APIs**

Las integraciones con otros sistemas, como sistemas de pago y sistemas de gestión de restaurantes, son comunes en los servicios de entrega de comida a domicilio. Estas integraciones se facilitan a menudo mediante el uso de APIs (interfaces de programación de aplicaciones) que permiten la comunicación entre diferentes sistemas de software.

### **Escalabilidad y Fiabilidad**

Dado el crecimiento rápido y la variabilidad en la demanda de servicios de entrega de comida a domicilio, la escalabilidad y la fiabilidad de la infraestructura tecnológica son críticas. Las empresas deben diseñar sus sistemas para manejar picos de demanda sin comprometer la calidad del servicio. Estrategias como la implementación de auto escalado y el uso de arquitecturas de microservicios pueden ayudar a lograr este objetivo.

## **1.2 Marco Contextual**

Descripción del contexto socioeconómico y cultural en Bolivia

Bolivia, como muchos otros países, ha experimentado un crecimiento significativo en su sector de entrega de comida a domicilio, impulsado por diversos factores económicos y sociales. En los últimos años, la demanda de servicios de entrega de alimentos ha aumentado

considerablemente, reflejando cambios en los hábitos de consumo y en el estilo de vida de la población boliviana.

El rápido crecimiento económico y la urbanización han contribuido a un aumento en la demanda de conveniencia y servicios a domicilio. Además, los cambios en los patrones de trabajo y el estilo de vida han generado una mayor preferencia por soluciones de entrega de comida rápida y práctica.

Sin embargo, el sector de entrega de comida a domicilio en Bolivia enfrenta desafíos operativos significativos, como la gestión eficiente de plataformas tecnológicas, la rápida evolución de las preferencias del consumidor y la seguridad de datos y el cumplimiento normativo. Estos desafíos requieren soluciones innovadoras y adaptadas al contexto boliviano para garantizar la eficiencia operativa y la satisfacción del cliente.

En el contexto específico de Bolivia, la situación socioeconómica ha influido significativamente en el mercado de deliveries de comida, especialmente en el último período marcado por la pandemia de COVID-19. A continuación, se analizan algunos aspectos relevantes:

### **Situación socioeconómica de Bolivia en el aspecto de deliveries de comida:**

La pandemia de COVID-19 ha tenido un impacto significativo en los hábitos de consumo de alimentos en Bolivia. Durante el período de restricciones de movilidad y distanciamiento social, se observó un aumento notable en la demanda de servicios de entrega a domicilio. Si bien la situación ha evolucionado y muchas restricciones se han levantado, el cambio en los hábitos de consumo persiste en cierta medida. Los consumidores continúan valorando la conveniencia y seguridad que ofrecen las entregas a domicilio, aunque en menor medida que durante el pico de la pandemia.

Además, el aumento de la penetración de smartphones e internet en la población boliviana ha facilitado el acceso a las plataformas de delivery, permitiendo que un mayor número de personas utilicen estos servicios de manera regular.

## **Tamaño del mercado.**

El mercado de deliveries de comida en Bolivia se encuentra en crecimiento acelerado. Este crecimiento se atribuye en gran medida al cambio en los hábitos de consumo impulsado por la pandemia, así como al aumento de la adopción de tecnologías digitales.

## **Tendencias.**

- **Crecimiento de los pedidos online:** Se espera que los pedidos online continúen creciendo en los próximos años en Bolivia. A medida que más personas se familiarizan con las plataformas de delivery y aprecian la comodidad que ofrecen, se espera que esta tendencia se mantenga al alza.
- **Diversificación de la oferta:** Las plataformas de delivery en Bolivia están ampliando su oferta de productos para satisfacer las necesidades cambiantes de los consumidores. Además de la entrega de comida, están incluyendo otros productos como bebidas, medicamentos, productos de supermercado, entre otros. Esta diversificación de la oferta busca mejorar la experiencia del usuario y capturar una mayor participación en el mercado.
- **Mayor competencia:** Con el crecimiento del mercado de deliveries de comida en Bolivia, se espera que la competencia entre las diferentes plataformas se intensifique. La entrada de nuevos competidores y la búsqueda de diferenciación por parte de las plataformas existentes generarán un entorno más competitivo en el que las empresas deberán innovar y mejorar constantemente para mantener o mejorar su posición en el mercado.

Se destaca la posición consolidada de PedidosYa en el mercado de delivery en comparación con la reciente incursión de PatioService. A pesar de su corta trayectoria, PatioService muestra potencial para competir y diversificar el mercado.

