

GIT



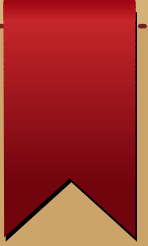
(github utiliza git)



git - ¿? (1)

- git es un sistema de control de versiones distribuído (scvd) escrito en C
- un sistema de control de versiones permite la creación de una 'historia' (aka **repository**) de un conjunto de archivos determinado, y en la que cada momento de la historia se guarda el estado de los archivos dicho momento, incluye la funcionalidad de revertir los archivos a un estado anterior de la 'historia'
- web oficial: <https://git-scm.com/>

git - ¿? (2)



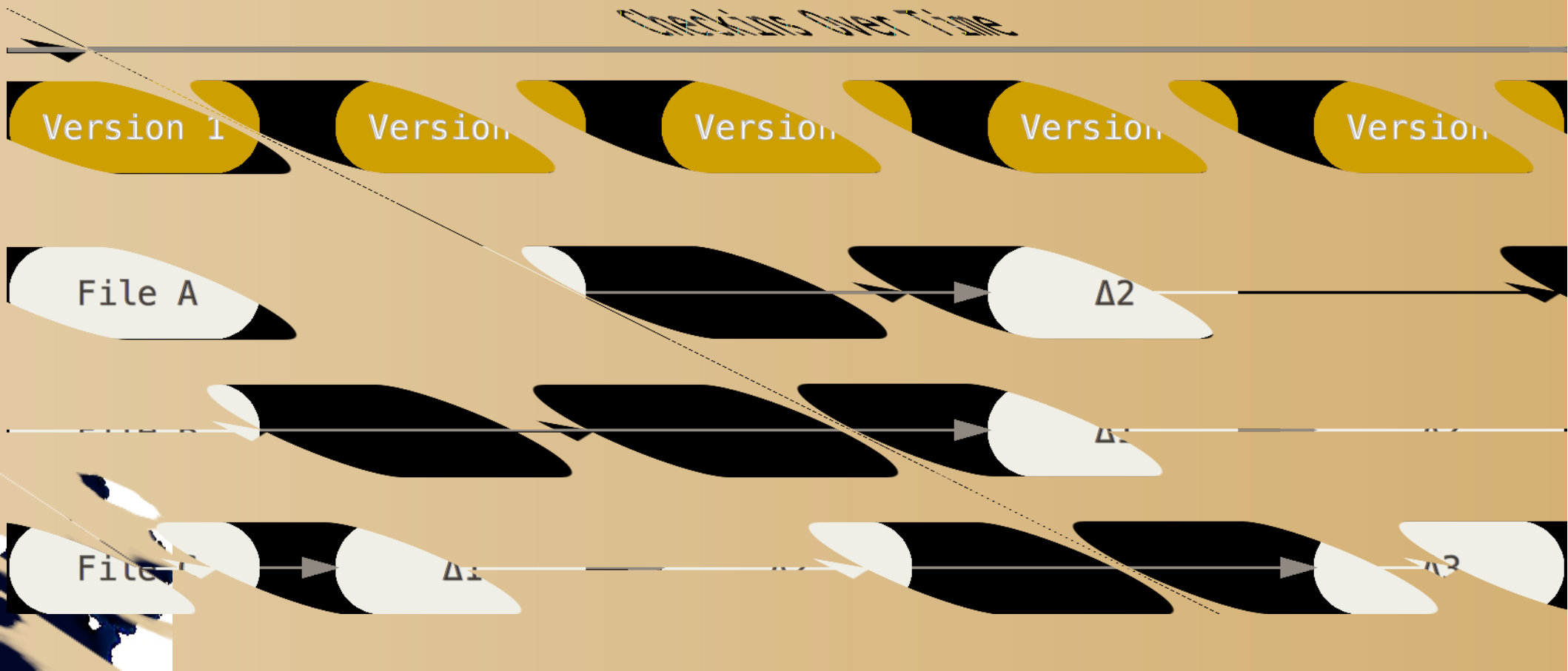
- dicho de otra forma: Git es un programita (escrito por Linus Torvalds, todo sea dicho) que:
 - te permite administrar-gestionar los cambios que haces en un proyecto
 - ver que ha ocurrido
 - juntar versiones distintas
 - retroceder a un estado anterior (a hace una semana, por ej.)



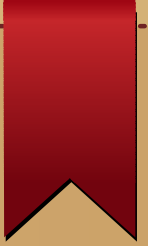
difs. con otros SCV (1)

(SCV: sistema de control de versiones)

- otros SCV (Subversion, Bazaar, etc.), guardan los cambios hechos en los ficheros



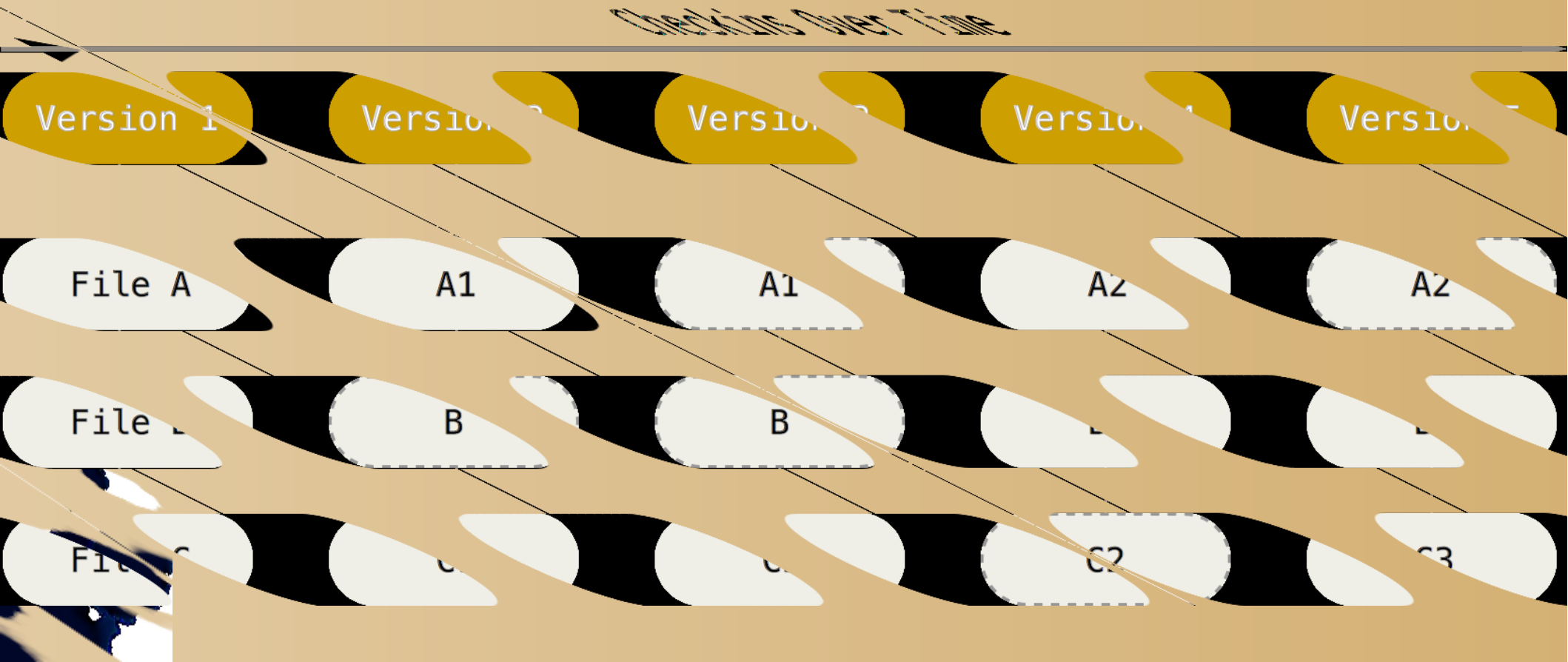
difs. con otros SCV (2)



