

Warum GitHub benutzen?

"**GitHub** ist ein webbasierter Hosting-Dienst für Software-Entwicklungsprojekte." (Quelle: <http://de.wikipedia.org/wiki/Github> 22.11.2013 11:19 Uhr)

Vorteile:

- + einfache Benutzung von Git durch, das Programm GitHub for Windows
- + Speicherplatz für Quellcode und Organisation
- + einfache Verwaltung von Aufgaben, Bugs, e.t.c. (Issues)
- + Fragen zu bestimmten Commits direkt an den Quellcode anpassen.

Nachteil:

- Mit kostenfreie Accounts kann man nur öffentliche Repositorys erstellen und verwalten.

eine Alternative:

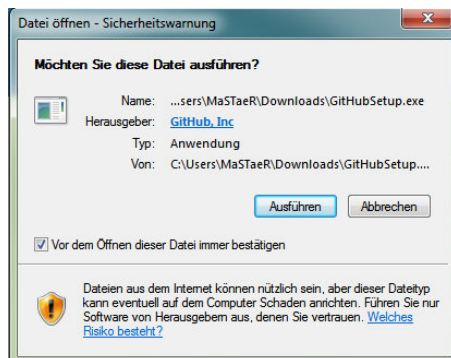
<http://gitlab.org/>

github
SOCIAL CODING

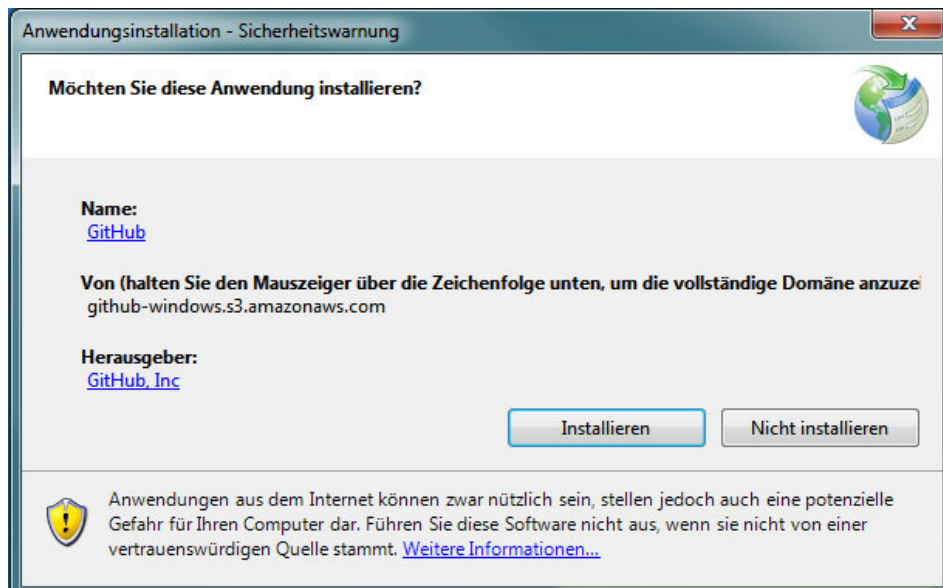


Installation von GitHub for Windows:

1. Download von GitHub for Windows unter: <http://windows.github.com/>
2. Die GitHubSetup.exe ausführen und den Anweisungen folgen:
 - Klicken Sie auf ausführen.



- Klicken Sie auf installieren.



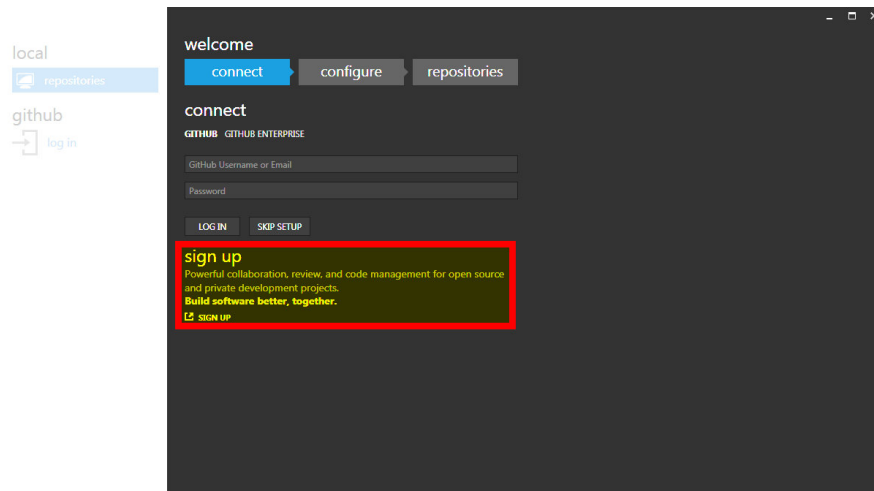
- Danach wird GitHub installiert, dies kann wenige Minuten dauern.



