

# Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

## Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

## Publicación de páginas web en GitHub

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

Publicación de páginas web en GitHub

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Trabajo en colaboración

Publicación de páginas web en GitHub

# "FINAL".doc



FINAL.doc!



FINAL\_rev.2.doc

<http://phdcomics.com/comics/archive.php?comiciid=1531>



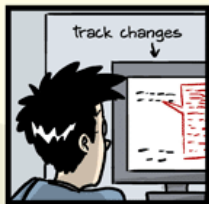
FINAL\_rev.6.COMMENTS.doc



FINAL\_rev.8.comments5.  
CORRECTIONS.doc



<http://phdcomics.com/comics/archive.php?comiciid=1531>



FINAL\_rev.18.comments7.  
corrections9.MORE.30.doc

FINAL\_rev.22.comments49.  
corrections.10. #@\$%WHYDID  
ICOMETOGRADSCHOOL?????.doc

<http://phdcomics.com/comics/archive.php?comiciid=1531>

# ¿Qué es el control de versiones y por qué debería importarte?

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

## Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

## Publicación de páginas web en GitHub

*El control de versiones es un sistema que **registra los cambios** realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que se puedan **recuperar** versiones específicas más adelante.<sup>1</sup>*

---

<sup>1</sup><https://git-scm.com/book/es/v1/>

# ¿Qué es el control de versiones y por qué debería importarte?

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

*El control de versiones es el cuaderno de laboratorio en el mundo digital.*

*Es lo que los profesionales usan para realizar un **seguimiento** de lo que han hecho y para **colaborar** con otras personas.*

***No sirve sólo para software:** libros, documentos, pequeños conjuntos de datos y cualquier cosa que cambie con el tiempo o que deba compartirse puede y debe almacenarse en un sistema de control de versiones.<sup>2</sup>*

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

---

<sup>2</sup><https://swcarpentry.github.io/git-novice/>



# Viajar en el tiempo

- ▶ Nada que haya sido sometido a un control de versiones se pierde jamás (*salvo que realmente quieras eliminarlo...*)
- ▶ **Todas** las versiones antiguas de un fichero se almacenan: un fichero se puede revertir a un estado anterior sin límites.

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# ¿Qué? ¿Cuándo? ¿Quién?

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

## Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

## Publicación de páginas web en GitHub

Un sistema de control de versiones registra:

- ▶ El detalle de los cambios realizados.
- ▶ La fecha y hora en la que fueron realizados.
- ▶ La persona que los realizó.

# Trabajo Colaborativo

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

- ▶ Cuando un equipo de personas trabaja conjuntamente en un proyecto, es posible que se produzcan cambios incompatibles en un mismo fichero.
- ▶ El sistema de control de versiones **impide** cambios simultáneos en un fichero. A cambio, permite la **resolución de conflictos** y los documenta.

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

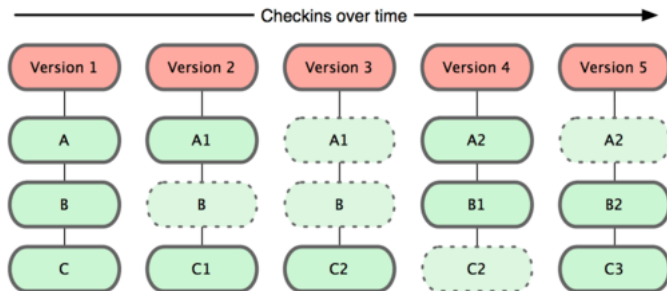
## Uso de git y GitHub

## Trabajo en colaboración

## Publicación de páginas web en GitHub

# Git es un Sistema de Control de Versiones

Git es una herramienta software (accesible mediante línea de comandos con git) que implementa un Sistema de Control de Versiones.



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

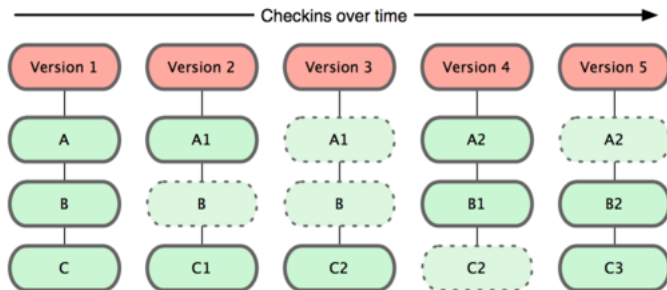
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

## Publicación de páginas web en GitHub

# Git es un Sistema de Control de Versiones

Cada vez que se ejecuta un cambio en una estructura de ficheros controlada con Git, realiza una «foto» del estado de los archivos en ese momento, y guarda una referencia a esa instantánea.



