



Manual de Git i de GitHub

Manual de Git i de GitHub



Govern de les Illes Balears

Vicepresidència i Conselleria
d'Innovació, Recerca i Turisme
Direcció General de Desenvolupament Tecnològic



Govern de les Illes Balears

Fundació Balear d'Innovació i Tecnologia



Informació general del document.

Descripció.

Títol:	Manual de GIT i Manual de GitHub
Estat:	Esborrany/Aprovat
Versió:	1.0
Autor/s:	Antoni Nadal Bennasar
Creat:	10/09/15
Modificat	25/04/2016
Fitxer:	ManualGit&GitHub.odt

Històric de modificacions.

Comentari:	Autor/s:	Data:
------------	----------	-------

Font documental.

<https://en.wikipedia.org/wiki/GitHub>
<https://github.com>



Index de Contingut

1.-Introducció.....	4
1.1.-Instal·lar Git i GitTortoise.....	4
1.1.1.-Instal·lació en Windows	4
1.1.2.-Instal·lació en Linux.....	4
1.1.3.-Notes sobre Git i GitHub.....	4
1.2.-Conèixer Git.....	4
2.-Guia Ràpida de GIT.....	5
2.1.-Branques.....	5
2.1.1.-Com es descarrega una branca.....	5
2.1.2.-Actualitzar codi del repositori remot.....	6
2.1.3.-Enviar canvis al repositori remot.....	6
2.2.-Tags.....	7
2.2.1.-Llistar tags.....	7
2.2.2.-Crear un tag.....	7
2.2.3.-Esborrar un tag.....	8
2.3.-Comandes d'Utilitat.....	8
2.3.1.-Estat de repositori: quina branca i fitxers pendents	8
2.3.2.-URL d'origen.....	8
2.3.3.-Guardar credencials TortoiseGit.....	8
3.-GitHub.....	10
3.1.-Rebre notificacions de cert repositori.....	10



1.- Introducció

1.1.- Instal·lar Git i GitTortoise

1.1.1.- Instal·lació en Windows

Primer instal·lar "Git Tortoise" després et donarà les indicacions per instal·lar el Git Command Line (Git via línia de comandes):

- Instal·lar TortoiseGit: <https://tortoisegit.org/>
- Instal·lar GIT Command Line: <https://github.com/git-for-windows/git/releases/tag/v2.5.2.windows.2>

1.1.2.- Instal·lació en Linux

<https://www.digitalocean.com/community/tutorials/how-to-install-git-on-ubuntu-14-04>

1.1.3.- Notes sobre Git i GitHub

NOTA 1: La diferencia entre SVN i Git més important és que el nostre PC funciona com un servidor de GIT, per això hem de fer un commit en local (commit) i si després un commit global (push) per a que els canvis es distribueixin de per tot.

NOTA 2: Es recomana no emprar GitHub Desktop ja que es una eina directament relacionada amb GitHub. Val més eines genèriques que facin feina directament amb Git i no en la plataforma o forja on estigui Git.

1.2.- Conèixer Git

A continuació hi ha dos enllaços a dos tutorials bastant interessants.

- <http://rogerdudler.github.io/git-guide/index.es.html>: tutorial molt esquemàtic i minimalista però bastant efectiu sobre comandes Git.
- <http://blog.openalfa.com/git-para-usuarios-de-svn>: Git per a usuaris de SVN.



2.- Guia Ràpida de GIT

2.1.- Branques

2.1.1.- Com es descarrega una branca

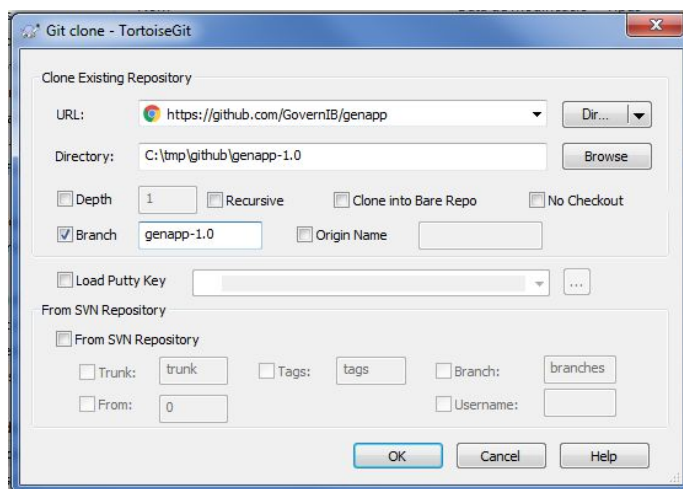
El primer es anar a un directori on volem el codi de la nostra branca. Anirem a aquest directori i executarem la següent comanda. Com es pot observar hi ha un nou paràmetre “--branch” amb el que podem elegir la branca i al final de la comanda indicam en quin subdirectori volem descarregar el codi (“genapp-1.0”).

```
$git clone --branch genapp-1.0 https://github.com/GovernIB/genapp genapp-1.0
Cloning into 'genapp'...
POST git-upload-pack (200 bytes)
remote: Counting objects: 1900, done.
remote: Compressing objects: 100% (1015/1015), done.
Receiving objects: 100% (1900/1900), 4.94 MiB | 1.25 MiB/s, done.
Resolving deltas: 100% (515/515), done.
Checking connectivity... done.
Checking out files: 100% (737/737), done.
```