## **CAPITULO II**

### **DIAGNOSTICO**

#### **2.1 Introducción**

En este capítulo, se explora un estudio de caso de éxito en la implementación de prácticas DevOps y el aprovechamiento eficiente de servicios en la nube, centrándose en la experiencia de PedidosYa, uno de los líderes en el mercado de delivery en América Latina. A medida que las empresas buscan mejorar su capacidad para entregar valor de manera rápida y confiable a sus clientes, la adopción de prácticas DevOps y el uso de tecnologías en la nube se han vuelto fundamentales.

El estudio de caso de PedidosYa proporciona una valiosa oportunidad para analizar cómo estas estrategias han contribuido al éxito operativo y al crecimiento sostenido de una empresa en un mercado altamente competitivo. Además, se identificarán áreas de mejora potenciales y se propondrán recomendaciones para fortalecer aún más su enfoque DevOps y su infraestructura tecnológica.

Este análisis detallado permitirá extraer lecciones clave y ofrecerá insights valiosos para otras organizaciones que buscan optimizar sus procesos de desarrollo de software, mejorar la calidad del producto y aumentar la satisfacción del cliente a través de la implementación efectiva de prácticas DevOps y el uso estratégico de servicios en la nube.

#### **2.1.1 Procesamiento y Análisis de Datos**

##### **Estudio de caso de éxito PedidosYa**

En esta sección, se detalla el conjunto de herramientas y servicios utilizados por PedidosYa en su proceso de desarrollo y operación, así como el análisis de datos relacionado con su infraestructura tecnológica.

PedidosYa, como empresa líder en el sector de entregas a domicilio, ha adoptado una serie de herramientas y servicios tecnológicos para optimizar sus operaciones y garantizar una

experiencia fluida para sus usuarios finales. A continuación, se presenta un desglose de las principales herramientas y servicios utilizados por PedidosYa (Nerd, 2021).

Para ordenar y ampliar según el ciclo de vida DevOps, podemos organizar las herramientas de la siguiente manera:

### **Planificación:**

- Git: Se emplea para el control de versiones del código fuente, permitiendo a los equipos de desarrollo colaborar de manera eficiente y mantener un historial de cambios.

### **Desarrollo:**

- Git: Además de su función en el control de versiones, Git también se utiliza para la colaboración y el seguimiento de cambios durante el desarrollo de nuevas características y funcionalidades.
- Helm: Facilita la gestión, implementación y actualización de aplicaciones en clústeres de contenedores, lo que es especialmente útil durante el desarrollo para desplegar y probar nuevas versiones de software de manera rápida y eficiente.
- Python y Bash: Utilizados para el desarrollo de scripts personalizados y la automatización de tareas durante el proceso de desarrollo.

### **Construcción:**

- Jenkins: Automatiza el proceso de compilación del código fuente, la ejecución de pruebas y la preparación de artefactos para su despliegue.

### **Pruebas:**

- Jenkins: Además de la construcción, Jenkins también se utiliza para automatizar el proceso de ejecución de pruebas, garantizando la calidad del software antes de su lanzamiento.
- Python y Bash: Pueden utilizarse para la automatización de pruebas y la creación de scripts de pruebas personalizados.

### **Embalaje:**

- Docker: En la etapa de embalaje, Docker se utiliza para crear contenedores que encapsulan la aplicación y todas sus dependencias en un entorno consistente y aislado. Esto asegura que la aplicación se ejecute de manera confiable en cualquier entorno donde Docker esté instalado.
- Kubernetes: Aunque Kubernetes se utiliza principalmente en la etapa de despliegue para orquestar y administrar los contenedores, también puede entrar en juego en la etapa de embalaje para definir cómo se empaquetan y distribuyen los contenedores. Kubernetes proporciona una manera de especificar cómo se ejecutan los contenedores, incluidas las reglas de escalabilidad, tolerancia a fallos, y configuraciones de red, lo que puede influir en el proceso de embalaje de la aplicación.
- Jenkins: Después de las pruebas, Jenkins puede ser configurado para empaquetar los artefactos de la aplicación en formatos específicos para su despliegue.

#### **Despliegue:**

- Kubernetes: Orquesta y automatiza el despliegue de la aplicación en contenedores, garantizando la disponibilidad y escalabilidad de la misma.
- Helm: Utilizado para la gestión e implementación de aplicaciones en clústeres de contenedores, facilitando el despliegue de nuevas versiones de software.
- Terraform: Permite la creación y gestión de la infraestructura necesaria para el despliegue de la aplicación, garantizando la consistencia y la reproducibilidad del entorno.
- Servicios de AWS: Proporcionan la infraestructura en la nube necesaria para ejecutar la aplicación de manera escalable y confiable.

#### **Monitoreo:**

- Amazon Elasticsearch: Se utiliza para el análisis y búsqueda de datos en tiempo real, lo que permite a PedidosYa obtener información valiosa sobre el comportamiento de los usuarios y el rendimiento de la plataforma.
- Servicios de AWS: AWS ofrece una serie de servicios de monitoreo y registro, como CloudWatch, que permiten a PedidosYa monitorear el rendimiento y la salud de su infraestructura en la nube.

