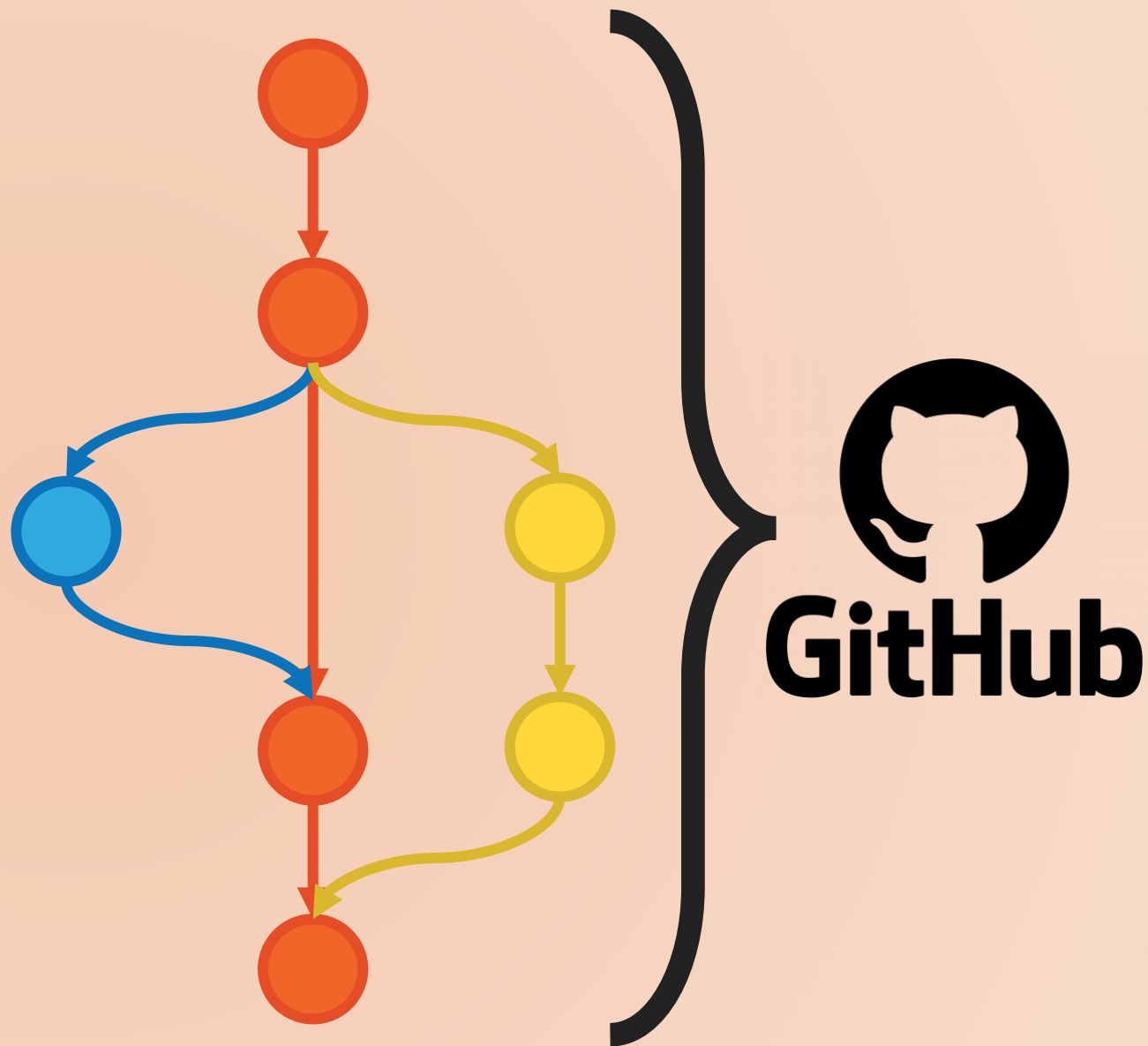


Guía básica de GitHub



Para subir un proyecto en GitHub necesitaremos tener un proyecto Git en nuestro equipo.

Rápidamente creare mi proyecto de ejemplo para refrescaros la memoria:

- 1. Creo el directorio del proyecto con mkdir y seguidamente establezco para git que este será mi proyecto:**

```
-VirtualBox:~/Desktop$ git init migithub  
-VirtualBox:~/Desktop$ mkdir migithub
```

- ✓ **Al hacer ls -a encontraremos que se creo un archivo oculto .git:**

```
alvaro@alvaro-VirtualBox:~/Desktop/migithub$ ls -a  
.  ..  .git
```



Antes de ponernos de seguido a subir el proyecto a GitHub realizaré algún que otro commit.

