



GRADO EN INGENIERÍA INFORMÁTICA

Calidad del Software

José Antonio Molins González - Curso 2014-15

**Tarea Obligatoria 4 – Uso de Git**

## Contenido

1.	Explicación manejo Git.....	2
1.1	Realizar un fork del repositorio original (bifurcación) .....	2
1.2	Descargamos el fork en nuestro equipo .....	4
1.3	Actualizar la copia local y el fork .....	5
1.4	Actualizamos nuestro repositorio de GitHub.....	6
1.5	Pull Request .....	8

## 1. Explicación manejo Git

En mi caso al trabajar en Windows, inicialmente he descargado el gestor de GIT oficial de la dirección <http://git-scm.com/download/win>. Este gestor incluye GIT bash, que es un emulador de terminal que permite ejecutar comandos GIT desde línea de comandos.

A continuación se resumen los pasos que se han seguido para completar el pull Request de la tarea:

### 1.1 Realizar un fork del repositorio original (bifurcación)

Accedemos a través de internet al repositorio definido por el profesor (<https://github.com/dcoloma/software-quality-unit3>). Realizamos un fork del directorio pulsando sobre el icono que dice fork. Esto crea una copia del repositorio en nuestra cuenta de GitHub

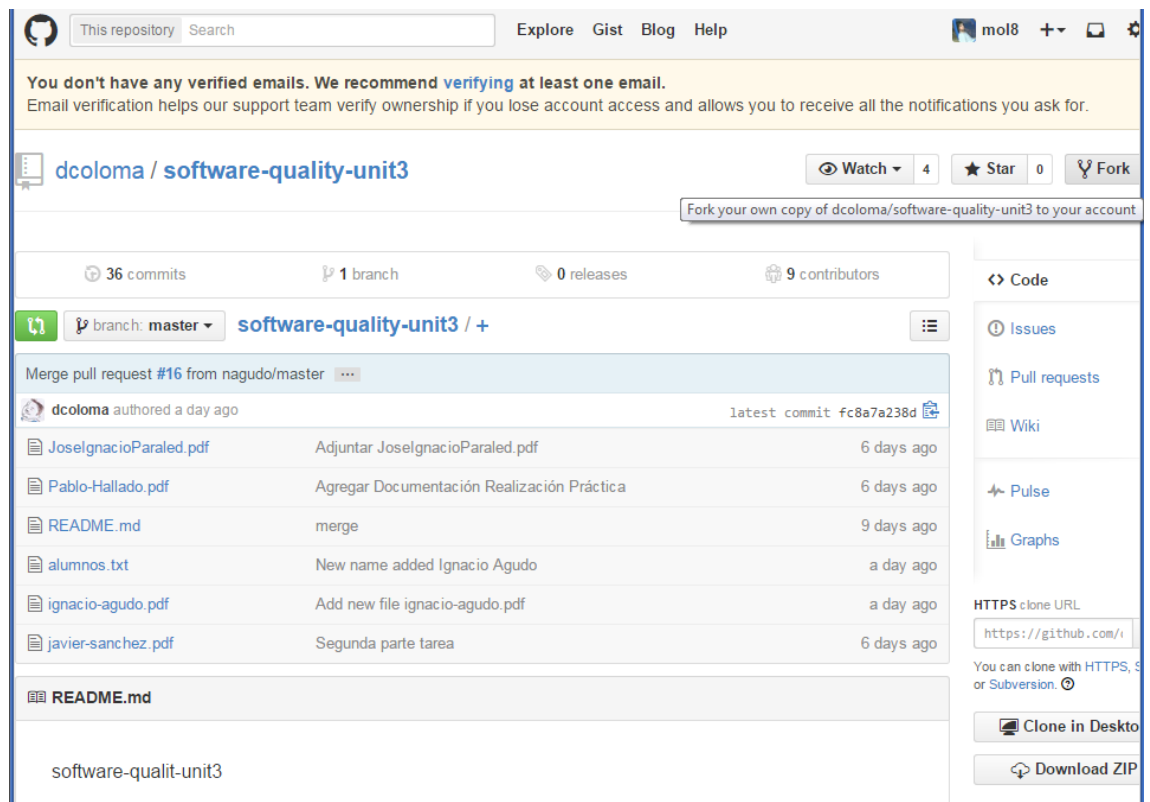
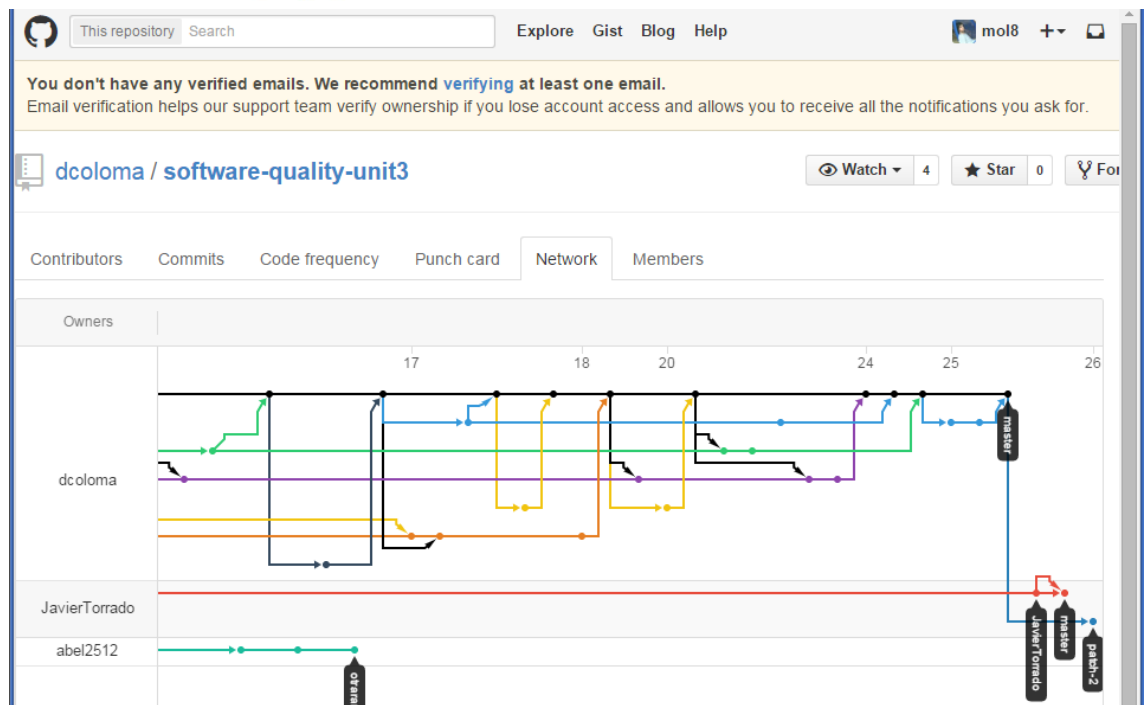
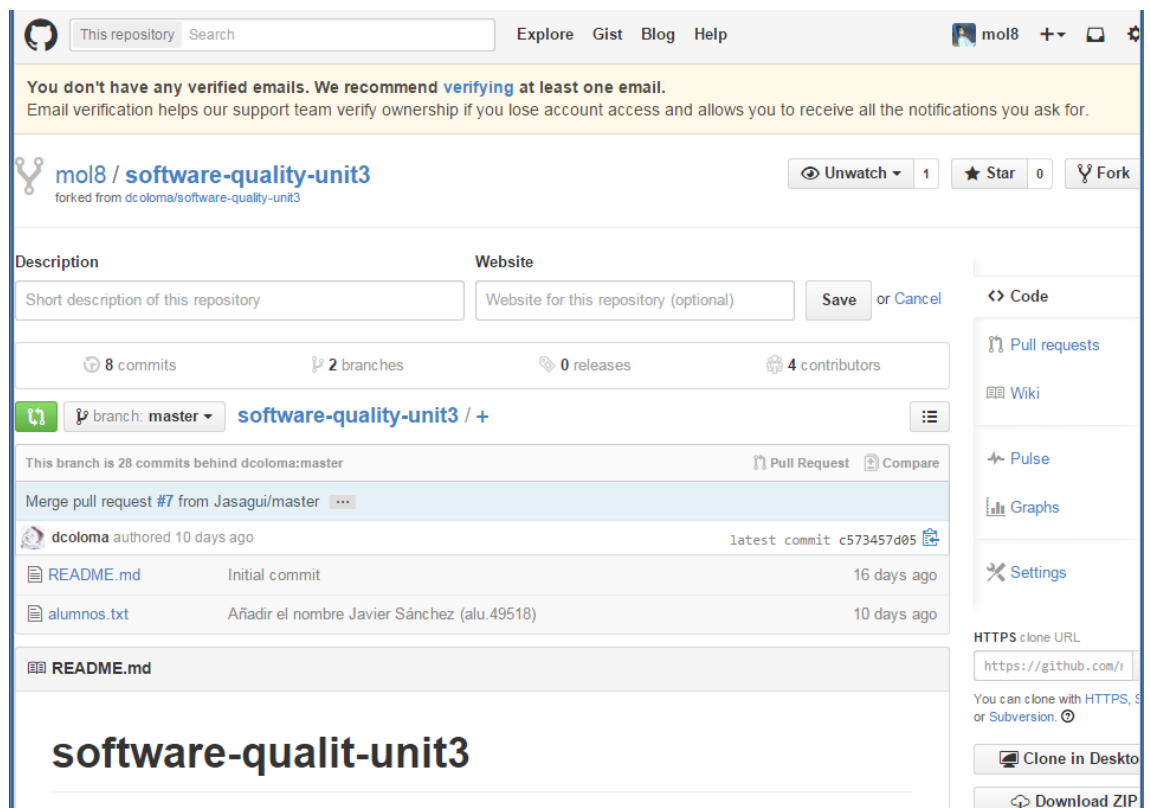