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

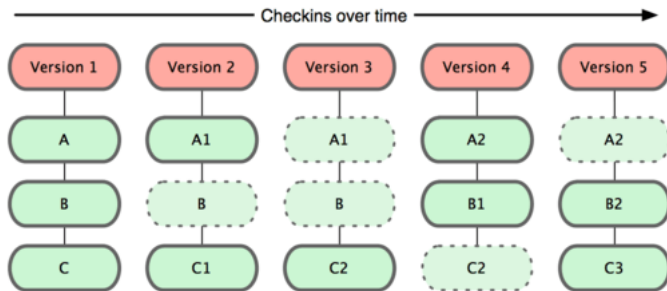
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Git es un Sistema de Control de Versiones

Por eficiencia, Git no almacena los archivos sino modificaciones sino un enlace al archivo anterior idéntico que ya está almacenado



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

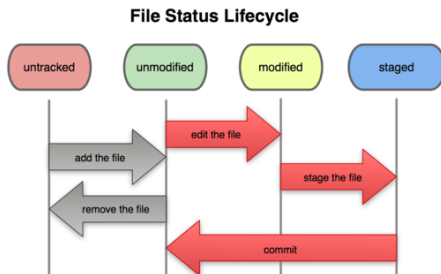
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Los estados de Git

- ▶ El desarrollador incorpora uno o varios ficheros al control de versiones. (*tracked*)
- ▶ Realiza modificaciones en los ficheros (*modified*).
- ▶ Incorpora esos ficheros modificados al área de preparación (*staged*).
- ▶ Finalmente, confirma todos los cambios del área de preparación: se realiza la instantánea de los ficheros. (*committed*)





# ¿Qué es GitHub?

- ▶ GitHub es la plataforma de alojamiento de código más importante a nivel mundial.
- ▶ Emplea el sistema de control de versiones git
- ▶ Ofrece una amplia variedad de funcionalidades
  - ▶ Alojamiento de código
  - ▶ Revisión de código
  - ▶ Trabajo colaborativo
  - ▶ Publicación de páginas web

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

## Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

## Publicación de páginas web en GitHub

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

Publicación de páginas web en GitHub

Conceptos básicos

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Publicación de páginas web en GitHub

# Creación de una cuenta en GitHub

<https://github.com/join>

The screenshot shows the GitHub registration page. At the top, it says 'Join GitHub' and 'The best way to design, build, and ship software.' Below this are three steps: 'Step 1: Set up your account' (active), 'Step 2: Choose your plan', and 'Step 3: Tailor your experience'. The 'Set up your account' section includes fields for 'Username', 'Email address', and 'Password', each with a red asterisk indicating it's required. Below the 'Email address' field, there's a note: 'We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.' Below the 'Password' field, there's a note: 'Make sure it's more than 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more](#).' At the bottom of this section is a link 'Verify account'. To the right of the form, there's a box titled 'You'll love GitHub' with the following text: 'Unlimited public repositories', 'Unlimited private repositories', and a list of three benefits: 'Limitless collaboration', 'Frictionless development', and 'Open source community', each preceded by a green checkmark.

Más información en [New GitHub account](#)

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

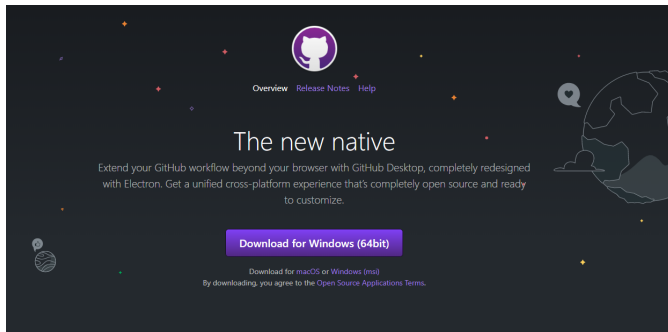
Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# Instalación de GitHub Desktop

<https://desktop.github.com/>



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Conectamos Git, GitHub y GitHub Desktop

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

- Una vez instalado comienza el proceso de autenticación, usando las credenciales del paso anterior<sup>3</sup>.

