

CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



PROGRAMMA

MODULO 1: GIT E GITLAB NEL WORKFLOW DI SVILUPPO

- Architettura e componenti principali di GitLab
- GitLab Projects, Groups e repository
- Repository locali e repository remoti
- Clone, fetch, pull e push
- Commit e gestione della cronologia
- Branch e tag
- HEAD e riferimenti Git
- Differenza tra merge e rebase
- Gestione dei conflitti
- Buone pratiche nella gestione dei commit

MODULO 2: BRANCHING STRATEGY E COLLABORAZIONE TRA SVILUPPATORI

- Organizzare il ciclo di sviluppo tramite branch
- Feature branch
- Development e release branch
- Hotfix
- Strategie trunk-based e GitFlow: differenze e casi d'uso
- Naming convention
- Gestione dei tag e delle versioni
- Branch Rules e Protected Branches
- Limitazione dei push diretti sui branch principali
- Gestione dei permessi di merge
- Strategie per evitare modifiche non controllate al codice

CONTATTI

 info@nexsys.it

 www.nexsys.it

CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



MODULO 3: MERGE REQUEST E CODE REVIEW

- Concetto di Merge Request
- Creazione e gestione delle Merge Request
- Draft Merge Request
- Assignee e Reviewer
- Commenti e discussioni sul codice
- Code Review
- Request changes e approvazione
- Merge checks
- Squash dei commit
- Rebase prima del merge
- Risoluzione dei conflitti
- Merge Request template
- Approval workflow
- Introduzione a CODEOWNERS
- Definizione delle responsabilità sul codice

MODULO 4: INTRODUZIONE A GITLAB CI/CD

- Concetti di Continuous Integration e Continuous Delivery
- Architettura GitLab CI/CD
- Pipeline
- Job
- Stage
- GitLab Runner
- Struttura del file .gitlab-ci.yml
- Sintassi YAML utilizzata nelle pipeline
- Creazione della prima pipeline
- Pipeline associate a push e Merge Request
- Analisi dello stato di una pipeline
- Log dei job

CONTATTI

 info@nexsys.it

 www.nexsys.it

CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



MODULO 5: COSTRUIRE PIPELINE GITLAB CI/CD

- Definizione di stage e job
- Script e comandi
- Dipendenze tra job
- Esecuzione parallela
- rules
- Condizionare l'esecuzione dei job
- Pipeline per branch differenti
- Pipeline per Merge Request
- Variabili CI/CD
- Variabili di progetto e di gruppo
- Gestione sicura delle informazioni sensibili
- Artifact
- Cache
- Gestione dei risultati di build e test
- Include e riutilizzo della configurazione
- Template di pipeline



CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



MODULO 6: GITLAB RUNNER

- Ruolo dei Runner nell'architettura GitLab
- Come vengono eseguiti i job
- Runner disponibili per progetto e gruppi
- Runner dedicati
- Tag dei Runner
- Assegnazione dei job
- Executor
- Shell Executor
- Docker Executor
- Isolamento degli ambienti di esecuzione
- Gestione e manutenzione dei Runner
- Analisi dei log
- Troubleshooting dei Runner

MODULO 7: ENVIRONMENT E DEPLOYMENT

- Gestione degli ambienti
- Development
- Test
- Staging
- Production
- Deployment tramite pipeline
- Job automatici e manuali
- Gestione delle variabili per ambiente
- Protezione degli ambienti critici
- Gestione delle release
- Rollback e strategie di ripristino
- Gestione degli artifact generati durante il processo di delivery



CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



MODULO 8: SICUREZZA E GOVERNANCE GITLAB

- Ruoli e permessi nei progetti
- Principio del least privilege
- Protezione dei branch
- Controllo dei Merge
- Approval rules
- CODEOWNERS
- Protezione delle variabili CI/CD
- Gestione di token e credenziali
- Personal Access Token e Project Access Token
- Sicurezza dei Runner
- Gestione dei secret nelle pipeline
- Protezione del codice e della supply chain software
- Introduzione alle funzionalità di security scanning disponibili nelle diverse edizioni GitLab

MODULO 9: TROUBLESHOOTING E OTTIMIZZAZIONE DELLE PIPELINE

- Analizzare una pipeline fallita
- Lettura dei log
- Errori YAML
- Errori nei job
- Problemi di Runner
- Gestione delle dipendenze
- Problemi con artifact e cache
- Retry dei job
- Pipeline duplicate o eseguite inutilmente
- Ottimizzazione dei tempi di esecuzione
- Parallelizzazione dei job

CONTATTI



info@nexsys.it



www.nexsys.it

CORSO GITLAB: DAL VERSIONAMENTO DEL CODICE ALLE PIPELINE CI/CD

Durata: 12 ore



OVERVIEW DEL CORSO

Quando più sviluppatori lavorano sullo stesso codice servono regole chiare: chi può modificare i branch principali, chi approva le modifiche, come si arriva dal commit al rilascio. GitLab permette di gestire tutto questo in un'unica piattaforma, ma spesso viene usato solo come repository remoto. Il corso accompagna il team in questo passaggio: si parte dal workflow Git quotidiano, si definiscono branching strategy e regole di code review con le Merge Request, poi si automatizzano build, test e deployment con le pipeline CI/CD. Il risultato è un processo di sviluppo più controllato e ripetibile: modifiche verificate prima del merge, review tracciate e rilasci che non dipendono da passaggi manuali. Le 4 ore di laboratorio facoltative permettono di applicare il percorso a un progetto guidato.

A CHI È RIVOLTO IL CORSO?

Il corso è rivolto a:

- Sviluppatori che vogliono collaborare sul codice con branch e Merge Request strutturati e automatizzare build e test
- Team leader e tech lead che devono definire regole di branching, review e approvazione per tutto il team
- DevOps engineer e sistemisti che costruiscono pipeline, gestiscono i Runner e rilasciano sugli ambienti
- Responsabili IT e security che governano permessi, token e secret e proteggono il codice e la supply chain

COSA SAPRAI FARE ALLA FINE DEL CORSO?

Al termine del corso i partecipanti saranno in grado di:

- Gestire repository, branch, tag e cronologia dei commit con Git e GitLab, e risolvere i conflitti
- Scegliere una strategia di branching e proteggere i branch principali
- Organizzare la code review con Merge Request, approval rules e CODEOWNERS
- Costruire pipeline CI/CD con stage, job, rules, variabili, artifact e cache
- Configurare e mantenere i GitLab Runner con Shell e Docker executor
- Automatizzare il deployment su più ambienti con job manuali, ambienti protetti e rollback
- Applicare il least privilege a ruoli, token e secret e diagnosticare le pipeline fallite