

















www.sena.edu.co

Pablo Ortiz

Versiones



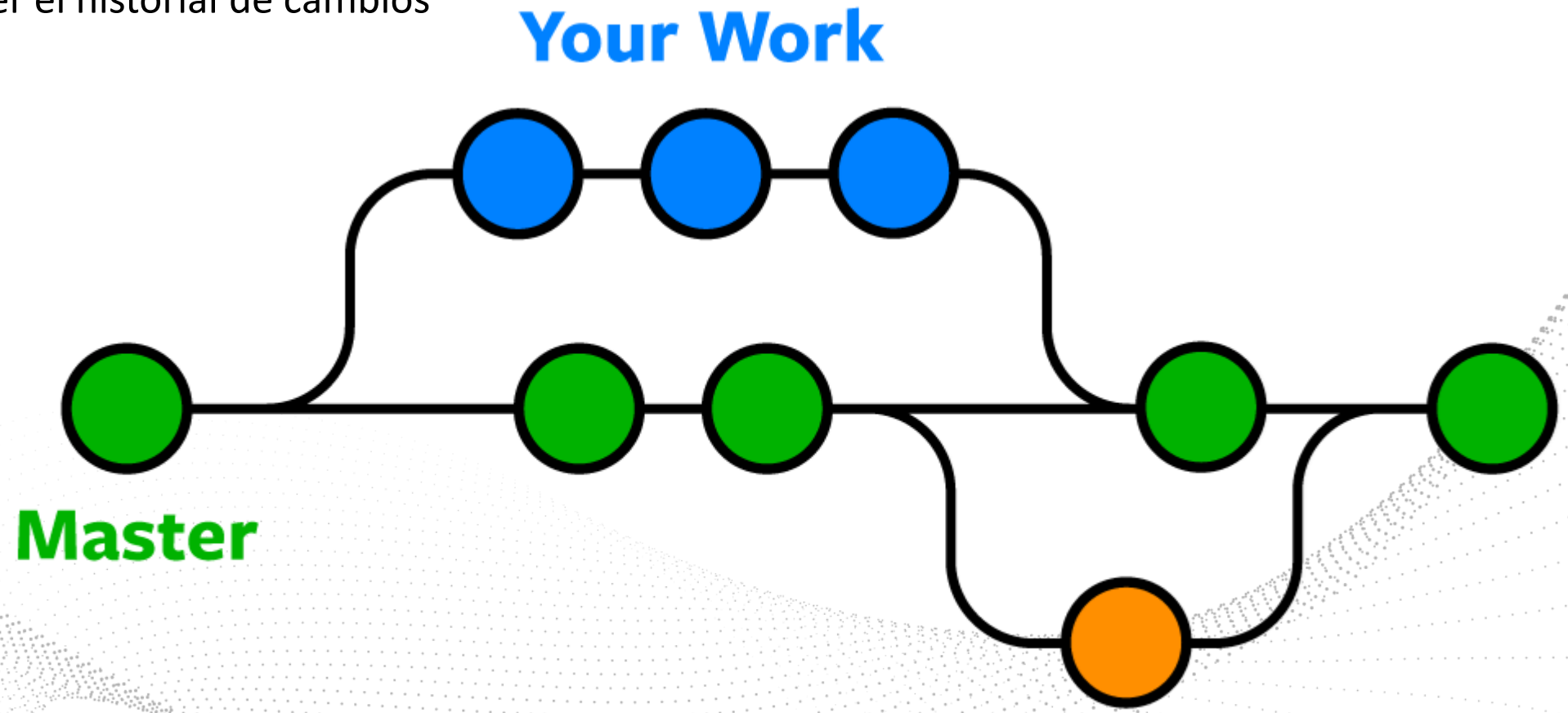
Documentos > Desarrollo Web - ADSI > git-prueba

	Nombre	Fecha de modificación	Tipo	Tama
✦	 Proyecto web - copia - copia (2)	18/03/2022 10:29	Documento de Mi...	
✦	 Proyecto web - copia - copia (3)	18/03/2022 10:29	Documento de Mi...	
✦	 Proyecto web - copia - copia (4)	18/03/2022 10:29	Documento de Mi...	
✦	 Proyecto web - copia - copia (5)	18/03/2022 10:29	Documento de Mi...	
✦	 Proyecto web - copia - copia (6)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia - copia	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (2)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (3)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (4)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (5)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (6)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia (7)	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web - copia	18/03/2022 10:29	Documento de Mi...	
	 Proyecto web	18/03/2022 10:29	Documento de Mi...	

