



# GitHub

## Configuración Básica

### Configurar Nombre que salen en los commits

```
git config --global user.name "GitUserName"
```

### Configurar Email

```
git config --global user.email "GitRegisteredEmail"
```

### Marco de colores para los comandos

```
git config --global color.ui true
```

## Iniciando repositorio

### Iniciamos GIT en la carpeta donde esta el proyecto

```
git init
```

### Clonamos el repositorio de github

```
git clone <url>
```

### Añadimos todos los archivos para el commit

```
git add .
```

### Hacemos el primer commit

```
git commit -m "Texto que identifique por que se hizo el commit"
```

### subimos al repositorio

```
git push origin master
```

## GIT CLONE

### Añadimos todos los archivos para el commit

```
git add .
```

### Añadimos el archivo para el commit

```
git add <archivo>
```

### Añadimos todos los archivos para el commit omitiendo los nuevos

```
git add --all
```

### Añadimos todos los archivos con la extensión especificada

```
git add *.txt
```

### Añadimos todos los archivos dentro de un directorios

```
git add docs/
```

## GIT COMMIT

Cargar en el HEAD los cambios realizados

```
git commit -m "Texto que identifique  
por que se hizo el commit"
```

**Agregar y Cargar en el HEAD los cambios realizados**

```
git commit -a -m "Texto que identifique  
por que se hizo el commit"
```

**De haber conflictos los muestra**

```
git commit -a
```

## GIT PUSH

**Subimos al repositorio**

```
git push <origien> <branch>
```

**Subimos un tag**

```
git push --tags
```

## GIT LOG

**Muestra los logs de los commits**

```
git log
```

**Muestras los cambios en los commits**

```
git log --oneline --stat
```

## GIT REMOTE

**Agregar repositorio remoto**

```
git remote add origin <url>
```

## GIT BRANCH

**Crea un branch**

```
git branch <nameBranch>
```

**Lista los branches**

```
git branch
```

**Comando -d elimina el branch y lo une al master**

```
git branch -d <nameBranch>
```

**Elimina sin preguntar**

```
git branch -D <nameBranch>
```

## OTROS COMANDOS

**Lista un estado actual del repositorio con lista de archivos modificados o agregados**

```
git branch <nameBranch>
```

**Quita del HEAD un archivo y le pone el estado de no trabajado**

```
git checkout -- <file>
```

**Crea un branch en base a uno online**

```
git checkout -b newlocalbranchname  
origin/branch-name
```

**Busca los cambios nuevos y actualiza el repositorio**

```
git pull origin <nameBranch>
```

**Cambiar de branch**

```
git checkout <nameBranch/tagname>
```

**Une el branch actual con el especificado**

```
git merge <nameBranch>
```

**Borrar un archivo del repositorio**

```
git rm <archivo>
```