Esta organización refleja cómo cada herramienta se utiliza en diferentes etapas del ciclo de vida DevOps, desde la planificación y el desarrollo hasta el despliegue y el monitoreo, contribuyendo así a la entrega continua y la mejora constante del software.

### **2.1.2 Comparación de caso de éxito frente a otra boliviana**

#### **PedidosYa:**

Nombre de la empresa: PedidosYa.

País de origen: Uruguay.

Fecha de lanzamiento de la aplicación: 2009 y en Bolivia en 2018.

Última actualización: 2 de mayo de 2024.

Número de usuarios activos y descargas: Más de 50 millones de descargas.

Calificaciones y comentarios de los usuarios: 4.5 estrellas basadas en 1.15 millones de opiniones en tiendas de aplicaciones.

(Google play, s.f.)

#### **PatioService:**

Nombre de la empresa: PatioService Delivery.

País de origen: Bolivia.

Fecha de lanzamiento de la aplicación: 2023.

Última actualización: 22 de abril de 2024.

Número de usuarios activos y descargas: Más de 10 mil descargas.

Calificaciones y comentarios de los usuarios: 3.8 estrellas basadas en 73 opiniones en tiendas de aplicaciones.

(Google play , s.f.)

#### **Funcionalidades y características:**

Ambas aplicaciones ofrecen servicios de delivery de comida y bebida, así como diferentes métodos de pago. Sin embargo, se destaca que PedidosYa tiene una presencia más consolidada en el mercado, con un mayor número de descargas y una calificación más alta por parte de los usuarios.

### **Análisis comparativo:**

- **Experiencia y presencia en el mercado:** PedidosYa tiene una trayectoria más larga y una presencia establecida en varios países, incluyendo Bolivia desde 2018. En contraste, PatioService es una empresa de origen boliviano que recientemente incursionó en el mercado de delivery.
- **Recepción de los usuarios:** Aunque ambas aplicaciones tienen una base de usuarios activos, PedidosYa cuenta con una calificación más alta y un mayor número de descargas, lo que sugiere una mayor satisfacción y aceptación por parte de los usuarios.
- **Actualizaciones y desarrollo continuo:** Ambas aplicaciones han recibido actualizaciones recientes, lo que indica un compromiso con el desarrollo continuo y la mejora de la experiencia del usuario.

Experiencia y presencia en el mercado: PedidosYa tiene una trayectoria más larga y una presencia establecida en varios países, incluyendo Bolivia desde 2018. En contraste, PatioService es una empresa de origen boliviano que recientemente incursionó en el mercado de delivery.

Recepción de los usuarios: Aunque ambas aplicaciones tienen una base de usuarios activos, PedidosYa cuenta con una calificación más alta y un mayor número de descargas, lo que sugiere una mayor satisfacción y aceptación por parte de los usuarios.

Actualizaciones y desarrollo continuo: Ambas aplicaciones han recibido actualizaciones recientes, lo que indica un compromiso con el desarrollo continuo y la mejora de la experiencia del usuario.

## 2.1.2 Tabulación y Codificación de datos

### Comparación de herramientas de uso en cada etapa del ciclo de vida DevOps

#### Planificación

*Tabla 1 Comparación de herramientas en la etapa de Planificación*

Herramienta de Planificación	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Jira	Herramienta de gestión de proyectos ágil popular entre equipos de desarrollo de software.	Soporta equipos de todos los tamaños, desde pequeños hasta grandes empresas.	Amplio soporte en línea, comunidad activa y recursos educativos disponibles .	Se integra con una amplia gama de herramientas de desarrollo de software, incluyendo GitHub, Bitbucket, GitLab, Slack, Confluence y Zapier.	Gratis para proyectos de código abierto, planes premium a partir de \$10 por usuario/meses.
Azure DevOps	Solución de gestión de DevOps de Microsoft que ofrece herramientas para la planificación, el seguimiento	Altamente escalable, adecuada para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y recursos educativos disponibles .	Se integra con una amplia gama de herramientas de Microsoft, incluyendo Azure, Visual Studio,	Gratis para planes básicos (hasta 5 usuarios), planes premium a partir de \$35 por

