



Fundamentos de GIT y GitHub

Fundamentos de GitHub (Parte I)

***Gestionar el código fuente
utilizando GitHub para
mantener un repositorio de
código remoto seguro y
permitir trabajo concurrente.***

- Unidad 1: Ambiente de desarrollo y sus elementos de configuración.
- Unidad 2: Elementos de la interfaz, navegación e interacción.
- Unidad 3: Fundamentos de GIT y GitHub.



Te encuentras aquí



¿Qué aprenderás en esta sesión?

Utilizar GitHub para compartir y respaldar tus proyectos

¿Qué ocurre si se elimina
la carpeta .git del
proyecto?

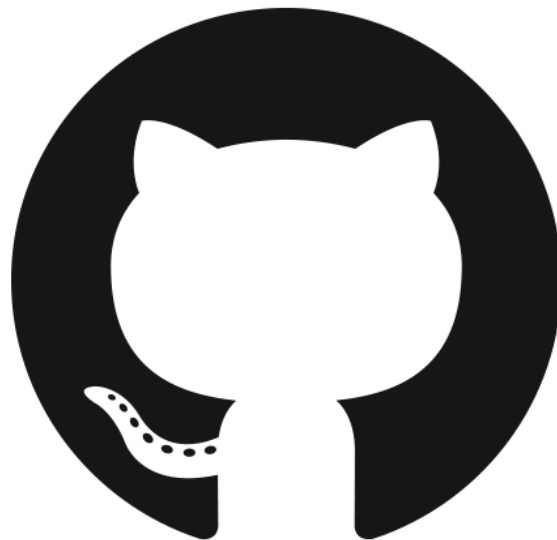


Plataformas de desarrollo colaborativo

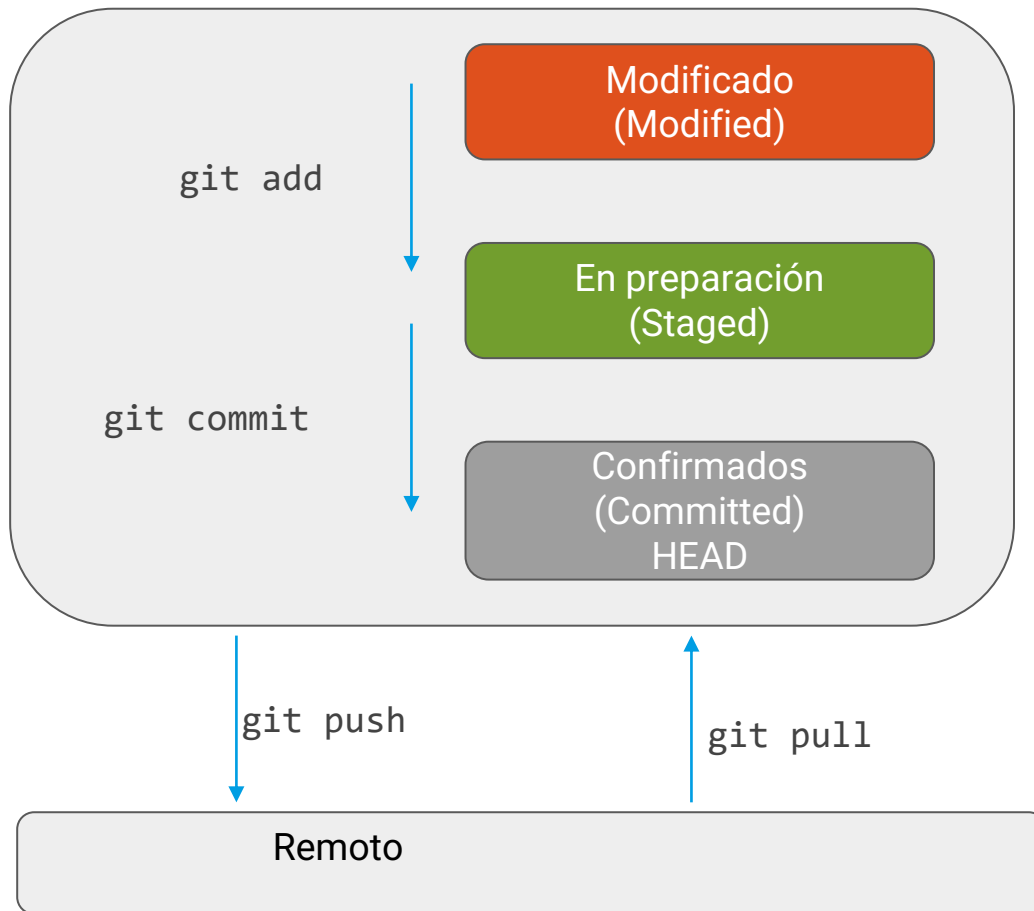
Las plataformas de desarrollo colaborativo nos permiten almacenar y gestionar proyectos Git, además de administrar los usuarios y sus permisos.

