



Biodata



IEB
INSTITUTO DE ECOLOGÍA
& BIODIVERSIDAD

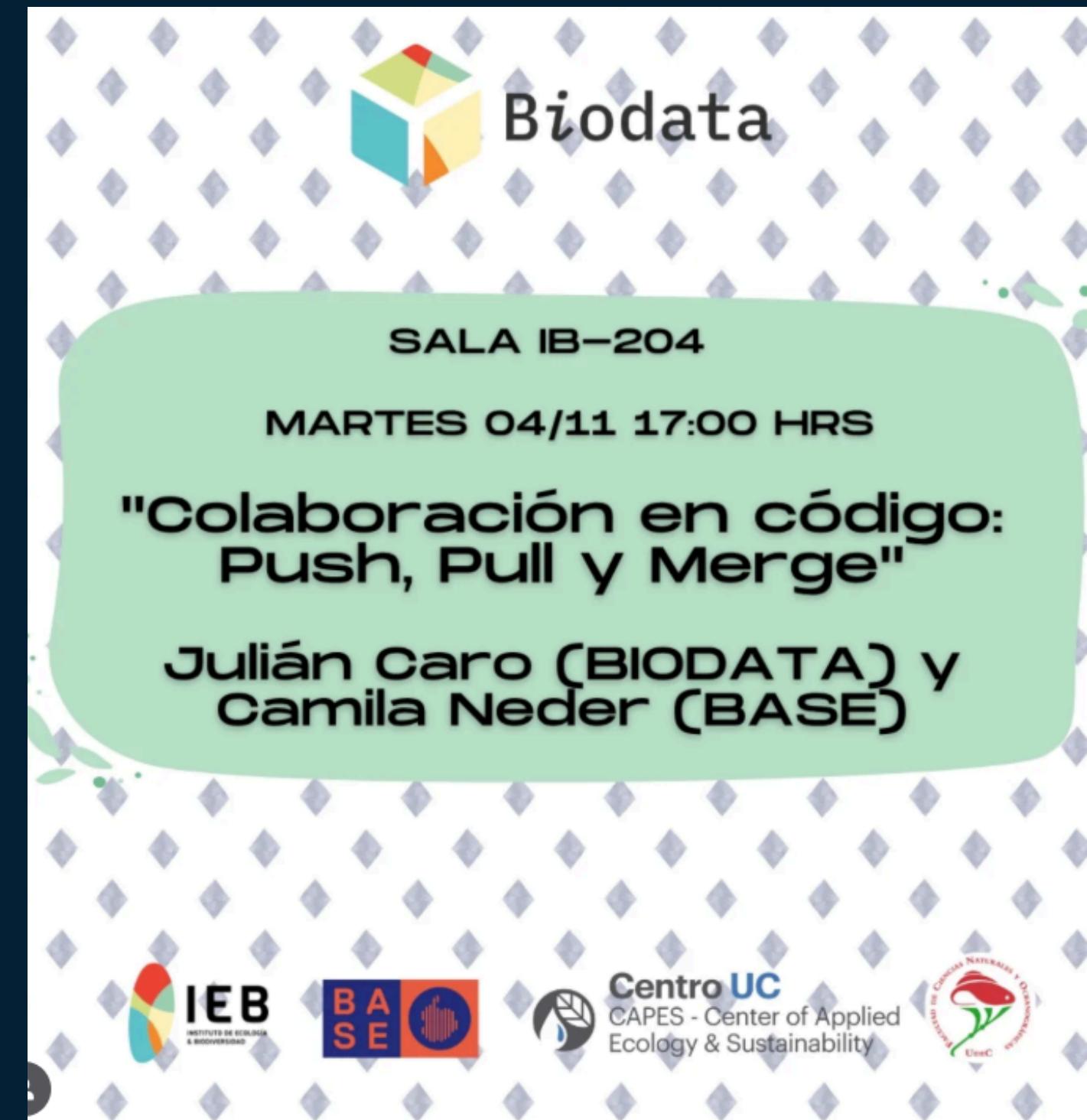


INSTITUTO
MILENIO
BIODIVERSIDAD
DE ECOSISTEMAS
ANTÁRTICOS Y
SUBANTÁRTICOS

PUSH, PULL, MERGE

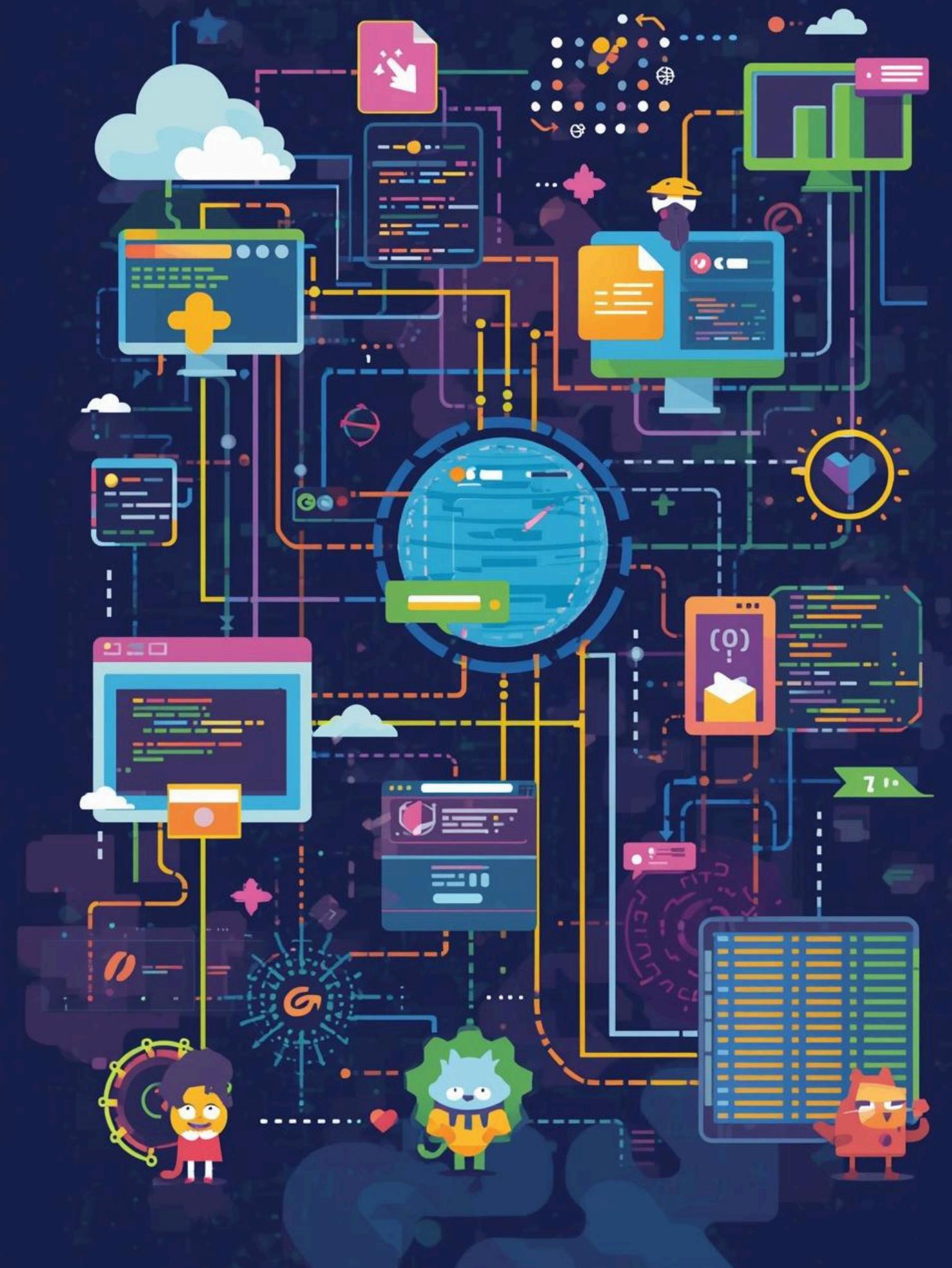
COLABORACIÓN EN CÓDIGO

CAMILA NEDER & JULIÁN CARO



Flujo de trabajo colaborativo

Sesión de hoy nace de una pregunta genuina para optimizar el flujo de trabajo colaborativo aplicando términos esenciales (repositorio, pull, push, merge, hacer un fork...) y comprendiendo las diferencias entre el trabajo local y en la website.



¿Qué es Git?

