

Sesión 8: Trazabilidad del código

Rastrea evolución, autores y razones de cambios; audita y encuentra regresiones rápido.

Qué vas a lograr

- Usar log/blame/bisect eficazmente.
- Encontrar cuándo y por qué cambió algo.
- Preparar auditorías y reportes.

Historial y cambios

```
# Visión general del repo
git log --oneline --graph --decorate --all
# Historial con estadísticas y seguimiento de renombres
git log --stat --follow -- path/archivo.ext
```

Quién cambió qué (blame)

```
git blame -L 10,60 path/archivo.ext
```

Buscar en el historial

```
# Añadió o eliminó ocurrencias de una cadena
git log -S "cadena" -- path/archivo.ext
# Cambios que coinciden con una expresión regular
git log -G "regex" -- path/archivo.ext
```

Encontrar el commit que introdujo un bug (bisect)

```
git bisect start
git bisect bad           # estado actual malo
git bisect good v1.2.0   # marca un punto bueno
# ejecuta pruebas manuales o script en cada paso
git bisect reset
```

Integración con plataformas

- Usa PRs/MRs descriptivos, vincula issues y CI.
- Revisa blame/anotaciones en la interfaz web para contexto.

Problemas comunes

- Renombres: usa `--follow`.
- Reformat masivo: `git blame -w` para ignorar espacios.