

GIT, Desktop, Visual Studio Code und Github

Moderne Software-Entwicklung erfordert zwingend den Einsatz eines Source-Code-Management-Systems, so wie GIT. Das ist für eine Team-Arbeit sehr wichtig.

<https://github.com/>

1. Github.com Konto anlegen inkl. repository
2. GIT installieren auf dem lokalen PC
3. Visual Studio Code – Projekt mit GIT verbinden
 - a. Kontrolle und Festlegen von 2 wichtigen Schritten
 - b. VS-Code und ein vorhandenes Projekt mit github.com verbinden
 - i. Übung: Projekt hochladen (VS-Code mit github.com verbinden) – repository automatisiert anlegen lassen
 - ii. Wenn man das Repository eigenhändig erstellen möchte:
4. desktop.github.com installieren
5. neues Projekt in VS-Code mit github.desktop verbinden
6. desktop.github mit dem GITHUB Konto von github.com verbinden

Theorie:

Git wurde vom Linux-Erfinder Linus Torvalds programmiert. Mittlerweile haben andere die Weiterentwicklung übernommen. Inzwischen hat Microsoft GitHub gekauft. GitHub ist eine kostenlos nutzbare Online-Plattform, die Git nutzt, um das Hosting von Software und deren Quellcode zu vereinfachen. Git spricht keine Sprache, sondern speichert nur Dateien.

Vorteile, wenn man Code in GitHub teilt: Manchmal bekommt man Feedback und Tipps und man hat den Code immer griffbereit. Auch Arbeitgeber sehen sich den Account an und freuen sich, wenn er gut aussieht.

GIT:

- ist eine Software für die Versionsverwaltung, d.h. mehrere Entwickler arbeiten an einem Projekt.
- Wird in allen IT-Industrien genutzt
- Dadurch kann man Softwarekonflikte vermeiden.
- Bei jeder Änderung am Projekt erstellt Git eine neue Version und man kann zu einer alten Version zurückspringen. Man muss keine Sorge haben, durch eine Änderung alles zu zerstören.
- Man fügt automatisch alle verschiedenen Programmteile zu einer Einheit zusammen
- Visual Studio enthält Git

Wie funktioniert Git:

- Man muss ein Repository (Projekt) anlegen
- Jeder Entwickler kann das Repository herunterladen und Veränderungen vornehmen
- Veränderungen kann man mit einem „Pull-Request“ anzeigen
- Der Master kann sich diese ansehen und aufnehmen, wenn er möchte

Was sind Branches?

- Man kann verschiedene Versionszweige laufen lassen – so zum Testen

- Neben der „masterbranch“ kann man neue Features probieren
- Ist ein Branch gut genug, kann man diesen mit einem Pull-Request in die Masterbranch aufnehmen