	y la entrega de software.			Office 365 y Slack.	usuario/mes.
GitLab	Plataforma de desarrollo de software de código abierto que incluye herramientas para la gestión de proyectos, el seguimiento de código y la implementación continua.	Altamente escalable, adecuada para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y recursos educativos disponibles .	Se integra con una amplia gama de herramientas de desarrollo de software, incluyendo GitHub, Bitbucket, Jenkins, Slack y Zapier.	Gratis para planes básicos (hasta 10 usuarios privados), planes premium a partir de \$4 por usuario/mes.
GitHub	Plataforma de alojamiento de código y gestión de versiones utilizada por equipos de desarrollo de software de todo el mundo.	Altamente escalable, adecuada para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y recursos educativos disponibles .	Se integra con una amplia gama de herramientas de desarrollo de software, incluyendo GitLab, Bitbucket, Jenkins, Slack y Zapier.	Gratis para planes básicos (hasta usuarios públicos ilimitados), planes premium a partir de \$4 por usuario/mes.
Trello	Herramienta de gestión de proyectos basada en el método	Altamente escalable, adecuada para equipos	Amplio soporte en línea, comunidad activa y	Se integra con una amplia gama de herramientas	Gratis para planes básicos, planes premium a

	Kanban que es popular entre los equipos ágiles.	de todos los tamaños.	recursos educativos disponibles .	de productividad, incluyendo Slack, Google Calendar, Power-Up y Zapier.	partir de \$10 por usuario/mes.
--	---	-----------------------	-----------------------------------	---	---------------------------------

Fuente: Elaboración propia

## Desarrollo

*Tabla 2 Comparación de herramientas en la etapa de Desarrollo*

Herramienta de Desarrollo	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Visual Studio	Entorno de desarrollo integrado (IDE) de Microsoft para el desarrollo de aplicaciones Windows, web y móviles.	Altamente escalable, adecuado para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y recursos educativos disponibles .	Se integra con una amplia gama de herramientas de Microsoft, incluyendo Azure, SQL Server, .NET y Xamarin.	Gratis para Community Edition, Suscripción de Professional 145 USD por usuario y por mes .
Visual Studio Code	Editor de código gratuito y de código abierto de Microsoft para el	Altamente escalable, adecuado para equipos de todos los tamaños y	Amplio soporte en línea, comunidad activa y	Se integra con una amplia gama de herramientas de	Gratis.

	desarrollo de software.	proyectos complejos.	extensiones de terceros.	desarrollo, incluyendo Git, GitHub, Docker y Kubernetes.	
IntelliJ IDEA	IDE de JetBrains para el desarrollo de Java, Kotlin, Scala, Groovy y otros lenguajes de programación .	Altamente escalable, adecuado para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y plugins de terceros.	Se integra con una amplia gama de herramientas de desarrollo, incluyendo Git, GitHub, Maven y Gradle.	Gratis para Community Edition, planes premium a partir de \$69.99 por usuario/año.
WebStorm	IDE de JetBrains para el desarrollo web y front-end.	Altamente escalable, adecuado para equipos de todos los tamaños y proyectos complejos.	Amplio soporte en línea, comunidad activa y plugins de terceros.	Se integra con una amplia gama de herramientas de desarrollo web, incluyendo HTML, CSS, JavaScript, Node.js y React.	Gratis para Community Edition, planes premium a partir de \$69.99 por usuario/año.
PyCharm	IDE de JetBrains para el desarrollo de Python.	Altamente escalable, adecuado para equipos de todos los	Amplio soporte en línea, comunidad activa y	Se integra con una amplia gama de herramientas	Gratis para Community Edition, planes premium a

		tamaños y proyectos complejos.	plugins de terceros.	de desarrollo de Python, incluyendo Django, Flask, Anaconda y Jupyter Notebook.	partir de \$69.99 por usuario/año.
--	--	--------------------------------	----------------------	---	------------------------------------

Fuente: Elaboración propia

## Construcción

*Tabla 3 Comparación de herramientas en la etapa de Construcción*

Herramienta de Construcción	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Jenkins	Servidor de integración continua y entrega continua (CI/CD) de código abierto popular para automatizar la construcción, pruebas e implementación de software.	Altamente escalable, adecuado para proyectos de todos los tamaños y equipos de todos los tamaños.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Gratis.

GitLab	Plataforma DevOps completa que incluye herramientas de CI/CD, gestión de código fuente, seguimiento de problemas, monitoreo y más.	Altamente escalable, adecuado para proyectos de todos los tamaños y equipos de todos los tamaños.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Plan gratuito para proyectos públicos y de código abierto, planes premium a partir de \$4 por mes por usuario.
CircleCI	Plataforma de CI/CD en la nube que ofrece una configuración rápida y sencilla, escalabilidad automática y una amplia gama de integraciones.	Altamente escalable, adecuado para proyectos de todos los tamaños y equipos de todos los tamaños.	Amplio soporte en línea, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Plan gratuito para proyectos públicos, planes premium a partir de \$20 por mes por usuario.
Travis CI	Plataforma de CI/CD en la nube popular	Altamente escalable, adecuado	Amplio soporte en línea, recursos	Se integra con una amplia gama	Plan gratuito para

	para proyectos alojados en GitHub.	para proyectos de todos los tamaños y equipos de todos los tamaños.	educativos disponibles.	de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	proyectos públicos, planes premium a partir de \$25 por mes por usuario.
Azure DevOps	Conjunto de herramientas de DevOps de Microsoft que ofrece CI/CD, gestión de código fuente, seguimiento de problemas, monitoreo y más.	Altamente escalable, adecuado para proyectos de todos los tamaños y equipos de todos los tamaños.	Amplio soporte en línea, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de Microsoft y de terceros, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Plan gratuito para proyectos públicos, planes premium a partir de \$35 por mes por usuario.