(SCV: sistema de control de versiones)

- git mantiene una imagen de todo en cada versión, lo punteado son links, no nuevas copias

checkins over time



primero descargar e instalar

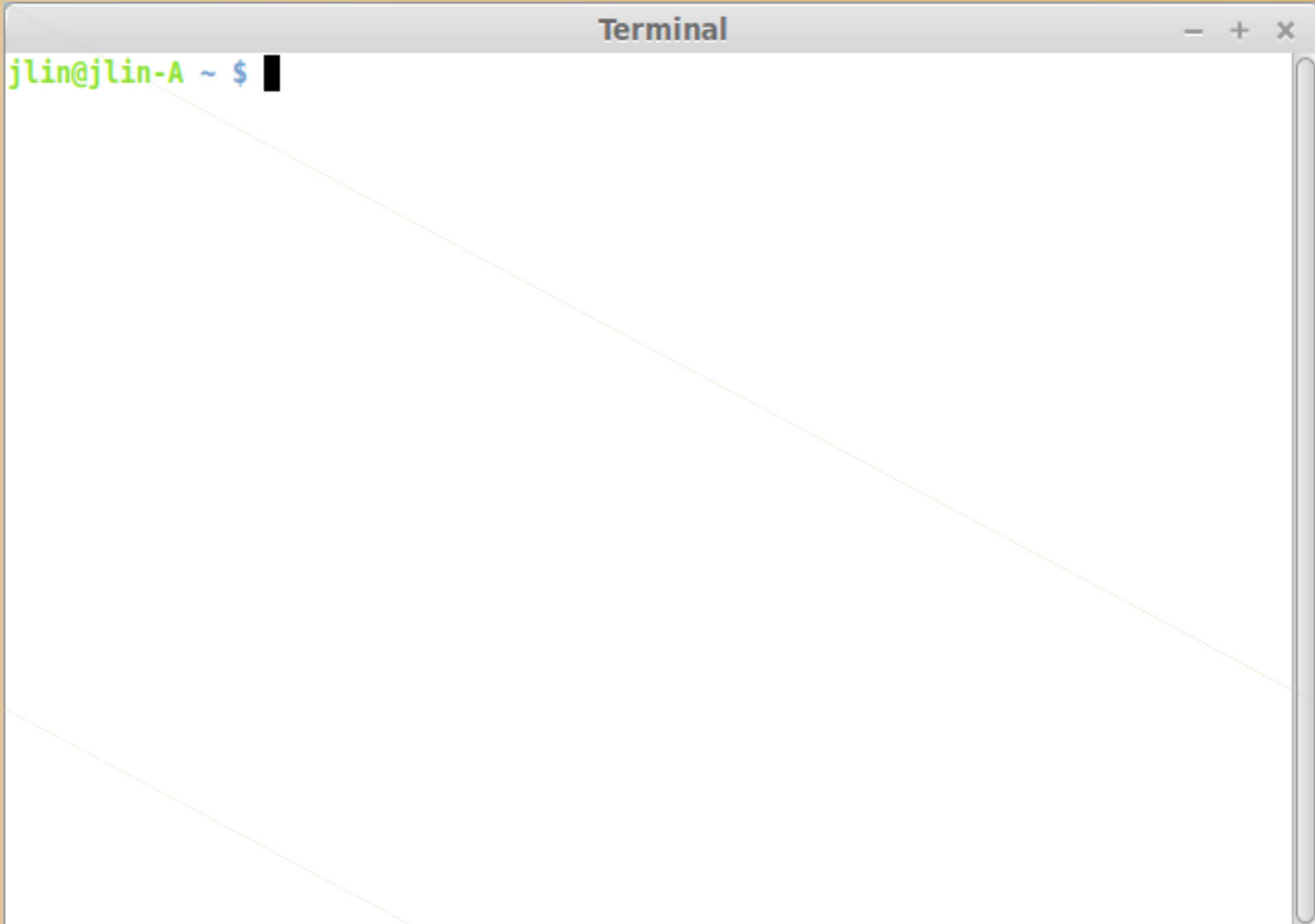
configuración

Descarga git para OSX

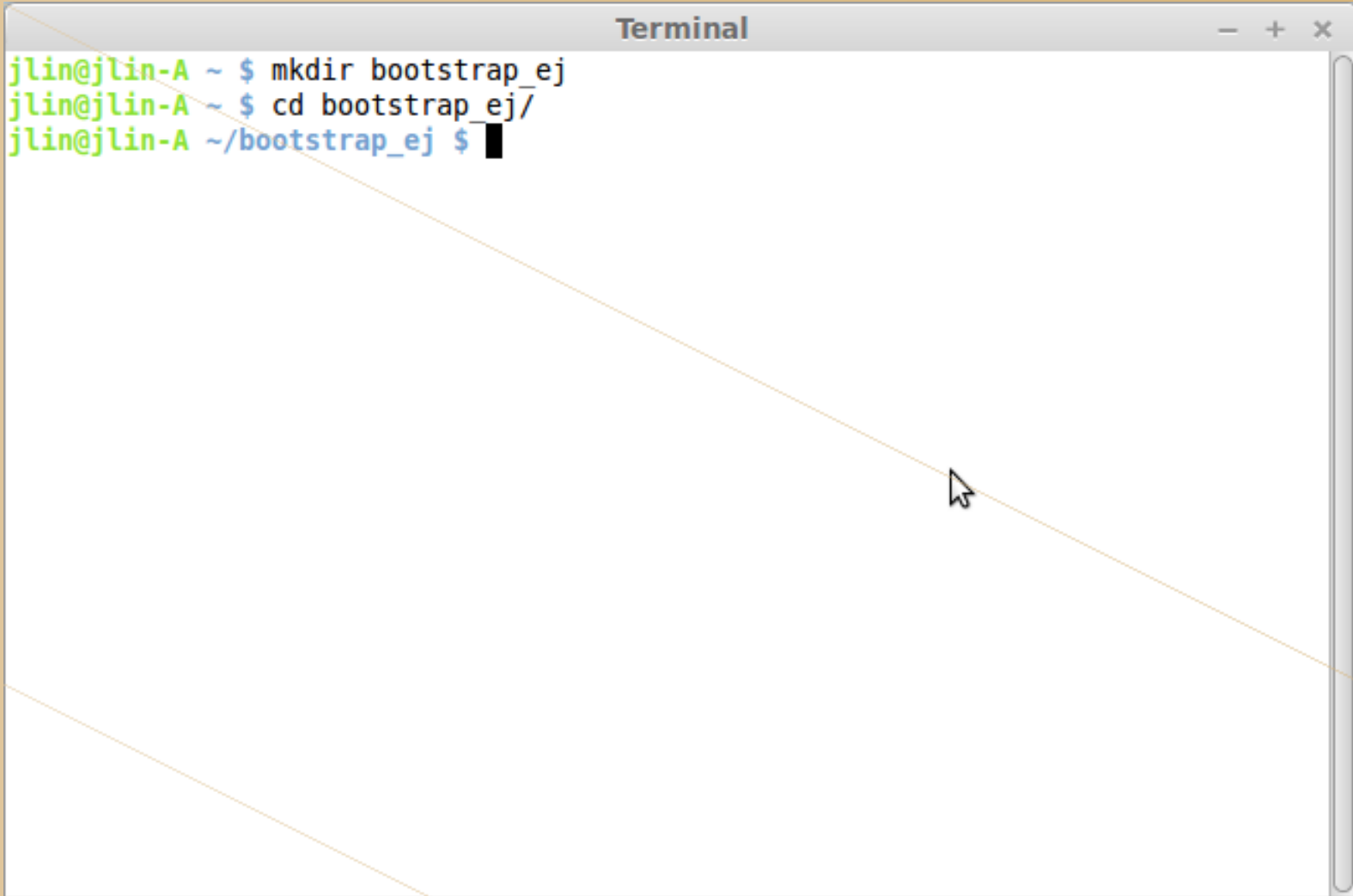
Descarga git para Windows

Descarga git para Linux

a la consola...



creamos carpeta de proyecto



```
Terminal
jlin@jlin-A ~ $ mkdir bootstrap_ej
jlin@jlin-A ~ $ cd bootstrap_ej/
jlin@jlin-A ~/bootstrap_ej $
```

A terminal window titled "Terminal" with standard window controls (-, +, x). It shows a user named "jlin" at a machine named "jlin-A" in the home directory "~". The user enters the command "mkdir bootstrap_ej", followed by "cd bootstrap_ej/", and the prompt changes to "~/bootstrap_ej \$". A mouse cursor is visible over the terminal area.

git init