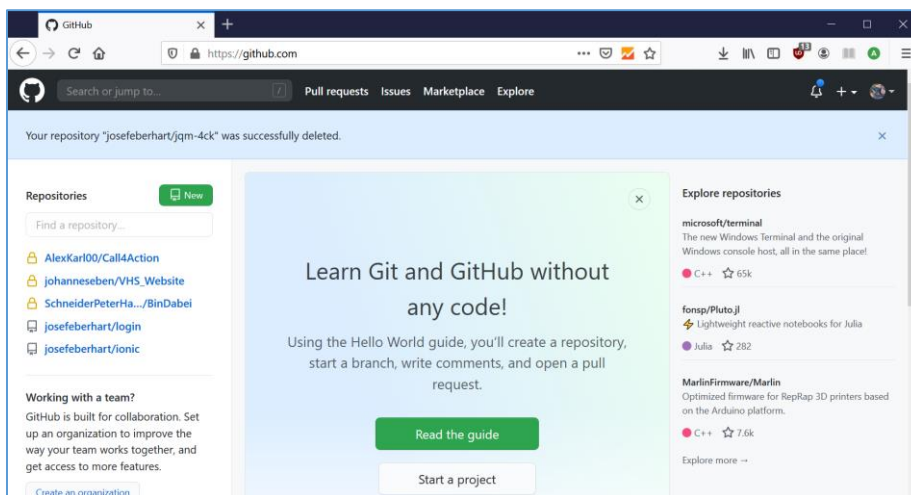
GitHub

- Ist eine Website, um Projekte zu hosten
- Eine benutzerfreundliche grafische Oberfläche
- Man kann dort in anderen Projekten stöbern und mitarbeiten#
- Große Community
- Sichere Speicherung des eigenen Projekts
- Man kann auch ToDo-Listen anlegen

1)Github.com Konto erstellen und repository anlegen

Erstelle ein Konto bei www.github.com

https://www.youtube.com/watch?v=3fUbBnN_H2c ab ca. 45:00





Neues Repository erstellen. Klicke auf „New“.

Repositories sind Container für spätere Projekte. Vergib einen Namen und einen Haken unten bei „Initialize with a README“. Dann auf „Create repository“

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)


Owner * **Repository name ***

 josefeberhart / testFuerGit 

Great repository names are short and memorable. Need inspiration? How about [stunning-journey?](#)

Description (optional)

 **Public**
Anyone on the internet can see this repository. You choose who can commit.

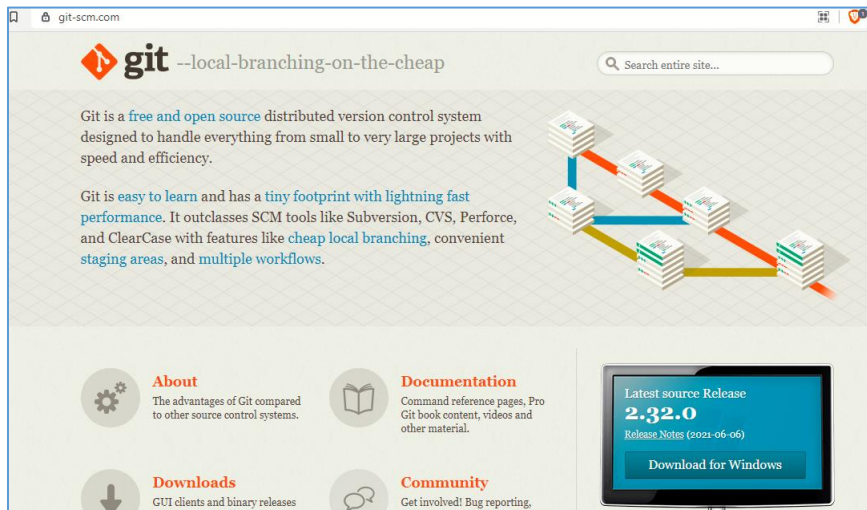
 **Private**
You choose who can see and commit to this repository.

2)GIT – lokal installieren

Damit man später Visual Studio Code direkt mit GITHUB verbinden kann, muss GIT installiert sein.

Daher downloade GIT von hier:

<https://git-scm.com/>



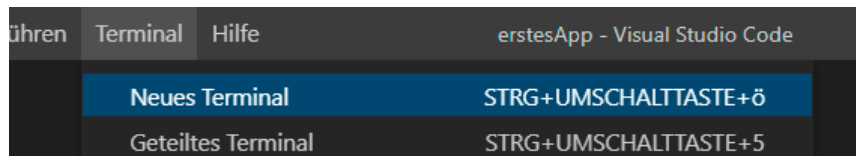
Nachdem GIT installiert ist, starte VS-Code neu.

3.) Visual Studio Code – Projekt mit GIT verbinden

3a) Kontrolle und Festlegen von 2 wichtigen Schritten:

Das muss man nur beim ersten Mal durchführen:

Öffne ein internes Terminal:



GIT Version abfragen – damit überprüft man nochmal, ob die Installation korrekt war

- git --version

```
PS D:\testFuerGit> git --version
git version 2.32.0.windows.2
PS D:\testFuerGit> |
```

Benutzername und E-Mail eingeben.