- Es ist nun installiert und öffnet eine Instanz von sich.

Account mit GitHub for Windows erstellen:

1. Wir starten GitHub (Nach der Installation wird es automatisch gestartet)
 - Wir klicken auf **SIGN UP** (Rot markiert im unteren Bild).



2. Wir werden weitergeleitet zu der Webseite: <https://github.com/pricing>
 - Klicken auf **Sign up now** (Rot markiert im unteren Bild).

GitHub

Search or type a command

Explore Features Enterprise Blog

Sign up

Plans and pricing

GitHub is free to use for public projects. Collaborate on private repositories with any of our paid plans.

Sign up now



Personal plans


For individuals looking to share their own projects and collaborate with others.


3. Wir folgern den Registrierungsschritten.


- Persondaten eingeben.

Join GitHub

The best way to design, build, and ship software.

 **Step 1:**
Set up a personal account

 **Step 2:**
Choose your plan

 **Step 3:**
Go to your dashboard

Create your personal account

Username

MaxMusterfrau ✓

This will be your username — you can enter your organization's username next

Email Address

mr.musterfrau.max@web.de ✓

We promise we won't share your email with anyone.

Password

..... ✓

Use at least one lowercase letter, one numeral, and seven characters.

Confirm your password

.....

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

Create an account

You'll love GitHub

Unlimited collaborators


Unlimited public repositories


✓ Great communication

✓ Friction-less development

✓ Open source community

- auswählen der kostenlosen Version

 **Completed**
Set up a personal account

 **Step 2:**
Choose your plan

Choose your personal plan

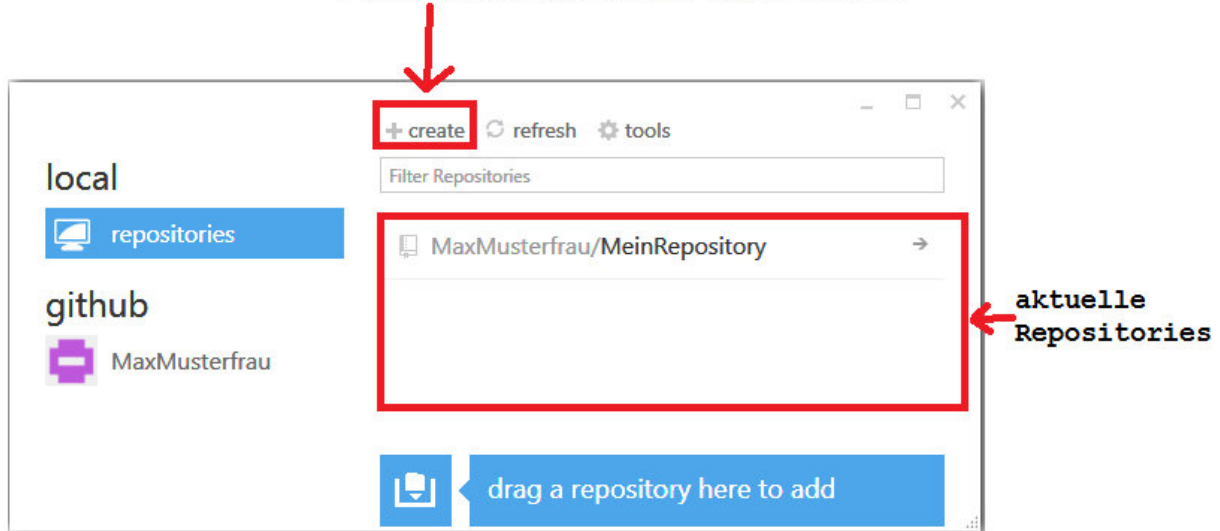
Plan	Cost	Private repos	
Large	\$50/month	50	<div>Choose</div>
Medium	\$22/month	20	<div>Choose</div>
Small	\$12/month	10	<div>Choose</div>
Micro	\$7/month	5	<div>Choose</div>
Free	\$0/month	0	<div>Chosen</div>

- Schon haben wir unseren Account eingerichtet und werden weitergeleitet zu unserem Dashboard.