Un sistema de control

Git es un sistema de control de versiones **local** que permite a los desarrolladores **guardar y gestionar** el historial de cambios en sus proyectos, facilitando el mantenimiento, comparación y recuperación de versiones que tengan varios códigos fuente.



git



Git permite



- Copiar archivo en el que estamos trabajando en conjunto (fork)
- Modificar archivos de forma independiente y segura en tu propia rama personal
- Fusionar mediante combinación y de forma inteligente los cambios específicos en la copia principal de archivos, de modo que los cambios no afecten a las actualizaciones de otras personas
- Seguimiento de tus cambios y los de otras personas, trabajo en versión más actualizada del proyecto.

¿Qué es GitHub?

Plataforma en red - Website

Es la plataforma para alojar proyectos utilizando el sistema de control de versiones Git.



<https://docs.github.com/es/get-started/start-your-journey/about-github-and-git>



GitHub permite

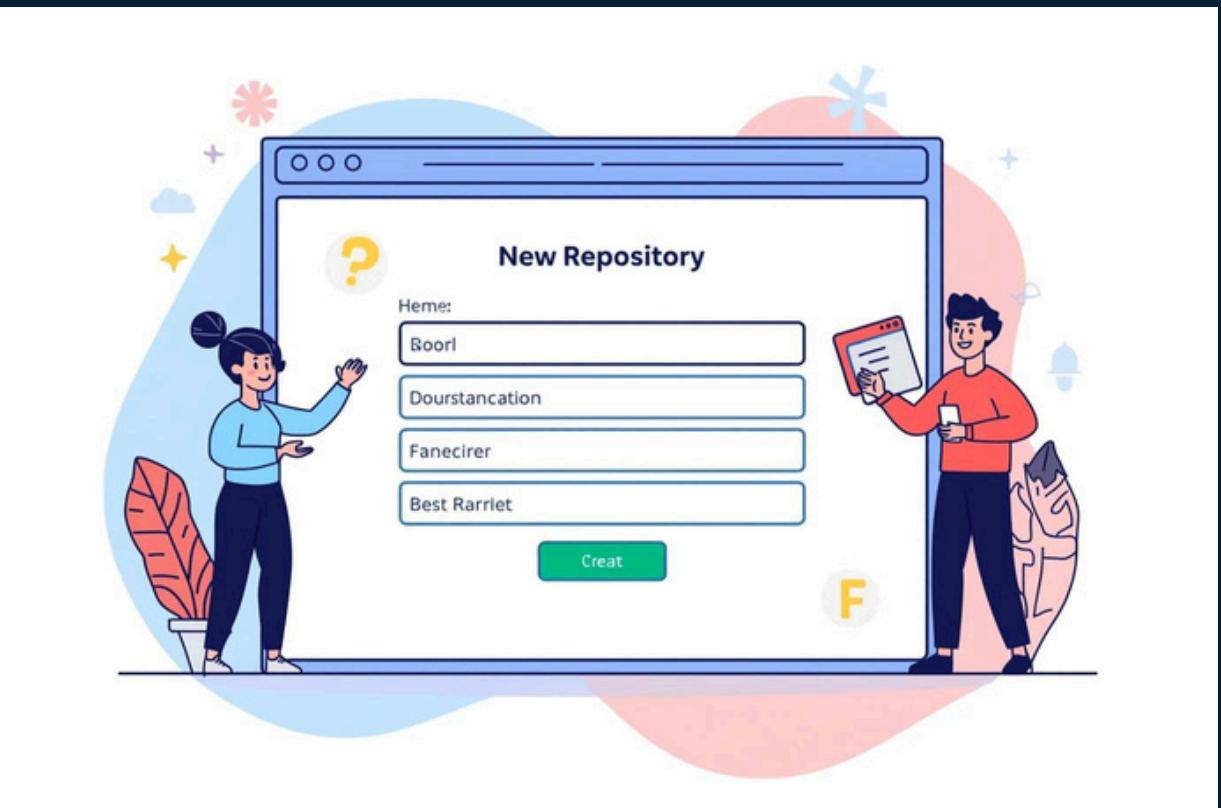


- Presentar o compartir el trabajo.
- Seguir y administrar los cambios en el código a lo largo del tiempo.
- Dejar que otros usuarios revisen el código y realicen sugerencias para mejorarlo.
- Colaborar en un proyecto compartido, sin preocuparse de que los cambios afecten al trabajo de los colaboradores antes de que esté listo para integrarlos.

