Tutorial: Versionverwaltung mit GIT Teil 1

1. Git auf dem Raspi installieren:

$ sudo apt-get install git

1. Git einrichten:

$ git config --global user.name “Andy Meissner”

$ git config --global user.email andy.meissner@mitutoyo-ctl.de

1. Neues Verzeichnis für ein Projekt erstellen und dorthin wechseln:

$ mkdir Projekt\_Gestensteuerung

$ cd Projekt\_Gestensteuerung

1. Leeres Repository initialisieren

$ git init

$ git clone <https://github.com/BogyMitutoyoCTL/OpenCV-gesture-recognition.git> D:\Test

1. Git status abfragen 🡪 Es wird nichts gefunden

$ git status

1. Neue Datei hinzufügen und anzeigen lassen

$ >test

$ ls

1. Git status abfragen 🡪 Eine unbeobachtete Datei wird gefunden
2. Datei stagen

$ git add test

1. Git status abfragen 🡪 Eine staged Datei wird gefunden
2. Änderungen commiten

Git commit –m „Erster Commit“

1. Git status abfragen 🡪 Keine Änderungen gefunden
2. Datei ändern

$ nano test 🡪 Text eingeben

Strg+O Datei speichern/ Strg+X Datei verlassen

1. Änderungen genau ansehen

$ git diff / $ git diff –staged

1. Dateien ignorieren, Beispiele

$ cat .gitignore # ein Kommentar - dieser wird ignoriert

# ignoriert alle Dateien, die mit .a enden

\*.a

# nicht aber lib.a Dateien (obwohl obige Zeile \*.a ignoriert)

!lib.a

# ignoriert eine TODO Datei nur im Wurzelverzeichnis, nicht aber

/TODO

# ignoriert alle Dateien im build/ Verzeichnis

build/

# ignoriert doc/notes.txt, aber nicht doc/server/arch.txt

doc/\*.txt

# ignoriert alle .txt Dateien unterhalb des doc/ Verzeichnis

doc/\*\*/\*.txt

1. Weitere Bearbeitungsmöglichkeiten

15.1 Alle Dateien aufeinmal stagen beim Commit

$ git commit –a –m “add new feature”

* 1. Datei löschen

$ rm test

$ git status

$ git rm test

$ git status

$ git rm --cached readme.txt

1. Commit Historie

$ git log

$ git log –p -2

1. Änderungen rückgängig machen

17.1 Letzten Commit anpassen

$ git commit -m 'initial commit'

$ git add forgotten\_file

$ git commit –amend

17.2 Alle lokalen Änderungen zurücksetzten

$ git fetch origin  
$ git reset --hard origin/master

17.4 Lokale Änderung einer Datei auf den Stand im HEAD zurücksetzten

$ git checkout -- <filename>

17.5 Branch auf einen anderen commit zurücksetzten

$ git reset -–soft <commit>