Esto crea un nuevo subdirectorio llamado `.git` que contiene todos los archivos necesarios del repositorio - un esqueleto de un repositorio Git. Todavía no hay nada en el proyecto que ponemos bajo seguimiento.

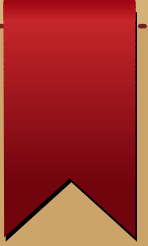


git init

Así creamos un repositorio (local)...

```
Terminal
jlin@jlin-A ~/bootstrap_ej $ # para crear un nuevo repositorio de git
jlin@jlin-A ~/bootstrap_ej $ git init
Initialized empty Git repository in /home/jlin/bootstrap_ej/.git/
jlin@jlin-A ~/bootstrap_ej $
```

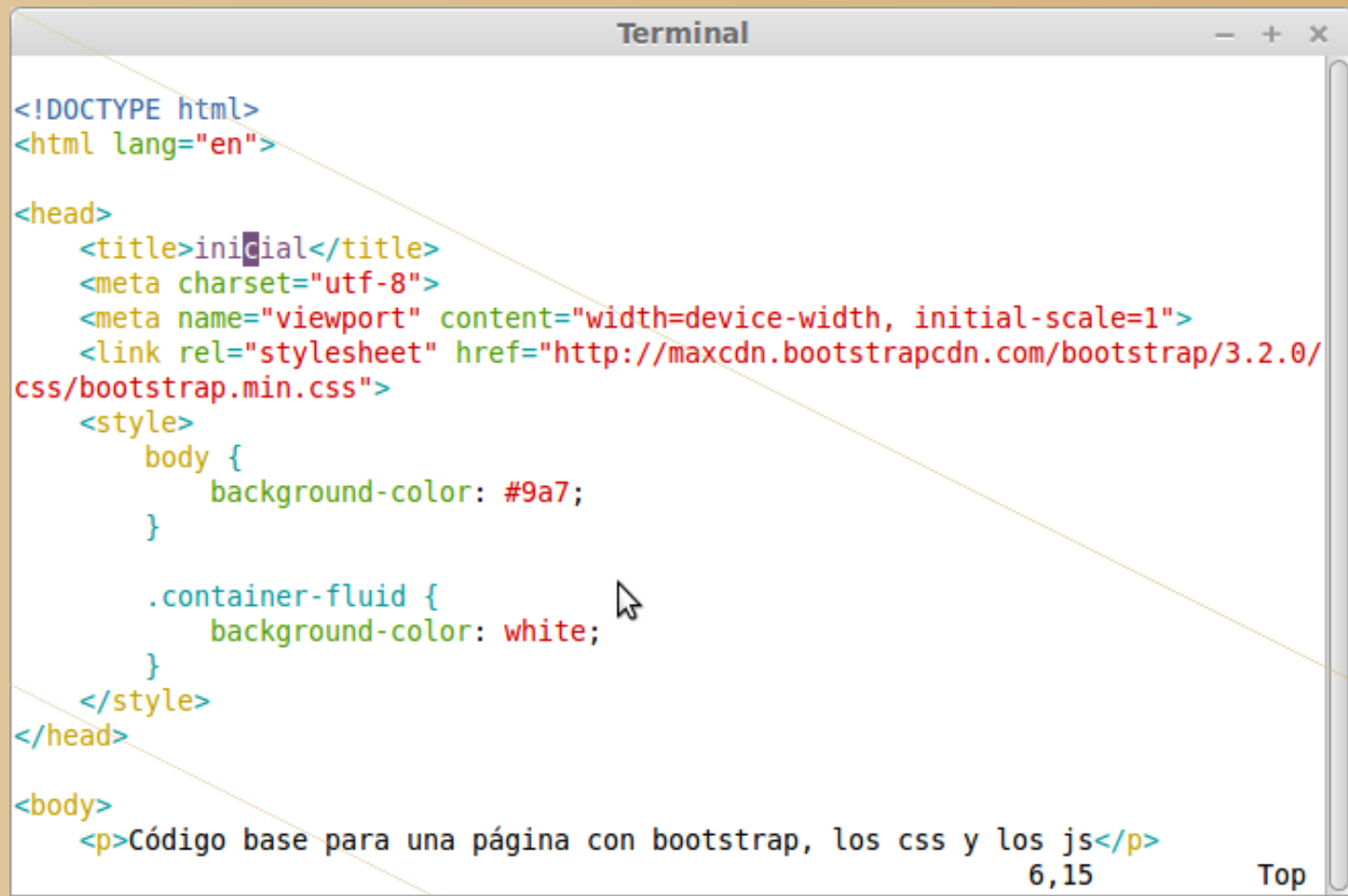
creamos un fichero...



```
Terminal
jlin@jlin-A ~/bootstrap_ej $ # para crear un nuevo repositorio de git
jlin@jlin-A ~/bootstrap_ej $ git init
Initialized empty Git repository in /home/jlin/bootstrap_ej/.git/
jlin@jlin-A ~/bootstrap_ej $ ls
jlin@jlin-A ~/bootstrap_ej $ vim index.html
jlin@jlin-A ~/bootstrap_ej $ ls
index.html
jlin@jlin-A ~/bootstrap_ej $ █
```

creamos un fichero...

Edítalo con el programa que quieras...