Ilustración 1 - Repositorio Original



**Ilustración 2 - Vista Forks y Commits del repositorio original**

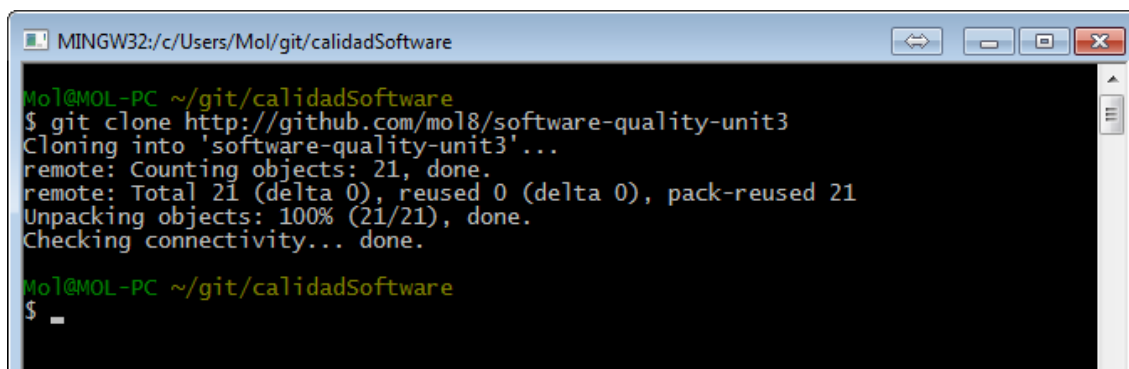


**Ilustración 3 - Rama generada en cuenta de GitHub**



## 1.2 Descargamos el fork en nuestro equipo

Para poder trabajar con los archivos contenidos dentro del proyecto que hemos traído a nuestra cuenta, descargamos estos archivos a nuestro equipo. Para esto primero nos situamos en el directorio en el que queremos clonar el proyecto y luego ejecutamos el comando `$ git clone https://github.com/mol8/software-quality-unit3`



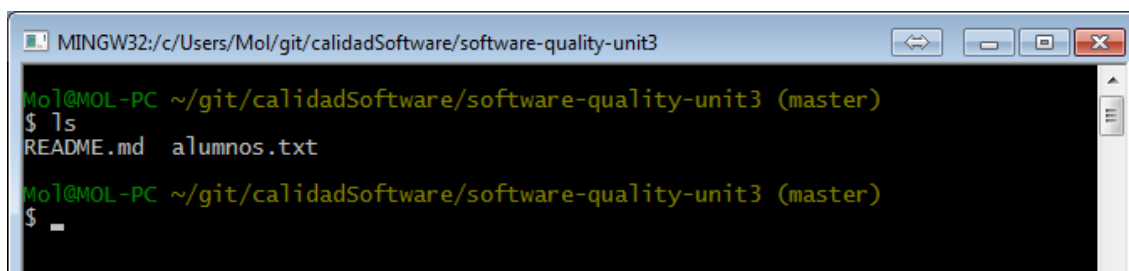
```
MINGW32:/c/Users/Mol/git/calidadSoftware

Mol@MOL-PC ~/git/calidadSoftware
$ git clone http://github.com/mol8/software-quality-unit3
Cloning into 'software-quality-unit3'...
remote: Counting objects: 21, done.
remote: Total 21 (delta 0), reused 0 (delta 0), pack-reused 21
Unpacking objects: 100% (21/21), done.
Checking connectivity... done.

Mol@MOL-PC ~/git/calidadSoftware
$
```

