


# SUBIR UNA PÁGINA A GITHUB

## Aloja tu sitio web en GitHub

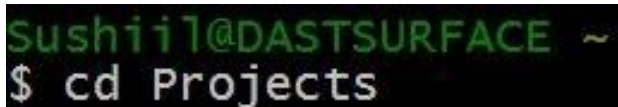
"Deployment" es el proceso de mover su código desde su computadora a un servidor web donde cualquiera puede acceder a él con un navegador. Siga estos pasos para implementar el sitio web que creó su equipo.

1. La pareja de tu equipo que está trabajando en la página de inicio de su sitio web DEBE nombrarla como *index.html*. Si no es así, deberían cambiarle el nombre. *Nota para todos los miembros del equipo: Cada vez que cambie el nombre de sus archivos HTML, debe actualizar los enlaces (<a>) en su código para que hagan referencia al nuevo nombre de archivo.*
2. Asegúrese de que el archivo esté guardado en una carpeta llamada **"Projects"** en su directorio de inicio.
3. Abra la Terminal (Mac/Linux) o GitBash (Windows).
4. Escribe **pwd** y presione Enter para verificar que se encuentra en su directorio de inicio. Debería mostrar algo como esto:



```
sushi1@DASTSURFACE ~  
$ pwd  
/c/Users/Sushi1
```

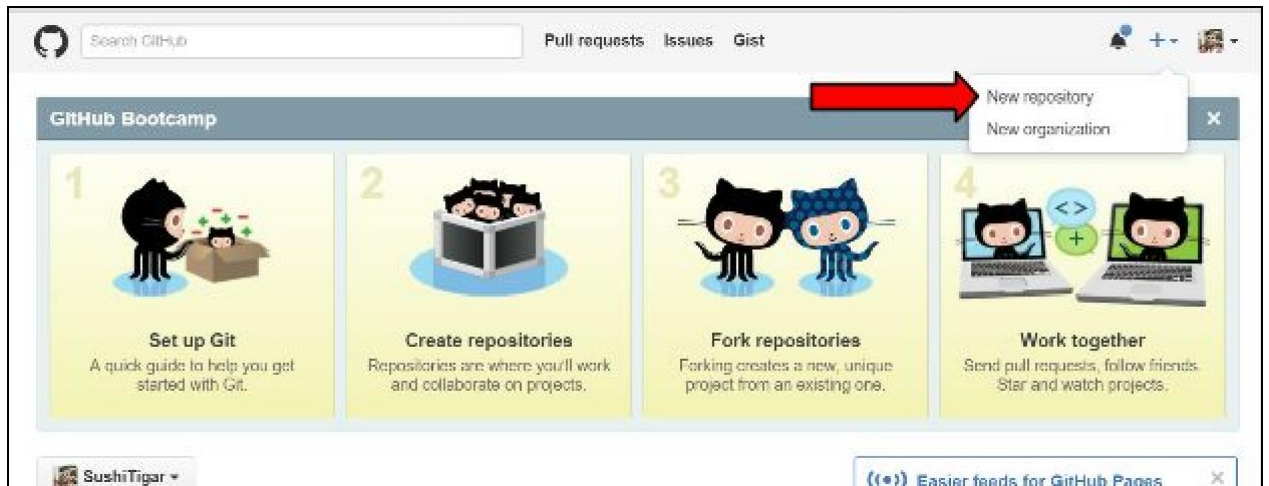
5. Ahora escribe **cd Projects** (Si le pusiste un nombre diferente a la carpeta de tu proyecto, cámbialo nombre aquí) y presione enter. El comando "cd" le dice al sistema que se mueva a una carpeta diferente. "Projects", en este caso, es la carpeta a la que deseas moverte.