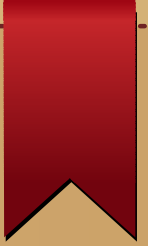
```
Terminal
<!DOCTYPE html>
<html lang="en">

<head>
  <title>inicial</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/
css/bootstrap.min.css">
  <style>
    body {
      background-color: #9a7;
    }

    .container-fluid {
      background-color: white;
    }
  </style>
</head>

<body>
  <p>Código base para una página con bootstrap, los css y los js</p>
6,15
Top
```

git add



Terminal

```
jlin@jlin-A ~/bootstrap_ej $ ls
index.html
jlin@jlin-A ~/bootstrap_ej $ # añadimos los ficheros html que tenemos para ser controlados por git
jlin@jlin-A ~/bootstrap_ej $ # en este caso solamente tenemos uno
jlin@jlin-A ~/bootstrap_ej $ git add *.html
jlin@jlin-A ~/bootstrap_ej $ █
```

Tu repositorio local esta compuesto por tres "árboles" administrados por git. El primero es tu **Directorio de trabajo** que contiene los archivos, el segundo es el **Index** que actua como una zona intermedia, y el último es el **HEAD** que apunta al último commit realizado.



git status

- Muestra el estado actual del árbol de ficheros que tenemos creado. Dicho de otra manera, te dice si los ficheros añadidos para ser vigilados están actualizados hasta su último cambio.



git status

- Vemos que falta por guardar (index.html)

```
Terminal
jlin@jlin-A ~/bootstrap_ej $ ls
index.html
jlin@jlin-A ~/bootstrap_ej $ # añadimos los ficheros html que tenemos en la carpeta para que git haga su seguimiento
jlin@jlin-A ~/bootstrap_ej $ # en este caso solamente tenemos uno
jlin@jlin-A ~/bootstrap_ej $ git add *.html
jlin@jlin-A ~/bootstrap_ej $
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   index.html

jlin@jlin-A ~/bootstrap_ej $ # con git status vemos que ficheros están bajo seguimiento y su estado
jlin@jlin-A ~/bootstrap_ej $
```


git commit -m “mensaje”



- Guardar los archivos añadidos mediante el comando ‘add’ en una nueva versión (de ahí que git sea un SCV, Sistema Control de Versiones), o en un momento de la historia.
- Cada commit es una imagen de lo que hacemos, y se puede recuperar posteriormente o juntar con otras versiones.



commit

- Realizamos un commit y se guardan los ficheros añadidos mediante el comando add en una nueva versión.

```
jlin@jlin-A ~/bootstrap_ej $ git commit -m "mi primer commit, plantilla inicial de bootstrap  
[master (root-commit) 4181374] mi primer commit, plantilla inicial de bootstrap  
1 file changed, 29 insertions(+)  
create mode 100644 index.html
```

git status

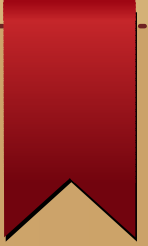
- Si modificamos el fichero con el comando 'git status' vemos que el index.html ha sido modificado. Si queremos guardarlo primero haremos un 'git add' y después un 'git commit'

```
jlin@jlin-A ~/bootstrap_ej $ # modificamos el archivo .html
jlin@jlin-A ~/bootstrap_ej $ vim index.html
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
jlin@jlin-A ~/bootstrap_ej $ # y vemos que git ve que ha habido un cambio
```

add + commit



```
Terminal
jlin@jlin-A ~/bootstrap_ej $ git commit -m "guardo en la 'rama principal' los ficheros guardados con el commit"
On branch master
Changes not staged for commit:
  modified:   index.html

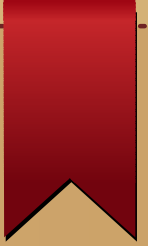
no changes added to commit
jlin@jlin-A ~/bootstrap_ej $ # como ves no hay nada que guardar (commit)
jlin@jlin-A ~/bootstrap_ej $ # si quiero guardar lo modificado tengo que hacer un 'git add' otra vez
jlin@jlin-A ~/bootstrap_ej $ git add *.html
jlin@jlin-A ~/bootstrap_ej $ git commit -m "he hecho una modificacion"
[master 783d6c3] he hecho una modificacion
 1 file changed, 1 insertion(+), 1 deletion(-)
jlin@jlin-A ~/bootstrap_ej $ # ahora si que ha funcionado
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
nothing to commit, working directory clean
jlin@jlin-A ~/bootstrap_ej $ # lo ves?
```



.gitignore

- Algunas veces deseamos que algunos archivos o directorios no sean incluidos en tu repositorio Git. Si los agregas al archivo .gitignore, Git comenzará a ignorarlos desde ese momento.

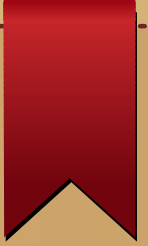
.gitignore



Terminal

```
jlin@jlin-A ~/bootstrap_ej $ ls
index.html
jlin@jlin-A ~/bootstrap_ej $ ls -la
total 16
drwxr-xr-x  3 jlin jlin 4096 may 15 23:08 .
drwxr-xr-x 85 jlin jlin 4096 may 15 23:08 ..
drwxr-xr-x  8 jlin jlin 4096 may 15 23:14 .git
-rw-r--r--  1 jlin jlin  769 may 15 23:08 index.html
jlin@jlin-A ~/bootstrap_ej $ touch .gitignore # creo el fichero .gitignore
jlin@jlin-A ~/bootstrap_ej $ vim .gitignore # lo edito, editalo con el editor de textos que quieras
```


añadiendo .gitignore...



Terminal

```
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .gitignore

nothing added to commit but untracked files present (use "git add" to track)
jlin@jlin-A ~/bootstrap_ej $ git add .gitignore
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   .gitignore

jlin@jlin-A ~/bootstrap_ej $ # añadimos el .gitignore al proyecto git
jlin@jlin-A ~/bootstrap_ej $ git commit -m "añadido .gitignore"
[master d1e742e] añadido .gitignore
1 file changed, 7 insertions(+)
create mode 100644 .gitignore
jlin@jlin-A ~/bootstrap_ej $ █
```


modificaciones (1) - diff

- Hacer una modificación nos obliga a hacer un 'add' y un commit si queremos guardarlo

```
Terminal
jlin@jlin-A ~/bootstrap_ej $ # modificamos otra vez el index.html
jlin@jlin-A ~/bootstrap_ej $ # mañana igual te vas de vacaciones y no sabes cuando vuelves, queremos un commit
jlin@jlin-A ~/bootstrap_ej $ vim index.html
jlin@jlin-A ~/bootstrap_ej $ █
```

modificaciones (2) - diff

- Modificando el index.html...

```
Terminal
background-color: #9a7;
}
.container-fluid {
background-color: white;
}
</style>
</head>
<body>
<p>Código base para una página con bootstrap, los css y los js</p>
<h1>un h1</h1>
<table>
  <thead>
    <tr>
      <th>encabez</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>cambio</td>
    </tr>
  </tbody>
</table>
```

git diff

```
Terminal
jlin@jlin-A ~/bootstrap_ej $ vim index.html
jlin@jlin-A ~/bootstrap_ej $
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
jlin@jlin-A ~/bootstrap_ej $ # pero yo quiero saber exactamente que ha cambiado!!
jlin@jlin-A ~/bootstrap_ej $ git diff
diff --git a/index.html b/index.html
index dada785..64659b8 100644
--- a/index.html
+++ b/index.html
@@ -21,6 +21,17 @@
<body>
  <p>Código base para una página con bootstrap, los css y los js</p>
  <h1>un h1</h1>
+  <table>
+    <thead>
+      <tr>
+        <th>encabez</th>
+      </tr>
+    </thead>
+    <tbody>
+      <tr>
+        <td>cambio</td>
+      </tr>
+    </tbody>
+  </table>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
  <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js"></script>
</body>
jlin@jlin-A ~/bootstrap_ej $ # vemos todas las lineas con +, significa que las he añadido nuevas
```

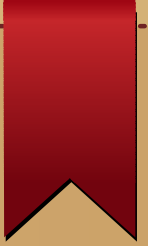
git add .