¿Dónde empiezo?

Desde cero en GitHub

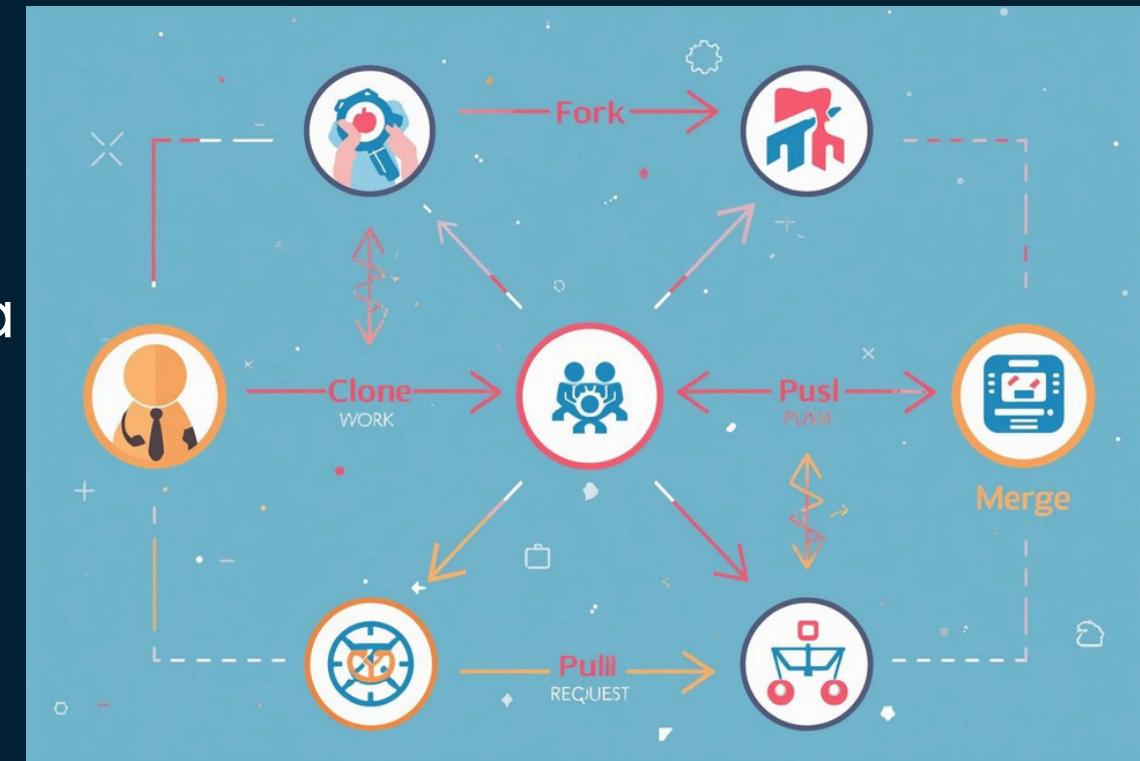
- 1) Creación de cuenta
- 2) Explorar otros usuarios y conectar
- 3) Descargar un código de interés
- 4) Cargar archivos en un repositorio "carpeta"
- 5) Aprender el "flujo de GitHub" y los términos claves del trabajo colaborativo (ramas, confirmaciones, solicitudes de cambios, fusiones mediante combinación).



