

---

# Conociendo GitHub Documentation

*Release 0.1*

**Luciano Castillo**

**Sep 27, 2017**



---

## Contents

---

<b>1</b>	<b>Índice:</b>	<b>3</b>
1.1	Introducción . . . . .	3
1.2	Aprendiendo a usar GitHub . . . . .	6







## Introducción

### ¿Qué es GitHub?

GitHub es una plataforma de **desarrollo colaborativo de software** para alojar proyectos utilizando el sistema de control de versiones [Git](#).



---

**Note:** El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.

---

### ¿Para que sirve?

GitHub aloja tu repositorio de código y te brinda **herramientas** muy útiles para el **trabajo en equipo**, dentro de un proyecto.



Además de eso, puedes **contribuir a mejorar el software de los demás**. Para poder alcanzar esta meta, GitHub provee de funcionalidades para hacer un **fork** y solicitar **pulls**.



Realizar un **fork** es simplemente **clonar un repositorio ajeno** (genera una copia en tu cuenta), **para eliminar algún bug o modificar cosas de él**. Una vez realizadas tus modificaciones puedes enviar un **pull** al dueño del proyecto. Éste **podrá analizar los cambios** que has realizado fácilmente, **y si considera interesante tu contribución, adjuntarlo con el repositorio original**.

## ¿Qué herramientas proporciona?

En la actualidad, GitHub es mucho más que un servicio de alojamiento de código. Además de éste, se ofrecen varias herramientas útiles para el **trabajo en equipo**. Entre ellas, caben destacar:





- Una **wiki** para el mantenimiento de las distintas versiones de las páginas.
- Un **sistema de seguimiento de problemas** que permiten a los miembros de tu equipo detallar un problema con tu software o una sugerencia que deseen hacer.
- Una **herramienta de revisión de código**, donde se pueden añadir anotaciones en cualquier punto de un fichero y debatir sobre determinados cambios realizados en un commit específico.
- Un **visor de ramas** donde se pueden comparar los progresos realizados en las distintas ramas de nuestro repositorio.

## ¿Qué uso le daremos?

En nuestra especialidad “*Programación*”, fuimos aprendiendo cosas y creando programas de código abierto, fomentando el **software libre**; es por eso que presentamos esta gran herramienta enfocada al **crecimiento de proyectos comunitarios** y libres.



En esta página podremos **crear una cuenta** gratuita y comenzar a **subir repositorios** de código (o crearlos desde 0), para que con la ayuda de todos ese proyecto mejore; así como también **fortalecer** los **proyectos** de los demás para crecer como grupo.

## Aprendiendo a usar GitHub

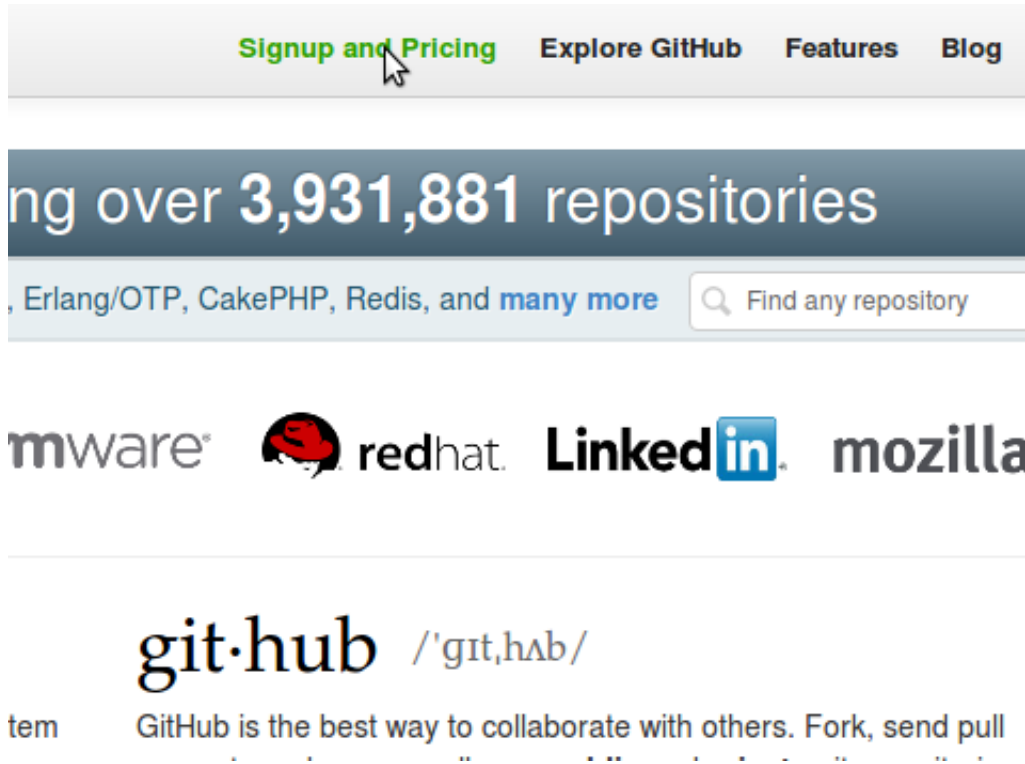
### Crear una cuenta

Para crear una cuenta GitHub vamos a seguir los siguientes pasos:

1. Entrá a [GitHub](#).



2. Hací clic en *Signup and Pricing*, de la barra de herramientas de la página.



- Ahora en *Create a free account*, para una cuenta gratuita.

- Finalmente, solo habrá que llenar un pequeño formulario con los siguientes datos:
  - Nombre de usuario.
  - Dirección email.

- Contraseña.
- Confirmar contraseña.

---

## Create your free personal account

**Username**

**Email Address**

We promise we won't share your email with anyone.


**Password**

Must contain one lowercase letter, one number, and be at least 7 characters long.

**Confirm Password**

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

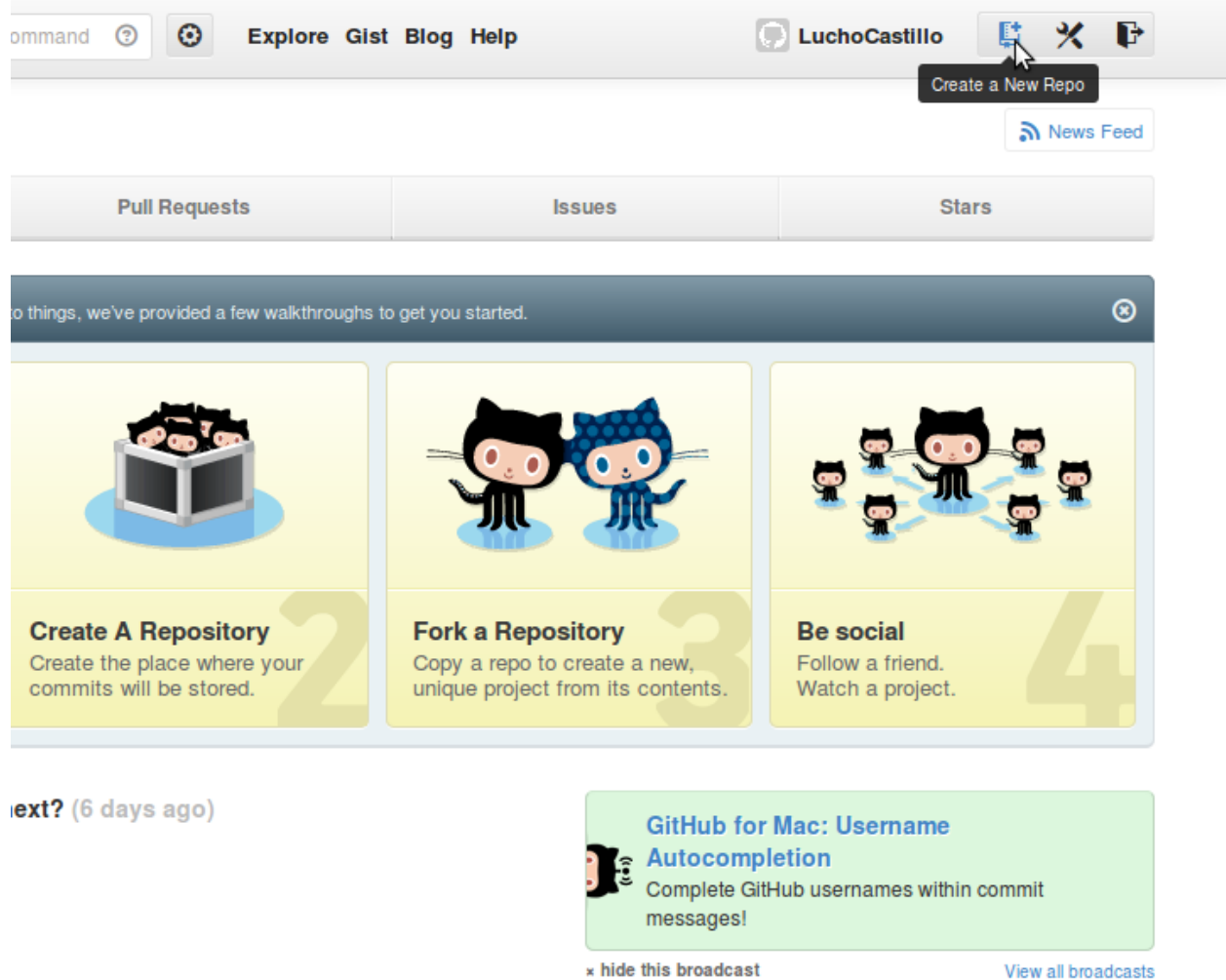
Create an account



## Manejo de repositorios

### ¿Cómo crear un repositorio?

Para crear un repositorio en GitHub, solo hay que seleccionar el botón “*Create a New Repo*”, de la barra de herramientas, habiendo entrado a [GitHub](#) con tu cuenta:



Ahora habrá que llenar dos datos:

1. Nombre del repositorio
2. Descripción del repositorio (opcional)

The screenshot shows the GitHub 'Create repository' form. At the top, there are two fields: 'Owner' with a dropdown menu showing 'LuchoCastillo' and a 'Repository name' field containing 'Repositorio' with a green checkmark. Below these is a hint: 'Great repository names are short and memorable. Need inspiration? How about **furry-octo-nemesis**.' The 'Description' field is optional and contains 'Soy un repositorio :)'. There are two radio buttons for visibility: 'Public' (selected) and 'Private'. Below these is a checkbox for 'Initialize this repository with a README', which is unchecked. At the bottom, there is a dropdown for 'Add .gitignore: None' and a large green 'Create repository' button with a mouse cursor hovering over it.

¡Listo! Repositorio creado, ahora lo vas a poder ver en tu perfil.

### Crear un proyecto

Al crear el repositorio, inmediatamente nos va a llevar a él. Como nuestro proyecto no tiene nada en su interior, no nos va a mostrar más que una ayuda para subir archivos y proyectos.

Para crear un proyecto desde cero, habrá que comenzar creando los archivos del mismo y luego subiéndolos a la página.

En el primer recuadro de la ayuda, verás una serie de comandos para el terminal.

```
touch README.md
git init
git add README.md
git commit -m "comentario"
git remote add origin https://github.com/LuchoCastillo/Repositorio.git
git push -u origin master
```

Antes de seguirlos, tendrás que instalar git:

```
sudo apt-get install git
```

### Subir proyecto

Para subir un proyecto ya realizado a GitHub, habrá que seguir exactamente los mismos pasos, ya que para iniciar un proyecto nuevo, habrá que subir los archivos creados y luego modificarlos en el editor de la página.

En este caso, los archivos junto con su contenido, ya están hechos. Solo hay que subirlos y editarlos si surge algún inconveniente.

Para subir un archivo hay que ubicarse en la carpeta del repositorio y seguir estos pasos:

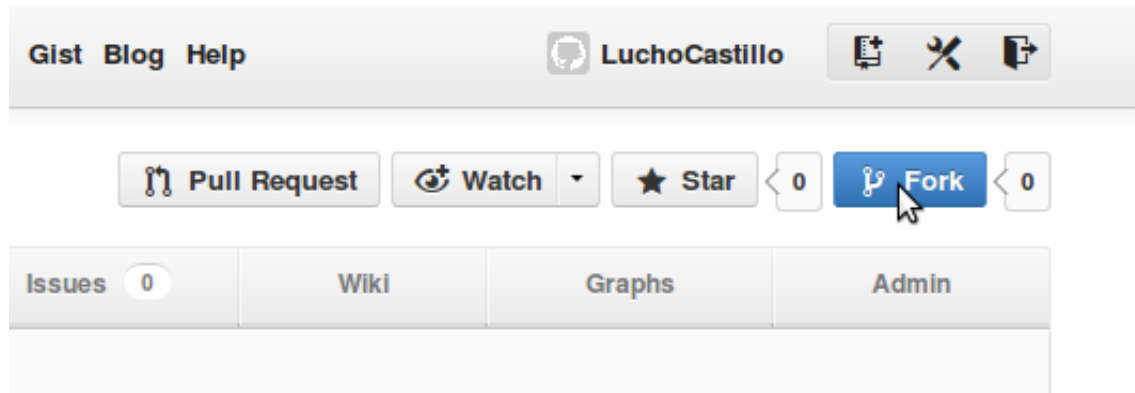
```
git add archivo
git commit -m "comentario"
git push
```

Es importante realizar los 3, ya que si no se ingresa un comentario, no se realiza el cambio.

Para adquirir un poco más de práctica y conocer más comandos de `git`, pueden ingresar a este [tutorial](#) bastante práctico.

## Colaborar en un proyecto ajeno

Para colaborar en un proyecto ajeno simplemente basta con buscarlo dentro de los repositorios, y luego presionar el botón *fork*. Esto genera automáticamente una copia del mismo en tu perfil.



Al terminar tus modificaciones podrás presionar *Pull Request* para enviárselo al creador del mismo.

