

Bro, ¿dónde está mi branch?

GIT, BRANCHES & FROKS

PLATFORM: GITHUB

Presentación

Antonio Calabuig Puigvert (a.calabuigpuigvert@edu.gva.es)

Carrera profesional:

- Arpem Networks S.L.
- Km77
- NavelSystems

Proyectos Educativos:

• docuCentres (https://github.com/buig/docusenia)

docuCentres by Antonio Calabuig Puigvert is licensed under CC BY-NC-SA 4.0

¿Esto de qué va?

- Git y Github
- ¿Qué es una rama (branch)?
- ¿Qué es un fork?
- ¿Cuándo usarlos?
- Ejemplo práctico

¿Qué es una rama (branch)?

- Una línea paralela de trabajo.
- Útil para desarrollar nuevas
 funcionalidades, arreglar bugs o
 experimentar sin afectar la rama principal
 (main o master)

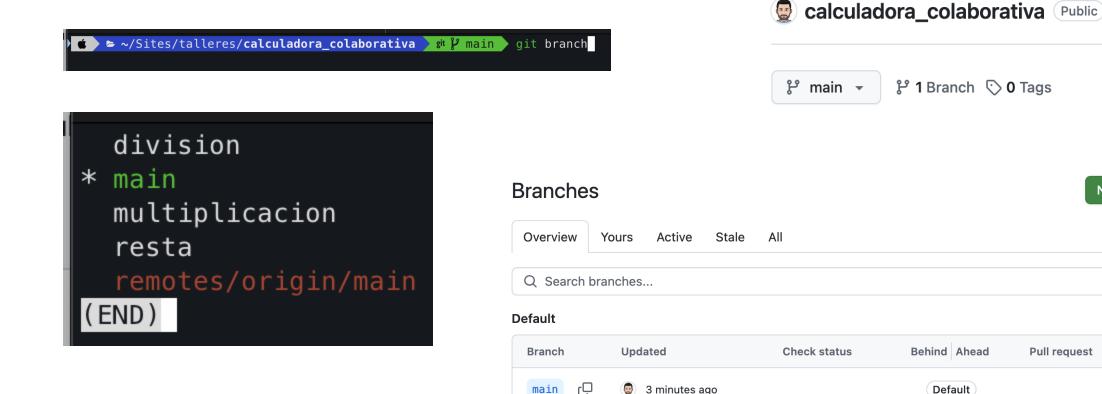
- ¿Por qué usar ramas?
 - Aislar trabajo.
 - Facilitar trabajo en equipo.
 - Permitir revisiones de código y pruebas antes de integrar.

Comandos básicos de ramas



- git branch –a # listado de ramas
- git branch nombre-de-la-rama # crear nueva rama
- git checkout -b nombre-de-la-rama # crear y cambiar a nueva rama
- git switch nombre-de-la-rama # cambiar de rama (Git 2.23+)
- git branch -m nombre-rama-viejo nombre-rama nuevo # renombrar la rama
- git merge nombre-de-la-rama # fusionar rama en la actual
- git branch -d nombre-de-la-rama. # eliminar rama

Visualización de ramas (cli / Github)



New branch

Pull request

Default

PR (Pull Request) en Github



Solicitud de fusión de ramas.



Aquí pueden intervenir otros componentes del equipo de desarrollo para:

Revisión de código Sugerencia de cambios Aceptación de los cambios Aceptación de la fusión de ramas



También pueden intervenir factores de CI/CD

Github Actions

¿Qué es un fork?

- Una copia de un repositorio en tu cuenta de GitHub.
- Desacoplado del original: puedes hacer cambios sin afectar el repositorio original



Cuando usar un fork:

- Cuando no tienes permisos para colaborar directamente.
- Para contribuir a proyectos open source.
- Para experimentar de forma segura.

Workflow con un fork

- 1. Hacer **fork** del repositorio.
- 2. Clonar el fork.
- 3. Crear una rama para cambios.
- 4. Hacer cambios y *commit*.
- 5. Push a tu fork.
- 6. Crear un *Pull Request* al repositorio original.

Repo Issues

Caso práctico

- Requisitos:
 - Conocimientos de GIT (clone, remote, commit, pull, push, branch, merge, checkout, switch)
 - o Cuenta de Github
 - Personal Access Token (Classic)
 - > https://USERNAME:TOKEN@github.com/buig/calculadora_colaborativa.git

Gestión de remotos y actualización del fork

- Se puede mantener el fork actualizado respecto al repositorio original.
- Para es necesario configurar dos remotos de forma local para el repositorio
 - o origin: url del repositorio original
 - o [forked_name]: url del fork del respositorio
- Comandos:
 - git fetch origin # obtener cambios del original
 - git merge origin/main. # fusionar la rama actual con la del original
- git push forked_name main # subir los cambios actualizados a tu fork
- o rebase [git pull origin main --rebase] # en ocasiones para poner tu rama actualizada respecto a los últimos cambios en el repositorio original

¿Qué más queréis saber?



GRÀCIES