Flujo típico de colaboración

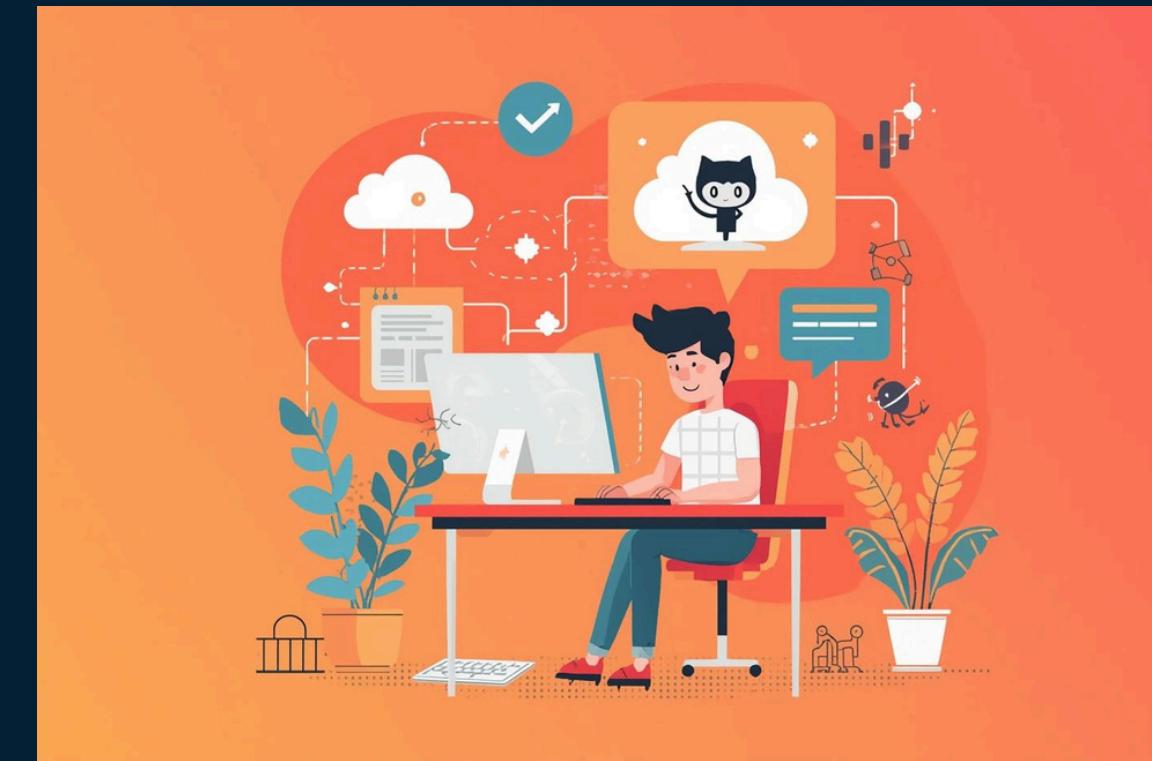
Fork del repo

Realiza una copia
del repositorio
original en tu
cuenta.



