

# Come funziona un progetto Open Source?

Samuele Valente

March 31, 2022

## 1 Obiettivo

L'obiettivo di questo esercizio è quello di dare una conoscenza di base di come accedere e visualizzare un progetto *open source*, come poter apportare delle modifiche ad esso e di aprire una *issue* in presenza di un problema.

## 2 Introduzione

Come appena descritto l'obiettivo è quello di dare all'utente un'idea su come fare delle operazioni di base all'interno di un progetto *Open Source*. Queste operazioni non sono del tutto banali e presumono che l'utente conosca una serie di concetti di base. Questo esercizio però, è creato in modo tale che anche l'utente meno esperto possa comprendere quali siano tali operazioni e cosa molto più importante, come aprire una *issue*, ovvero come segnalare un errore a coloro che mantengono il progetto. Quest'ultima operazione è molto importante e abbastanza semplice e sarebbe auspicabile che chiunque voglia utilizzare progetti *open source* conosca il modo in cui farla.

## 3 A chi e' rivolto

Questo esercizio è rivolto a tutti coloro che vogliono capire il funzionamento di un progetto *open source*.

## 4 Durata

La durata di questo esercizio è di circa 30 minuti

## 5 Livello arcobaleno

Questo esercizio si riferisce al livello 3 dell'arcobaleno

## 6 Difficoltà

La difficoltà di questo esercizio è medio-alta

## 7 Strumenti necessari

Gli strumenti necessari per svolgere tale esercizio sono:

- Un computer con sistema operativo Windows, Linux o MacOS
- Un browser a scelta (in questo esercizio verrà utilizzato firefox)
- Un account alla piattaforma GitHub <sup>1</sup>.  
Se si è già in possesso di un account si può procedere direttamente dalla seconda sezione del paragrafo "Istruzioni Dettagliate"

## 8 Costi

Questo esercizio non ha nessun costo.

## 9 Istruzioni dettagliate

### 9.1 Creare un account GitHub

- Recarsi al seguente link <https://github.com/signup?source=login>
- Inserire la propria mail dove richiesta
- Inserire la password dove richiesta
- Inserire un *nickname* dove richiesto
- premere il tasto conferma
- Andare nella propria casella di posta elettronica e verificare che sia arrivata una mail da GitHub, come la richiesta di premere il link di verifica della mail.

### 9.2 Creare una nuova versione del progetto e visionare le differenze

1. Aprire una finestra di ricerca sul proprio browser preferito
2. Effettuare l'accesso al sito <https://github.com>
3. Collegarsi al seguente link <https://github.com/samuvale95/EsercizioCDT>. La pagina a cui vi sarete collegati rappresenta un esempio di progetto Open Source sulla piattaforma Github.
4. Effettuare il fork del progetto premendo il tasto fork in alto a destra. Questo permetterà di avere una copia personale del progetto nella quale si potranno fare tutte le modifiche che si riterranno opportune.
5. Per prima cosa si vedrà come accedere alle varie versioni del codice. Per fare ciò bisogna premere il tasto *commits* in alto a destra. Si aprirà una schermata che mostrerà tutta l'evoluzione del progetto con tutte le modifiche effettuate.

---

<sup>1</sup><https://github.com/>

6. Premere il link 'Aggiunta modifica alla bozza con correzioni' per vedere le differenze tra il file *Lettera.md* tra la versione attuale e quella precedente.
7. Si aprirà una schermata che mostra appunto le differenze, in rosso le modifiche della versione precedente e in verde quelle della versione attuale.
8. Premere il tasto Browse files in alto a destra per vedere i file contenuti nella versione attualmente visionata. Come si può notare rispetto all'ultima versione manda un file *tabellina\_del\_9.py*
9. Tornare all'ultima versione del progetto, premendo sempre sul tasto *commits* e poi in alto a sinistra premere il *dropdown button* che ha come nome *d354c08797* e una volta premuto selezionare la voce *main*.
10. Premere in alto a sinistra la voce *code*, per ritornare al codice del progetto all'ultima versione.
11. Premete sul file *Lettera.md*
12. Premete in alto a destra sull'icona per editare il file.
13. Il file come si può notare leggendolo e confrontandolo con la versione precedente, presenta molti errori di battitura e grammaticali. Ciò che si deve fare ora è correggerli e fare un nuovo *commit* (ovvero aggiungere una nuova versione al progetto.)
14. Una volta effettuate le modifiche al file ritenute necessarie, scorrere la pagina verso il basso fino a vedere la voce *Commit changes*.
15. Se si vuole, modificare il messaggio che c'è scritto, altrimenti premere il tasto *Commit changes*. Questo creerà una nuova versione del progetto con il file *Lettera.md* contenente le modifiche appena effettuate.

### 9.3 Creare una *issue* al progetto

1. Collegarsi al link <https://github.com/samuvale95/EsercizioCDT>
2. Nella colonna di destra sotto la voce *Release* selezionare quella relativa al proprio sistema operativo e scaricarla.  
In alternativa per chi ha installato Python sul proprio PC può eseguire direttamente il programma *tabellina\_del\_9.py* con il seguente comando  
*python tabellina\_del\_9.py*
3. Eseguire il programma il quale scopo è stampare la tabellina del 9 e controllare se sono presenti eventuali errori (guardare bene perchè gli errori sono presenti :D )
4. Trovati gli errori, tornare sul progetto al link <https://github.com/samuvale95/EsercizioCDT>, premere il tasto *issue* sulla destra rispetto al tasto *code*. Si aprirà una pagina contenente tutti i problemi aperti e lo storico di quelli risolti. In un vero progetto *open source*, non sarebbe permessa la creazione di due *issue* rivolte allo stesso problema. In questo caso ai fini dell'esercizio saranno presenti molte *issue* simili, ma non lo considereremo un problema.

5. Premere *New issue* in alto a destra, per creare e notificare al proprietario del progetto che è stato trovato un nuovo problema.
6. Compilare il *form* spiegando con più dettagli possibili il problema riscontrato, anche inserendo degli *screen shot* se fosse necessario. La descrizione di un problema dovrebbe essere il più dettagliata possibile, inserendo il maggior numero di informazioni per poter far capire ai mantenitori del progetto come poter fare per replicare il problema e risolverlo di conseguenza.
7. Una volta inviata la *issue*, verrà notificata al gestore della repository e una volta risolta verrà chiusa.

## 10 suggerimenti per variazioni

Una variazione di questo esercizio potrebbe essere quella di cercare un progetto open source su un'altra piattaforma come GitLab <sup>2</sup> provare a effettuare gli stessi passaggi fatti con questo esercizio.

## 11 Risultati attesi

Al termine dell'esercizio si sarà in grado di aprire un progetto open source e navigare tra i suoi file. Si sarà in grado di visionare versioni differenti del progetto e in fine si sarà in grado di aprire una *issue* al proprietario della repository.

## 12 Licenza

CC-BY

---

<sup>2</sup><https://about.gitlab.com/>