

Git desde Visual Studio Code

Interfaz Gráfica, Ramas y Conflictos

Prof. Yazmin Rubiela Cote

Universidad Industrial de Santander

Matemáticas Computacionales

¿Por qué usar Git en VS Code?

Ya conocemos la **Terminal** (la cocina), ahora pasaremos al **Comedor** (la interfaz gráfica).

- **Visual:** Ves los colores de los archivos modificados.
- **Rápido:** Haces `add` y `commit` con clics.
- **Integrado:** No necesitas salir del editor de código.
- **Resolución de Conflictos:** Es mucho más fácil entender las diferencias visualmente.

1. El Centro de Mando (Source Control)

Todo ocurre en la pestaña de ****Source Control**** (Icono de la rama a la izquierda o `Ctrl + Shift + G`).

Los Estados de los Archivos (Colores)

- **U (Verde):** Untracked (Nuevo archivo).
- **M (Naranja):** Modified (Archivo modificado).
- **D (Rojo):** Deleted (Archivo borrado).

2. Hacer un Commit sin Terminal

En la Terminal:

- 1 `git add .`
- 2 `git commit -m "mensaje"`

En VS Code:

- 1 Pasar el mouse sobre el archivo o "Changes".
- 2 Dar clic en el botón + (Stage Changes).
- 3 Escribir el mensaje en la caja de texto superior.
- 4 Dar clic en el botón ✓ **Commit** (o Ctrl+Enter).

3. Crear Ramas (Branching)

Olvídate de escribir comandos largos. En VS Code la rama está en la ****Barra de Estado**** (esquina inferior izquierda).

- 1 Busca donde dice **main** (o master) abajo a la izquierda.
- 2 Haz clic ahí. Se abrirá un menú arriba.
- 3 Selecciona: **+ Create new branch...**
- 4 Escribe el nombre (ej. experimento) y dale Enter.

Nota: Al hacerlo, VS Code te cambia automáticamente a la nueva rama (hace el switch).

4. Resolver Conflictos Visualmente

Cuando haces un merge y hay conflicto, VS Code pinta el código de colores:

¡Conflicto Detectado!

VS Code te mostrará 3 botones mágicos encima del texto conflictivo:

- **Accept Current Change (Verde):** Se queda con lo que tú tienes en tu rama actual.
- **Accept Incoming Change (Azul):** Se queda con lo que viene de la otra rama (lo nuevo).
- **Accept Both Changes:** Guarda las dos versiones (una debajo de la otra).

Paso final: Al elegir una opción, guarda el archivo y haz un Commit para cerrar el conflicto.

Tip Pro: La Extensión "Git Graph"

VS Code nativo es bueno, pero para ver el "mapa del metro" (historial gráfico) recomendamos instalar una extensión.

Pasos:

- 1 Ve a la pestaña de Extensiones (Cuadritos a la izquierda).
- 2 Busca "**Git Graph**".
- 3 Instálala.
- 4 Ahora tendrás un botón en la barra inferior para ver todo el historial de colores, ramas y merges visualmente.

Resumen: Terminal vs VS Code

Acción	VS Code (Clics)
git add	Botón + al lado del archivo
git commit	Escribir mensaje y dar a ✓
git branch	Clic en la esquina inferior izquierda
git push	Botón "Sync Changes" (Flechas circulares)
Resolver Conflicto	Clic en "Accept Current" o "Incoming"