Existen diversas plataformas para este objetivo, como por ejemplo:

- [GitHub](#)
- [GitLab](#)
- [Bitbucket](#)
- [SourceForge](#)



Workflow



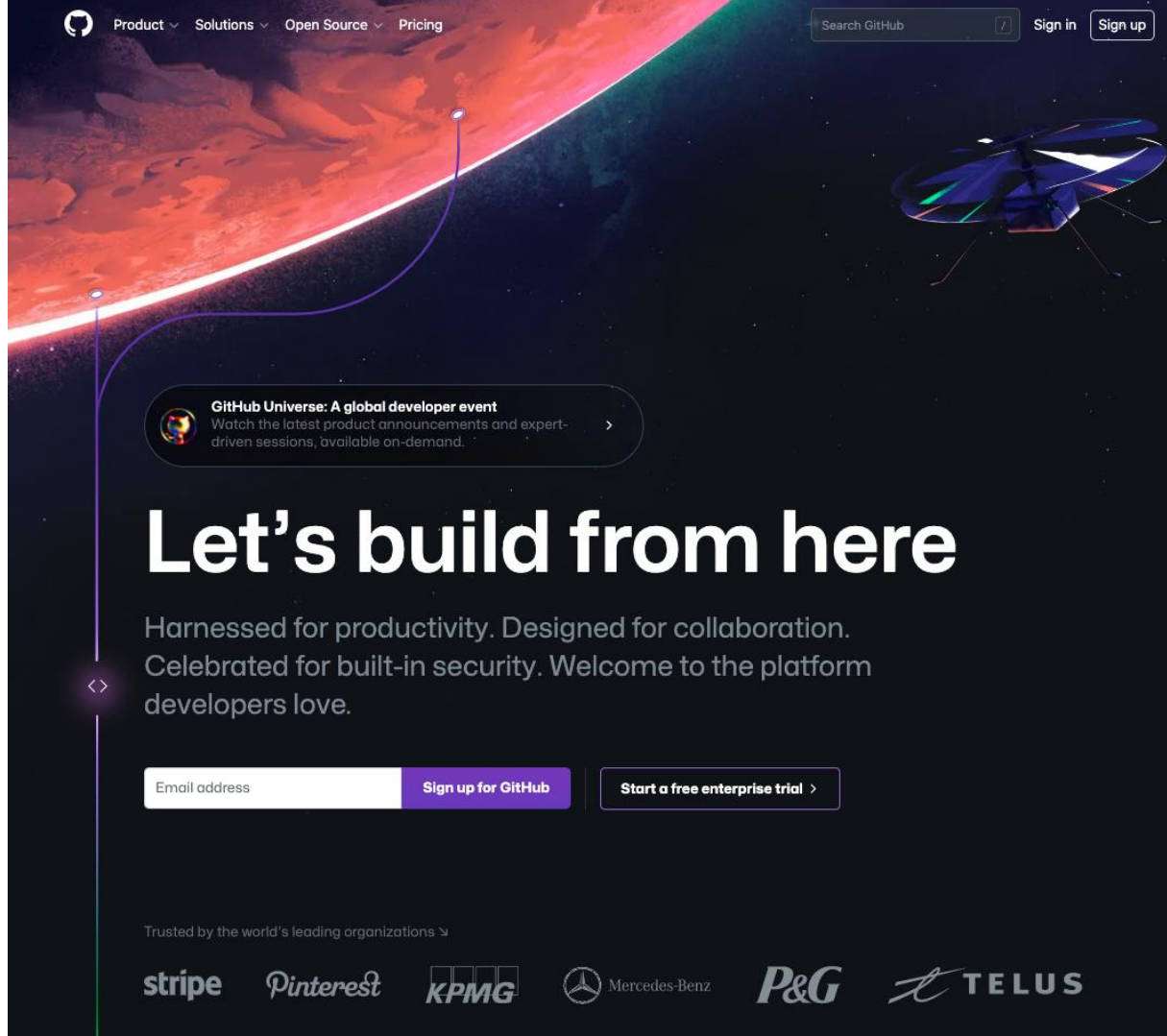
`/* GitHub */`

GitHub

¿Qué es?

Es una plataforma de desarrollo colaborativo que nos permite almacenar y gestionar proyectos Git, diseñado para proyectos en equipo.

{desafío}
latam_

The banner features a dark space background with a large, glowing red planet on the left and a satellite or probe on the right. A purple line with circular nodes connects the planet to a central event box and then down to a code symbol. The event box is titled 'GitHub Universe: A global developer event' and includes a small globe icon and a right arrow. The main headline 'Let's build from here' is in large white font, followed by a sub-headline in smaller white font. Below this is a registration section with an email input field, a purple 'Sign up for GitHub' button, and a 'Start