Diese sollen unbedingt gleich sein mit denen, die man in Github-Desktop und bei der Github-Anmeldung beim Hosten angibt. Dadurch ist gewährleistet, dass Änderungen der richtigen Person zugeordnet werden.

- git config --global user.name "josefeberhart"

```
PS D:\testFuerGit> git config --global user.name "josefeberhart" |
```

- git config --global user.email josefeberhart@geheimeskonto.at
- Zur Überprüfung kann man folgendes eingeben und erhält die Werte:

```
<command> [<args>]
PS D:\ionic_start\erstesApp> git config --global --list
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
user.name=josefeberhart
```

Quelle: <https://www.youtube.com/watch?v=F2DBSH2VoHQ>

Eine Verbindung zu GITHUB ist aber somit noch nicht vorhanden.

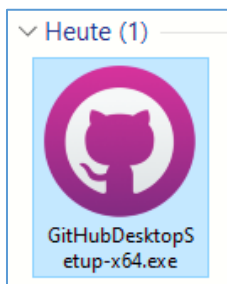
4)desktop.github installieren – Bindeglied zwischen lokalem Projekt und github.com

Dann lade den Desktop-Client herunter. Das ist die Schnittstelle zum Upload auf das Online github-Repository.

Dazu folgende URL: <https://desktop.github.com/>



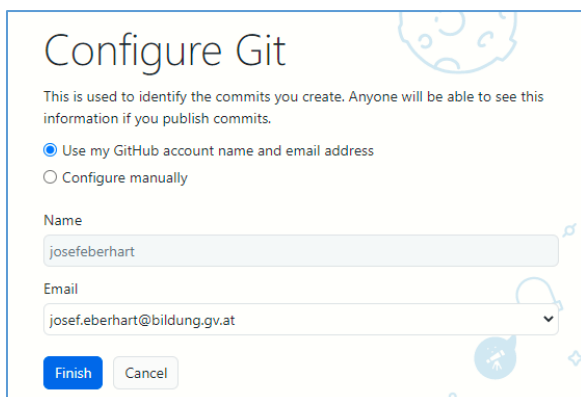
Installiere den Download



Hier spätestens muss man ein Konto anlegen bzw. ein bestehendes Konto eingeben.

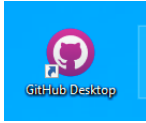
Wichtig:

Wenn der Name und die E-Mail angezeigt werden, dann sollten diese mit den Daten übereinstimmen, die man oben bei GIT und dem Github.com angegeben hat. Das ist wichtig, da man spätere Änderungen an Code auch identifizieren können muss.

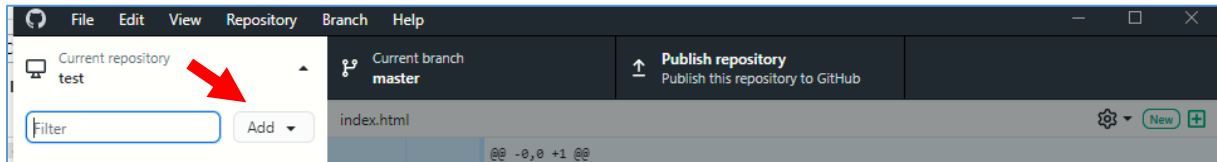


5) neues Projekt in VS-Code mit github.desktop verbinden

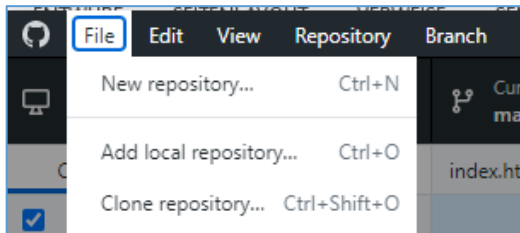
Nun öffne github.desktop mit dem Desktop-Icon.