Verificam que el checkout s'ha realitzat de la branca correcte executant un “git status”

```
$ cd genapp-1.0
$ git status
On branch genapp-1.0
Your branch is up-to-date with 'origin/genapp-1.0'.
nothing to commit, working directory clean
```

Podem fer el mateix, però amb GitTortoise tal i com es mostra en la captura de l'esquerra. Hem d'elegir la URL, el directori on descarregar el codi i marcar que volem una branca i escriure que la branca que volem és la “genapp-1.0”





2.1.2.- Actualitzar codi del repositori remot

Una de les coses que hem de saber fer, es actualitzar el codi que tenim “descarregat” amb les modificacions que altres usuaris han enviat al servidor (commits). Això es fa amb la comanda `git pull` (la comanda més o manco equivalent amb SVN seria “`svn up`”):

```
$ git pull
Already up-to-date.
```

En el cas anterior ens indica que no hi ha modificacions pendents en el servidor. Un exemple que extreu canvis del repositori remot es mostra a continuació:

```
$ git pull
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 2), reused 3 (delta 2), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/GovernIB/genapp
   93e0e92..190a5e7  genapp-1.0 -> origin/genapp-1.0
Updating 93e0e92..190a5e7
Fast-forward
 pom.xml | 96 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++-----
 1 file changed, 89 insertions(+), 7 deletions(-)
```

2.1.3.- Enviar canvis al repositori remot

Si hem afegit algun fitxer llavors abans d'enviar els canvis haurem d'indicar a git que hem d'incloure un nou fitxer al repositori:

```
$git add <filename>
```

Un cop ens funcionin els canvis realitzats en la branca, el primer de tot serà actualitzar el nostre projecte amb els canvis d'altres usuaris fent el que es diu al punt “2.1.2.-Actualitzar codi del repositori remot”.

Després ja podem continuar amb l'enviament en si:

- Indicarem els fitxers que volem commitar (`git add`)
- Enviarem els canvis al repositori local (`git commit`)
- Enviarem els canvis al repositori remot (`git push`)

El codi per fer l'anterior es mostra a continuació:

```
$ git add .
$ git commit -m "Adaptar pom a GitHub"
[genapp-1.0 190a5e7] Adaptar pom a GitHub
 1 file changed, 89 insertions(+), 7 deletions(-)
$ git push
warning: push.default is unset; its implicit value has changed in
```



Git 2.0 from 'matching' to 'simple'. To squelch this message and maintain the traditional behavior, use:

```
git config --global push.default matching
```

To squelch this message and adopt the new behavior now, use:

```
git config --global push.default simple
```

When push.default is set to 'matching', git will push local branches to the remote branches that already exist with the same name.

Since Git 2.0, Git defaults to the more conservative 'simple' behavior, which only pushes the current branch to the corresponding remote branch that 'git pull' uses to update the current branch.

See 'git help config' and search for 'push.default' for further information. (the 'simple' mode was introduced in Git 1.7.11. Use the similar mode 'current' instead of 'simple' if you sometimes use older versions of Git)

```
Username for 'https://github.com': anadal-fundaciobit
Password for 'https://anadal-fundaciobit@github.com':
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.14 KiB | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
To https://github.com/GovernIB/genapp
 93e0e92..190a5e7  genapp-1.0 -> genapp-1.0
```

2.1.4.- Crear Nova branca

Estant dins un repositori hem d'executar la següent comanda:

```
$ git checkout -b genapp-1.1
Switched to a new branch 'genapp-1.1'
```

La nova branca ja està creada en local, però l'hem d'enviar al repositori:

```
$ cd genapp
$ git push origin genapp-1.1
Username for 'https://github.com': kitanibaladmin
Password for 'https://kitanibaladmin@github.com':
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/GovernIB/genapp
 * [new branch]      genapp-1.1 -> genapp-1.1
```



2.2.- Tags

2.2.1.- Llistar tags

La comanda és molt senzilla:

```
$ git tag
```

2.2.2.- Crear un tag

Només s'ha d'indicar el nom del nou tag i un missatge de commit. Un cop afegit el tag l'hem d'enviar al servidor remot (git push):

```
$ git tag -a genapp-1.0.0 -m 'GenApp Tag 1.0.0'
$ git push --tags
```

També es pot crear emprant l'entorn web de GitHub.

