

NOME DELLA GUIDA

Informazioni Documento

Nome documento	Guida a TO DO
Versione	v.1.0.0
Data redazione	2015-01-29
Redattori	Tesser Paolo
Uso	ERRORE

Sommario

TO DO

Diario Revisioni

Modifica	Autore	Data	Versione
Inizio stesura del documento	Tesser Paolo	2015-01-29	v0.0.1

Indice

1	Introduzione	1
1.1	Informativi	1
2	Impostare git	2
3	Creare un repository	3
3.1	Creazione in locale	3
4	Lavorare sul repository	4
4.1	Lavorare in locale	4
4.1.1	Visualizzare le differenze	4
4.1.2	Rimuovere i file dalla staging area	4
4.1.3	Annullare i cambiamenti	4
4.2	Lavorare in remoto	4

Elenco delle figure

1 Introduzione

Git è un sistema di controllo di versione distribuito.

Distribuito (DVCS) significa che ogni componente, del gruppo di lavoro, possiede una copia locale completa del repository.

Vantaggi:

- operazioni di commit più veloci da effettuare;
- si riesce a lavorare offline;
- ognuno ha in locale un backup del repository.

1.1 Informativi

- CodeSchool: www.codeschool.com;
- CERN - Best Practice: PDF;

2 Impostare git

La prima volta che si installa git è buona pratica impostare da subito alcune credenziali.

Da linea di comando eseguire i seguenti comandi:

```
path: git config --global user.name "Nome Cognome"  
path: git config --global user.email mail@mail.com  
path: git config --global color.ui true
```

3 Creare un repository

Un repository può essere creato in locale e usato solo nel computer dell'utente, anche se questo non permetterà un lavoro collaborativo, oppure, dopo la creazione in locale, può essere associato ad un repository creato su un server (o attraverso un servizio di hosting).

Per creare un repository su un servizio di hosting basta seguire le istruzioni che il sito propone.

3.1 Creazione in locale

Di seguito viene illustrata la sequenza di comandi che dovranno essere eseguiti:

```
path: mkdir nome_repo  
path: cd nome_repo  
path: git init
```

L'ultima istruzione creerà tutti i metadati necessari al repository. Verranno salvati nella cartella nascosta:

nome_repo/.git/

4 Lavorare sul repository

4.1 Lavorare in locale

Le tre azioni che vengono più spesso eseguite quando si lavora in locale sono:

- creare un file;
- aggiungerlo alla **staging area** attraverso il comando:

```
path: git add nome_del_file.ext
```

Se si vogliono aggiungere più file contemporaneamente o solo alcuni, possiamo eseguire in esclusione questi comandi:

```
path: git add --all # aggiunge tutti i file presenti nella unstaging area
path: git add <list of file>
path: git add *.ext
path: git add dir/
```

- committare i cambiamenti attraverso il comando:

```
path: git commit -m "msg"
```

Quest'ultima operazione può essere effettuata anche senza l'opzione -m, in tal caso la procedura seguirà lo schema illustrato nella sezione TO DO;

Per osservare i cambiamenti avvenuti dall'ultima operazione di commit basta lanciare il comando:

```
path: git status
```

4.1.1 Visualizzare le differenze

Per osservare le differenze apportate tra un file e la sua precedente versione:

```
path: git diff nome_del_file
# oppure
path: git diff --staged nome_del_file
# per osservare le modifiche su i file già inseriti nella staging area
```

4.1.2 Rimuovere i file dalla staging area

Per rimuovere dei file aggiunti nella staging area, tramite il comando **git add**, bisogna lanciare il seguente comando:

```
path: git reset HEAD # per rimuovere tutti i file
path: git reset HEAD nome_del_file # per rimuovere solo quello specificato
```

4.1.3 Annullare i cambiamenti

4.2 Lavorare in remoto