a free enterprise trial' button with a right arrow. At the bottom, a row of logos for various companies is displayed, preceded by the text 'Trusted by the world's leading organizations'.

Product ▾ Solutions ▾ Open Source ▾ Pricing

Search GitHub Sign in Sign up

GitHub Universe: A global developer event
Watch the latest product announcements and expert-driven sessions, available on-demand. >

Let's build from here

Harnessed for productivity. Designed for collaboration.
Celebrated for built-in security. Welcome to the platform developers love.

<>

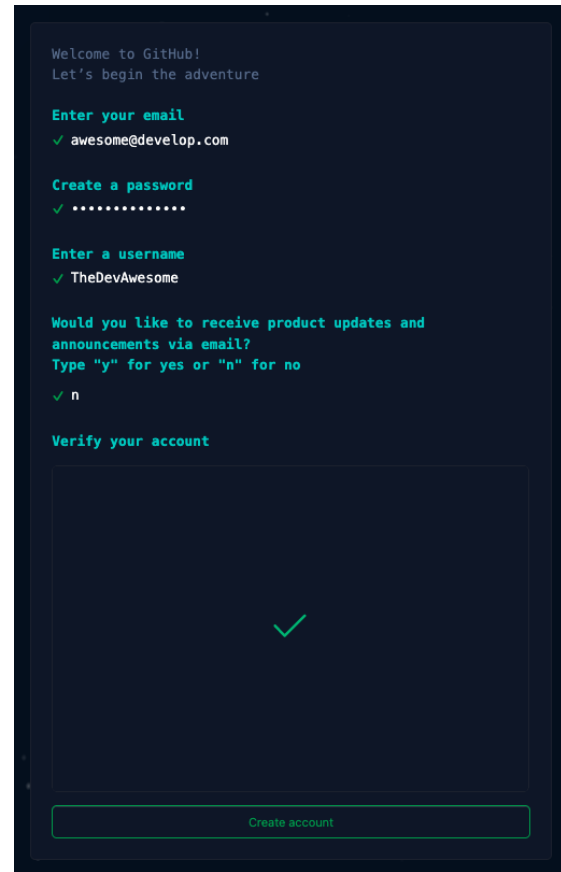
Email address **Sign up for GitHub** **Start a free enterprise trial >**

Trusted by the world's leading organizations ▾

stripe Pinterest KPMG Mercedes-Benz P&G TELUS

Creando una cuenta en GitHub

- Ingresa a <https://github.com/>.
- En la parte superior derecha están las opciones para ingresar (*Sign in*) o para crear una nueva cuenta (*Sign up*).