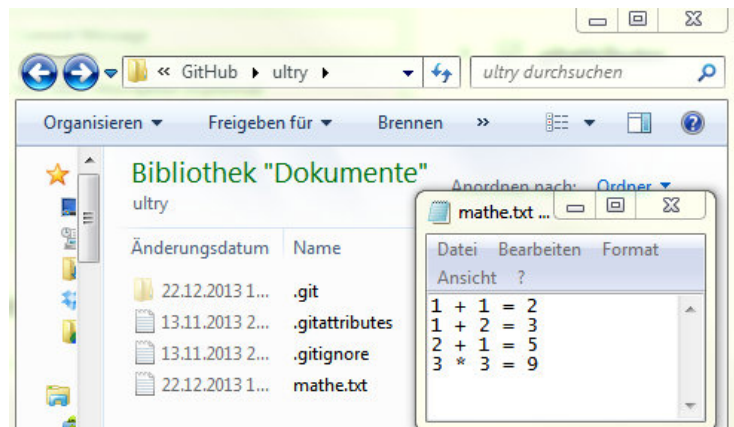
1 x 1 der Benutzung anhand eines Beispiels:

Erstellen eines neuen Repositorys:

Erstellen eines neuen Repositories

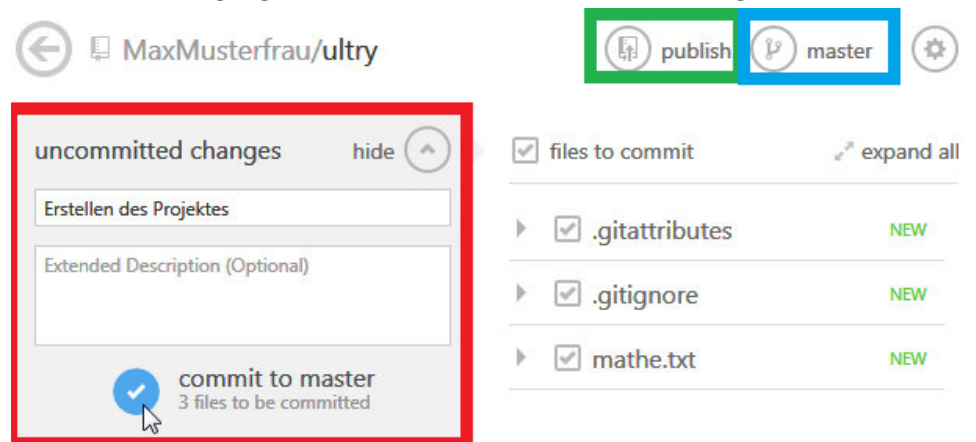


Nach dem Erstellen eines Repository wird ein Ordner in `Dokumente/GitHub/{Name_des_erstellten_Repositorys}` erstellt. Ich habe ein Repository Namens `ultry` erstellt. Erstelle nun eine `txt` in diesen Ordner.

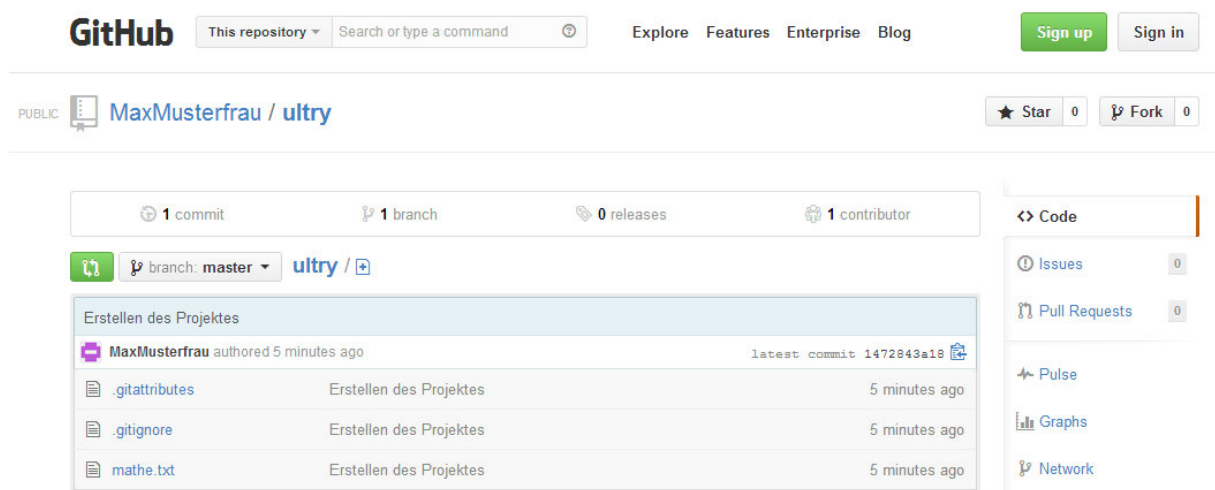


Um dieses Repository auch für andere zur Verfügung zu stellen, müssen wir die Veränderungen zum Server senden. Dafür Committen wir die Veränderungen Local (rot) und veröffentlichen ihn (grün).

Das blaue Kästchen zeigt an, in welchen Repository wir uns gerade befinden.



