

# **My *git*-Guide**

Helena Nawrath  
h.nawrath@yahoo.de

Carl Arne Thomann  
arnethomann@me.com

3. Oktober 2014

TU Dortmund – Fakultät Physik

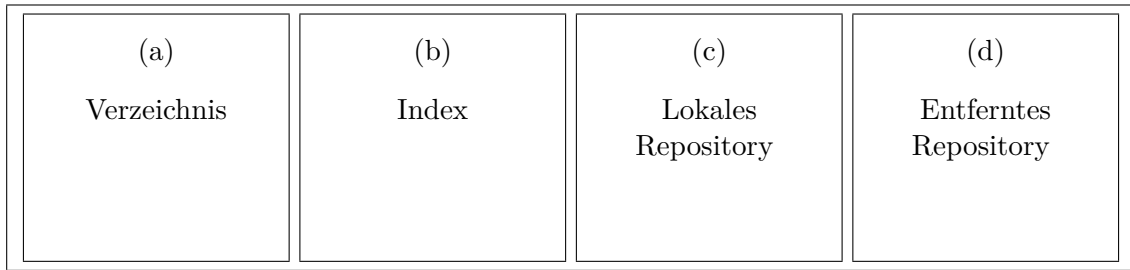
## **Inhaltsverzeichnis**

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Einleitung</b>                           | <b>3</b> |
| <b>2</b> | <b>Grundlagen</b>                           | <b>4</b> |
| <b>3</b> | <b>Erstellen von entfernten Repositorys</b> | <b>5</b> |

# 1 Einleitung

An dieser Stelle möchte noch eine Anleitung stehen ☺

## 2 Grundlagen



**Abbildung 1:** Die vier Instanzen

Das

### 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`
-