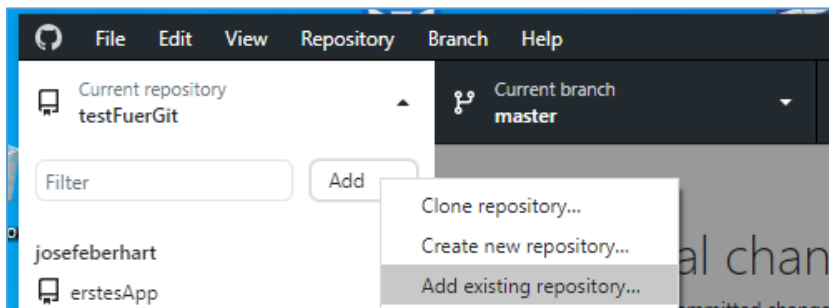
- Klicke auf „Add“ um einen Ordner vom lokalen Laptop zu suchen, der später auf github.com geschoben werden soll.



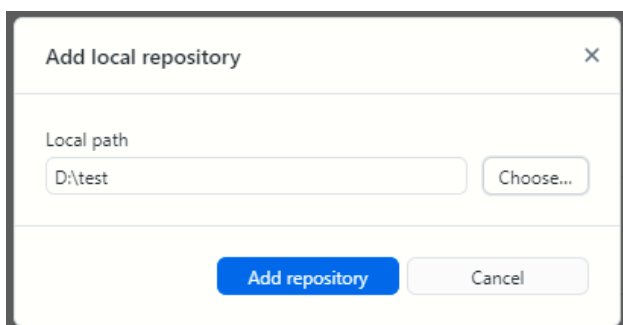
- oder auf „File“ und dann „Add local repository...“



Dann auf „Add existing repository...“



Suche den Pfad und wähle daher den Ordner aus:



Nun ist dieses eben angelegte Projekt integriert und bereit auf github.com geschoben zu werden.

6)desktop.github mit dem GITHUB Konto von github.com verbinden

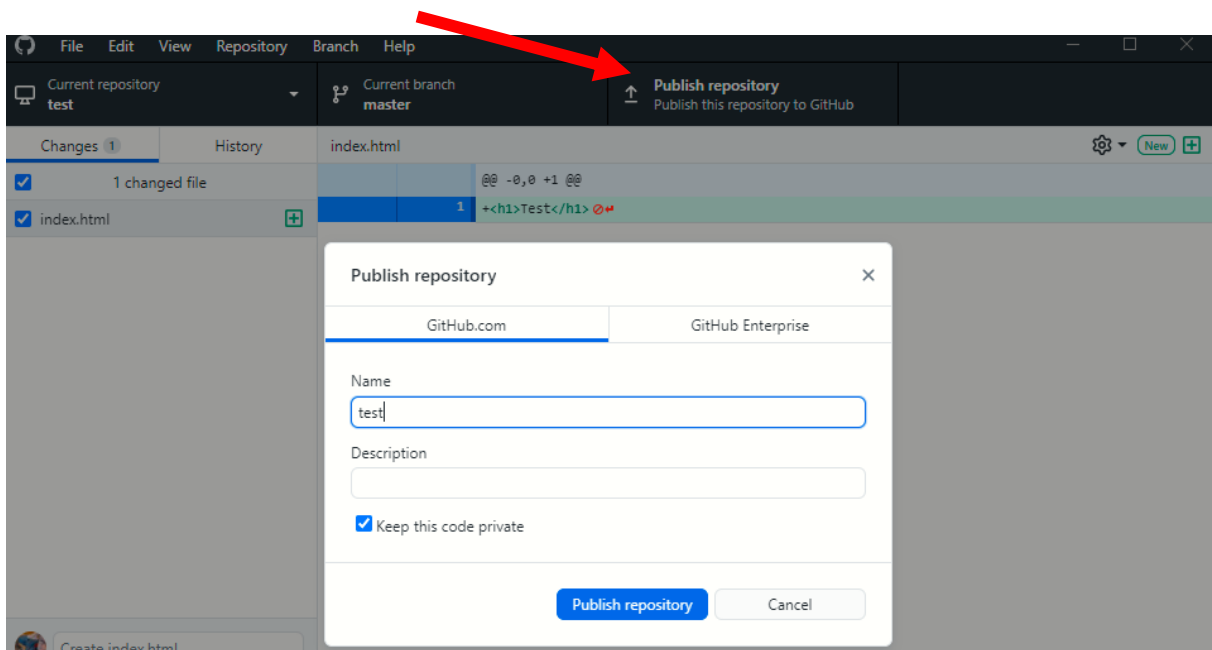
Da man bereits beides zusammenpassend (Name und E-Mail) eingerichtet hat, also