*File > Options > Accounts > Sign In*

- A continuación, conectamos la información de usuario con Git<sup>4</sup>.

*File > Options > Git*

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

---

<sup>3</sup>Más información en [Authenticating to GitHub](#).

<sup>4</sup>Más información en [Configuring Git](#).

# Remoto y Local

- ▶ **Github.com** aloja los **repositorios remotos** (nube).
- ▶ En tu ordenador trabajas con una **copia local** del repositorio. Otros desarrolladores tendrán sus propias copias locales.
- ▶ La(s) copia(s) local(es) y el repositorio remoto deben estar **sincronizados** mediante diferentes comandos de git.

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

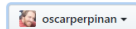
# Nuevo repositorio *remoto* desde github.com

<https://github.com/new>

## Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



Repository name \*

Great repository names are short and memorable. Need inspiration? How about **refactored-spoon**.

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▼

Add a license: **None** ▼



Create repository

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

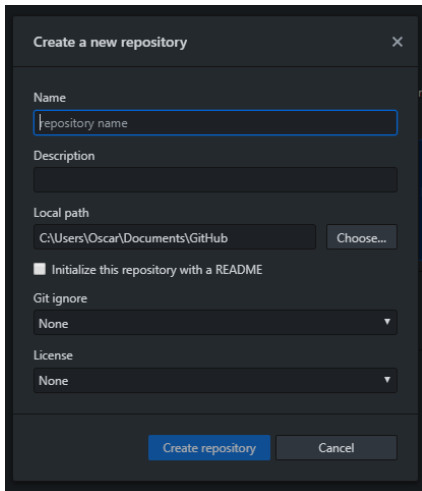
Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub



# Nuevo repositorio *local* desde GitHub Desktop

*File > New Repository*



The screenshot shows the 'Create a new repository' dialog box in GitHub Desktop. The dialog has a dark theme and a close button (X) in the top right corner. It contains the following fields and options:

- Name:** A text input field with the placeholder text 'Repository name'.
- Description:** A text input field.
- Local path:** A text input field showing 'C:\Users\Oscar\Documents\GitHub' and a 'Choose...' button to the right.
- Initialize this repository with a README:** A checkbox that is currently checked.
- Git ignore:** A dropdown menu with 'None' selected.
- License:** A dropdown menu with 'None' selected.

At the bottom of the dialog, there are two buttons: 'Create repository' (highlighted in blue) and 'Cancel'.

Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Decisiones al crear un repositorio

- ▶ Elige un `.gitignore` adecuado al proyecto: Véase <https://github.com/github/gitignore>.
- ▶ No olvides inicializar y cumplimentar el `README.md`. Para el formato véase [Formatting syntax](#).
- ▶ Elige una **licencia** adecuada a tu proyecto y a tus intereses actuales y futuros. Véase <https://choosealicense.com>.

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

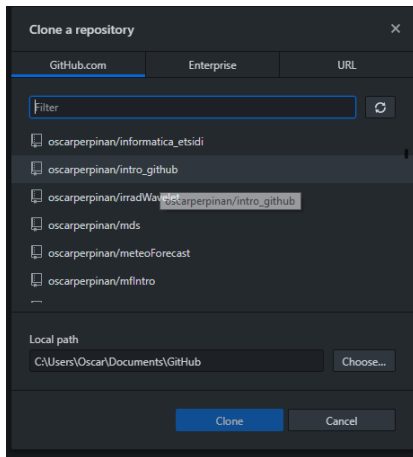
Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# Clonar un repositorio remoto

Si hemos creado el repositorio desde github.com (*repositorio remoto*), hay que clonarlo para poder trabajar con él (*copia local*).

*File > Clone Repository*



Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

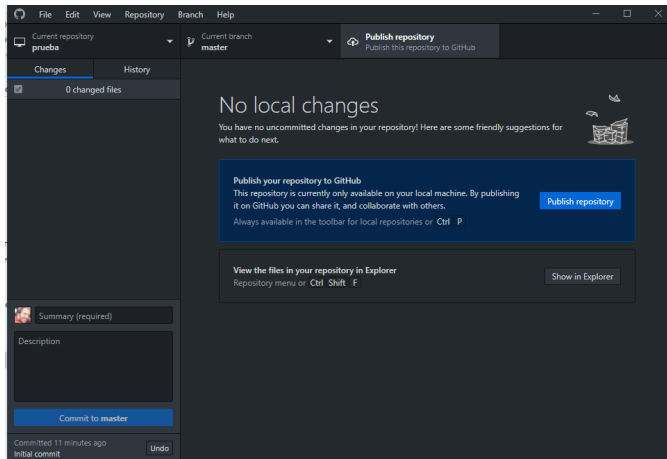
Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# Publicar un repositorio local