Para recordar el minucioso proceso que se ha de seguir para no fastidiarla en ningún momento.

working
directory

Staging area

Git directory
(repository)

1. Creación de archivos:

```
~/Desktop/migithub$ nano index.html  
~/Desktop/migithub$ nano styles.css  
~/Desktop/migithub$ nano app.js
```

2. Comparar con workingdirectory:

```
~/Desktop/migithub$ git status -s  
?? app.js  
?? index.html  
?? styles.css
```

git add

3. Añadir al staging area:

```
/migithub$ git add src
```

4. Comparar con workingdirectory:

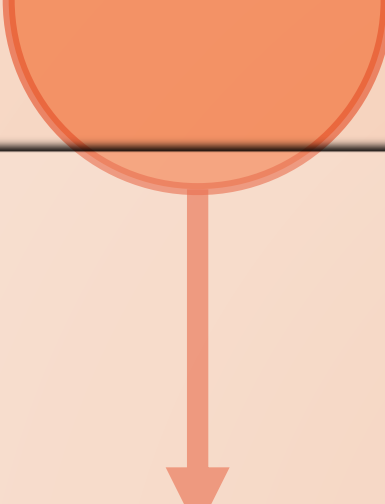
```
~/Desktop/migithub$ git status -s  
A src/app.js  
A src/index.html  
A src/styles.css
```

git commit

5. Subir un commit:

```
alvaro@alvaro-VirtualBox:~/Desktop/migithub$ git commit -m "  
[master (root-commit) 2429e4b] Creación de 3 archivos (html,  
3 files changed, 3 insertions(+)
```





En este siguiente commit modifiko el archivo html para que una vez subido a GitHub, quien vaya a trabajar sobre los otros archivos pueda ver el resultado

Código de index.html:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>MiGitHub</title>
8   <link rel="stylesheet" href="styles.css">
9   <script src="app.js"></script>
10 </head>
11 <body>
12   <h1>¿Desea subir sus archivos a github?</h1>
13   <form action="" method="post">
14     <p><input type="radio" name="github" id="si">si</p>
15     <p><input type="radio" name="github" id="no">no</p>
16     <input type="submit" value="submit">
17     <input type="reset" value="reset">
18   </form>
19 </body>
20 </html>
21
```



Llego el momento de subir nuestro proyecto a GitHub.

Para ello deberás crear una cuenta en:

<https://github.com/join>

Git repository

GitHub repository

git push

- 1. Tras crear la cuenta crearemos un nuevo repositorio y daremos a las indicaciones de nuestra elección.**
- 2. Luego siguiendo los pasos que nos indican para pushear el repositorio que tenemos en nuestro equipo:**

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

```
git remote add origin https://github.com/huertin03/migithub.git  
git branch -M main  
git push -u origin main
```

Tras en la consola haber intentado hacer el push e indicado nuestras credenciales saltará el siguiente fallo relacionado con el autenticador de la contraseña:

```
alvaro@alvaro-VirtualBox:~/Desktop/migithub$ git push -u origin main  
Username for 'https://github.com': huertin03  
Password for 'https://huertin03@github.com':  
remote: Support for password authentication was removed on August 13, 2021.  
remote: Please see https://docs.github.com/en/get-started/getting-started-with-git/about-remotes#cloning-with-https-urls for information on currently recommended modes of authentication.  
fatal: Authentication failed for 'https://github.com/huertin03/migithub.git/'
```

En la siguiente diapositiva remarcaré la compleja sucesión de pasos a realizar para realizar el push exitosamente.



1. `alvaro@alvaro-VirtualBox:~/Desktop/migithub$ sudo apt install curl`
2. `alvaro@alvaro-VirtualBox:~/Desktop/migithub$ curl -LO https://aka.ms/gcm/linux-install-source.sh`
3. `alvaro@alvaro-VirtualBox:~/Desktop/migithub$ sh ./linux-install-source.sh`
4. `alvaro@alvaro-VirtualBox:~/Desktop/migithub$ git-credential-manager configure`
5. `alvaro@alvaro-VirtualBox:~/Desktop/migithub$ git config --global credential.helper gpg`

Después de seguir estos pasos habremos habilitado el git-credential-manager. Este método está obsoleto ya que no es seguro.

Pero con esto podremos iniciar sesión cada vez que indicaremos un push.

```
alvaro@alvaro-VirtualBox:~/Desktop/migithub$ git push -u origin main
```

Para terminar realizaremos el push:



Para añadir colaboradores en tu proyecto podremos hacerlos desde la página web.

Después de seguir los pasos de la anterior diapositiva en el nuevo equipo cada nuevo colaborador deberá introducir un comando fetch para bajar el directorio en su equipo.

Añadir nuevo colaborador:

Manage access



You haven't invited any collaborators yet

Add people

Bajar última actualización subido a GitHub:

```
:~/Desktop/migithub$ git pull origin <branchname>
```

Esto te volverá a pedir usuario y contraseña



En el nuevo usuario he realizado un commit que cambia el archivo css.

No sin antes cambiar de rama, y seguir todos los pasos de la diapositiva 3.

Cuando nos estemos seguros que no queremos realizar más cambios en esta rama subimos un push para que todos tengan este commit.

Código de styles.css:

```
1 body {
2   margin: 0;
3   background-color: #383838;
4   height: 100vh;
5   width: 100vw;
6   display: grid;
7   place-items: center;
8   overflow: hidden;
9 }
10
11 h1,p {
12   color: white;
13   background: red;
14   padding-top: 1em;
15   padding-bottom: 1em;
16   padding-right: 0.5em;
17   padding-left: 0.5em;
18   border-radius: 150px;
19   transition: all 1s;
20 }
21 h1:hover,p:hover {
22   background: rgb(139, 0, 0);
23 }
24
```

Listar ramas:

```
~/Desktop/migithub$ git branch
```

Nueva rama:

```
~/Desktop/migithub$ git branch css
```