- github.com
- Desktop Github

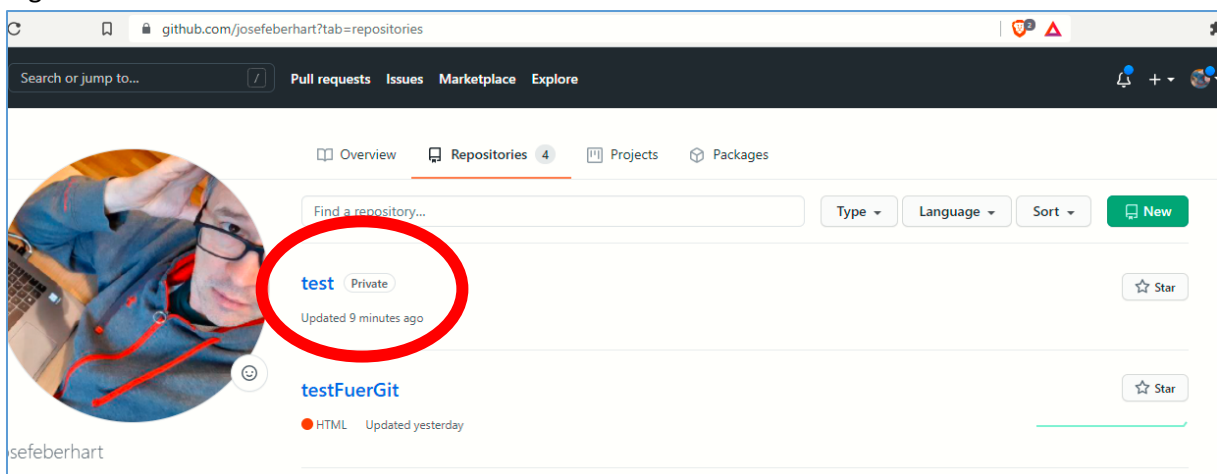
kann man das eben erstellte Projekt vom desktop.github auf github.com übertragen.

Nun kann es auf Github publiziert werden. Dafür klicke rechts oben im Menü auf

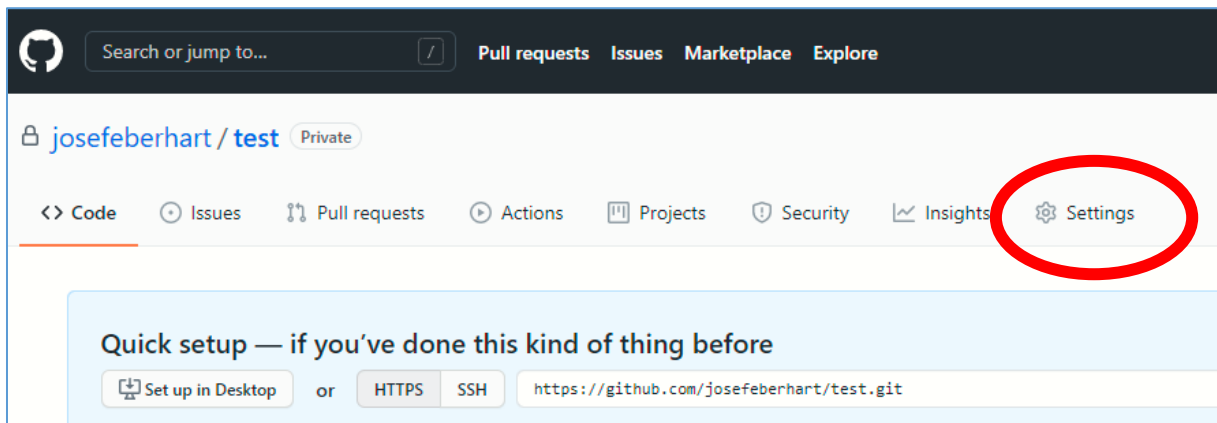
- Publish repository
- Gibt dann den Namen ein, den er dann haben soll.



