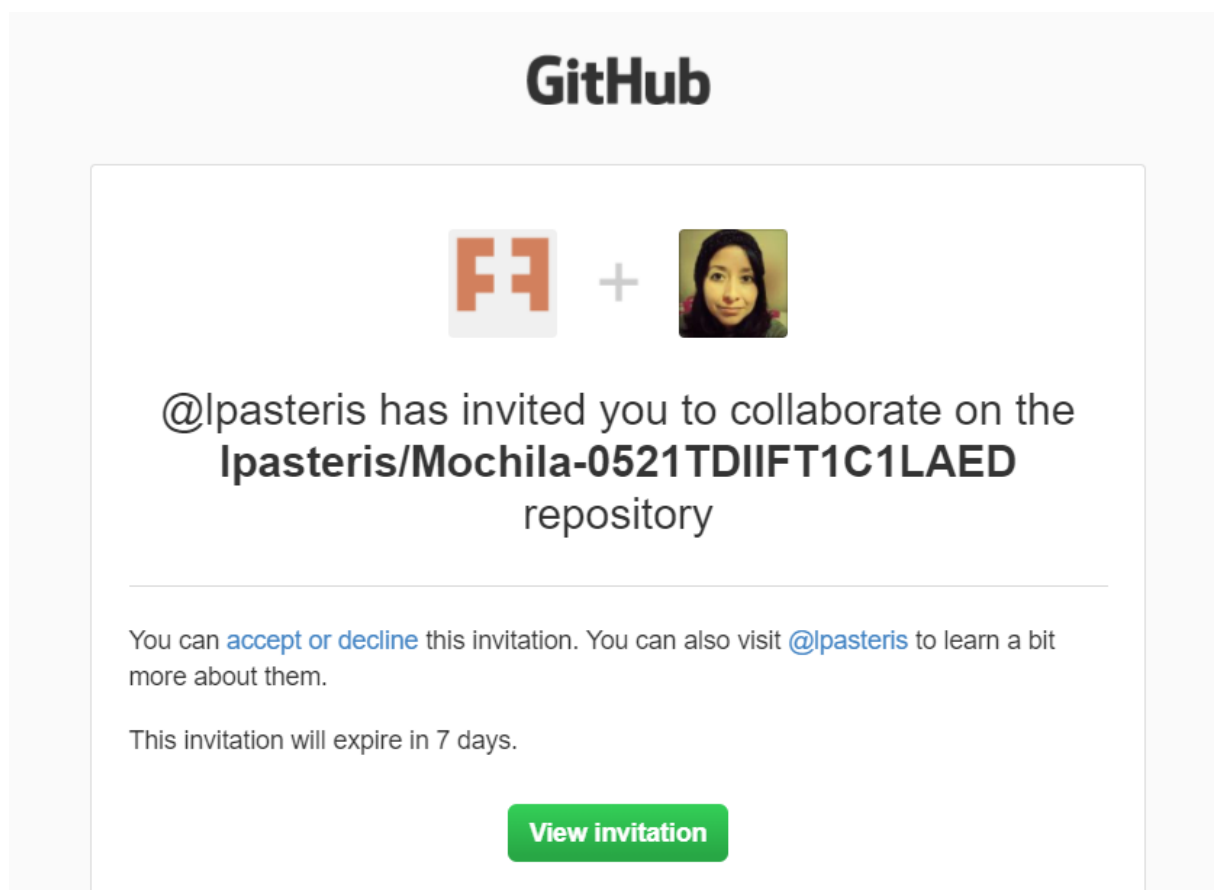


Git + GitHub

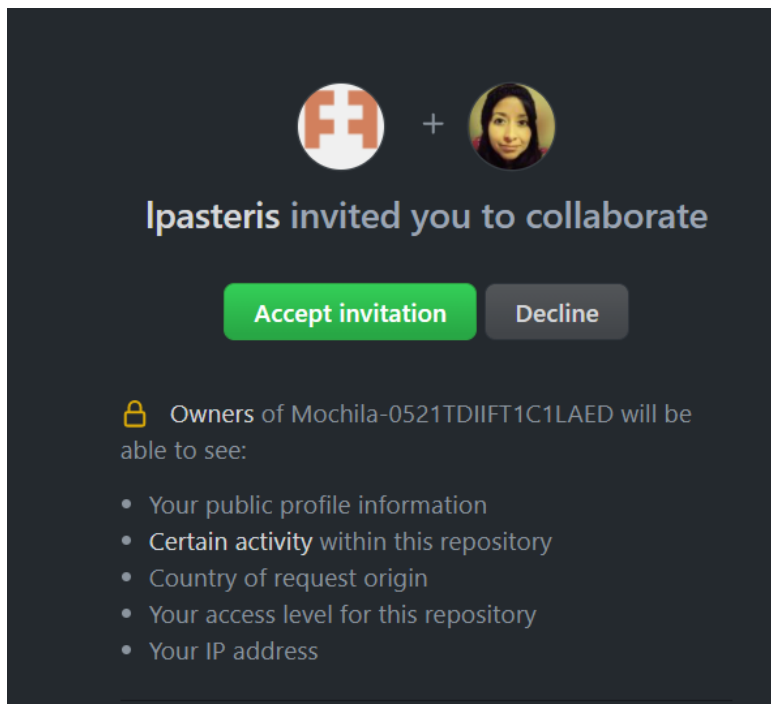
Guia paso a paso para trabajar en la mochila de la cursada.

Primeros pasos

1. Lo primero que tenés que hacer es proporcionarle tu nombre de usuario a tu profesor o profesora para que te agregue como colaborador al repositorio.