Nun können wir das Projekt unter `github.com/{Benutzername}/{Repository_name}/` erreichen. In diesen Fall unter `https://github.com/MaxMusterfrau/ultry/`.

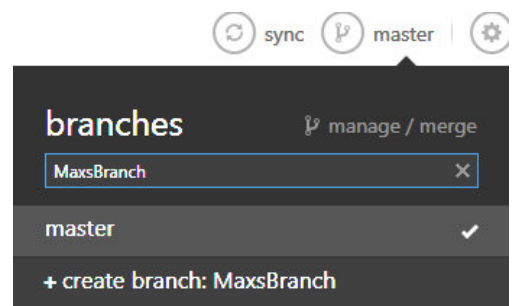
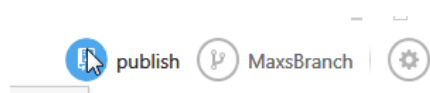


Hier können wir auf das Projekt und die Branches online zugreifen. Beachte, es ist keine Anmeldung auf GitHub dafür von nöten. Also jeder kann den Quellcode und die Issues einsehen.

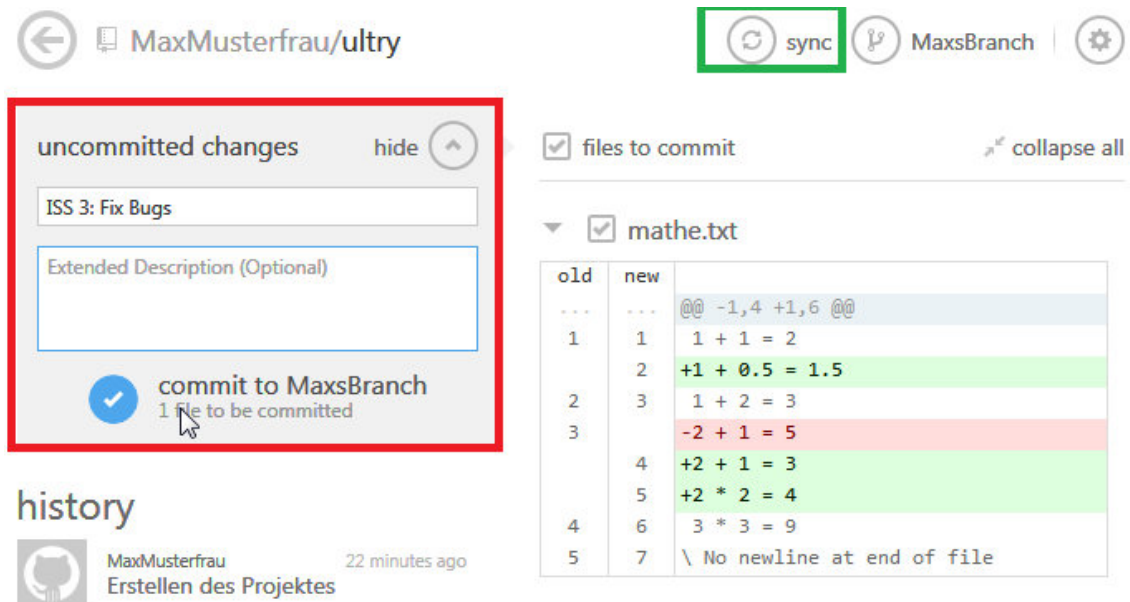
Unter den Issues versteht man das Verwalten von Aufgaben die noch erledigt werden müssen. Es ist auch möglich diese als Bug oder ähnliches zu bezeichnen.

Jeder sollte in sein eigenen Branch arbeiten und erst in den master pushen, wenn es zu 100 % läuft.

Man erstellt einfach ein neuen Branch in den man im Programm GitHub for Windows auf master klickt und ein neuen Branch einrichtet. Damit dieser wieder auf dem Server gelangt und für andere Verfügbar ist, müssen wir ihn wieder veröffentlichen.



Nachdem wir nun in diesen Branch gearbeitet haben und sichergestellt haben, dass alles richtig ist, committen (rot) wir diesen Branch und Synchronisieren (grün) ihn danach.



Schauen wir uns die History an können wir die Veränderungen in diesen Commit sehen. Klicken wir auf github (rot), dann können wir ein Kommentar zu einer bestimmten Zeile hinzufügen und so Nachfragen, was es mit einer bestimmten Zeile auf sich hat.

Wollen wir nun unsere Veränderungen nun in den master-Branch rein schieben (mergen). Klicken wir auf "manage / merge" (siehe Bild). Nun können wir durch ziehen der Branches auswählen welcher in welchen gemergt werden soll. In diesen Fall wollen wir von MaxBranch in den master-Branch hinein mergen.