Terminal

```
jlin@jlin-A ~/bootstrap_ej $ touch ejemplo.o
jlin@jlin-A ~/bootstrap_ej $ ls -la
total 20
drwxr-xr-x  3 jlin jlin 4096 may 15 23:33 .
drwxr-xr-x 85 jlin jlin 4096 may 15 23:27 ..
-rw-r--r--  1 jlin jlin   0 may 15 23:33 ejemplo.o
drwxr-xr-x  8 jlin jlin 4096 may 15 23:33 .git
-rw-r--r--  1 jlin jlin  182 may 15 23:21 .gitignore
-rw-r--r--  1 jlin jlin  891 may 15 23:27 index.html
jlin@jlin-A ~/bootstrap_ej $ git commit -m "he añadido una tabla"
On branch master
Changes not staged for commit:
  modified:   index.html

no changes added to commit
jlin@jlin-A ~/bootstrap_ej $ git add .
jlin@jlin-A ~/bootstrap_ej $ # con esto añado todos los ficheros en la nueva version
jlin@jlin-A ~/bootstrap_ej $ █
```

teniendo .gitignore...



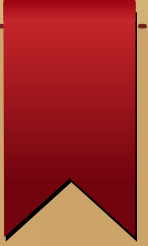
Terminal

```
jlin@jlin-A ~/bootstrap_ej $ # añadimos a ejemplo.o
jlin@jlin-A ~/bootstrap_ej $ git add ejemplo.o
The following paths are ignored by one of your .gitignore files:
ejemplo.o
Use -f if you really want to add them.
fatal: no files added
jlin@jlin-A ~/bootstrap_ej $ # pero como en .gitignore le hemos dicho que no lo mire nos lo dice
```

añadimos otro fichero...

```
Terminal
jlin@jlin-A ~/bootstrap_ej $ # añadimos a ejemplo.o
jlin@jlin-A ~/bootstrap_ej $ git add ejemplo.o
The following paths are ignored by one of your .gitignore files:
ejemplo.o
Use -f if you really want to add them.
fatal: no files added
jlin@jlin-A ~/bootstrap_ej $ # pero como en .gitignore le hemos dicho que no lo mire nos lo dice
jlin@jlin-A ~/bootstrap_ej $
jlin@jlin-A ~/bootstrap_ej $ touch navbar.html # añado un fichero navbar.html
jlin@jlin-A ~/bootstrap_ej $
jlin@jlin-A ~/bootstrap_ej $ git add .
jlin@jlin-A ~/bootstrap_ej $ # así hemos añadido todos los ficheros en la carpeta local en el proyecto
```