2. Una vez que te agregó como colaborador, te llegará un correo electrónico con la invitación:

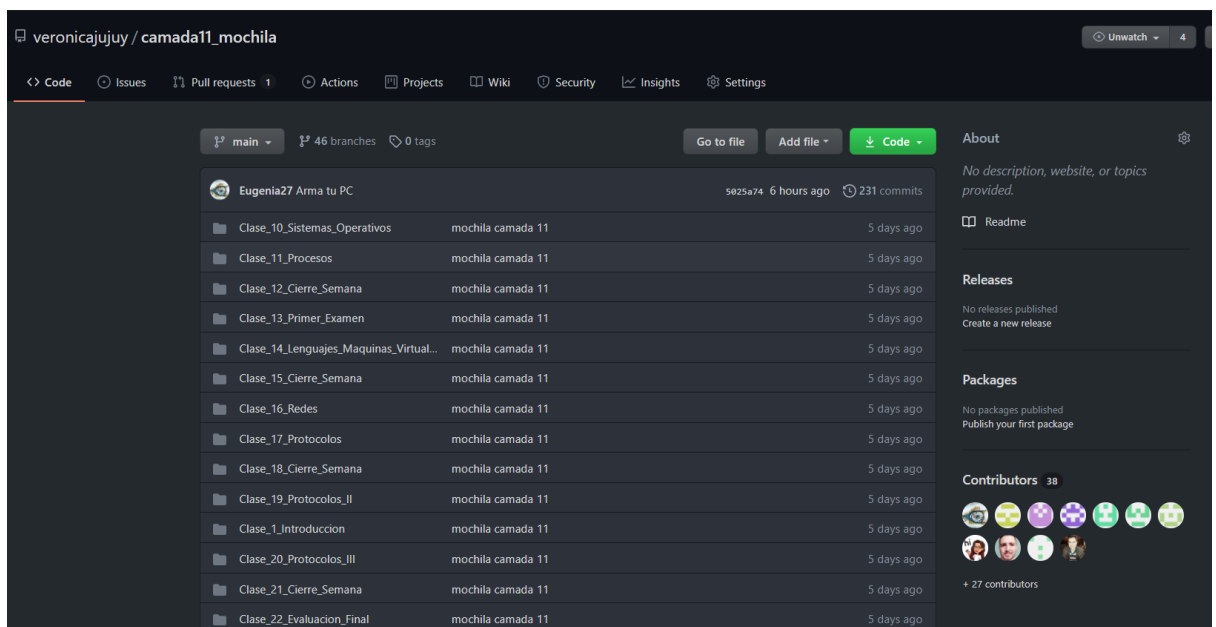


3. Al hacer clic en el botón "view invitation" te llevará a tu github para que puedas aceptar la invitación:



Es importante que antes de este paso ya te hayas logueado a github con tu usuario y contraseña.