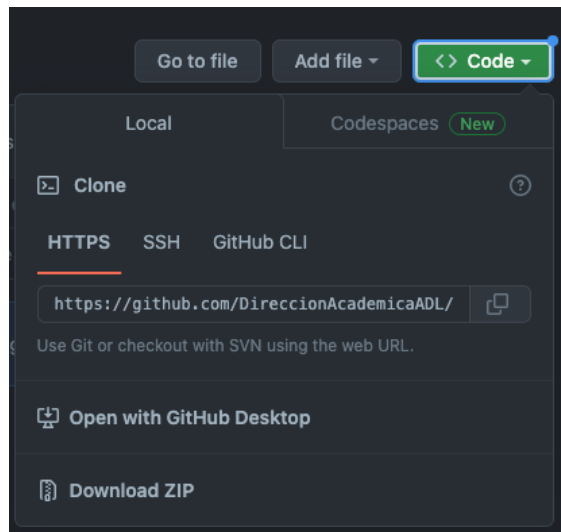
/* Clonar un repositorio */

Clonando un repositorio remoto desde GitHub

Obtener un repositorio

Al clonar un repositorio remoto, se obtiene una copia local con toda la historia Git registrada. En la página principal de cada proyecto se encuentra la estructura del proyecto, una descripción y otra información asociada.

- Por ejemplo, [entramos al proyecto Landing en GitHub.](#)
- Encima de la lista de archivos, el botón  permite descargar el código de distintas formas.
- Para clonar el repositorio con HTTPS, en «HTTPS» haz clic en  para copiar la dirección web.



Clonando un repositorio remoto desde GitHub

Remotes

En un terminal, cambiamos la ubicación en donde se quiera clonar el repositorio para luego ejecutar el comando `git clone` junto con la dirección copiada anteriormente.

```
$ git clone https://github.com/DireccionAcademicaADL/Landing.git
Cloning into 'Landing'...
remote: Enumerating objects: 40, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 40 (delta 6), reused 5 (delta 5), pack-reused 30
Receiving objects: 100% (40/40), 2.91 MiB | 2.00 MiB/s, done.
Resolving deltas: 100% (6/6), done.
```

Clonando un repositorio remoto desde GitHub

Remotes

Al clonar un repositorio se descarga una copia completa de toda la información del repositorio, y se pueden empujar (**push**) los cambios hacia el repositorio remoto en GitHub o se pueden traer (**pull**) los cambios de otras personas.

```
$ cd Landing
$ git status
On branch master
Your branch is up to date with 'origin/master'.


nothing to commit, working tree clean

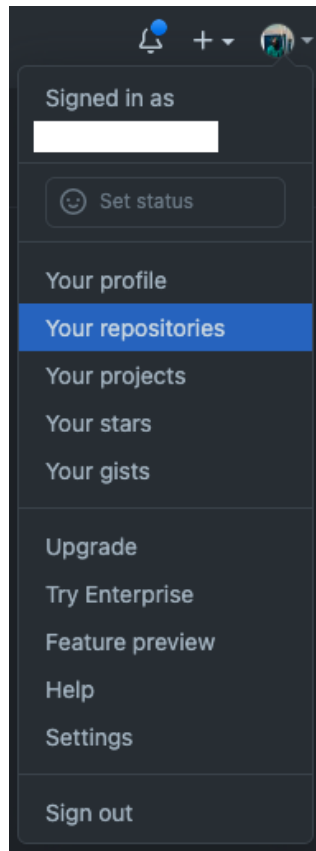
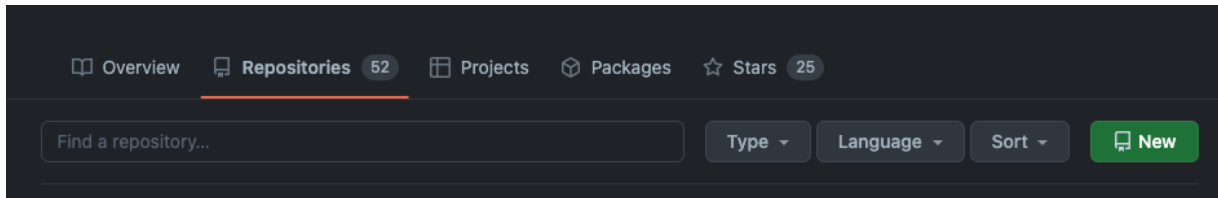
$ git remote -v
origin  https://github.com/DireccionAcademicaADL/Landing.git (fetch)
origin  https://github.com/DireccionAcademicaADL/Landing.git (push)
```

/* Crear un repositorio */

Acceder a tus repositorios

En la parte superior derecha encontramos toda la información relacionada con el usuario registrado.

