

# Práctica 1 – Git

## Introducción

En esta primera práctica se emplean los conocimientos adquiridos a lo largo del comienzo del curso sobre Git, probando y explicando una serie de acciones importantes de este sistema de control de versiones.

## Git clone

El comando `git clone` es utilizado para clonar un repositorio. Un repositorio clonado es una copia de esta local que se puede modificar, realizar *commits* y empujar al repositorio original remoto (asumiendo que se tienen los requisitos necesarios para ello). Se teclea el comando seguido por el enlace del repositorio remoto que se desea copiar. Dado que al comienzo de la práctica se pide hacer *fork* de un repositorio proporcionado por la universidad (p1-fork), lo siguiente sería clonar p1-fork para continuar trabajando esta de manera local.

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub>git clone https://github.com/zonkoholland/p1-fork.git
Cloning into 'p1-fork'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 5 (from 1)
Receiving objects: 100% (6/6), done.

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub>cd /p1-fork
The system cannot find the path specified.

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub>cd p1-fork

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork>
```

## Git add y Git commit

Git add prepara archivos cualquier cambio seleccionado para el próximo commit. A continuación, el commit realiza un guardado del repositorio local actual, creando una “captura” actual a la cuál se puede incluso volver si hiciese falta. Se teclea el comando de add seguido de los archivos modificados que se quieren preparar. Como ejemplo, aquí se utiliza para añadir la captura previa del clone al repositorio con el tal de incluirlo posteriormente al archivo README (el prefijo -m después del `git commit` permite añadir un mensaje explicando el propósito de esos cambios).

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork\photos>git add git_clone.png

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork\photos>git commit -m "Captura de clonar"
[main 2375a4b] Captura de clonar
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 photos/git_clone.png

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork\photos>
```

## Git checkout

Git checkout es utilizado para cambiar a una rama ya existente. Sin embargo, se puede acompañar de un prefijo -b para directamente crear una nueva rama por si esta no existiese. Eso mismo se hace aquí, creando una nueva rama documentos que se utilizará para subir los archivos necesarios para la práctica.

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork>git checkout -b documentos
```

```
Switched to a new branch 'documentos'
```

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork>git branch
```

```
* documentos
```

```
main
```

## Git status

Este comando muestra todo cambio añadido y preparado para hacer un commit. Después de hacer modificaciones al archivo README, se llama a este comando y muestra lo siguiente (esto es previo a realizar un commit):

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork>git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

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

C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones Telemáticas\GitHub\p1-fork>
```

## Git push

Este comando empuja todos los cambios guardados en local al remoto. En este instante se ha añadido y hecho commit del documento entorno.docx a la rama documentos y se empuja para guardarlo en remoto:

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones  
Telemáticas\GitHub\p1-fork>git branch
```

```
* documentos
```

```
main
```

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones  
Telemáticas\GitHub\p1-fork>git push
```

```
info: please complete authentication in your browser...
```

```
Enumerating objects: 4, done.
```

```
Counting objects: 100% (4/4), done.
```

```
Delta compression using up to 4 threads
```

```
Compressing objects: 100% (3/3), done.
```

```
Writing objects: 100% (3/3), 140.18 KiB | 11.68 MiB/s, done.
```

```
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
```

```
To https://github.com/zonkoholland/p1-fork.git
```

```
b9821a7..28907f8 documentos -> documentos
```

```
C:\Users\holla\Documents\ICAI\AAA Tercero\Programación de Aplicaciones  
Telemáticas\GitHub\p1-fork>
```