

Icono Training Consulting



Formación

Curso GIT

Indra



Formador

Alejandro Cerezo



CONSULTOR / DESARROLLADOR y FORMADOR TÉCNICO en el área de PROGRAMACIÓN y DESARROLLO DE APLICACIONES FULLSTACK, cuenta con + de 15 años de experiencia tanto en el análisis y programación de aplicaciones web (Ciclo de vida 360: consultoría, creación de aplicaciones desde cero, actualizaciones, implementación de nuevas funcionalidades, mantenimiento, maquetación, diseño, etc), como en la formación técnica de esta área.

Especialista FULLSTACK, imparte formación en tecnologías tanto del BACKEND (Node, Express, Nest) y Bases de Datos (SQL y NoSQL Mongo); como del FRONTEND en diversos lenguajes, frameworks y librerías (Angular desde v. JS a 18x), React, Redux, Next, Vue JS, JS, Typescript Web Components, HTML, CSS, SASS, Bootstrap) y en herramientas de TESTING (Jest, Jasmine, Vitest, Cypress,) CI/CD (Husky, Sonar) y GIT/GITHUB, entre otras tecnologías.

Entrega formación en diversos formatos (planes de formación programada in company, formación ad hoc para proyectos IT,...), tanto en modalidad on-line como en modalidad presencial, capacitando a profesionales técnicos de entidades del sector público y del privado (fabricantes e integradores tecnológicos nacionales e internacionales, entidades financieras, aseguradoras, telecomunicaciones,...)



training@iconotc.com

□ **Duración:** 20 horas

□ **Modalidad:** On-line

□ **Fechas/Horario:**

- Días 3, 4, 5, 6 y 7 Noviembre 2025
- Horario Días 3, 4, 6 y 7 Noviembre 9:00 – 13:30 hs. / Día 5 Noviembre 9:00 – 11:00

□ **Contenidos:**

- Introducción
 - Qué es un SCV y qué un SCV distribuido
 - Historia de GIT: C, kernel linux, contexto (SVN, Mercurial, ...)
 - Anatomía de un SCV distribuido | diferencias/parecidos con centralizados
 - Instalación en Windows
 - CheatSheets y Libros recomendados
- Quick Start
 - Primer repo (init), primer commit
 - Configuración inicial: email y name
 - add/commit y status/log/show
 - Mensajes de commit
 - Anatomía de un repositorio git: staging area, index and cache
- Aprendiendo a referenciar revisiones y paths
 - Anatomía de comandos típicos, referencias VS paths
- Herramientas para preparar un buen commit en cualquier situación
 - HEAD, master, HEAD~1 y otras referencias útiles
 - Números de commit: SHA1, subcadena de SHA1
 - Nombres de tags, de heads y de branches
 - Referencias por mensaje de commit (:/cadena)
 - Para saber más: SPECIFYING REVISIONS en "man gitreparse"
- Rescribiendo la historia
 - amend
 - checkout

- reset
 - stash
 - git clean n | git clean f
 - revert
 - rebase
 - git bisect
 - Trabajando en paralelo
 - branches
 - o crear, borrar, intercambiar
 - o crear desde ref (git checkout b mybranch master~1)
 - tags
 - o crear, usar
 - patches
 - o crear, aplicar
 - remotes:
 - o remote v
 - o push/pull
 - o clones
 - o repos bare
 - o push branch, push tag
 - resolución de conflictos
 - merge VS rebase VS cherrypick
 - Pull Request
- o Creación
- o Uso
- o Merging
- o Cierre
- Utilidades
 - GitK, GitG y git gui | git log graph | formato git log
 - IntelliJ
- Configuración de git
 - alias
 - .gitconfig
 - o editor
 - o coloreado comandos
 - o formato salida comandos
 - o otras opciones
 - Hooks
 - o cómo crear
 - o hooks de lado cliente: commits, emails, rebase, ...
 - o hooks de lado servidor: prereceive, postreceive, update
- Subproyectos
 - crear submodules
 - workflow de commits
 - git submodule status recursive
 - git submodule foreach ...

- Integración con otras herramientas y entornos

- SourceTree
- Github
- Gitlab
- Bitbucket

- Buenas prácticas

- Commits atómicos
- Commits frecuentes
- No commits de trabajo a medias
- Test antes de commit
- Buenos mensajes de commit
- Usar branches, featurebranching
- Workflows
 - o Presentar las opciones más usadas
 - o Fijar un workflow común

Icono Training Consulting

GIT



Completa nuestra encuesta
de satisfacción a través del QR



GRACIAS

indra