Git



Sistema de control de versiones para llevar control de los cambios a los archivos y documentos de un repositorio y conocer el historial de cambios



Someone Else's Work

Instructor: Pablo Ortiz

Historia



- C.V.S: Concurrent Version System.
- SBN: Sub Version.
- Git: Gestión de Infraestructuras de Telecomunicaciones Linux torval - ramificación de versiones.

Redes Sociales

- GitHub: Red Social de Código. Microsoft
- GitLab: Red Social de Código Open Source

GitHub vs GitLab

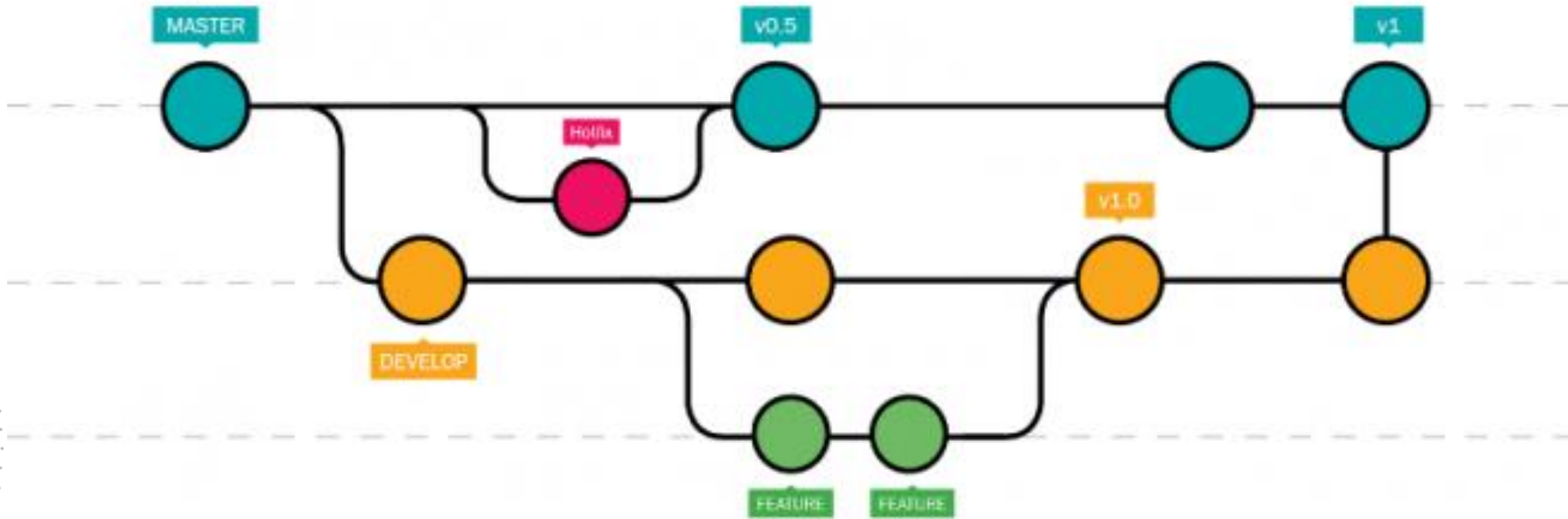


Aprende gestionar tus repositorios en GitHub y GitLab en:

 ed.team/cursos/git-workflow

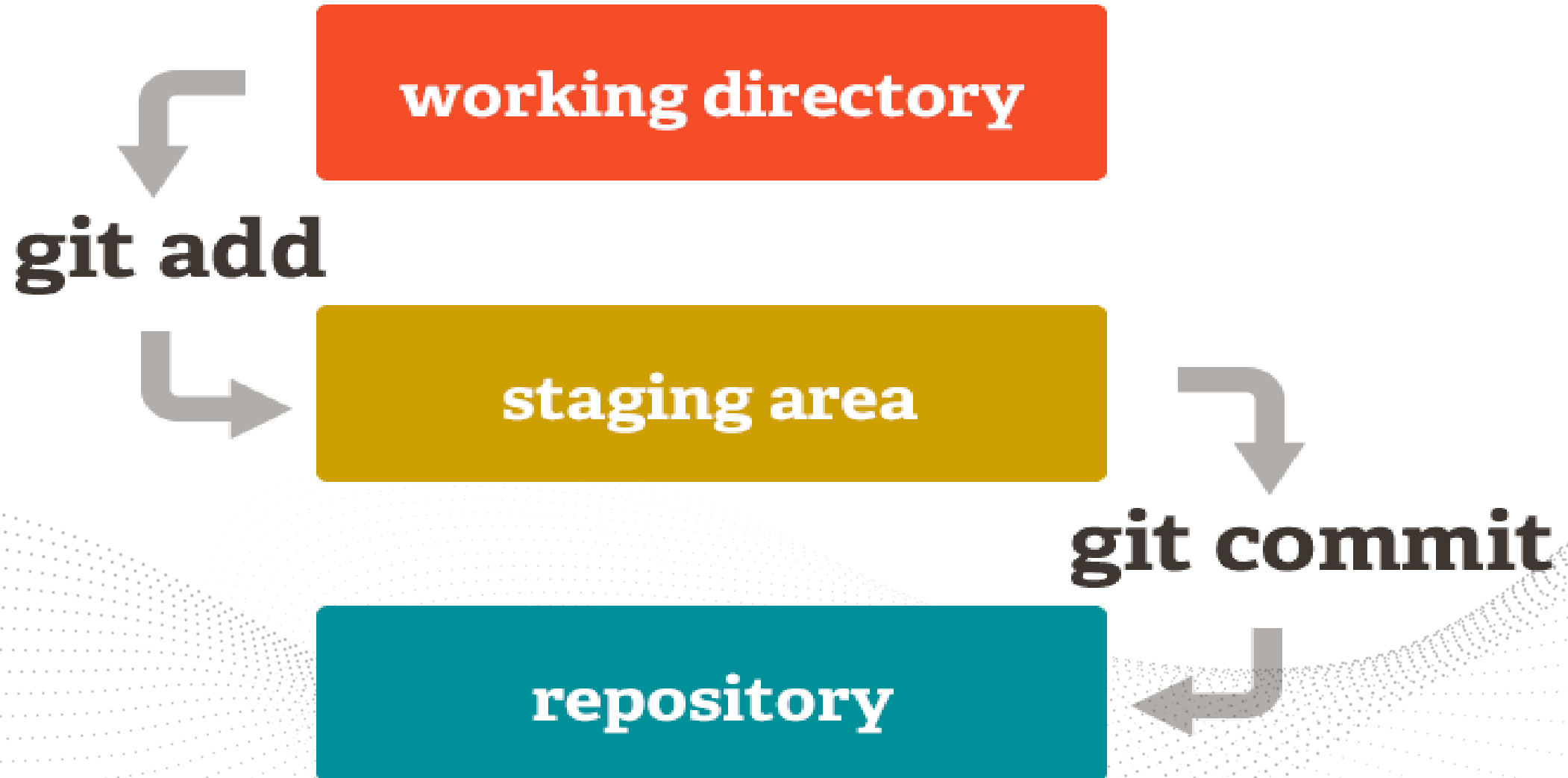


Flujo





Estados de Git



Creando Repositorio Local

- Ubicarse en la carpeta donde se creara el proyecto.
- Crear carpeta para el proyecto.
- Acceder a la carpeta creada (del proyecto). (dar clic derecho en la carpeta e iniciar con Git bash)
- Iniciar Git (`git init`).
- Crear archivos para el repositorio (code .).
- Verificar el estado de los archivos o cambios (`git status`).
- Colocar los archivos en estado git preparación (`git add nombre del archivo`)o (`git add -A` todos).
- Verificar estado (`git status`).
- Llevar archivos al repositorio (`git commit -m "mensaje"`).
- Subir el repositorio a GitHub (`git remote add origin http://....`).
- Enviar los archivos al repositorio (`git push origin master`)

Actualizar Cambios al Repositorio



- Agregar los cambios (Git add -A).
- Llevar archivos al repositorio (git commit -m "mensaje").
- Enviar los archivos al repositorio (git push origin master)

Ingresar Credenciales



- `Git config --global user.name "nombre del usuario"`
- `Git config --global user.email "correo del usuario"`



Crear Ramas o Bifurcaciones

- Verificar rama actual (Git log –online).
- Crear rama nueva (git branch nombre_rama).
- Mirar las ramas creadas (git branch).
- Moverse a la rama creada (checkout nombre_rama).

Guardar cambios en la nueva rama

- Colocar los archivos en estado git preparación (git add nombre del archivo) o (git add -A todos).
- Verificar estado (git status).
- Llevar archivos al repositorio (git commit -m "mensaje").
- Subir el repositorio a GitHub (git remote add origin [http://....](http://...)).
- Enviar los archivos al repositorio (git push origin nueva_rama)

Merge



- Moverse a la rama a unir (`git checkout nombre rama original`).
- Unir las ramas (`git merge nombre rama a unir`).
- Borrar la rama de los cambios (`git Branch -d nombre rama`).