otro add y un commit



Terminal

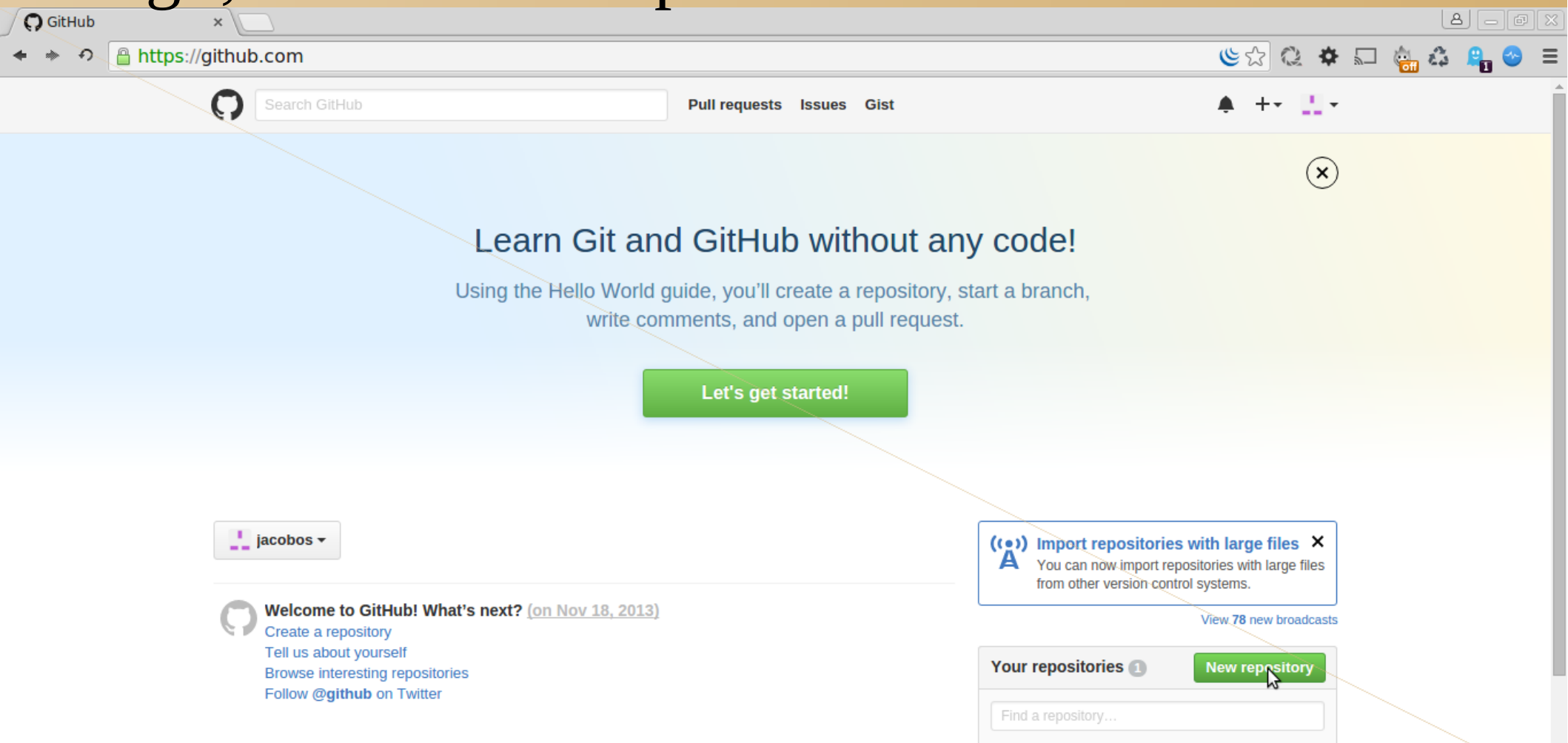
```
jlin@jlin-A ~/bootstrap_ej $ git add .
jlin@jlin-A ~/bootstrap_ej $ # así hemos añadido todos los ficheros en la carpeta local en el proyecto
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   index.html
        new file:   navbar.html

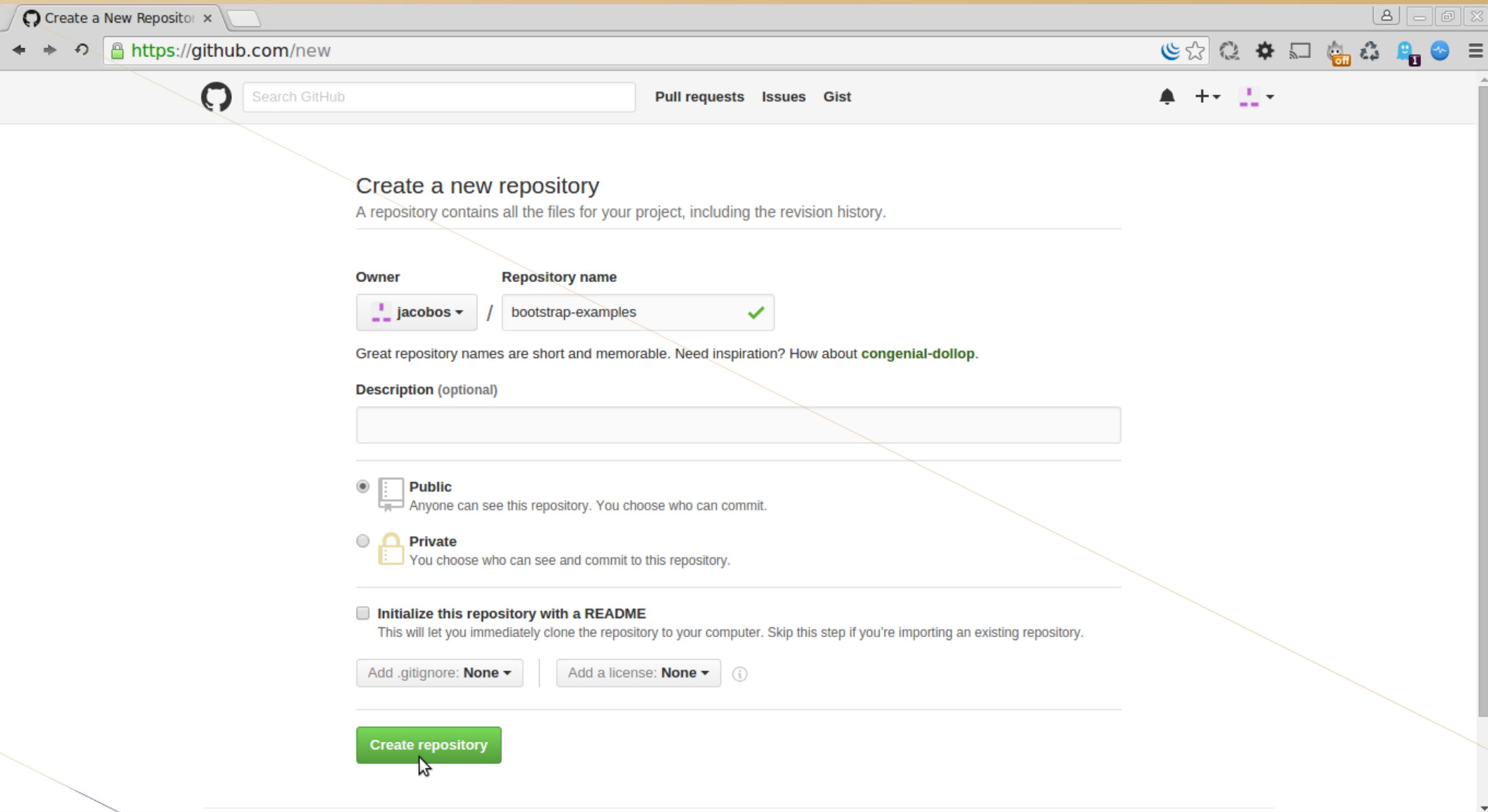
jlin@jlin-A ~/bootstrap_ej $ # nos muestra que hemos añadido uno nuevo
jlin@jlin-A ~/bootstrap_ej $
jlin@jlin-A ~/bootstrap_ej $ git commit -m "he añadido todos los ficheros nuevos, solo teníamos navbar.html"
[master e786e52] he añadido todos los ficheros nuevos, solo teníamos navbar.html
2 files changed, 11 insertions(+)
create mode 100644 navbar.html
jlin@jlin-A ~/bootstrap_ej $
```

y ahora lo subimos a github...

- Hasta este momento todo ha sido solamente git, creamos un repositorio nuevo



creamos un repositorio



Create a New Repository x

https://github.com/new

Search GitHub

Pull requests Issues Gist

Create a new repository

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

Owner **Repository name**

jacobos / bootstrap-examples ✓

Great repository names are short and memorable. Need inspiration? How about **congenial-dollop**.

Description (optional)

☒ **Public**
Anyone can see this repository. You choose who can commit.

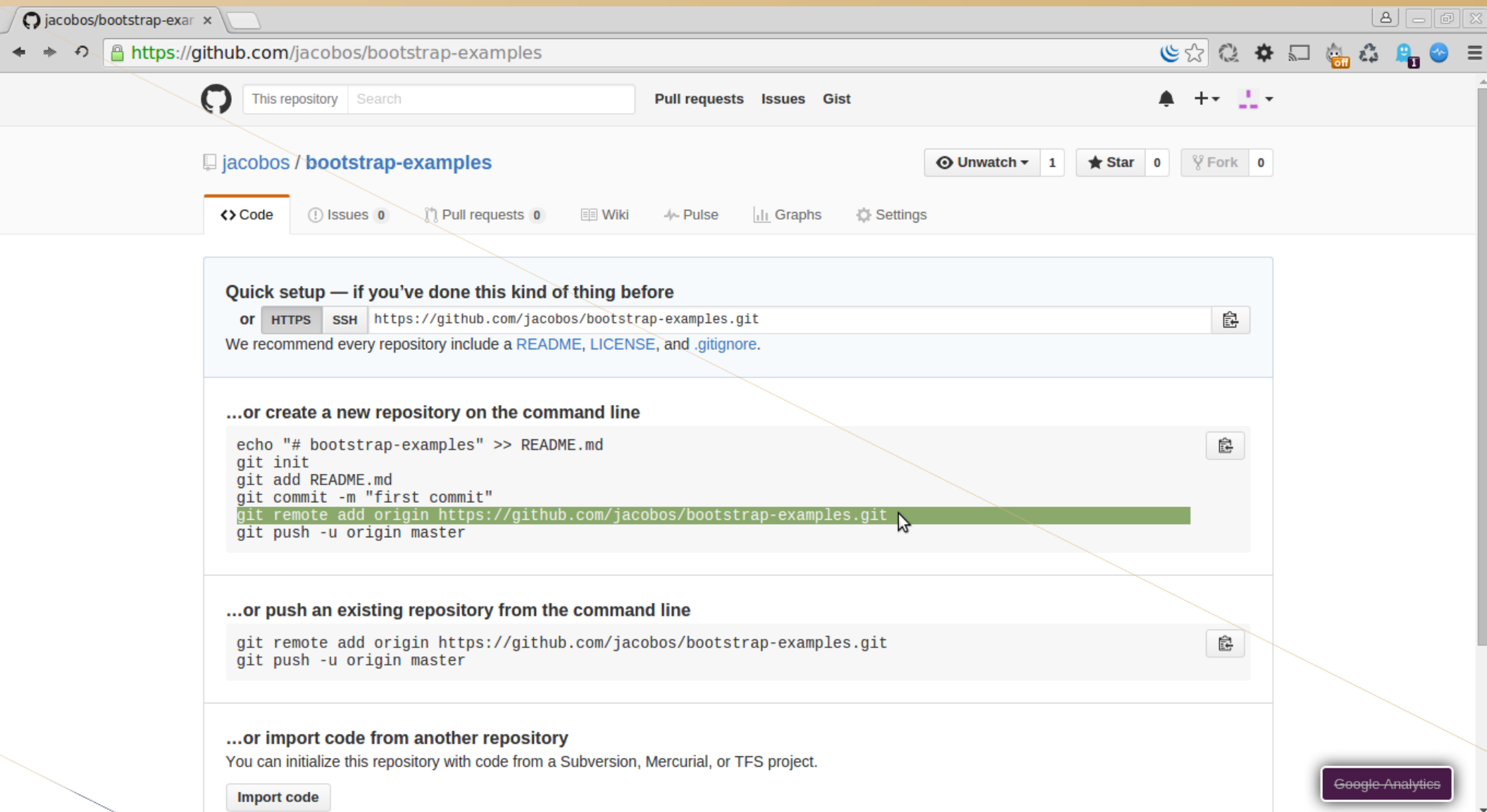
☐ **Private**
You choose who can see and commit to this repository.