4. Una vez que hayas aceptado la invitación te aparecerá el repositorio correspondiente a tu camada:



Antes de clonar el repositorio

5. Chequeá que esté configurado tu usuario y mail en el github con los comandos:

```
git config user.name
```

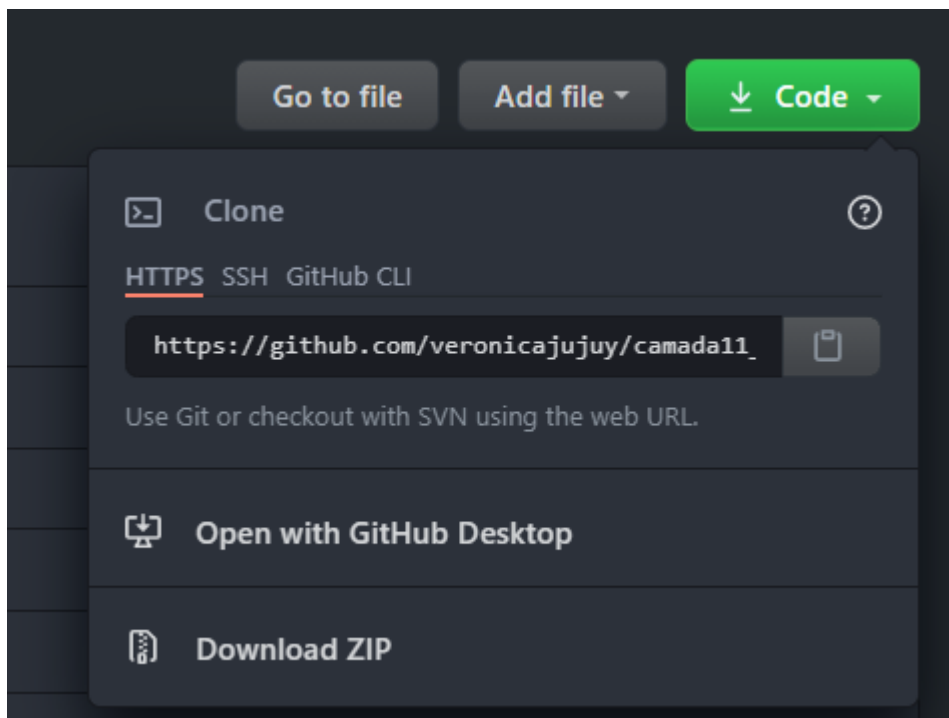
```
git config user.email
```


6. Si no estuviese configurado, es momento que lo hagas, y que el usuario sea el que le pasaste a tu profesor o profesora:

```
git config user.name "nombre_usuario"  
git config user.email "email@email.com"
```

Clonando el repositorio en mi computadora

7. Luego, para clonar el repositorio en tu computadora debés hacer clic en el botón **Code** ubicado en el margen superior. Allí se desplegará una ventana con la dirección URL correspondiente al repositorio. ¡Es muy importante que la opción **HTTPS** esté subrayada (Verificá que no esté subrayada la opción SSH).



8. Copia la dirección haciendo clic en en el icono de la carpeta: 
9. En tu computadora, elegí una carpeta para clonar el repositorio. Puede ser la carpeta de la materia, ya que una vez que se ejecute el comando de clonación se creará una carpeta dentro de ella que se llamará como el repositorio de la camada. No hace falta que inicialices esa carpeta con git (no hace falta que hagas git init en esa carpeta).

