

My *git*-Guide

Helena Nawrath
h.nawrath@yahoo.de

Carl Arne Thomann
arnethomann@me.com

3. Oktober 2014

TU Dortmund – Fakultät Physik

Inhaltsverzeichnis

| | | |
|----------|---|----------|
| 1 | Einleitung | 3 |
| 2 | Grundlagen | 4 |
| 2.1 | Verzeichnis | 4 |
| 2.2 | Staging | 4 |
| 2.3 | Repositorys | 4 |
| 3 | Erstellen von entfernten Repositorys | 5 |

1 Einleitung

Diese Version befindet sich im Aufbau.

Die Produktionsdateien befinden sich im nicht veröffentlichen Branch "BuildingBranch"
unter Konstruktion durch Carl Arne Thomann ☺

2 Grundlagen

Verzeichnis Ordner, dessen Inhalt bearbeitet wird

Index Zwischenstufe, in welchem *git* die Änderung festhält

Lokales Repository Kopie des Systemordners von *git* auf dem Rechner

Lokales Repository Systemordner von *git* auf dem Server

2.1 Verzeichnis

Der Ordner, dessen Inhalt bearbeitet wird, heißt Arbeitsverzeichnis. Sein Inhalt sind die Dateien eines Projektes, *git* kann nur darauf zugreifen und bearbeiten, wenn dies durch einen ausdrücklichen Befehl angeordnet wird (*checkout*, *reset*, *rm*). Änderungen dieses Ordners wird von *git* erkannt, sofern sich diese Dateien im **Index** befinden.

2.2 Staging

Der **Index**, auch **Staging**, ist eine Zwischenstufe, in welche bearbeitete Dateien hinzugefügt werden, dies geschieht über den Befehl *git add {Datei}*.

2.3 Repositories

Das **lokale Repository** ist ein versteckter Systemordner, in welchen *git* die Änderungen festhält, analog befindet sich ein **entferntes Repository** auf dem Server.

3 Erstellen von entfernten Repositorys

1. Erstelle einen Ordner mit dem Namen Deines (zukünftigen) Repositorys.
2. Öffne den Ordner in der Kommandozeile
3. Melde bei GitHub das Repos an
4. Verwende in der Kommandozeile die Befehle, die auf GitHub angezeigt werden:
 - a) `touch README.md`
 - b) `git init`
 - c) `git add README.md`
 - d) `git add {andere Dateien}`
 - e) `git commit -m "first commit"`
 - f) `git remote add origin {Adresse von GitHub.com}`
 - g) `git push -u origin master`

Dabei stehen die Befehle für folgende Aktivitäten:

- a** Erstellt eine leere Datei, die als erste Datei ins entfernte Repository geladen wird.
 - b** Initialisiert das *git*-System
 - c** Fügt die Dateien zu dem Staging hinzu, ab jetzt kann git darauf Einfluss nehmen.
Vor dem "Adden" war das nicht möglich
 - d** Veranlasst das Übernehmen der Änderungen zu einem Datencommit, welcher bereit ist, auf den Server gelegt zu werden.
 - e** Hiermit wird das Ur-Repository vom Rechner auf den Server geladen.
Damit ist git in gewohnter Weise betriebsbereit.
 - f** Das ist der erste Push-Befehl. Alle folgenden sind `git push origin master` oder `git push`
-