




ACTIVIDAD 1

DESARROLLO DE SOFTWARE

Hecho por:

Chacón Roque, Leonardo Alexander

Chipana Calderón, Guido Anthony





Parte 1



1. ¿Por qué surgió la necesidad de DevOps en el desarrollo de software?

DevOps surgió para solucionar los problemas de los modelos tradicionales, donde el desarrollo y las operaciones trabajaban separados, haciendo entregas lentas, acumulando errores y dificultándose a adaptarse a los cambios. La necesidad de ciclos de entrega más ágiles y eficientes llevó a integrar los equipos con procesos automatizados y una cultura de colaboración.

2. Explica cómo la falta de comunicación y coordinación entre los equipos de desarrollo y operaciones en el pasado llevó a la creación de DevOps.

Antes, los equipos del desarrollo y de las operaciones tenían objetivos distintos: los desarrolladores buscaban rapidez, mientras que operaciones priorizaba estabilidad. Esta falta de alineación generaba retrasos, lanzamientos problemáticos y un ambiente de culpabilidad entre equipos. Para reducir estas barreras, se creó DevOps, promoviendo la colaboración y la automatización de procesos.

3. Describe cómo el principio de mejora continua impacta tanto en los aspectos técnicos como en los culturales de una organización.

El principio de mejora continua impulsa optimizaciones constantes en código, infraestructura y metodologías. A nivel técnico, permite detectar y corregir errores rápidamente, mientras que a nivel cultural fomenta una mentalidad de aprendizaje, experimentación y adaptación, mejorando la eficiencia y la motivación de los equipos.

4. ¿Qué significa que DevOps no se trata sólo de herramientas, individuos o procesos?

Significa que DevOps no es sólo un conjunto de tecnologías o un rol específico dentro de la empresa, sino una transformación de la organización de la empresa. Su éxito depende de un cambio en la mentalidad y en la forma de trabajo, promoviendo comunicación, responsabilidad compartida y procesos flexibles.

5. ¿Cómo contribuyen los equipos autónomos y multifuncionales a una implementación exitosa de DevOps?

Los equipos autónomos (desarrollo, QA, operaciones), permiten el intercambio de conocimientos y la resolución conjunta de los problemas, mejorando la calidad del producto. Cada equipo contribuye de la siguiente forma:

- Desarrollo: Realiza cambios en el software frecuente y rápidamente, asegurando que el código esté optimizado y listo para su despliegue continuo.
- QA: Integra pruebas automatizadas en el flujo de trabajo para detectar errores.
- Operaciones: Gestiona la infraestructura como código, automatiza despliegues y supervisa el rendimiento del software en producción.

Parte 2



¿Qué significa "desplazar a la izquierda" en el contexto de DevSecOps y por qué es importante?

“Desplazar a la izquierda” en el contexto de DevSecOps significa implementar la seguridad desde el principio del desarrollo y no al final del desarrollo como normalmente se realizaba.

Es importante por las siguientes razones:

- Detecta y corrige anomalías de forma temprana, lo que reduce el riesgo de incidentes en producción.
- La colaboración entre desarrolladores y expertos en seguridad se fortalece.
- Mejora la resiliencia del sistema ante ataques o fallos inesperados.

Explica cómo IaC mejora la consistencia y escalabilidad en la gestión de infraestructuras.

La infraestructura como Código (IaC) mejora la consistencia y escalabilidad en la gestión de infraestructuras al permitir que los entornos de desarrollo sean idénticos lo que minimiza los errores y conflictos

¿Cuál es la diferencia entre monitoreo y observabilidad? ¿Por qué es crucial la observabilidad en sistemas complejos?

El monitoreo se basa en la recopilación/recolección de métricas del sistema (uso de CPU, memoria, latencia de respuestas, tasa de errores) mientras que la observabilidad trata de analizar estas métricas para detectar problemas antes de que afecten a los usuarios finales.

¿Cómo puede la experiencia del desarrollador impactar el éxito de DevOps en una organización?

El papel del desarrollador es importante en el éxito de DevOps en una organización. Un entorno de trabajo optimizado aumenta la productividad y la calidad del software ya que herramientas y procesos eficientes reducen la frustración y minimizan el tiempo. También cuando el desarrollador cuenta con un flujo de trabajo ágil y bien estructurado, se fomenta la colaboración y la adopción de buenas prácticas dentro del equipo de desarrollo.

Describe cómo InnerSource puede ayudar a reducir silos dentro de una organización.

InnerSource es una estrategia que puede ayudar a reducir silos dentro de una organización ya que promueve la colaboración entre equipos al permitir compartir código y poder contribuir a proyectos internos lo que aumenta la transparencia y también facilita la reutilización de software ya que los desarrolladores tienen acceso al código de otros equipos.

¿Qué rol juega la ingeniería de plataformas en mejorar la eficiencia y la experiencia del desarrollador?

La ingeniería de plataformas juega un rol clave en la mejora de la eficiencia y la experiencia del desarrollador al proporcionar herramientas y entornos optimizados, los equipos pueden enfocarse en la creación de software sin preocuparse por la infraestructura. La automatización de procesos como despliegues, pruebas y monitoreo reduce la carga operativa lo que permite a los desarrolladores concentrarse en la innovación y entrega. Además al estandarizar la infraestructura y las prácticas de desarrollo, se garantiza una mayor seguridad y estabilidad en las aplicaciones.

Gracias