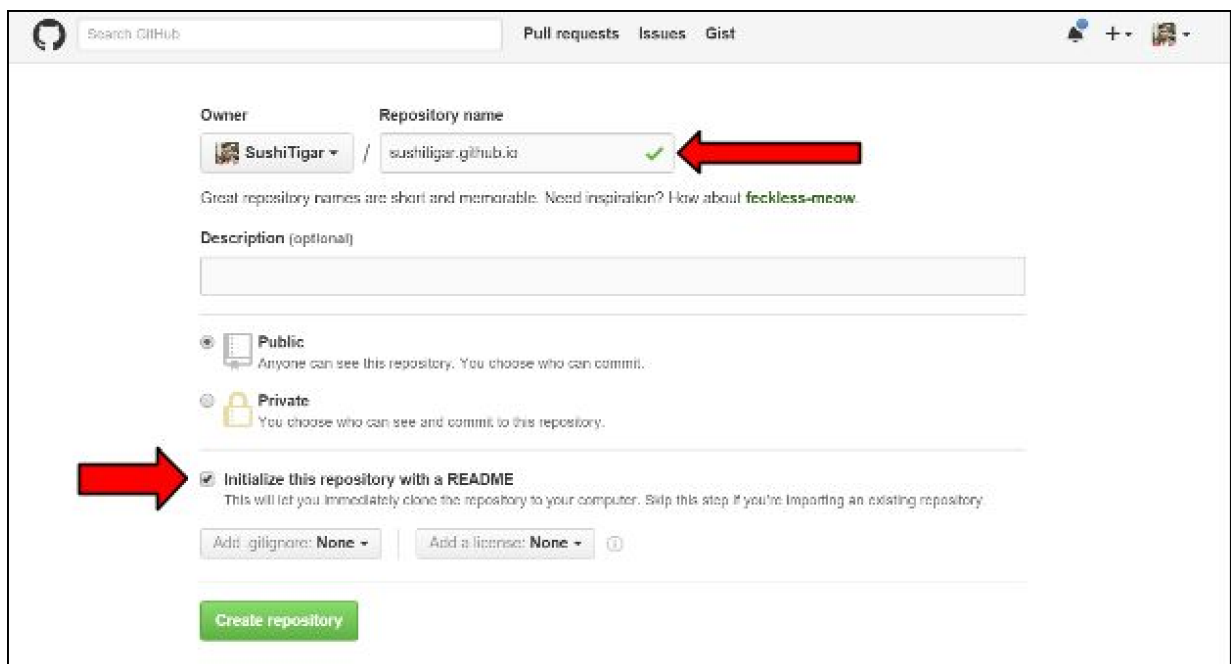
```
sushi1@DASTSURFACE ~  
$ cd Projects
```

# SUBIR UNA PÁGINA A GITHUB

- Ahora abre tu navegador web y ve a github.com. Inicia sesión si aún no lo has hecho. Selecciona el icono + en la parte superior derecha y luego selecciona **New repository**.



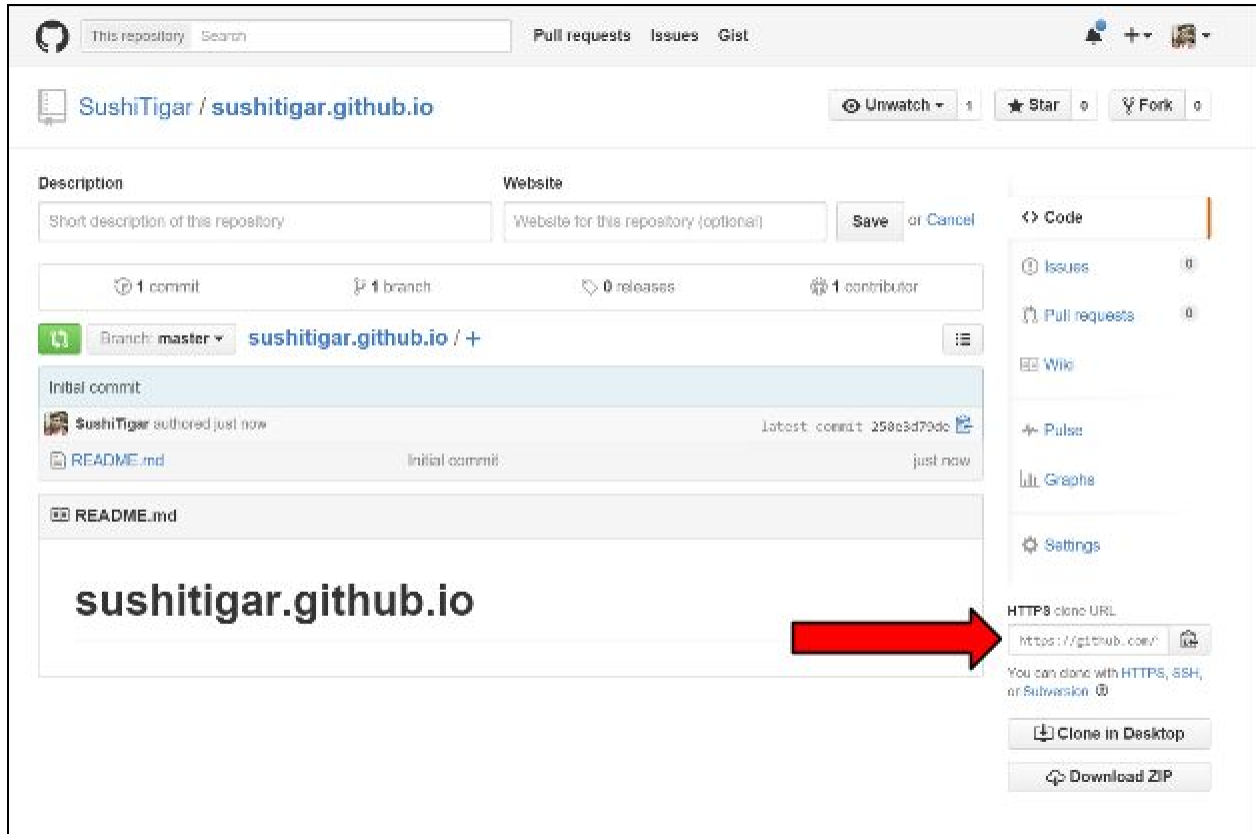
- En el campo denominado "Repository name", escribe **tu username.github.io** reemplazando "username" con su nombre de usuario de GitHub. En el siguiente ejemplo, mi nombre de usuario es "sushitigar", así que escribí "sushitigar.github.io".
- Asegúrese de estar creando un repositorio "público". Agregue una descripción si lo desea.
- IMPORTANTE** seleccione la casilla de verificación junto a **Initialize this repository with a README**.