10. ¡Ahora podés clonar!

Ejecutá el comando `git clone "url_del_repositorio"` en la carpeta elegida:

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH
$ git clone https://github.com/veronicajujuy/camada11_mochila.git
```

11. Si es la primera vez que clonas un repositorio, github te pedirá que te loguees.

En ese caso deberás introducir tu usuario y contraseña donde te indique o te redireccionará a una página en donde deberás aceptar la conexión.

12. Una vez clonado te aparecerá en tu terminal algo como esto:

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH
$ git clone https://github.com/veronicajujuy/camada11_mochila.git
Cloning into 'camada11_mochila'...
remote: Enumerating objects: 1000, done.
remote: Counting objects: 100% (493/493), done.
remote: Compressing objects: 83% (200/240) eceiving objects: 0% (1/10)
remote: Compressing objects: 93% (224/240) eceiving objects: 1% (10/1)
remote: Compressing objects: 100% (240/240), done.
remote: Total 1000 (delta 325), reused 405 (delta 250), pack-reused 507R

Receiving objects: 100% (1000/1000), 11.83 MiB | 1.67 MiB/s, done.
Resolving deltas: 100% (490/490), done.

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH
$
```

13. ¡Ahora ya puedes acceder a la mochila! Si ejecutas el comando `ls`, te darás cuenta que habrá una nueva carpeta con el nombre del repositorio clonado.

Para acceder a él, solo deberás cambiar de carpeta y posicionarte en la nueva:

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH
$ cd camada11_mochila/

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (main)
$
```

14. Te darás cuenta que estas en la carpeta correcta porque al lado del nombre de la mochila aparecerá en color celeste la rama principal "main".

Si listas las carpetas te aparecerán todas las carpetas de las clases donde deberás, a partir de ahora, subir la práctica realizada en las mismas:

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (main)
$ ls
Clase_10_Sistemas_Operativos/
Clase_11_Procesos/
Clase_12_Cierre_Semana/
Clase_13_Primer_Examen/
Clase_14_Lenguajes_Maquinas_Virtuales/
Clase_15_Cierre_Semana/
Clase_16_Redes/
Clase_17_Protocolos/
Clase_18_Cierre_Semana/
Clase_19_Protocolos_II/
Clase_1_Introduccion/
Clase_20_Protocolos_III/
Clase_21_Cierre_Semana/
Clase_22_Evaluacion_Final/
Clase_23_Amenazas_Informaticas/
Clase_24_Cierre_Semana/
Clase_25_Etica_Informatica/
Clase_26_Cierre/
```

Creando mi rama y subiendo las ejercitaciones

15. Para crear una rama utilizamos el comando

```
git branch <nombre-rama>
```

Para la mochila crearemos una rama con nuestro apellido y nombre: Ejemplo:

```
git branch Potter_Harry
```

Luego nos cambiamos a esa rama utilizando el comando

```
git checkout <nombre-rama>
```

En nuestro caso:

```
git checkout Potter_Harry
```

16. Podremos visualizar que estamos en nuestra rama porque en vez de (main) nos figurará nuestro apellido y nombre:

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (main)
$ git branch Potter-Harry

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (main)
$ git checkout Potter-Harry
Switched to branch 'Potter-Harry'