☒ **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

git remote add origin



[jacobos/bootstrap-examples](#) x

[https://github.com/jacobos/bootstrap-examples](#)

This repository Search Pull requests Issues Gist

[jacobos](#) / [bootstrap-examples](#) Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

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

or **HTTPS** **SSH** `https://github.com/jacobos/bootstrap-examples.git`

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

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

```
echo "# bootstrap-examples" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/jacobos/bootstrap-examples.git
git push -u origin master
```

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

```
git remote add origin https://github.com/jacobos/bootstrap-examples.git
git push -u origin master
```

...or import code from another repository

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

[Import code](#)

Google Analytics

git remote add origin

- Con este comando le decimos cuál es el repositorio remoto (el link que aparece) con el que queremos compartir los datos que tenemos en el repositorio local

Terminal

```
lin@jlin-A ~/bootstrap_ej $ git remote add origin https://github.com/jacobos/bootstrap-examples.git  
lin@jlin-A ~/bootstrap_ej $ █
```

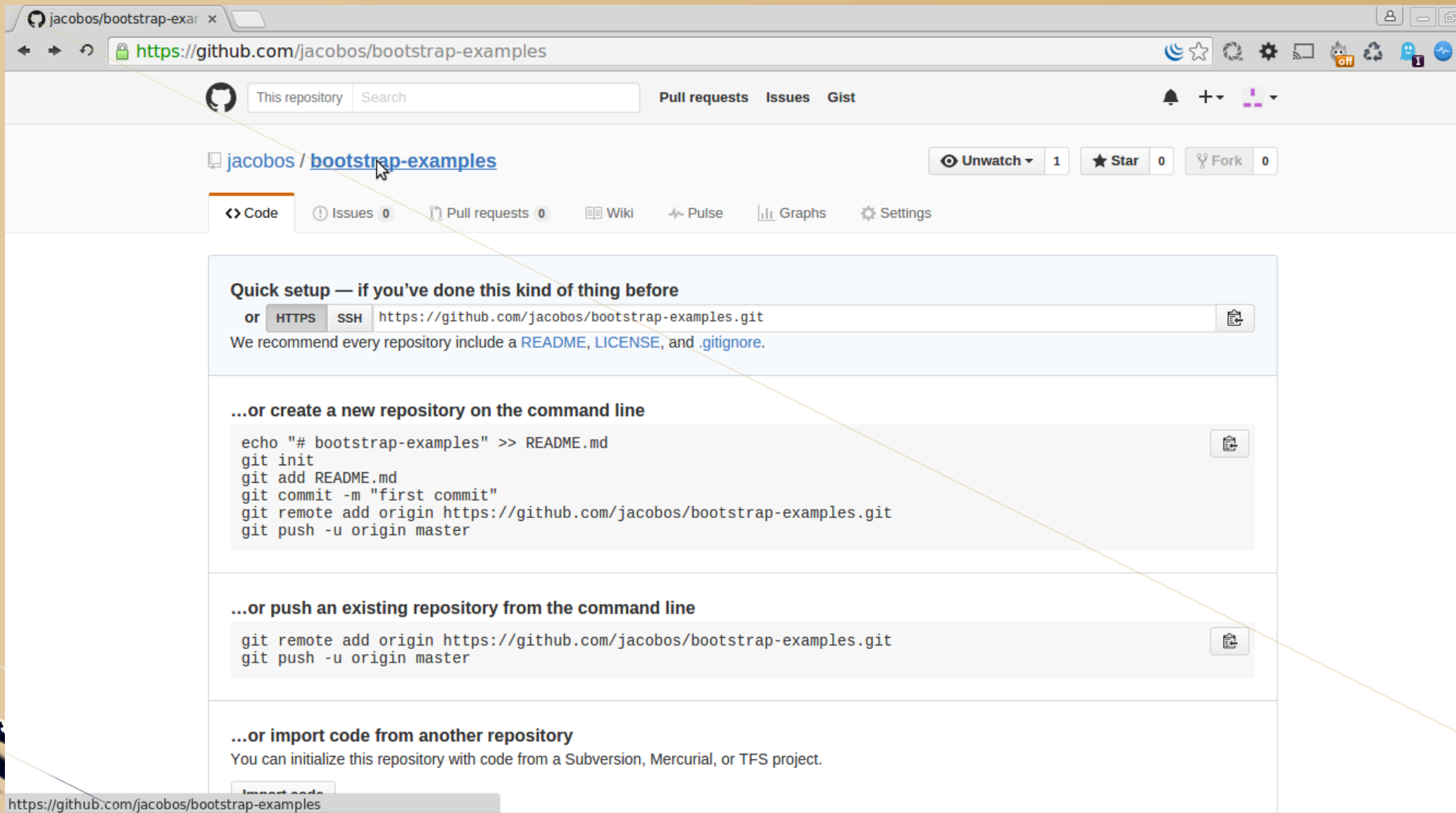
git push -u origin master

- Con este comando guardamos lo que tenemos en local en el remoto, en la rama principal (master)

Terminal

```
jlin@jlin-A ~/bootstrap_ej $ git remote add origin https://github.com/jacobos/bootstrap-examples.git
jlin@jlin-A ~/bootstrap_ej $ git status
On branch master
nothing to commit, working directory clean
jlin@jlin-A ~/bootstrap_ej $ git push -u origin master
Username for 'https://github.com': jacobos.solvam@gmail.com
Password for 'https://jacobos.solvam@gmail.com@github.com':
Counting objects: 13, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (13/13), 1.60 KiB | 0 bytes/s, done.
Total 13 (delta 2), reused 0 (delta 0)
To https://github.com/jacobos/bootstrap-examples.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
ilin@ilin-A ~/bootstrap_ej $
```

vamos al repositorio...



jacobos/bootstrap-examples x

https://github.com/jacobos/bootstrap-examples

This repository Search Pull requests Issues Gist

jacobos / bootstrap-examples Unwatch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

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

or **HTTPS** **SSH** https://github.com/jacobos/bootstrap-examples.git

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

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

```
echo "# bootstrap-examples" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/jacobos/bootstrap-examples.git
git push -u origin master
```

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

```
git remote add origin https://github.com/jacobos/bootstrap-examples.git
git push -u origin master
```

...or import code from another repository

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

Import code

https://github.com/jacobos/bootstrap-examples

y vemos los ficheros subidos...

