

Git

# GitHub

---

GitHub è un servizio web e cloud-based che aiuta gli sviluppatori ad archiviare e gestire il loro codice e a tracciare e controllare le modifiche.

Per capirne il funzionamento devono essere compresi due elementi:

- Controllo versioni
- Git

# Controllo versioni

Il controllo delle versioni aiuta gli sviluppatori a tracciare e gestire le modifiche al codice di un progetto software. Man mano che un progetto software cresce, questo processo diventa essenziale.

Il controllo delle versioni permette agli sviluppatori di lavorare in sicurezza attraverso il **branching** (ramificazione) e il **merging** (fusione).

Con il branching, uno sviluppatore duplica parte del codice sorgente (chiamato **repository**). Lo sviluppatore può quindi apportare in modo sicuro modifiche a quella parte del codice senza influenzare il resto del progetto.

Poi, una volta che lo sviluppatore fa funzionare correttamente la sua parte di codice, può fondere quel codice nel codice sorgente principale e renderlo ufficiale (merging).

# Git

Git è uno specifico sistema di controllo versioni open-source creato da Linus Torvalds nel 2005.

In particolare, Git è un sistema di controllo versioni distribuito, il che significa che l'intero codice base e la cronologia sono disponibili sul computer di ogni sviluppatore, il che permette di creare facilmente ramificazioni e fusioni.

Esistono diversi sistemi Git, GitHub è privato ma ne esistono anche di liberi come GitLab.

# GitHub

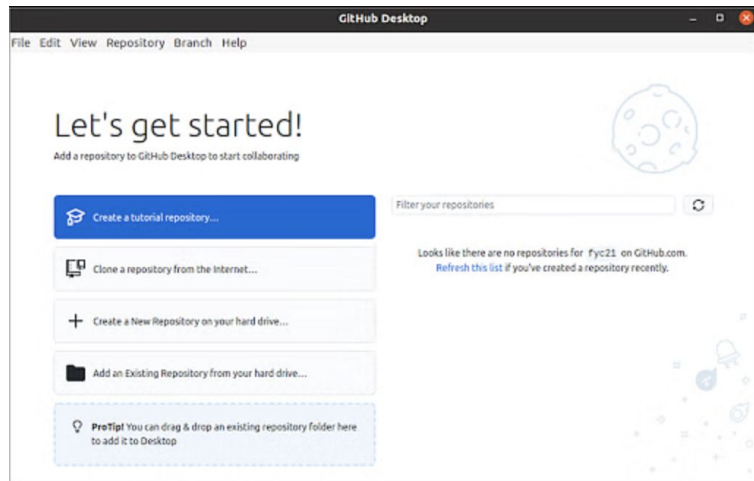
GitHub è un'azienda a scopo di lucro che offre un servizio di hosting di repository Git basato su cloud. In sostanza, rende molto più facile per individui e team utilizzare Git per il controllo delle versioni e la collaborazione.

L'interfaccia di GitHub è abbastanza facile da usare, quindi anche i programmatori alle prime armi possano sfruttare le funzionalità di Git. Senza GitHub, l'utilizzo di Git richiede generalmente una maggiore esperienza tecnica e l'utilizzo della linea di comando.

GitHub è così facile da usare che viene utilizzato anche per gestire altri tipi di progetti, come la scrittura di libri.

# GitHub Desktop

# GitHub Desktop



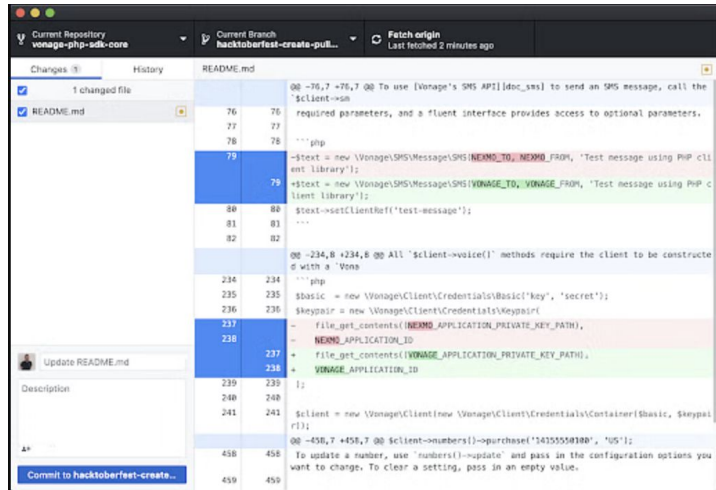
Puoi usarlo per creare quattro diversi tipi di repository:

- Repository di tutorial: questa opzione crea un repository di tutorial nel computer locale, che può quindi essere pubblicato nell'account GitHub.
- Clona un repository da Internet: questa opzione clona un repository dall'account GitHub o dall'URL.
- Nuovo repository sul disco rigido del tuo dispositivo locale: questa opzione crea un nuovo repository locale.
- **Aggiungi un repository esistente dal tuo disco rigido:** questa opzione aggiunge qualsiasi repository creato in precedenza dal disco locale.

Puoi anche utilizzare la barra dei menu nella parte superiore della finestra dell'applicazione per eseguire diverse attività relative al repository.

# Pull Request

Le richieste **pull**, note anche come richieste di unione (merge) si verificano quando uno sviluppatore o un collaboratore è pronto per avviare il processo di unione delle modifiche al codice con il repository del progetto principale.



- Sul lato sinistro dell'immagine vediamo i file che sono stati modificati, aggiunti o rimossi. Selezioniamo i file che desideriamo spostare nel repository.
- Nell'angolo in basso a sinistra dello schermo. Sono presenti due caselle di testo e un comando Commit che richiedono il nome del ramo.
- Dopo aver compilato i due campi, fai clic sul pulsante "Commit".
- Una volta inviate le modifiche, otterrai un'altra casella di testo che ti chiederà se desideri creare una richiesta pull. Se sei pronto, fai clic sul pulsante "Create pull request".
- Dopo aver fatto clic sul pulsante, verrà visualizzata una schermata di creazione della richiesta pull.