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (Potter-Harry)
$
```

17. Ingresamos a la clase correspondiente del día. Si no está especificado un nombre en particular crearemos la carpeta:

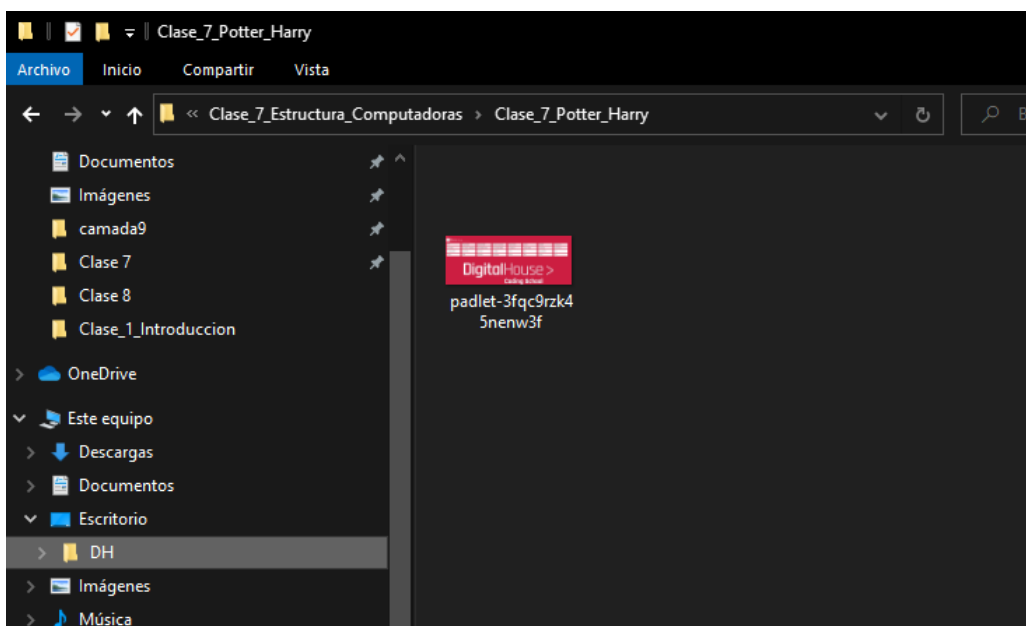
Clase_X_Apellido_y_Nombre

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila (Potter-Harry)
$ cd Clase_7_Estructura_Computadoras/

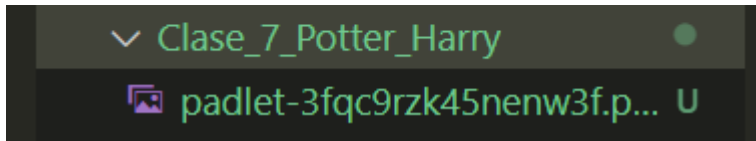
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Computadoras (Potter-Harry)
$ mkdir Clase_7_Potter_Harry

User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Computadoras (Potter-Harry)
$
```

18. Allí crearemos o subiremos el contenido trabajado en clases. Si se trata de un archivo .jpg .png o .pdf podés copiarlo directamente desde el explorador de archivos:



19. En el Visual Studio code, nos aparecerá que tenemos un nuevo archivo Untracked:



20. Si corremos el comando git status nos aparecerá la nueva carpeta creada lista para hacerle un add:

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Computadoras (Potter-Harry)
$ git status
On branch Potter-Harry
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Clase_7_Potter_Harry/

nothing added to commit but untracked files present (use "git add" to track)
```

21. Es hora de hacer el git add . y el commit

```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Computadoras (Potter-Harry)
$ git add .

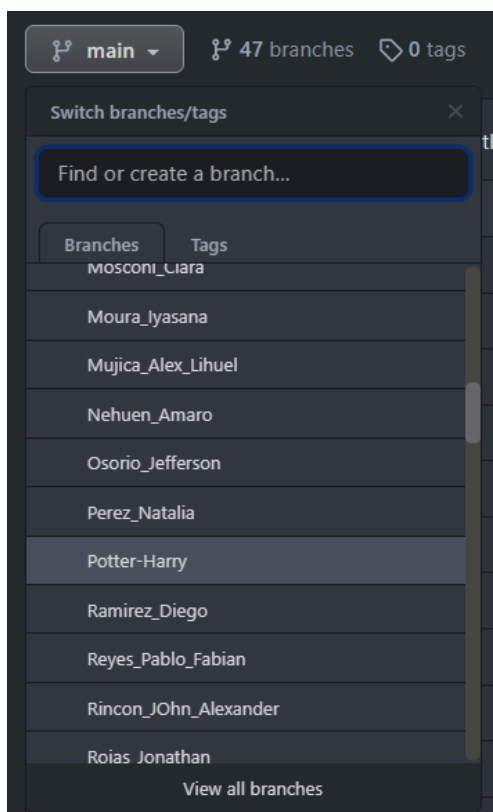
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Computadoras (Potter-Harry)
$ git commit -m "archivo clase 7 agregado"
[Potter-Harry c4e76a3] archivo clase 7 agregado
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Clase_7_Estructura_Computadoras/Clase_7_Potter_Harry/padlet-3fqc9rzk45nenw3f.png
```

22. Una vez que hicimos el commit podemos pushear a nuestra rama con el comando:

```
git push origin <nombre-rama>
```

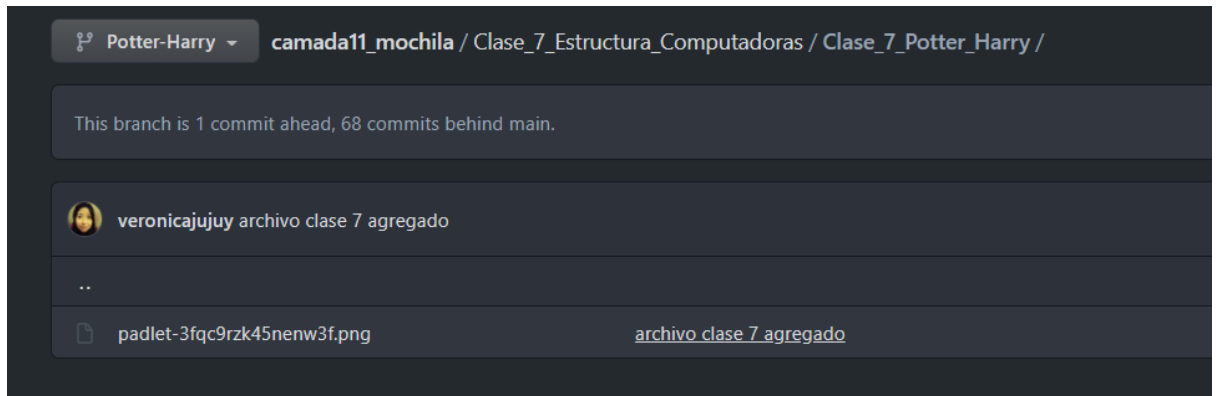
```
User@DESKTOP-8JN8B10 MINGW64 ~/Desktop/DH/camada11_mochila/Clase_7_Estructura_Compu
tadoras (Potter-Harry)
$ git push origin Potter-Harry
Enumerating objects: 90, done.
Counting objects: 100% (85/85), done.
Delta compression using up to 4 threads
Compressing objects: 100% (40/40), done.
Writing objects: 100% (67/67), 1004.04 KiB | 363.00 KiB/s, done.
Total 67 (delta 30), reused 60 (delta 26), pack-reused 0
remote: Resolving deltas: 100% (30/30), completed with 8 local objects.
remote:
remote: Create a pull request for 'Potter-Harry' on GitHub by visiting:
remote:   https://github.com/veronicajujuy/camada11_mochila/pull/new/Potter-Harry
remote:
To https://github.com/veronicajujuy/camada11_mochila.git
 * [new branch]      Potter-Harry -> Potter-Harry
```

23. Podemos ver en el repositorio remoto que nuestra rama fue creada y está listada junto a la rama de nuestros compañeros:



Si selecciono la rama, puedo ver el último commit que hice y el archivo en la

carpeta correspondiente a la clase:



24. Ya tengo mi archivo en el repo de la camada!!

Agregando más archivos a mi rama

Cuando tengamos que agregar más archivos porque vamos avanzando en las clases, vamos a dirigirnos a la rama que creamos anteriormente con el comando:

```
git checkout <nombre-rama>
```

Ahora puedo repetir el proceso desde el punto **17**!!

¡Esperamos que esta guía te ayude a trabajar correctamente en tu mochila!