Fuente: Elaboración propia

## Pruebas

*Tabla 4 Comparación de herramientas en la etapa de Pruebas*

Herramienta de Pruebas	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Selenium	Framework de código abierto para pruebas de automatización web que permite a los usuarios crear y ejecutar scripts de prueba para navegadores web.	Altamente escalable, adecuado para pruebas de sitios web y aplicaciones web de todos los tamaños.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI y Docker.	Gratis.
Appium	Framework de código abierto para pruebas de automatización móvil que permite a los usuarios crear y ejecutar scripts de prueba para aplicaciones móviles	Altamente escalable, adecuado para pruebas de aplicaciones móviles en dispositivos iOS y Android.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI y Docker.	Gratis.

	nativas, híbridas y web.				
Cypress	Framework de pruebas de automatización JavaScript moderno para pruebas de aplicaciones web de una sola página y aplicaciones web tradicionales.	Altamente escalable, adecuado para pruebas de aplicaciones web modernas y dinámicas.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI y Docker.	Gratis y de pago (planes premium a partir de \$25 por mes por usuario).
K6	Herramienta de código abierto para pruebas de carga y rendimiento que permite a los usuarios simular cargas de usuarios en aplicaciones web y APIs.	Altamente escalable, adecuado para pruebas de carga y rendimiento de aplicaciones web y APIs de alto tráfico.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de monitoreo y análisis, incluyendo Grafana, Prometheus y Datadog.	Gratis.

Fuente: Elaboración propia

## Embalaje

Tabla 5 Comparación de herramientas en la etapa de Empaquetado

Herramienta de Empaquetado	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Docker	Plataforma de contenedores de código abierto popular para empaquetar y ejecutar aplicaciones en entornos aislados.	Altamente escalable, adecuado para empaquetar y ejecutar aplicaciones de todos los tamaños y en entornos de nube y locales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI, Kubernetes y Ansible.	Gratis.
Kubernetes	Plataforma de código abierto para orquestar y administrar contenedores Docker.	Altamente escalable, adecuado para la gestión de aplicaciones distribuidas y microservicios en entornos de nube y locales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins,	Gratis.

				GitLab, CircleCI, Travis CI, Docker y Ansible.	
Packer	Herramienta de código abierto para crear imágenes de máquina virtual (VM) automatizadas y consistentes.	Altamente escalable, adecuado para la creación de imágenes de VM para entornos de nube y locales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI, Docker y Ansible.	Gratis.
Vagrant	Entorno de desarrollo local para crear y administrar máquinas virtuales.	Altamente escalable, adecuado para el desarrollo local y la prueba de aplicaciones en entornos virtuales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI,	Gratis y de pago (planes premium a partir de \$10 por mes por usuario).

				Docker y Ansible.	
Ansible	Herramienta de código abierto para la automatización de la configuración de sistemas y la implementación de aplicaciones.	Altamente escalable, adecuado para la automatización de tareas de infraestructura y DevOps en entornos de nube y locales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI, Docker y Kubernetes.	Gratis y de pago (planes premium a partir de \$2 por mes por nodo).

Fuente: Elaboración propia

## Despliegue

Herramienta de Despliegue	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Kubernetes	Plataforma de código abierto para orquestar y administrar contenedores Docker.	Altamente escalable, adecuado para el despliegue y la gestión de aplicaciones distribuidas y microservicios en entornos	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins,	Gratis.

		de nube y locales.		GitLab, CircleCI, Travis CI, Docker y Ansible.	
Ansible	Herramienta de código abierto para la automatización de la configuración de sistemas y la implementación de aplicaciones.	Altamente escalable, adecuado para el despliegue automatizado de aplicaciones en entornos de nube y locales.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Jenkins, GitLab, CircleCI, Travis CI, Docker y Kubernetes.	Gratis y de pago (planes premium a partir de \$2 por mes por nodo).
Jenkins	Servidor de integración continua y entrega continua (CI/CD) de código abierto popular para automatizar la construcción, pruebas e implementación de software.	Altamente escalable, adecuado para el despliegue automatizado de aplicaciones desde pipelines de CI/CD.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle,	Gratis.

