



Introducción a Git

Porque Drive no
es un gestor de cambios

Daniel Rodríguez
@guluc3m

Disponible en cloud.gul.es/ftp
Noviembre 2021

Tabla de contenidos

Introducción a Git

- ¿Qué es Git?
- ¿Para qué se usa?
- Instalación

Conceptos básicos

- Repositorios
- Ramas
- Commit
- Etiquetas

Plataformas

Archivos especiales

Ejemplo práctico

Preguntas

¿Qué es Git?

Herramienta de gestión de archivos que permite trabajar de forma colaborativa con ficheros de texto plano.
Existen otras alternativas como SVN, pero lo que se lleva ahora es Git.



¿Para qué se usa?

- ▶ Trabajo colaborativo. Desde 1 a N personas
- ▶ Control de versiones
- ▶ Workflow completo, desde actualizar código hasta testeo y despliegue
- ▶ Poder echarle la culpa a la gente :)



Instalación

Windows

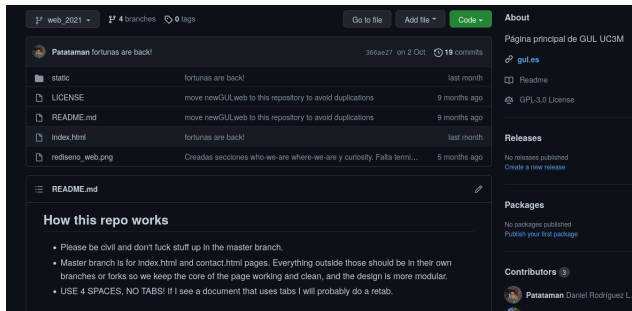
- ▶ Git: <https://git-scm.com/downloads/guis>
- ▶ Solo Github:
<https://desktop.github.com/>

Linux

- ▶ Consola: `apt-get install git`
- ▶ GUI:
<https://git-scm.com/download/gui/linux>

Repositorio

Un repositorio va a ser donde almacenemos nuestros archivos y código. Esa carpeta de Dropbox o Drive donde subes tus prácticas



Repositorio

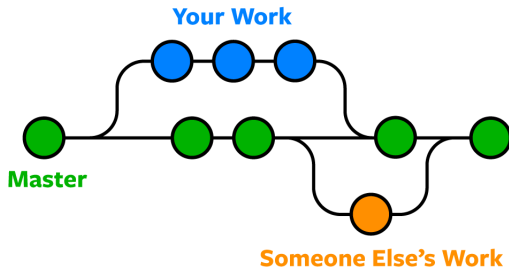
Un repositorio va a ser donde almacenemos nuestros archivos y código. Esa carpeta de Dropbox o Drive donde subes tus prácticas



Ramas / Branches

Ya tenemos toda nuestra porquería, digo código, metido dentro de un repositorio y es posible que queramos organizar a un nivel mayor.

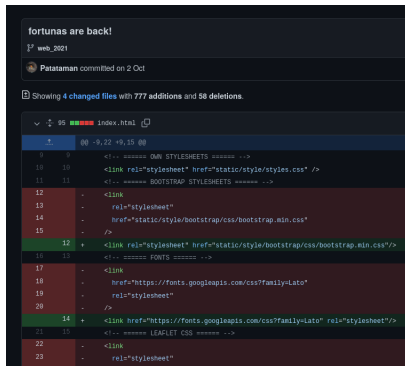
Separar código de desarrollo y producción, separar por personas... Esto son las ramas.



Commit

Ya tenemos un sitio donde guardar cosas (repositorio), podemos organizar los archivos que suba (branches), ¡pero no sabemos cómo subir ficheros!

Un commit es esta subida de ficheros que vamos a realizar y siempre va acompañado de un comentario



The screenshot shows a Git commit interface. At the top, the commit message is "fortunas are back!" and it was committed by "Patataman" on "2 Oct". Below this, it states "Showing 4 changed files with 777 additions and 58 deletions." The main part of the image shows a diff for the file "index.html". The diff highlights changes between two versions of the file. Line 12 shows a new line added: `<link rel="stylesheet" href="static/style/bootstrap/css/bootstrap.min.css"/>`. Line 14 shows a new line added: `<link href="https://fonts.googleapis.com/css?family=Lato" rel="stylesheet"/>`. The diff uses color coding: green for additions and red for deletions.

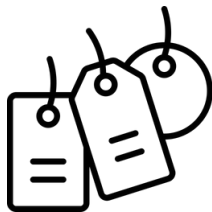
```
fortunas are back!
web_2021
Patataman committed on 2 Oct

Showing 4 changed files with 777 additions and 58 deletions.

index.html
@@ -9,22 +9,15 @@
 9 9      <!-- ===== OWL STYLESHEETS ===== -->
10 10      <link rel="stylesheet" href="static/style/styles.css" />
11 11      <!-- ===== BOOTSTRAP STYLESHEETS ===== -->
12 +      <link
13 -      rel="stylesheet"
14 -      href="static/style/bootstrap/css/bootstrap.min.css"
15 -      />
12 +      <link rel="stylesheet" href="static/style/bootstrap/css/bootstrap.min.css"/>
16 15      <!-- ===== FONTS ===== -->
17 -      <link
18 -      href="https://fonts.googleapis.com/css?family=Lato"
19 -      rel="stylesheet"
20 -      />
14 +      <link href="https://fonts.googleapis.com/css?family=Lato" rel="stylesheet"/>
21 15      <!-- ===== LEAFLET CSS ===== -->
22 -      <link
23 -      rel="stylesheet"
```

Etiquetas / Tags

Las etiquetas o tags son complementos que existen para marcar ciertos commits que hemos realizado a lo largo del tiempo. Normalmente se utilizan para marcar versiones o releases del programa en cuestión



Plataformas

- ▶ Github <- [link](#)
- ▶ Gitlab <- [link](#)
- ▶ Bitbucket <- [link](#)

 **r/learnprogramming**
u/Amantulsyan35 • 12h



Difference between GitHub and Git

If you were to explain a 10 year old what git and github are and the differences between them what would your answer be?

 **308** 

 76

 Share

 Award

 BEST COMMENTS ▾

 **raalag** 10h

 8 Awards

Git is porn GitHub is pornhub





 Reply

 **1.0k** 

Archivos especiales

`.gitignore`

Si creamos un archivo `.gitignore` en la raíz del repositorio podemos especificar ahí los archivos que no queremos que se suban.

`.gitkeep`

Si creamos un archivo `.gitkeep` dentro de una carpeta vacía, esta se subirá al repositorio a pesar de estar vacía.



Ejemplo práctico

Chuleta

Lista de comandos más frecuentes. Todos van precedidos de `git`

- ▶ **pull**: Para actualizar los cambios en el repositorio local
- ▶ **add**: Para añadir los ficheros que se quieren cambiar/añadir al repositorio
- ▶ **commit**: Para añadir un comentario de cambios sobre los ficheros añadidos con `add`
- ▶ **push**: Para subir los cambios al repositorio (recomendado hacer `pull` antes, por si acaso)
- ▶ **log**: Para ver el histórico de commits en el repositorio
- ▶ **status**: Para ver archivos cambiados, rama actual...
- ▶ **checkout**: Para cambiar de rama o deshacer cambios
- ▶ **diff** : Para ver explícitamente los cambios realizados en los archivos a subir

Preguntas, sugerencias y reclamos