Si hemos creado el repositorio desde GitHub Desktop (*repositorio local*), hay que publicarlo en github.com (*repositorio remoto*)



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

## Publicación de páginas web en GitHub

Conceptos básicos

Uso de git y GitHub

Primeros Pasos

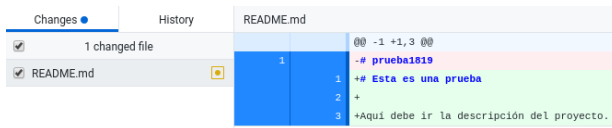
Flujo de Trabajo

Trabajo en colaboración

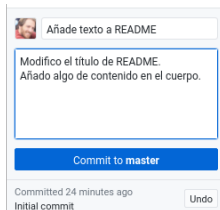
Publicación de páginas web en GitHub

# Cambios en la copia local

1. Modifica los ficheros de la copia local.
2. Añade los cambios realizados a la siguiente «instantánea» del repositorio (`git add`)



3. Confirma los cambios, escribiendo un resumen de lo realizado (`git commit`)



# Como escribir un buen mensaje de commit

- ▶ *Commit early and often*: cada commit debe incluir cambios pequeños y coherentes.
- ▶ Escribir un mensaje de calidad al ejecutar cambios facilita tanto el trabajo personal como la colaboración en equipo con Git.<sup>5</sup>

## Recomendaciones

- ▶ El **título** debe ser **conciso** (50 caracteres) y escrito en imperativo (*Do something...*)
- ▶ El **cuerpo** debe explicar **el qué y el por qué del cambio**, no el cómo, comparando con el comportamiento anterior al cambio.
- ▶ Se pueden incluir referencias a *issues* (ver a continuación) con #XX siendo XX el número de la *issue*.

<sup>5</sup><https://chris.beams.io/posts/git-commit/>

# Histórico de cambios

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Los cambios confirmados con commit se anotan en la historia (git log)

*View > History*

Changes History

No branches to compare

**Añade texto a README**  
Oscar Perpiñán Lamigueiro committed 7 ...

**Initial commit**  
Oscar Perpiñán Lamigueiro committed 3...

**Añade texto a README**  
Oscar Perpiñán Lamigueiro committed eb5a029 1 changed file

Modifico el título de README.  
Añado algo de contenido en el cuerpo.

README.md

	@@ -1 +1,3 @@
1	-# prueba1019
	+# Esta es una prueba
2	+
3	+Aquí debe ir la descripción del proyecto.

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub



# Publicar cambios al repositorio remoto

- Para sincronizar los cambios realizados **desde la copia local hasta el repositorio remoto** hay que publicar mediante `git push`.

*Repository > Push*

- A partir de este punto, la copia local y el repositorio remoto están sincronizados.

## Importante

En el caso de repositorios compartidos, antes de un `git push` es imprescindible actualizar la copia local incorporando los cambios del repositorio con `git pull`.

Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Recibir cambios de un repositorio remoto

Para obtener en la copia local los cambios recientes que existan en el repositorio hay que emplear `git pull`, que es la combinación de la secuencia:

1. `git fetch`, para obtener los cambios recientes del repositorio remoto.
2. `git merge`, para combinarlos con la copia local.

*Repository > Pull*

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

1. Realiza modificaciones en los ficheros de la copia local.
2. **COMMIT** :: Confirma los cambios con un mensaje informativo.
3. **PULL** :: Incorpora los cambios del repositorio remoto a la copia local.
4. **PUSH** :: Publica los cambios de la copia local al repositorio remoto.

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

Publicación de páginas web en GitHub

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

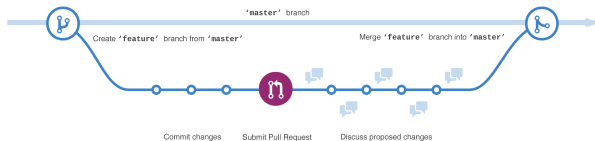
Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Rama master



En un repositorio de GitHub existe una rama (*branch*) que se usa por defecto: **master**<sup>6</sup>.

Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

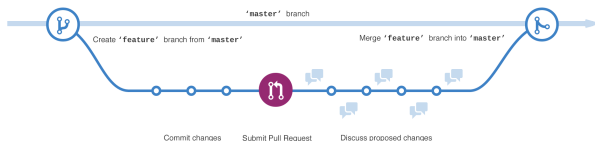
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

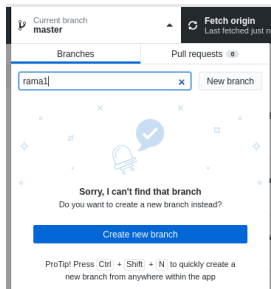
<sup>6</sup>Understanding the GitHub Flow

# Ramas para facilitar la colaboración



Cuando hay varias personas trabajando sobre un mismo repositorio, es necesario crear nuevas ramas para evitar conflictos.

De esta forma, cada persona implementa **cambios** en una **rama determinada** de forma paralela al resto del equipo.



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

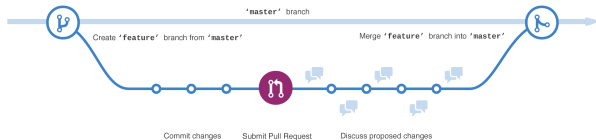
Ramas

Persiguiendo a los bichos

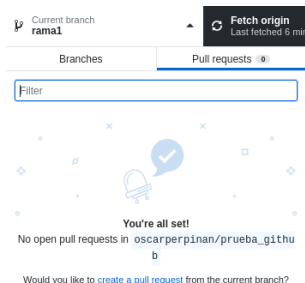
Herramientas gráficas para el análisis de un repositorio

## Publicación de páginas web en GitHub

# Combinación de código

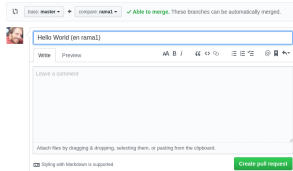


Cuando los cambios están listos y confirmados (*commit* + *push* en la rama específica), se realiza una petición (*pull request*) para combinar estos cambios en la rama **master**.



## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

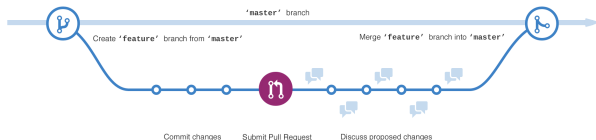
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

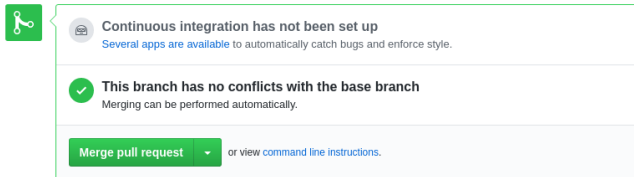
## Publicación de páginas web en GitHub



# Combinación de código



El coordinador del proyecto es el encargado de revisar cada petición y, si todo está correcto, incluir los cambios (*merge*) en la rama **master**.



Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

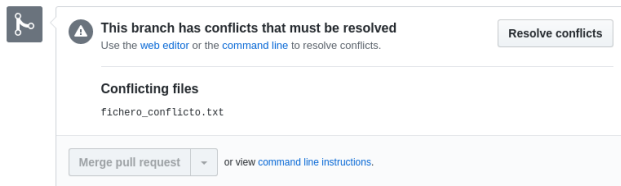
Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Resolución de conflictos

Si no se pueden combinar los cambios automáticamente se produce un conflicto (por ejemplo, cuando dos usuarios modifican un mismo fichero).



Un conflicto se debe resolver manualmente.



Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# Consejos

- ▶ **No olvides hacer *pull* antes de iniciar una nueva interacción con el repositorio.**
- ▶ Recuerda las recomendaciones sobre un buen mensaje de `git commit`.
- ▶ **Organización previa:** las **tareas** asignadas a un rama deben ser **independientes** de las otras ramas para evitar conflictos.
- ▶ Cuando el trabajo en una rama ha concluido, hay que **combinar cambios con master lo antes posible** para reducir la posibilidad de conflicto. Las ramas accesorias utilizadas se pueden eliminar una vez finalizado el proceso.
- ▶ Este proceso se debe repetir tantas veces como sea necesario para realizar cambios de forma colaborativa.