				Docker y Kubernetes.	
GitLab	Plataforma DevOps completa que incluye herramientas de CI/CD, gestión de código fuente, seguimiento de problemas, monitoreo y más.	Altamente escalable, adecuado para el despliegue automatizado de aplicaciones desde pipelines de CI/CD integrados.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Plan gratuito para proyectos públicos y de código abierto, planes premium a partir de \$4 por mes por usuario.
CircleCI	Plataforma de CI/CD en la nube que ofrece una configuración rápida y sencilla, escalabilidad automática y una amplia gama de integraciones.	Altamente escalable, adecuado para el despliegue automatizado de aplicaciones desde pipelines de CI/CD en la nube.	Amplio soporte en línea, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de desarrollo y DevOps, incluyendo Git, GitHub, Bitbucket, Maven, Gradle, Docker y Kubernetes.	Plan gratuito para proyectos públicos, planes premium a partir de \$20 por mes por usuario.

Fuente: Elaboración propia

## Monitoreo

*Tabla 6 Comparación de herramientas en la etapa de Monitoreo*

Herramienta de Operaciones y Monitoreo	Descripción	Escalabilidad	Soporte y Comunidad	Integraciones	Costo
Nagios	Sistema de monitoreo de infraestructura de código abierto popular para servidores, redes y aplicaciones.	Altamente escalable, adecuado para equipos de todos los tamaños y entornos complejos.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de infraestructura y DevOps, incluyendo Jenkins, Puppet, Chef y Docker.	Gratis para Community Edition, planes premium a partir de \$595 por año.
Zabbix	Sistema de monitoreo de infraestructura de código abierto y empresarial que ofrece una amplia gama de funciones para el monitoreo del rendimiento, la	Altamente escalable, adecuado para equipos de todos los tamaños y entornos complejos.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de infraestructura y DevOps, incluyendo Jenkins, Puppet, Chef y Docker.	Gratis para Community Edition, planes premium a partir de \$1,800 por año.

	disponibilidad y la seguridad.				
Prometheus	Sistema de monitoreo de código abierto y altamente personalizable que utiliza un modelo de datos basado en series temporales.	Altamente escalable, adecuado para equipos de todos los tamaños y entornos complejos.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de infraestructura y DevOps, incluyendo Kubernetes, Grafana y Alertmanager.	Gratis.
Grafana	Plataforma de visualización de datos de código abierto popular para crear paneles de control e informes personalizados.	Altamente escalable, adecuado para equipos de todos los tamaños y entornos complejos.	Amplia comunidad de usuarios y desarrolladores, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de monitoreo y análisis, incluyendo Prometheus, Zabbix, Nagios y Datadog.	Gratis para Community Edition, planes premium a partir de \$35 por mes.
Datadog	Plataforma de monitoreo e inteligencia observacional SaaS que ofrece una amplia gama	Altamente escalable, adecuado para equipos de todos los tamaños y	Amplio soporte en línea, recursos educativos disponibles.	Se integra con una amplia gama de herramientas de infraestructura, DevOps y	Gratis para planes básicos (hasta 5 GB de datos por mes),

	de funciones para el monitoreo del rendimiento, la disponibilidad, la seguridad y los registros.	entornos complejos.		SaaS, incluyendo AWS, Azure, Google Cloud Platform, Kubernetes, Docker y GitHub.	planes premium a partir de \$15 por mes.
--	--	---------------------	--	--	--

Fuente: Elaboración propia

### 2.1.3 Análisis y Discusión de Resultados

#### Propuesta de modelo de Implementación

En base a los datos analizados y recolectados en la sección previa, considerando el contexto de Bolivia se propone el siguiente modelo de implementación teniendo en cuenta cada etapa del ciclo de vida DevOps.

#### 1. Planificación:

- **Herramienta:** GitLab (Plan gratuito)  
GitLab ofrece un plan gratuito completo que incluye gestión de código fuente, seguimiento de problemas, CI/CD básico y monitoreo, lo que lo convierte en una opción ideal para equipos con recursos limitados.

#### 2. Desarrollo:

- **Metodología:** Scrum  
Scrum es una metodología ágil sencilla y flexible que se adapta bien a equipos pequeños y proyectos con recursos limitados.
- **Herramienta de automatización de pruebas:** Selenium (Gratuito)  
Selenium es una herramienta de código abierto popular para pruebas de automatización web, que permite crear y ejecutar scripts de prueba de forma gratuita.
- **Herramienta de integración continua:** Jenkins (Gratuito)

Jenkins es un servidor de CI/CD de código abierto ampliamente utilizado, que ofrece una amplia gama de funcionalidades para automatizar la construcción y las pruebas del código.