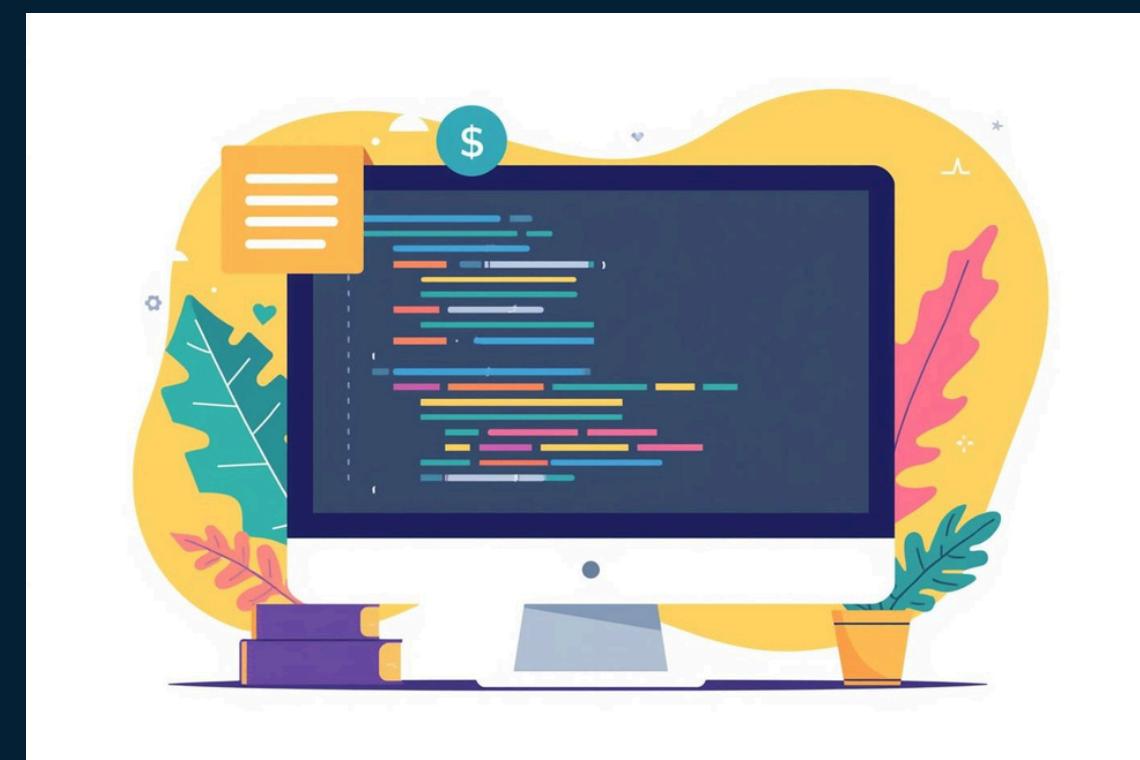
Clonar a
computadora

Descarga el
repositorio en tu
máquina local
para trabajar.



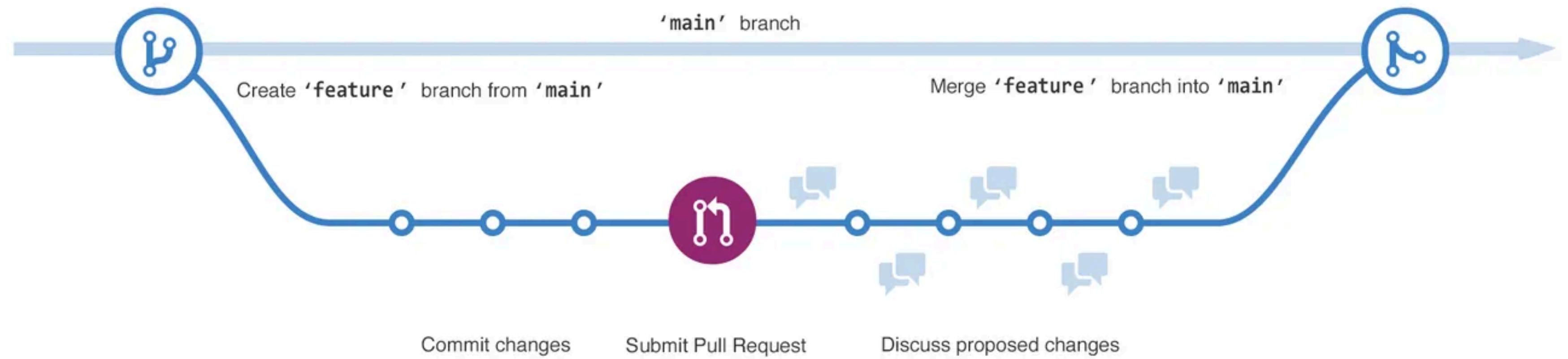
Crear rama de
trabajo

Utiliza git
checkout -b para
iniciar una nueva
función.



Hacer cambios
commit y
contribuir
Agrega tus
modificaciones y
envíalas al
repositorio remoto.





Fork, Pull, Push, Merge

Claves para colaboración efectiva

Fork

Forkear un repositorio te permite crear una copia del repositorio original en tu cuenta. Esto facilita realizar cambios sin afectar el trabajo de otros, promoviendo la colaboración.

Pull

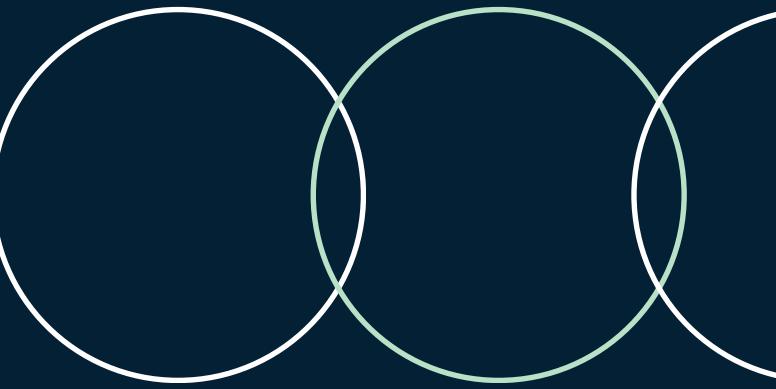
El comando Pull permite traer los cambios o modificaciones realizados en el repositorio remoto a tu copia local. “Jalar o descargar” los cambios que nosotros no tenemos. Así, puedes mantener tu trabajo actualizado y evitar conflictos en el código.