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Issues

Todos los repositorios de GitHub tienen una sección denominada «Issues»<sup>7</sup> a modo de *bug tracker*.

Pueden usarse para seguimiento de fallos, mejoras, tareas, etc.

The screenshot shows the GitHub Issues interface for the 'cluster' repository. At the top, there's a search bar with the query 'is:open label:cluster' and buttons for 'Filters', 'Labels', and 'Milestones'. Below the search bar is a link to 'Clear current search query, filters, and sorts'. The main content area displays a list of 13 open issues. Each issue entry includes a status icon (e.g., 'S-confirmed-bug'), the issue title, a list of labels (e.g., 'cluster', 'iojs-backport', 'net'), and a comment count. The issues listed are:

- net::Server.unref() failed on cluster mode (2 comments)
- ENOTSUP errorException in NodeJS using mssql in cluster worker (1 comment)
- listen() doesn't work with cluster on "node -e" (4 comments)
- cluster round robin could know about ipc send back pressure (feature-request, 4 comments)
- cluster: v0.11.13, cluster crashed in windows 2008r2 running on VMWare. (asynclistener, 10 comments)
- Cluster worker.kill(signal) does not pass the signal parameter (10 comments)
- cluster: implement API for pluggable distribute() for round-robin scheduler (feature-request, 6 comments)

<sup>7</sup><https://guides.github.com/features/issues/>

Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub

# Estructura de una issue

Una issue es un tablero de discusión en el que pueden participar los responsables del repositorio y cualquier usuario de GitHub.

**Debe** contener un título y una descripción.

**Puede** contener etiquetas, metas, y responsables.

Running cluster "hello world" on Win7 x64 returns -1073741819  
#4707

 **Open** turanuk opened this issue on Feb 3, 2013 · 3 comments



turanuk commented on Feb 3, 2013

Running code listed here: [http://nodejs.org/api/cluster.html#cluster\\_how\\_it\\_works](http://nodejs.org/api/cluster.html#cluster_how_it_works) seems to cause the worker to die with an error code of -1073741819 referring to an access violation. I assume this is a port conflict because if I change the code to only spawn a single worker the problem goes away and I'm able to access the web server. I tried port 3000 and ran into the same thing. Thoughts/debugging tips?



bnoordhuis commented on Feb 10, 2013

Member

Ah, good old 0xc0000005. If you're using the official binary with no native add-ons, that obviously shouldn't happen. Try hooking up the processes to a debugger to see where the access violation happens.



bnoordhuis commented on Feb 10, 2013

Member

Also, what happens when you run the test suite? You invoke it with `python tools/test.py simple`.



turanuk commented on Feb 12, 2013

Assignees

No one assigned

Labels

cluster

windows

Projects

None yet

Milestone

No milestone

Notifications

 **Subscribe**

You're not receiving notifications from this thread.

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

Uso de git y  
GitHub

Primeros Pasos

Flujo de Trabajo

Trabajo en  
colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

Publicación de  
páginas web en  
GitHub

# Contenido de una issue

- ▶ En la descripción de una issue se debe suministrar toda la información posible para el responsable del repositorio, **incluyendo un ejemplo mínimo, completo y verificable**<sup>8</sup>.
- ▶ El contenido será formateado como Markdown (incluye un *preview*)<sup>9</sup>.
- ▶ Se pueden incluir referencias al código y a otras issues<sup>10</sup>.

---

<sup>8</sup><https://stackoverflow.com/help/mcve>

<sup>9</sup>Veáse la guía [Basic Writing and Formatting syntax](#).

<sup>10</sup>Veáse la guía [Autolinked references and urls](#).

Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

- Ramas

- Persiguiendo a los bichos

- Herramientas gráficas para el análisis de un repositorio

Publicación de páginas web en GitHub



Toda la actividad realizada en un repositorio puede verse de manera gráfica a través del botón *Insights* en la web del repositorio en GitHub<sup>11</sup>. Por ejemplo,

- ▶ Contribución de los integrantes del equipo
- ▶ Estructuras de ramas de un repositorio
- ▶ Histórico de cambios en un repositorio

---

<sup>11</sup>Más detalles en [Ver información del repositorio de forma gráfica.](#)