### 3. Construcción:

- **Herramienta de contenedores:** Docker (Gratuito)

Docker es una plataforma de contenedores de código abierto líder, que permite empaquetar y ejecutar aplicaciones de forma aislada y reproducible, lo que facilita la implementación y el mantenimiento del software.

- **Herramienta de infraestructura como código:** Ansible (Gratuito)

Ansible es una herramienta de automatización de código abierto para la configuración de sistemas y la implementación de aplicaciones, que permite definir y provisionar la infraestructura de forma automatizada y consistente.

### 4. Pruebas:

- **Pruebas unitarias:** JUnit (Gratuito)

JUnit es un framework de código abierto para pruebas unitarias en Java, que permite crear y ejecutar tests de forma sencilla y eficiente.

- **Pruebas de integración:** TestNG (Gratuito)

TestNG es otro framework de código abierto para pruebas unitarias y de integración en Java, que ofrece funcionalidades adicionales como pruebas parametrizadas y manejo de dependencias.

- **Pruebas de sistemas:** Cucumber (Gratuito)

Cucumber es un framework de código abierto para pruebas de aceptación basadas en comportamiento (BDD), que permite crear tests de forma clara y comprensible para el negocio.

### 5. Embalaje y Despliegue:

#### **Herramienta de empaquetado:** Docker (Gratuito)

Justificación: Docker se utiliza para empaquetar las aplicaciones en contenedores ligeros y autosuficientes, que pueden ejecutarse en diferentes entornos sin necesidad de modificar la infraestructura subyacente.

### **Herramienta de despliegue:** Ansible (Gratis)

Justificación: Ansible se utiliza para automatizar el despliegue de los contenedores Docker en servidores locales o en la nube. Esto permite una implementación eficiente y consistente de las aplicaciones.

### **Integración de Kubernetes:**

A medida que los proyectos maduran y las necesidades aumentan, se puede considerar la adopción de Kubernetes para la orquestación a gran escala y la gestión de aplicaciones distribuidas más complejas.

## **6. Monitoreo:**

- **Herramienta de monitoreo de infraestructura:** Prometheus (Gratis)  
Prometheus es un sistema de monitoreo de código abierto que permite recopilar y analizar métricas de infraestructura de forma eficiente.
- **Herramienta de monitoreo de software:** Grafana (Gratis)

Grafana es un dashboarding tool de código abierto que permite visualizar las métricas recopiladas por Prometheus de forma intuitiva y accesible.

## **2.2 Conclusiones y Recomendaciones**

### **2.2.1 Conclusiones**

Se ha demostrado que DevOps, es una herramienta valiosa para las empresas incluso de delivery.

Se pudo notar que la infraestructura y costos relacionados a la implementación de DevOps, lo hace una opción viable ya que los costos de estas son asequibles y algunas o en su gran mayoría son gratuitas con sus limitantes, que dependiendo del contexto de puede optar por otras de pago que nos puede dar más funcionalidades

### **Infraestructura y Costos de Implementación:**

Aunque muchas herramientas DevOps son accesibles e incluso gratuitas, es importante tener en cuenta que la inversión inicial en infraestructura y capacitación del personal puede ser significativa. Es esencial realizar un análisis completo de los costos involucrados y considerar tanto los beneficios a corto plazo como los ahorros a largo plazo que puede proporcionar la implementación de DevOps.

**Importancia de DevOps y la Nube:** El estudio de caso de PedidosYa destaca la importancia de la implementación efectiva de prácticas DevOps y el uso estratégico de servicios en la nube para optimizar los procesos de desarrollo de software y mejorar la experiencia del cliente.

**Éxito y Consolidación de PedidosYa:** PedidosYa ha logrado establecerse como un líder en el mercado de delivery en América Latina, con una presencia sólida y una alta aceptación por parte de los usuarios, lo que demuestra el impacto positivo de su enfoque DevOps y el aprovechamiento de la nube.

**Potencial de Competidores Locales:** La entrada de empresas locales como PatioService muestra un crecimiento y diversificación en el mercado de delivery en Bolivia, lo que indica un aumento en la competencia y la necesidad de adaptación por parte de los líderes establecidos como PedidosYa.

### 2.2.2 Recomendaciones

#### **Ampliación del Alcance de DevOps:**

Se recomienda no limitar la aplicación de DevOps exclusivamente al sector de entrega de comida rápida. DevOps ofrece beneficios significativos que pueden ser aprovechados por una variedad de industrias, incluyendo delivery de otros tipos de productos como farmacéuticos, electrodomésticos, y más. Explorar la implementación de DevOps en estas áreas podría conducir a mejoras sustanciales en la eficiencia operativa y la satisfacción del cliente en un espectro más amplio de servicios de entrega en Bolivia.