Todo lo relacionado con los repositorios se encuentra agrupado en la opción: *Your repositories*, la cual lista todos los repositorios, permite buscar uno en particular y crear uno nuevo con el botón 



Creando un repositorio remoto

- **Public:** Cualquiera en internet puede ver este repositorio. Tu eliges quién puede hacer cambios.
- **Private:** Tu eliges quién puede ver y cambiar este repositorio.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



Repository name *

Great repository names are short and memorable. Need inspiration? How about [verbose-potato?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License: None ▾



You are creating a public repository in your personal account.

Create repository

Agregando el remoto

- **origin** es el nombre por defecto del repositorio remoto por defecto.
- El comando `git remote` permite eliminar un remoto con el comando `git remote rm <name>` y cambiar el nombre con el comando `git remote rename <old-name> <new-name>`


Quick setup — if you've done this kind of thing before

 Set up in Desktop or **HTTPS** **SSH** `https://github.com/awesomedev/gitexample.git` 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).


...or create a new repository on the command line

```
echo "# gitexample" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/[username]/gitexample.git
git push -u origin main
```




...or push an existing repository from the command line

```
git remote add origin https://github.com/[username]/gitexample.git
git branch -M main
git push -u origin main
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.



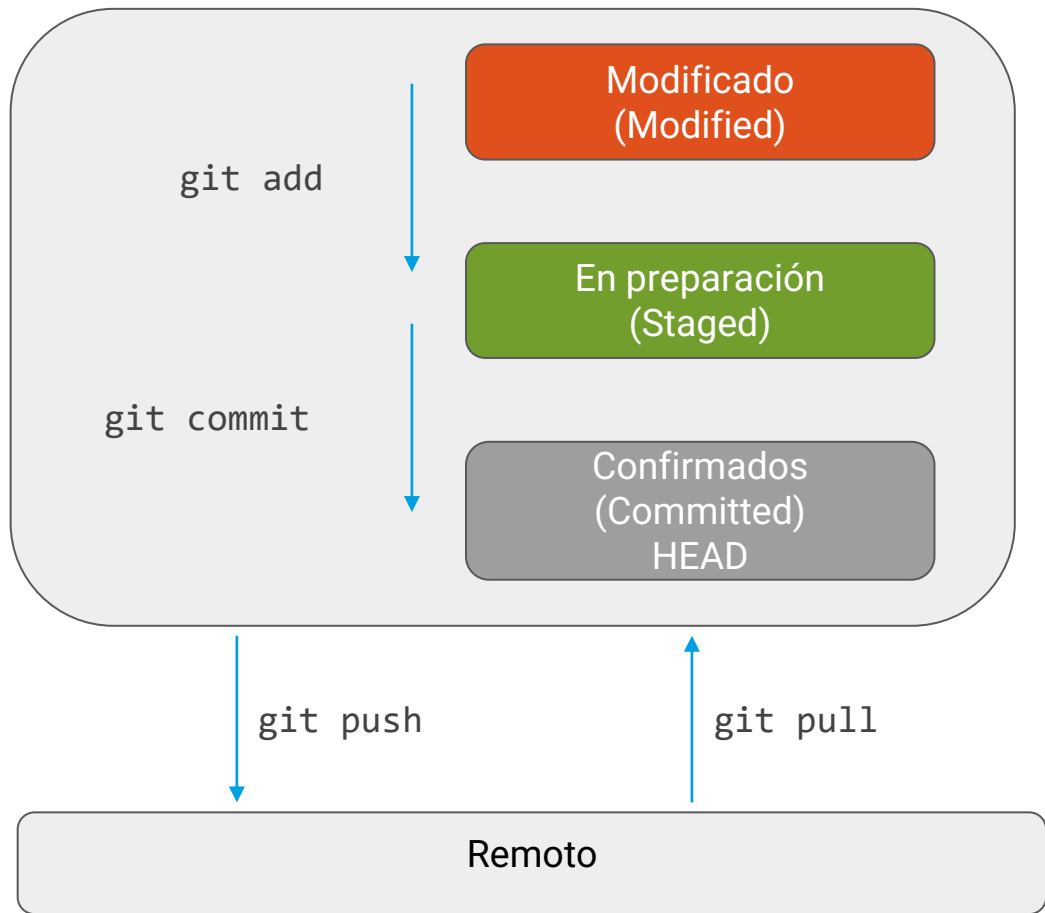
/* Interacción con el remoto */

Comandos básicos

- `git fetch` es el comando que le dice a tu git local que recupere la última información de los metadatos del remoto y verifica si hay algún cambio disponible, aunque no actualiza archivos.
- `git pull` además de hacer el “fetch” del remoto, trae los cambios del repositorio hacia el local.
- `git push` empuja los cambios locales hacia el repositorio remoto.



Workflow

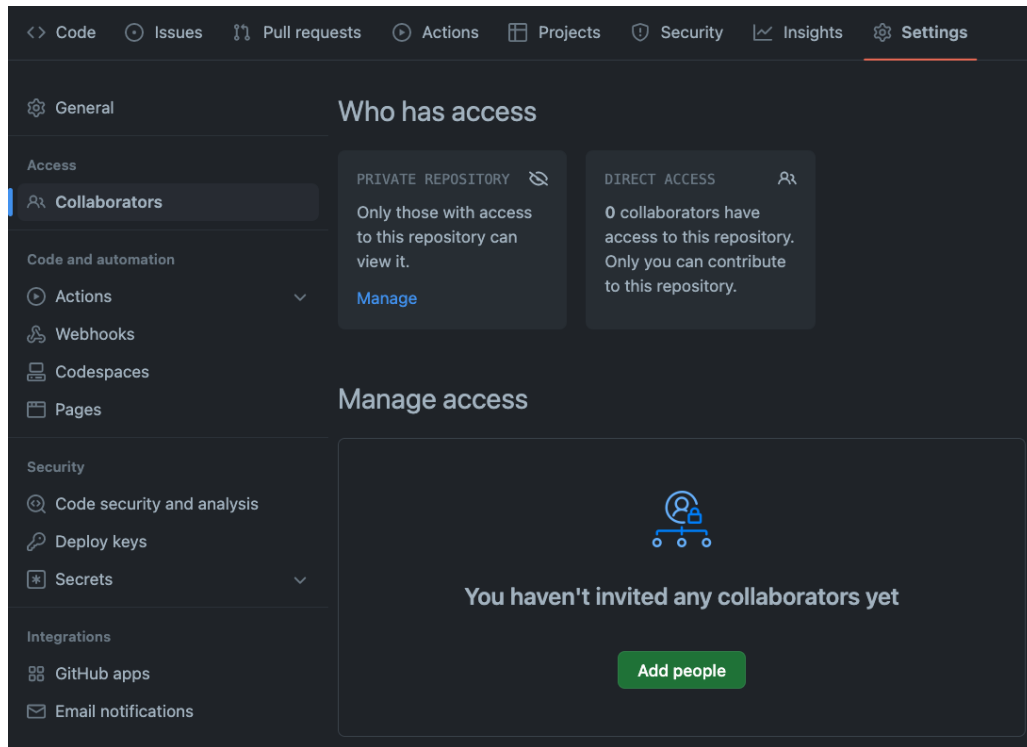


Permisos en el proyecto

Colaboradores

¿Podemos modificar el contenido de un repositorio público?

No, a no ser que estemos en la lista de colaboradores que se administra en la pestaña de configuraciones (Settings).



Alternativas



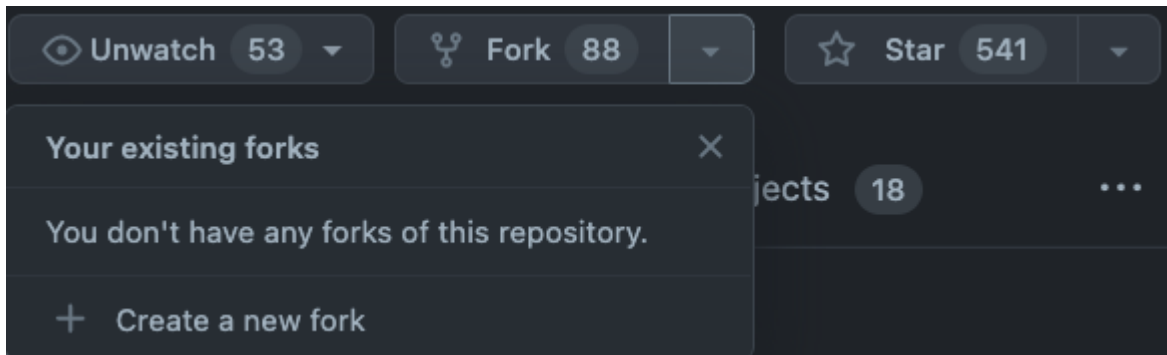
¿Qué podemos hacer si queremos modificar un proyecto?

- Generar una copia del repositorio público.
- Descargar el repositorio, generar cambios y subirlos.
- Crear una copia y asociarla a nuestra cuenta (*fork*).

Fork & Pull request

Fork

Fork es la acción de generar una copia exacta del repositorio en ese momento y asociar esa copia a nuestra cuenta, manteniendo la referencia al autor. Si el autor agrega más commits, estos no se ven reflejados en nuestro proyecto *“forkeado”*.

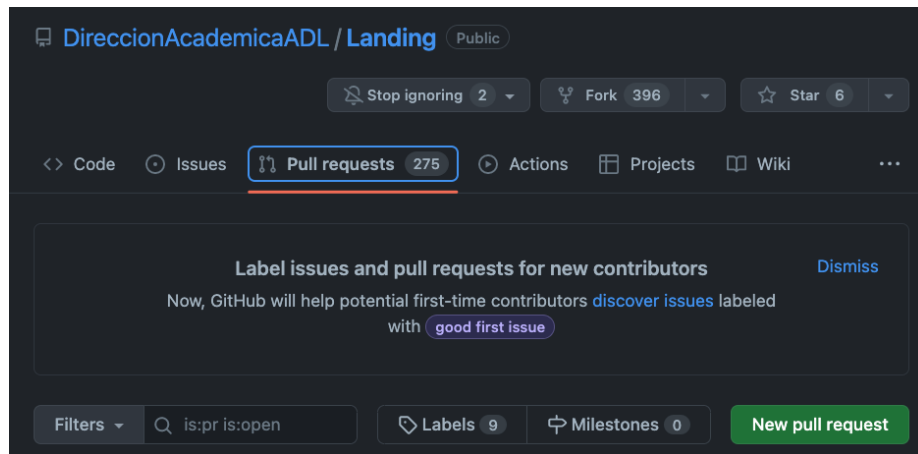


Pull request

PR