**Ilustración 4 - Clone al equipo local del repositorio de GitHub**

Luego podemos acceder al directorio que se ha creado al hacer el clone y ver los archivos que se han insertado.



```
MINGW32:/c/Users/Mol/git/calidadSoftware/software-quality-unit3

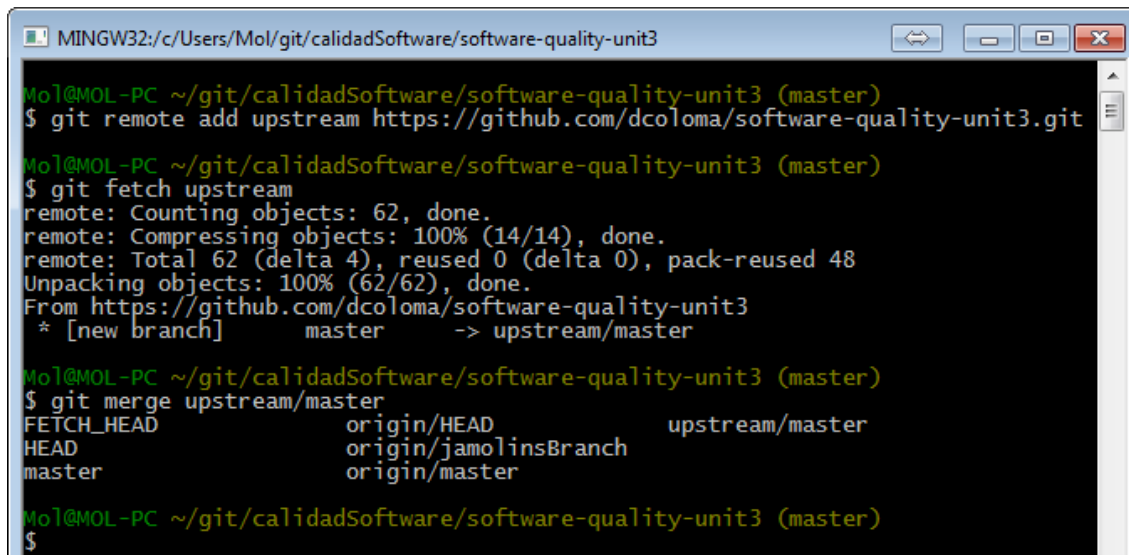
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ ls
README.md  alumnos.txt

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$
```