#### **Promoción de la Adopción de DevOps:**

Es esencial fomentar una mayor adopción de prácticas DevOps entre las empresas de entrega en Bolivia. Se recomienda la promoción de programas de capacitación y concientización sobre DevOps para ayudar a las empresas a comprender los beneficios y las mejores prácticas

de esta metodología. Esto podría incluir talleres, seminarios y recursos en línea para facilitar la transición hacia una cultura DevOps en el sector de entrega en Bolivia.

**Colaboración y Compartir Experiencias:**

Se alienta a las empresas de entrega a colaborar entre sí y compartir experiencias sobre la implementación de DevOps. Esto podría llevarse a cabo a través de grupos de discusión, comunidades en línea o eventos de networking donde las empresas puedan compartir sus desafíos, éxitos y lecciones aprendidas en la implementación de DevOps. El intercambio de conocimientos y mejores prácticas puede acelerar el proceso de adopción y maximizar los beneficios para todas las empresas involucradas en el sector de entrega en Bolivia.

## BIBLIOGRAFÍA

- Anderson, W. (9 de junio de 2023). *Medium*. Obtenido de GUIDE ON DEVOPS IMPLEMENTATION: STRATEGY, PLAN, ROADMAP:  
<https://stechsolutions.medium.com/guide-on-devops-implementation-strategy-plan-roadmap-f5bfb431f7d9>
- AWS. (s.f.). *Caso práctico de Rappi*. Obtenido de AWS:  
<https://aws.amazon.com/es/solutions/case-studies/Rappi/>
- Fernández, R. (5 de febrero de 2024). *El mercado online mundial de entrega de comida (food delivery) – Datos estadísticos*. Obtenido de Statista:  
<https://es.statista.com/temas/8657/el-mercado-online-mundial-de-reparto-de-comida-food-delivery/#topicOverview>
- Gallagher, D., & Lennon, R. G. (07 de febrero de 2023). *IEEE*. Obtenido de Architecting Multi-Cloud Applications for High Availability using DevOps:  
<https://ieeexplore.ieee.org/abstract/document/10035095>
- Google play . (s.f.). *PatioService Delivery*. Obtenido de Google play :  
[https://play.google.com/store/apps/details?id=com.movapps.patioservice&hl=es\\_MX&gl=US](https://play.google.com/store/apps/details?id=com.movapps.patioservice&hl=es_MX&gl=US)
- Google play. (s.f.). *PedidosYa - Delivery Online*. Obtenido de Google play:  
[https://play.google.com/store/apps/details?id=com.pedidosya&hl=es\\_BO&gl=US](https://play.google.com/store/apps/details?id=com.pedidosya&hl=es_BO&gl=US)
- Kalinin, K. (6 de abril de 2023). *Why DevOps Strategy Is a Must for Your Business: Implementation Steps, Plan and Examples*. Obtenido de topflight:  
<https://topflightapps.com/ideas/how-to-implement-devops/>
- Kantar. (3 de noviembre de 2023). *Comer fuera vuelve a los niveles prepandemia a nivel global*. Obtenido de Kantar: <https://www.kantar.com/es/inspiracion/gran-consumo/comer-fuera-vuelve-a-los-niveles-prepandemia-a-nivel-global>

- Microsoft Azure. (s.f.). *¿Qué es DevOps?* Obtenido de Azure Microsoft:  
<https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-devops#tools>
- Nerd, P. (1 de junio de 2021). *La INFRAESTRUCTURA en la NUBE de PEDIDOS YA!* - *Podcast*. Obtenido de Youtube:  
[https://www.youtube.com/watch?v=i\\_fq7siKslo&t=210s](https://www.youtube.com/watch?v=i_fq7siKslo&t=210s)
- Salas, G. (27 de diciembre de 2022). *PedidosYa presenta informe de tendencias de consumo de 2022*. Obtenido de La Voz: <https://lavoz.digital/2022/12/27/pedidosya-presenta-informe-de-tendencias-de-consumo-de-2022/>
- SES.EUS. (20 de junio de 2023). *DevOps y los desafíos que afrontan las empresas*. Obtenido de Noticias SES: <https://sesitdigital.com/devops-y-los-desafios-que-afrontan-las-empresas/>
- Villa, M. (6 de mayo de 2018). *En 10 años, consumo en restaurantes creció un 300%*. Obtenido de La Razón: <https://www.la-razon.com/lr-article/en-10-anos-consumo-en-restaurantes-crecio-un-300-2/>
- Xeridia. (02 de mayo de 2018). *El éxito de la Teoría de las Limitaciones en DevOps*. Obtenido de Xeridia: <https://www.xeridia.com/blog/el-exito-de-la-teoria-de-las-limitaciones-en-devops>