Ergebnis:

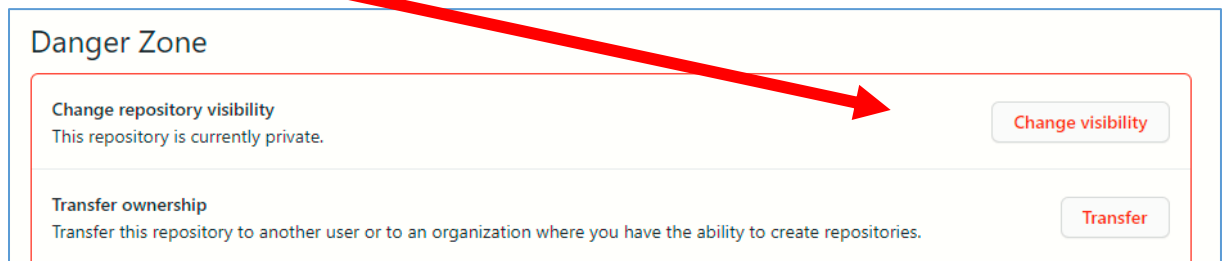


Wenn „Privat“ eingestellt ist, soll dies auf „public“ abgeändert werden.
Klicke dafür auf den Namen, hier z.B. „test“ und dann auf „settings“:

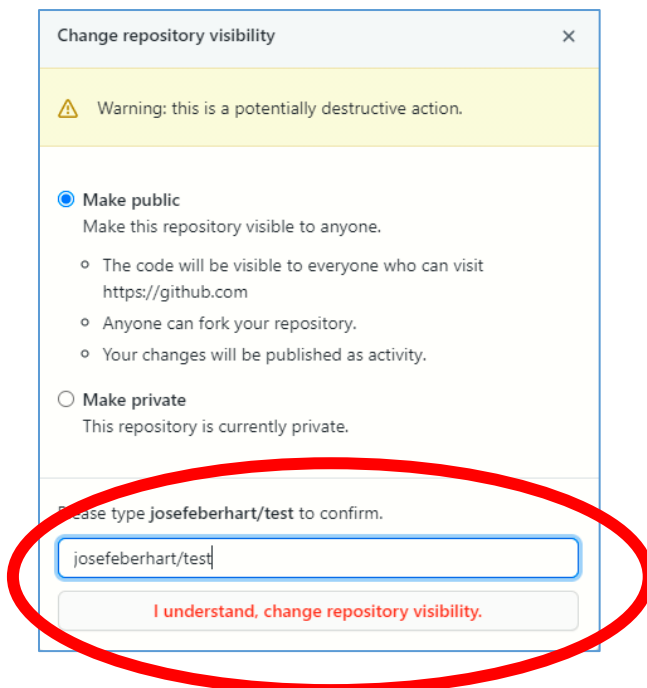


Dann ganz unten in rotem Rahmen:

- Change visibility



In dieser Zone (roter Rahmen) muss man immer speziell die Änderungen bestätigen. Dafür ganz unten den Namen auf github usw. eingeben.



Weitere gute Info:

Ohne Desktop.git.com: https://www.youtube.com/watch?v=lvF_9pf5p3Q