**Ilustración 5 - Archivos dentro del repositorio descargado**

### 1.3 Actualizar la copia local y el fork

A continuación se ha de proceder a actualizar la copia local y el fork de nuestra cuenta. Para eso, añadimos la rama remota (add), nos traemos su contenido (fetch) y lo juntamos con la rama master en local (merge).



```

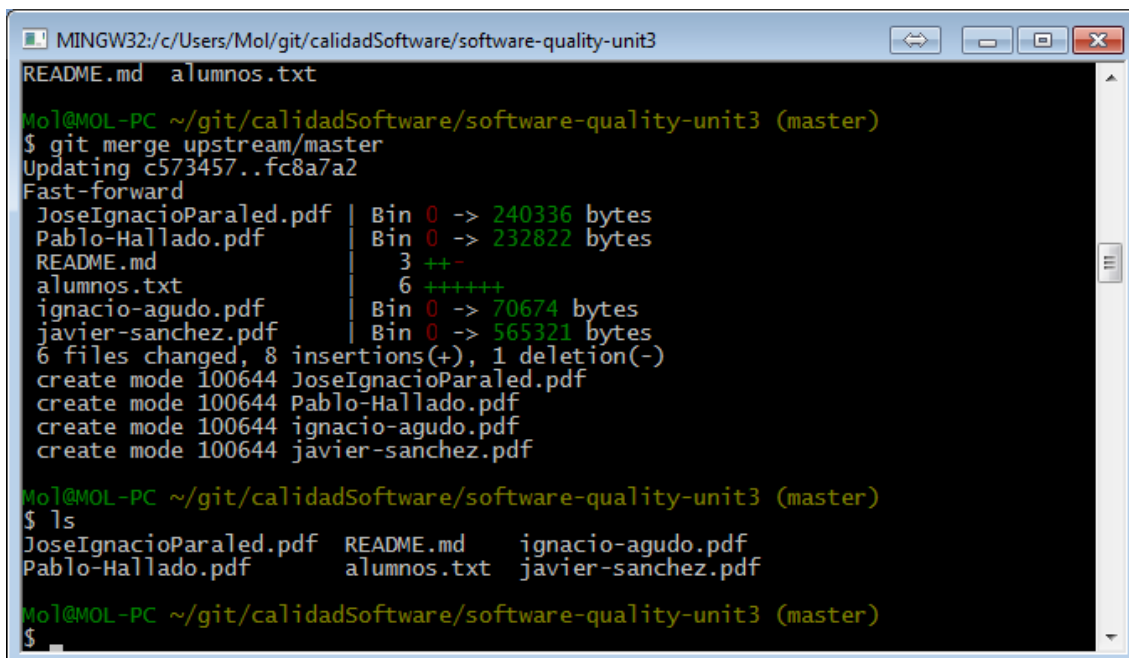
MINGW32:/c/Users/Mol/git/calidadSoftware/software-quality-unit3
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git remote add upstream https://github.com/dcoloma/software-quality-unit3.git

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git fetch upstream
remote: Counting objects: 62, done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 62 (delta 4), reused 0 (delta 0), pack-reused 48
Unpacking objects: 100% (62/62), done.
From https://github.com/dcoloma/software-quality-unit3
* [new branch]      master       -> upstream/master

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git merge upstream/master
FETCH_HEAD          origin/HEAD          upstream/master
HEAD                origin/jamolinsBranch
master              origin/master

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$
  
```

Ilustración 6 - Generación de rama remota



```

MINGW32:/c/Users/Mol/git/calidadSoftware/software-quality-unit3
README.md  alumnos.txt

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git merge upstream/master
Updating c573457..fc8a7a2
Fast-forward
 JoseIgnacioParaled.pdf | Bin 0 -> 240336 bytes
 Pablo-Hallado.pdf      | Bin 0 -> 232822 bytes
 README.md              | 3 ++-
 alumnos.txt            | 6 ++++++
 ignacio-agudo.pdf      | Bin 0 -> 70674 bytes
 javier-sanchez.pdf     | Bin 0 -> 565321 bytes
 6 files changed, 8 insertions(+), 1 deletion(-)
 create mode 100644 JoseIgnacioParaled.pdf
 create mode 100644 Pablo-Hallado.pdf
 create mode 100644 ignacio-agudo.pdf
 create mode 100644 javier-sanchez.pdf

Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ ls
JoseIgnacioParaled.pdf  README.md  ignacio-agudo.pdf
Pablo-Hallado.pdf      alumnos.txt  javier-sanchez.pdf

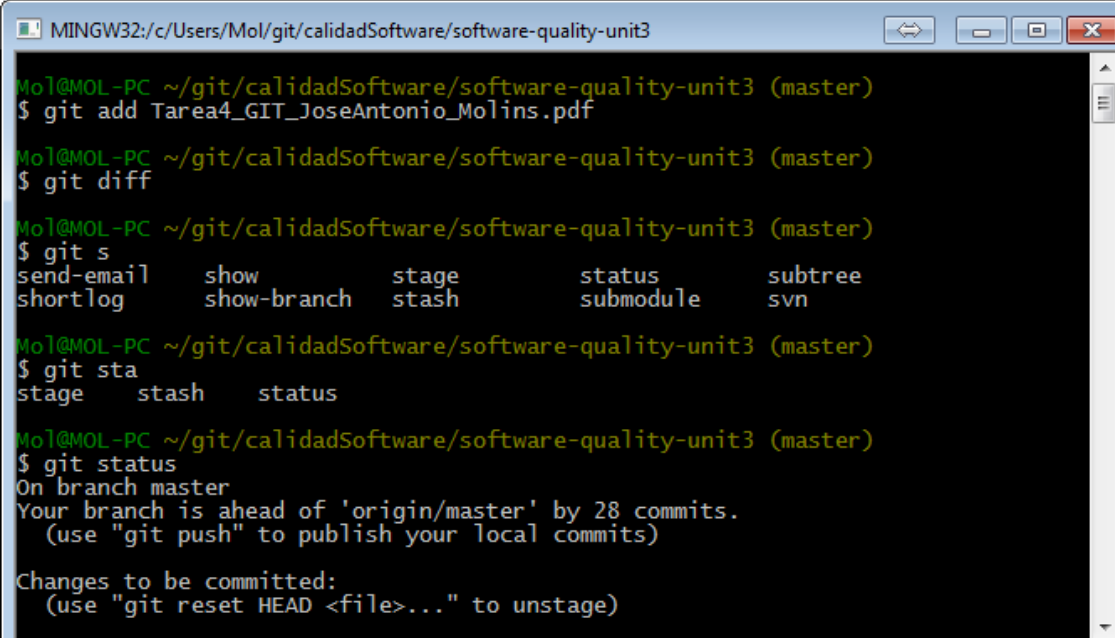
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$
  