# Contribución de los integrantes del equipo

Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

## Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio

## Publicación de páginas web en GitHub

Aug 14, 2011 - Jul 23, 2014

Contributions to master, excluding merge commits

Contributions: Commits ▾



# Estructura de ramas de un repositorio

# Introducción al control de versiones y trabajo colaborativo con GitHub

Oscar Perpiñán  
Lamigueiro

## ¿Qué es el control de versiones?

## ¿Qué son Git y GitHub?

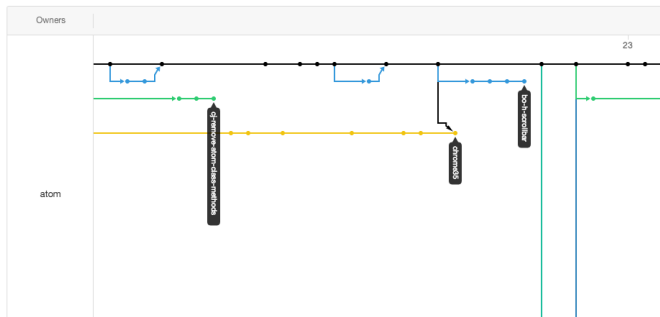
## Primeros Pasos

Flujo de Trabajo

Ramas

## Persiguiendo a los bichos

Herramientas gráficas para el análisis de un repositorio



# Cambios en un repositorio

Introducción al  
control de  
versiones y trabajo  
colaborativo con  
GitHub

Oscar Perpiñán  
Lamigueiro

## Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

## Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

## Trabajo en colaboración

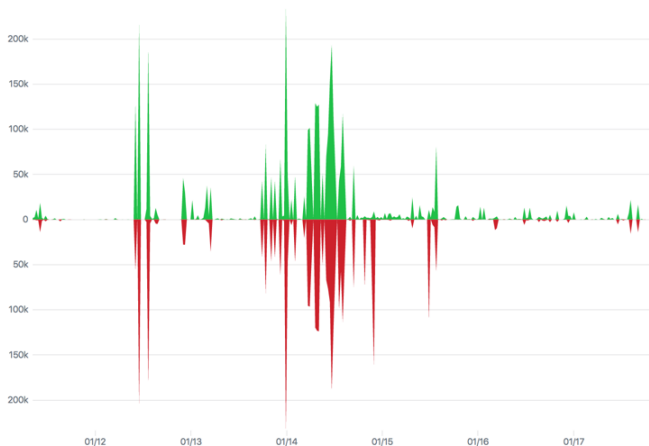
Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

## Publicación de páginas web en GitHub

Additions and Deletions per week



Conceptos básicos

Uso de git y GitHub

Trabajo en colaboración

Publicación de páginas web en GitHub

# Página web de usuario u organización

1. Crea un repositorio nuevo con el nombre `<username>.github.io`<sup>12</sup>.
2. Sube (commit + push) un fichero `index.html` a la rama master con código HTML.
3. Con un navegador ve a la dirección `https://<username>.github.io`

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Hello World</title>
  </head>
  <body>
    Hello World!
  </body>
</html>
```

<sup>12</sup>Siendo `<username>` tu nombre de usuario en GitHub.

## Si no sabes HTML

- En la página del repositorio:

*Settings > GitHub Pages > Source > master branch*

*Settings > GitHub Pages > Theme Chooser*

- Modifica el fichero README.md<sup>13</sup> (commit + push).
- Con un navegador ve a la dirección  
<https://<username>.github.io/<repository>>

### Conceptos básicos

¿Qué es el control de  
versiones?

¿Qué son Git y GitHub?

### Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

### Trabajo en colaboración

Ramas

Persiguiendo a los bichos

Herramientas gráficas para  
el análisis de un repositorio

### Publicación de páginas web en GitHub

---

<sup>13</sup>Más información sobre formato Markdown

<https://guides.github.com/features/mastering-markdown/>.

## Si sabes HTML

- ▶ Crea una carpeta docs en la rama master del repositorio.
- ▶ En esta carpeta docs crea/modifica un fichero `index.html` (`commit + push`).
- ▶ En la página del repositorio:

*Settings > GitHub Pages > Source > docs folder*

- ▶ Con un navegador ve a la dirección  
<https://<username>.github.io/<repository>>

### Conceptos básicos

¿Qué es el control de versiones?

¿Qué son Git y GitHub?

### Uso de git y GitHub

Primeros Pasos

Flujo de Trabajo

### Trabajo en colaboración

Ramas

Perseguendo a los bichos

Herramientas gráficas para el análisis de un repositorio

### Publicación de páginas web en GitHub