The screenshot shows a web browser displaying the GitHub repository page for 'jacobos/bootstrap-examples'. The browser's address bar shows the URL 'https://github.com/jacobos/bootstrap-examples'. The repository page includes a search bar, navigation links for 'Pull requests', 'Issues', and 'Gist', and a header section with 'Unwatch', 'Star', and 'Fork' buttons. Below the header, there are tabs for 'Code', 'Issues', 'Pull requests', 'Wiki', 'Pulse', 'Graphs', and 'Settings'. The main content area shows the repository's commit history, with a table listing files: '.gitignore', 'index.html', and 'navbar.html'. A blue banner at the bottom of the commit history encourages adding a README. The footer of the page contains copyright information and links to 'Status', 'API', 'Training', 'Shop', 'Blog', and 'About'.

Browser tabs: jacobos/bootstrap-exar x

Address bar: <https://github.com/jacobos/bootstrap-examples>

Repository: jacobos / bootstrap-examples

Buttons: Unwatch 1, Star 0, Fork 0

Navigation: Code, Issues 0, Pull requests 0, Wiki, Pulse, Graphs, Settings

Description: No description or website provided. — Edit

Repository stats: 4 commits, 1 branch, 0 releases, 1 contributor

Branch: master | New pull request

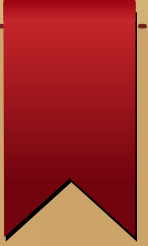
Actions: Create new file, Upload files, Find file, Clone or download

File	Commit Message	Time
.gitignore	añadido .gitignore	a day ago
index.html	he añadido todos los ficheros nuevos, solo teniamos navbar.html	a day ago
navbar.html	he añadido todos los ficheros nuevos, solo teniamos navbar.html	a day ago

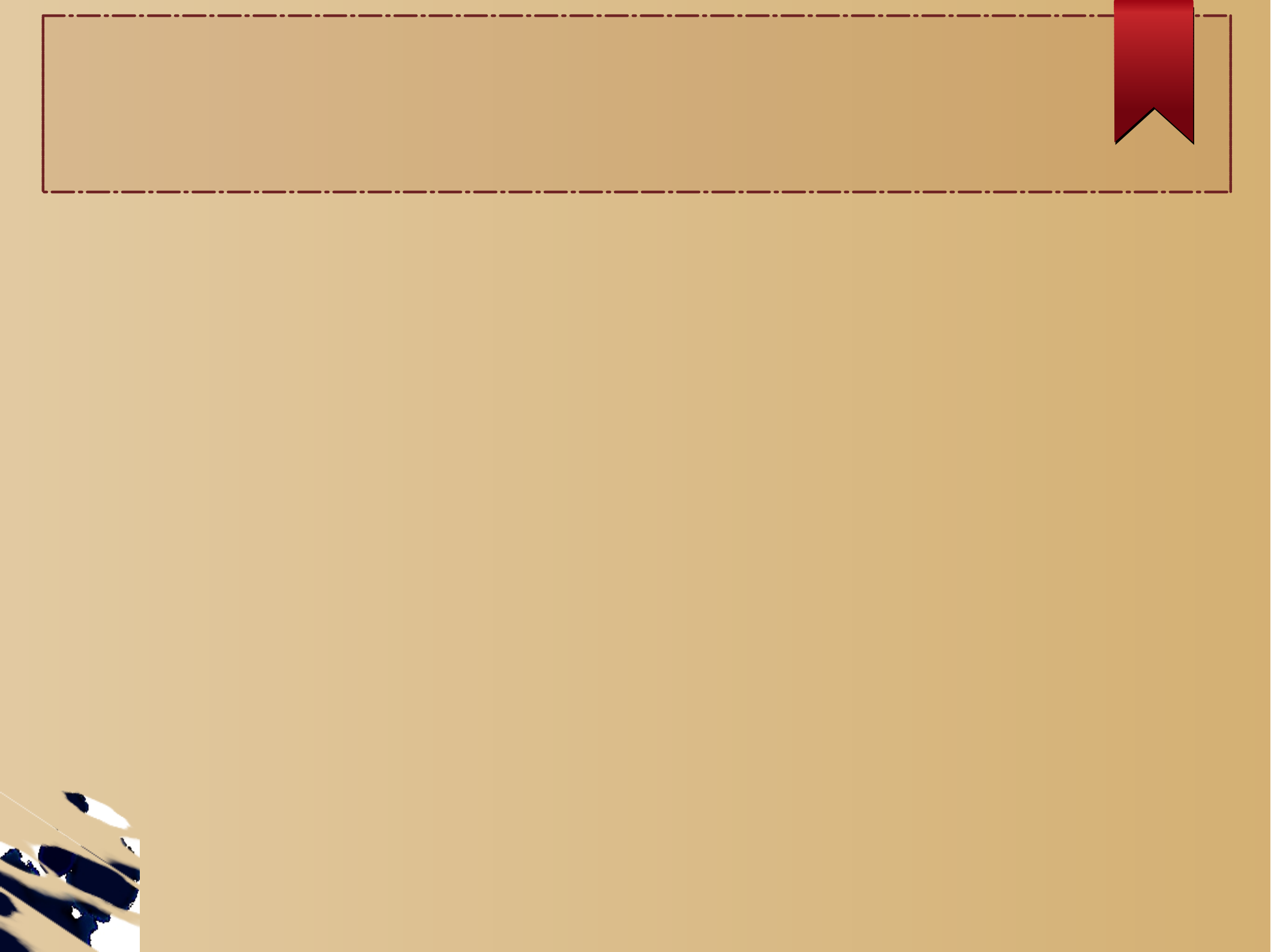
Help people interested in this repository understand your project by adding a README. [Add a README](#)

Footer: © 2016 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Contact](#) [Help](#) [Status](#) [API](#) [Training](#) [Shop](#) [Blog](#) [About](#)

git clone



```
Terminal
jlin@jlin-A ~ $ # para clonar un repositorio (un proyecto entero)
jlin@jlin-A ~ $ git clone https://github.com/twbs/bootstrap.git
Cloning into 'bootstrap'...
```



git clone



```
Terminal
jlin@jlin-A ~ $ # para clonar un repositorio (un proyecto entero)
jlin@jlin-A ~ $ git clone https://github.com/twbs/bootstrap.git
Cloning into 'bootstrap'...
```

un cursillo... (hay +)

https://www.codeschool.com



[Courses](#)

[Screencasts](#)

[Support](#)

[Upgrade](#)



What would you like to learn?



SUGGESTED:

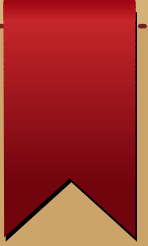
[Angular](#)

[React](#)

[Ruby](#)

[Python](#)

Links



- [Página oficial Set up git \(github\)](#)
 - [Understanding git](#)
 - [Understanding git\(2\)](#)
 - [Git, guía sencilla](#)
 - [Tutorial de git](#)
 - [Git courses \(codeschool\)](#)
- [Otro 'github', bitbucket, gratis para para peques](#)