2.2.3.- Esborrar un tag

Suposem que hem creat un tag genapp-1.0.0 però que ens hem equivocat i l'hem d'esborrar. Llavors heu de seguir les següents passes:

```
$ git tag -d genapp-1.0.0
Deleted tag '1.0.0' (was 190a5e7)

$ git push origin :refs/tags/genapp-1.0.0
Username for 'https://github.com': kitanibaladmin
Password for 'https://kitanibaladmin@github.com':
To https://github.com/GovernIB/genapp
- [deleted]          genapp-1.0.0
```

Després haurem d'anar via web a l'apartat de Tags/Releases del repositori i eliminar el draft de Releases associat al que acabam d'esborrar.

2.3.- Comandes d'Utilitat

2.3.1.- Estat de repositori: quina branca i fitxers pendents

Ens informa en quina branca ens trobam i l'estat de commit dels fitxers.


```
$ git status
On branch portafib-1.1
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    all.bat

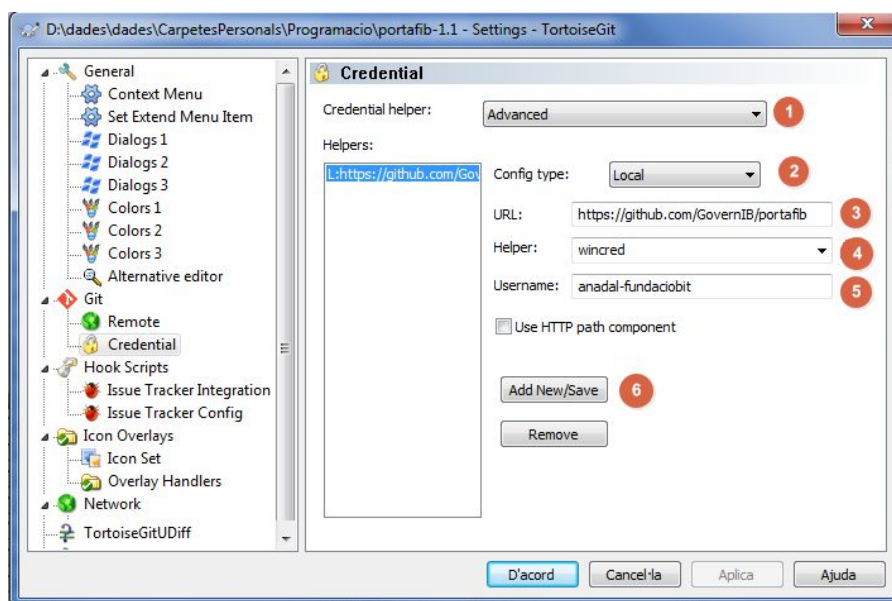
nothing added to commit but untracked files present (use "git add" to track)
```

2.3.2.- URL d'origen

Ens retorna la URL des d'on s'ha obtingut el repositori

```
$ git config --get remote.origin.url
https://github.com/GovernIB/portafib
```

2.3.3.- Guardar credencials TortoiseGit



- (1) Elegir Advanced
- (2) Elegir Local
- (3) Escriure la URL d'origen del repositori GitHub
- (4) Elegir wincred
- (5) Posar el vostre username de GitHub
- (6) Guardar



En principi qualsevol push que es realitzi sobre aquest repositori emprarà les credencials guardades.

2.3.4.- Recuperar Fitxers Perduts (Esborrats en Local)

A vegades esborram fitxers sense voler en la nostra còpia del repositori. Aquests fitxers no es recuperen fent un “git pull”. S'ha d'executar un “git checkout .” per recuperar aquests fitxers perduts o esborrats en local.

2.3.5.- *IgnoreFiles* només en Local (.git/info/exclude)

Hi ha fitxers que només tenim en el nostre repositori i que no volem afegir al fitxer .gitignore ja que és de àmbit global. Per això existeix el fitxer `.git/info/exclude` que és com el .gitignore però que només s'aplica al nostre repositori local.

3.- GitHub

3.1.- Rebre notificacions de cert repositori

Per rebre correus dels canvis que es realitzen en algun repositori podeu seguir les instruccions que es detallen en aquest enllaç <https://help.github.com/articles/receiving-email-notifications-for-pushes-to-a-repository/>.