Push

Hacer un Push envía tus cambios locales al repositorio remoto. “Empuja” los cambios para compartir tu trabajo con otros colaboradores y asegurar que todos tengan acceso a las últimas actualizaciones.

Merge

Merge sirve para Unir



HELLO WORLD

- 1) Crear repositorio
- 2) Crear Read.me
- 3) Crear branch
- 4) Editar Read.me o realizar cambio
- 5) Confirmar cambios
- 6) Solicitud de incorporación de cambios (pull request)
- 7) Revisar e incorporar (pull)
- 8) Fusionar cambios (merge)

Ejercicio simple

 GitHub Docs

The AI powered developer platform



Hola mundo - Documentación de GitHub

Sigue este ejercicio de Hola mundo para aprender el flujo de trabajo de solicitud de cambios de GitHub.

 GitHub Docs

Buenas prácticas

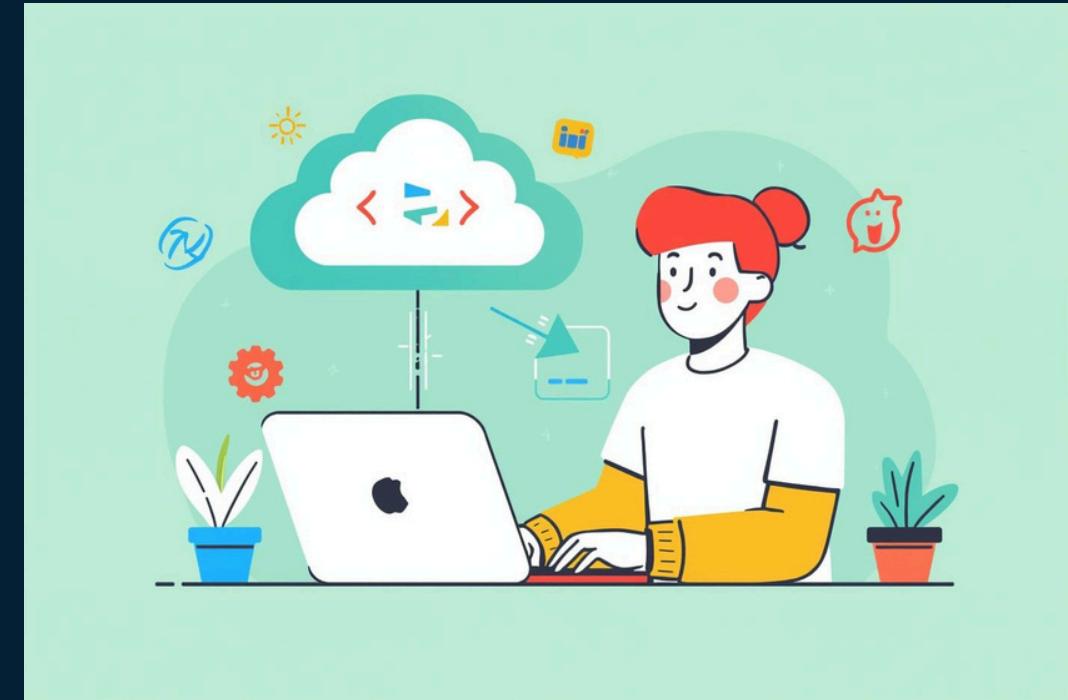
Commits claros

Realizar *commits* pequeños ayuda a mantener un buen historial.



Usar ramas

Para nuevas funciones promueve flujo trabajo ordenado.



Sincronizar frecuentemente

Usar git pull a menudo evita conflictos en el código.

Revisar pull requests

Revisar pull requests asegura la calidad antes de hacer merge.



Biodata



IEB
INSTITUTO DE ECOLOGÍA
& BIODIVERSIDAD



INSTITUTO
MILENIO
BIODIVERSIDAD
DE ECOSISTEMAS
ANTÁRTICOS Y
SUBANTÁRTICOS

PUSH, PULL, MERGE COLABORACIÓN EN CÓDIGO

EJEMPLO EN VIVO CON
REPOSITORIOS BIODATA

