



DEVOPS SENIOR



CURSO:

DEVOPS SENIOR

- **Módulo 1: DEVOPS ESTRATÉGICO Y GITOPS**
- Módulo 2: AUTOMATIZACIÓN CON IA EN DEVOPS
- Módulo 3: SEGURIDAD AVANZADA Y DEVSECOPS
- Módulo 4: OBSERVABILIDAD AVANZADA
- Módulo 5: KUBERNETES AVANZADO
- Módulo 6: SERVICE MESH & NETWORKING MODERNO



Te encuentras aquí

CURSO:

DEVOPS SENIOR

- Módulo 7: INFRAESTRUCTURA COMO CÓDIGO AVANZADA
- Módulo 8: PLATFORM ENGINEERING & INTERNAL DEVELOPER PLATFORMS (IDP)
- Módulo 9: FINOPS & COST OPTIMIZATION
- Módulo 10: AIOPS & INCIDENT MANAGEMENT
- Módulo 11: SOFT SKILLS PARA ROLES DEVOPS SENIOR
- Módulo 12: PROYECTO FINAL INTEGRADOR

Módulo 1: DevOps Estratégico y GitOps.



OBJETIVO ESPECÍFICO DEL MÓDULO

- IDENTIFICAR CONCEPTOS DE DEVOPS ESTRATÉGICO Y GITOPS, GITOPS AVANZADO Y EVOLUCIÓN DEL ROL DEVOPS, SEGUN LAS PRACTICAS AVANZADAS DE GITOPS, DEVSECOPS, KUBERNETES, OBSERVABILIDAD, IAC, FINOPS Y AIOPS.



¿Cómo cree que cambiaría el trabajo del equipo de operaciones si toda la infraestructura y las configuraciones de aplicaciones pudieran gestionarse únicamente desde un repositorio Git?



DEVOPS MODERNO

- El DevOps actual trasciende la simple automatización. Integra colaboración continua entre desarrollo, operaciones, seguridad y calidad para entregar software más rápido, estable y seguro. Las prácticas modernas incluyen integración continua (CI), entrega continua (CD), infraestructura como código (IaC), y monitoreo constante.
- El enfoque está en ciclos de retroalimentación cortos, despliegues confiables, y alta capacidad de recuperación, permitiendo adaptarse rápidamente al cambio sin comprometer la estabilidad del sistema.



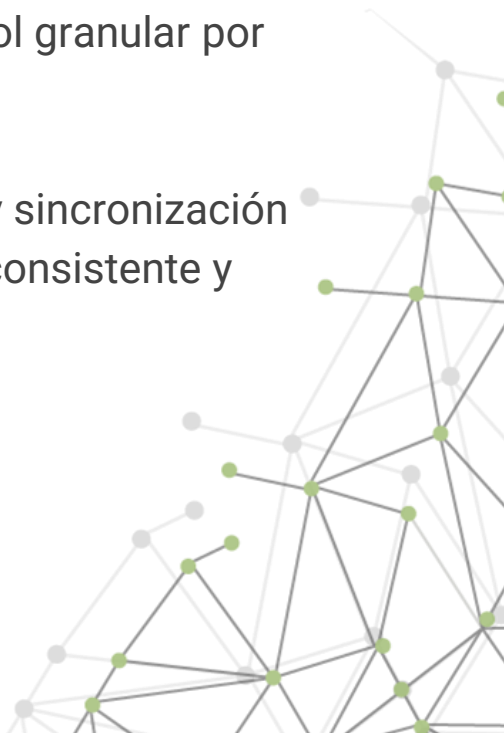
GITOPS PATTERNS

- GitOps se basa en el principio de usar Git como única fuente de verdad. Los patrones modernos incluyen sincronización automática de manifiestos declarativos con el entorno de producción, rollbacks automáticos, validación continua del estado del clúster y gestión de múltiples entornos desde un mismo repositorio.
- Estos patrones garantizan trazabilidad, gobernanza y control de cambios en infraestructuras complejas, permitiendo despliegues seguros y auditables.



ARGOCD AVANZADO

- ArgoCD permite la implementación continua declarativa en Kubernetes mediante sincronización desde repositorios Git. En su uso avanzado, soporta múltiples aplicaciones, hooks personalizados, rollbacks automáticos, y control granular por RBAC.
- Ofrece dashboards detallados, integración con Helm y Kustomize, y sincronización automática de estados deseados, haciendo que el despliegue sea consistente y confiable.



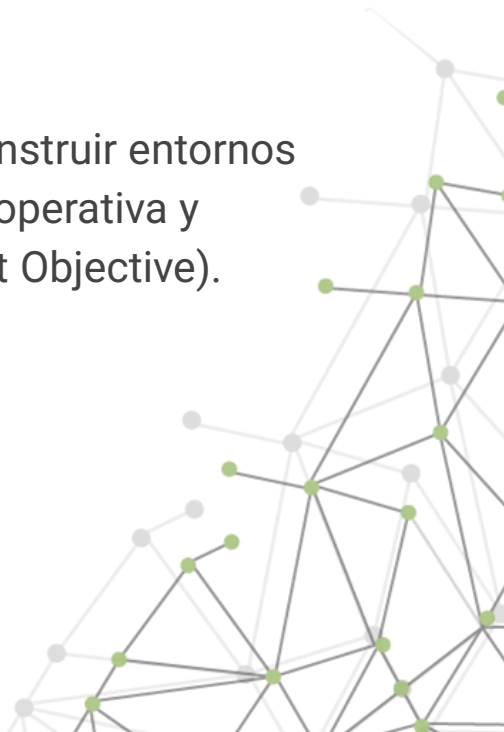
GESTIÓN DE SECRETOS

- La gestión segura de secretos en entornos automatizados es clave. Herramientas como HashiCorp Vault, Sealed Secrets o External Secrets permiten almacenar y entregar credenciales de forma cifrada.
- Las prácticas modernas implican integración con CI/CD, expiración controlada, auditoría de accesos y rotación automática. Esto minimiza la exposición y evita filtraciones accidentales.



DR (DISASTER RECOVERY)

- Un plan de recuperación ante desastres en DevOps contempla respaldos periódicos, replicación de datos, infraestructura automatizada y validación constante de procedimientos de restauración.
- Con herramientas como Velero, Terraform y GitOps, es posible reconstruir entornos rápidamente ante fallos catastróficos, manteniendo la continuidad operativa y reduciendo el RTO (Recovery Time Objective) y RPO (Recovery Point Objective).





**No olvide desarrollar los ejercicios que
contiene el Módulo...**

Después de explorar GitOps, ArgoCD y la gestión segura de secretos, ¿qué prácticas implementaría en su equipo para mejorar la trazabilidad, seguridad y automatización del ciclo de vida de sus aplicaciones?



**Éxito en la evaluación parcial y
en la Prueba Final...**

{desafío}
latam_

*Academia de
talentos digitales*