# SUBIR UNA PÁGINA A GITHUB

10. Finalmente, selecciona el botón verde **Create repository** en la parte inferior.

11. En la siguiente página, en la parte inferior de la columna derecha (que se muestra a continuación), resalte y copie el texto en el campo llamado **HTTPS clone URL**.



12. Ahora regrese a su terminal o Git Bash. Antes de continuar, necesitas saber cómo pegar el texto en tu terminal:

- Git Bash en Windows: haga clic derecho en la línea de comando, seleccione **Paste**.
- Mac: **Cmd+V**.
- Linux: presione **Ctrl + Shift + V**.

# SUBIR UNA PÁGINA A GITHUB

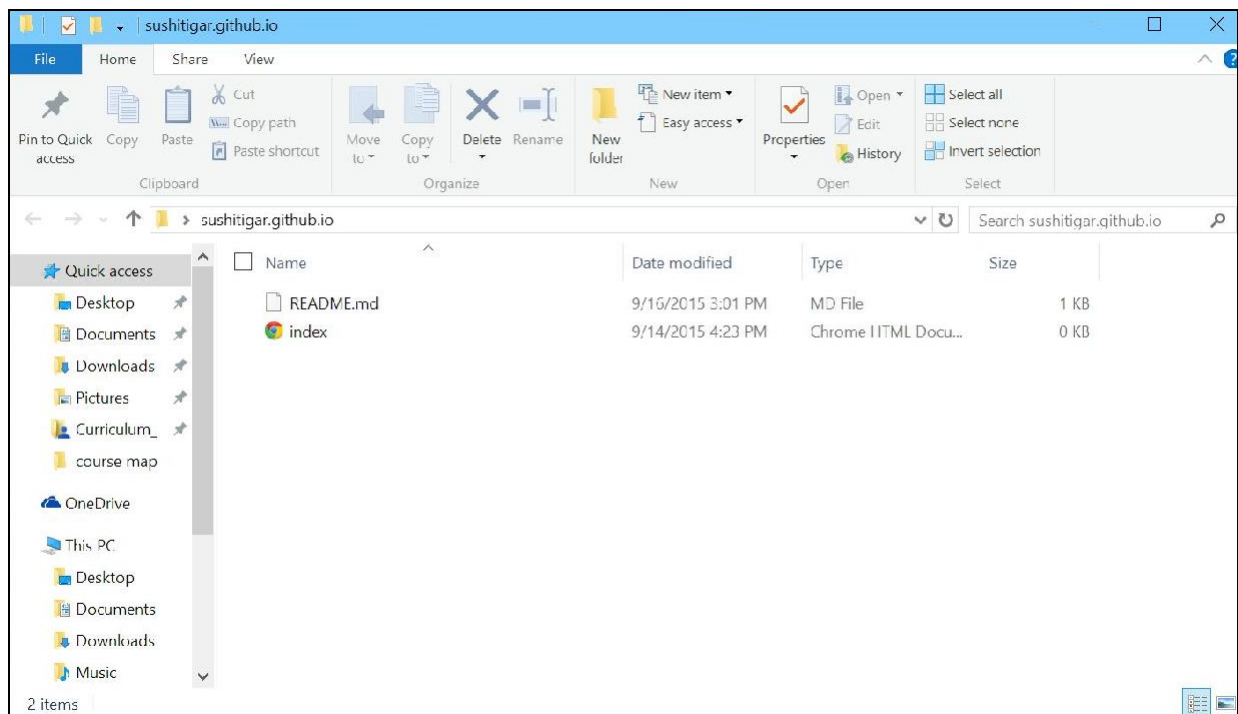
13. A continuación, vas a **clonar** ese repositorio que acabas de crear en GitHub en tu computadora. Cuando clonas algo, estás haciendo una copia de ese repositorio ("repo") en tu computadora y vinculando el repositorio de tu computadora al de GitHub. Para clonar, escriba **git clone** en su terminal, luego pegue el enlace que copió de GitHub. Debería verse así:(reemplazando "username" con su nombre de usuario)