Pull request (PR) es la acción de abrir una solicitud para que nuestros cambios sean aceptados dentro del proyecto. Esta solicitud debe ser revisada antes de ser aprobada para mezclar nuestros cambios en el proyecto.

En la pestaña de Pull requests del proyecto, se encuentra la opción de crear una nueva petición para integrar los cambios.



Actividad



Actividad

1. Crear un fork del repositorio.
2. Clonar el repositorio *forkeado* a nuestro computador.
3. Generar algún cambio simple en el proyecto.
4. Commitear los cambios.
5. Hacer push a nuestro repositorio *forkeado*.
6. Crear un pull request describiendo los cambios implementados.



Git & GitHub

Git & GitHub

Git	GitHub
Software de versionamiento de archivos que registra los cambios en el tiempo.	Plataforma de desarrollo colaborativo que permite almacenar y gestionar proyectos Git. Se debe crear una cuenta.
Se instala y funciona de manera local en el computador. Toda la información del proyecto se encuentra en la carpeta .git dentro del proyecto.	Es una plataforma web en la que se debe crear una cuenta y permite tener una copia de nuestro proyecto.
Al ser local NO necesita de una conexión a internet para funcionar.	Al ser una plataforma web se necesita una conexión a internet.
La carpeta .git es visible solo para el usuario dueño del proyecto.	Administra los usuarios y los permisos para leer, escribir y modificar el proyecto compartido.

¿Cuáles son los beneficios de
las plataformas
colaborativas?





Próxima sesión...

Utilizar Git y Android Studio

{desafío}
latam_



{desafío}
latam_

*Academia de
talentos digitales*