```

Ilustración 7 - Descarga de nuevos ficheros de la rama remota



## 1.4 Actualizamos nuestro repositorio de GitHub

Aquí subimos los cambios realizados en local en la rama master de nuestro repositorio. Añadimos el archivo pdf que hemos generado.



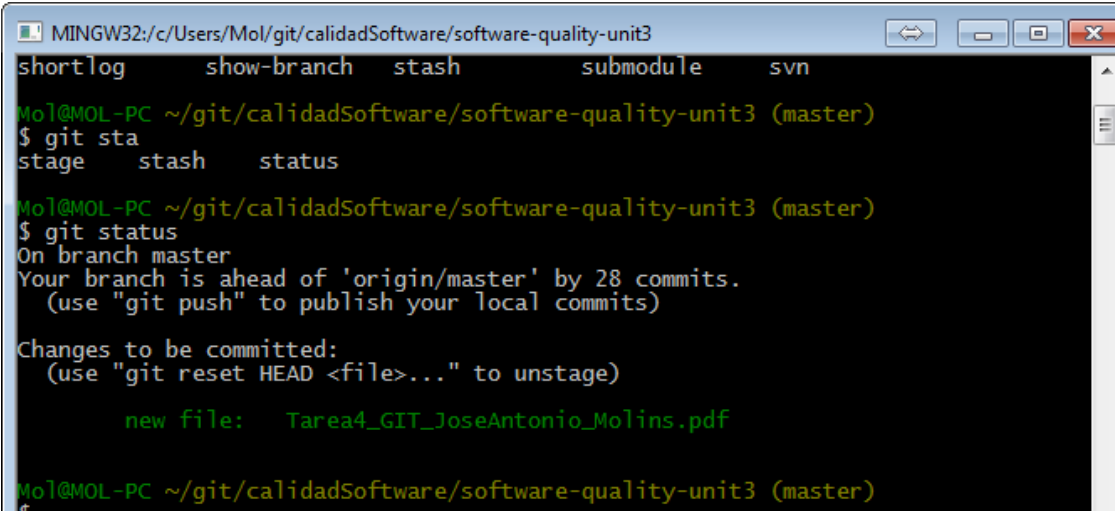
```
MINGW32:/c/Users/Mol/git/calidadSoftware/software-quality-unit3
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git add Tarea4_GIT_JoseAntonio_Molins.pdf
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git diff
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git s
send-email      show      stage      status      subtree
shortlog       show-branch  stash      submodule   svn
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git sta
stage  stash  status
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 28 commits.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

       new file:   Tarea4_GIT_JoseAntonio_Molins.pdf
```

Ilustración 8 - Añadimos el archivo a subir

Comprobamos con status que existe un nuevo archivo para subir al repositorio.



```
MINGW32:/c/Users/Mol/git/calidadSoftware/software-quality-unit3
shortlog      show-branch  stash      submodule   svn
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git sta
stage  stash  status
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 28 commits.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

       new file:   Tarea4_GIT_JoseAntonio_Molins.pdf
Mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$
```