**git clone https://github.com/username/username.github.io.git.**

Presione Enter y debería recibir un mensaje similar a la imagen de la página siguiente.

```
sushit@DASTSURFACE ~/Projects/sushitigar.github.io (master)
$ git clone https://github.com/sushitigar/sushitigar.github.io.git
Cloning into 'sushitigar.github.io'...
remote: Counting objects: 15, done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 15 (delta 5), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (15/15), done.
Checking connectivity... done.
```

14. Ahora, si miras en tu carpeta Proyectos, debería haber una nueva carpeta llamada **username.github.io** ("username" siendo su nombre de usuario de GitHub). Copie su archivo **index.html** de su carpeta de Proyectos a la nueva carpeta "username.github.io".



# SUBIR UNA PÁGINA A GITHUB

15. Ahora escribe `cd username.github.io` (Como siempre, reemplace "username" con su nombre de usuario) y luego presione enter. Para facilitar la escritura, comience a escribir su username y luego presione **tab** ; Si la carpeta está allí, la terminal completará el resto del nombre por usted.

```
sushi1@DASTSURFACE ~/Projects
$ cd sushitigar.github.io
```

16. Escribe `git status`. Git le mostrará qué archivos en su computadora han cambiado desde la última vez que realizó un **Commit**. Los commits son una instantánea de cómo se ven sus archivos en un momento específico. En este caso, aún no ha hecho ninguno, por lo que le indicará qué ha cambiado desde que clonó el repositorio. Aquí le indica que ha agregado el archivo `index.html`.

```
sushi1@DASTSURFACE ~/Projects/sushitigar.github.io (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes not staged for commit:
  Untracked files:
    (use "git add <file>..." to include in what will be committed)

        index.html

no changes added to commit (use "git add" and/or "git commit -a")
```

17. Ahora escribe `git add index.html` y presiona enter. Esto le dice a git que pronto querrás tomar una instantánea de este archivo. Repite el comando `git add` para cada archivo indicado por Git (en azul) como "untracked".

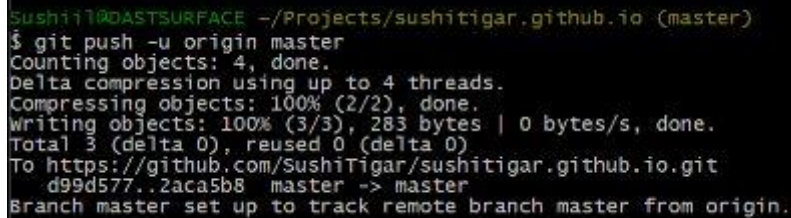
```
sushi1@DASTSURFACE ~/Projects/sushitigar.github.io (master)
$ git add index.html
```

18. A continuación, escriba `git commit -m "first commit"` y presiona enter. **Commit** le dice a git que tome la instantánea. El `-m` le dice que desea guardar un mensaje con esa instantánea (piense en un título). El texto entre comillas es el **Commit message**(el título). Debería Ver una respuesta similar a la siguiente.

```
sushi1@DASTSURFACE ~/Projects/sushitigar.github.io (master)
$ git commit -m "first commit"
[master 2aca5b8] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 index.html
```

# SUBIR UNA PÁGINA A GITHUB

19. Ahora escribe **git push origin master** y presiona enter. Se le pedirá su username y password de GitHub y luego le dará una respuesta similar a la siguiente. **git push** envía su código a GitHub y hará que su repositorio de GitHub tenga los mismos archivos, con los mismos cambios, que la confirmación que acaba de realizar.



```
sushit1@DASTSURFACE ~/Projects/sushitigar.github.io (master)
$ git push -u origin master
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 283 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/SushiTigar/sushitigar.github.io.git
   d99d577..2aca5b8  master -> master
Branch master set up to track remote branch master from origin.
```

20. Si todo salió bien, puedes ingresar esta dirección en tu navegador y ver tu sitio web, ¡en vivo en Internet!

(Reemplace “username” con su nombre de usuario)

**<http://username.github.io>**

Comparte esta dirección con tus amigos y familiares y ellos también podrán ver tu trabajo.