Ilustración 9 - Comprobamos el estado del repositorio local en rama master

Por último, generamos un commit con un comentario para subir los cambios y subimos los cambios realizados al repositorio de GitHub por medio del comando *push*.

```

new file:   Tarea4_GIT_JoseAntonio_Molins.pdf

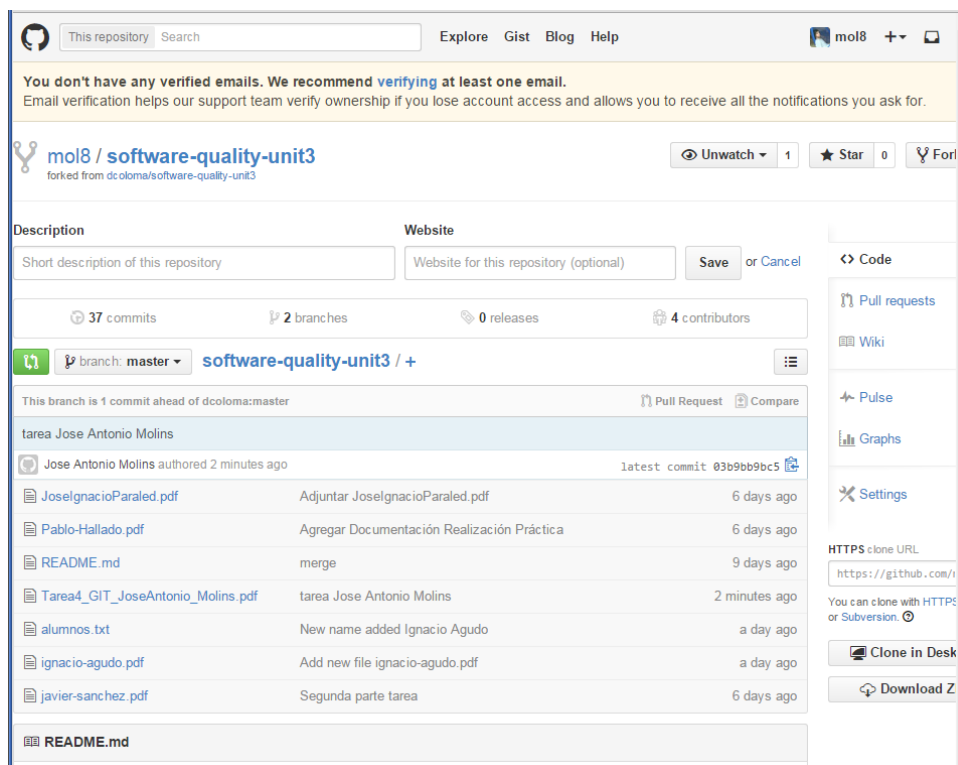
mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git commit -m "tarea Jose Antonio Molins"
[master 03b9bb9] tarea Jose Antonio Molins
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Tarea4_GIT_JoseAntonio_Molins.pdf

mol@MOL-PC ~/git/calidadSoftware/software-quality-unit3 (master)
$ git push -u origin master
Username for 'https://github.com': mol8
Password for 'https://mol8@github.com':
Counting objects: 74, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (62/62), done.
Writing objects: 100% (65/65), 1.71 MiB | 38.00 KiB/s, done.
Total 65 (delta 13), reused 0 (delta 0)
To http://github.com/mol8/software-quality-unit3
 c573457..03b9bb9  master -> master
Branch master set up to track remote branch master from origin.

```

**Ilustración 10 - Commit y Push**

Además al hacer esto hemos subido los modificaciones que nos habíamos traído del repositorio original (<http://github.com/dcoloma/software-quality-unit3>). Ahora mi rama master de mi repositorio en GitHub, contiene los cambios que se han hecho en local.



**Ilustración 11 - Repositorio GitHub**





## 1.5 Pull Request

El último paso es solicitar un pull request al propietario del repositorio original (David Coloma) para que acepte o no en el repositorio original los cambios que hemos hecho.

o

This repository Search Explore Gist Blog Help mol8 +

You don't have any verified emails. We recommend [verifying](#) at least one email. Email verification helps our support team verify ownership if you lose account access and allows you to receive all the notifications you ask for.

dcoloma / software-quality-unit3 Watch 4 Star 0 Fork

### Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base fork: dcoloma/software-quality-... base: master ... head fork: mol8/software-quality-unit3 compare: master

✓ Able to merge. These branches can be automatically merged.

Create pull request Discuss and review the changes in this comparison with others.

1 commit 1 file changed 0 commit comments 1 contributor

Commits on Apr 26, 2015

Jose Antonio Molins tarea Jose Antonio Molins 03b9bb9

Showing 1 changed file with 0 additions and 0 deletions. Unified Split

BIN Tarea4\_GIT\_JoseAntonio\_Molins.pdf View

Binary file not shown

No commit comments for this range