

---

A04

# Continuous Integration mit Jenkins

---

Softwareentwicklung  
5BHIT 2017/18, GruppeA

Nicolaus Rotter

Note:  
Betreuer: Dolezal

Version 0.2  
Begonnen am 13. Oktober 2017  
Beendet am 13. Oktober 2017

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Aufgabenstellung . . . . .	1
<b>2</b>	<b>Grundanforderungen</b>	<b>2</b>
2.1	Installation Jenkins . . . . .	2
2.2	Installation Violations, Corbertura . . . . .	2
2.3	Installation Nose, Coverage, Pylint . . . . .	2

# 1 Einführung

Lass das Bruch-Projekt mithilfe von Jenkins automatisch bei jedem Build testen!

## 1.1 Aufgabenstellung

Grundanforderungen:

- Installiere auf deinem Rechner bzw. einer virtuellen Instanz das Continuous Integration System Jenkins
- Installiere die notwendigen Plugins für Jenkins (Violations, Cobertura)
- Installiere Nose, Coverage und Pylint (mithilfe von pip)
- Integriere dein Bruch-Projekt in Jenkins, indem du es mit Git verbindest
- Überlege dir und argumentiere eine sinnvolle Pull-Strategie
- Konfiguriere Jenkins so, dass deine Unit Tests automatisch bei jedem Build durchgeführt werden inkl. Berichte über erfolgreiche / fehlgeschlagene Tests und Coverage
- Protokolliere deine Vorgehensweise (inkl. Zeitaufwand, Konfiguration, Probleme) und die Ergebnisse (viele Screenshots!)

Erweiterungen:

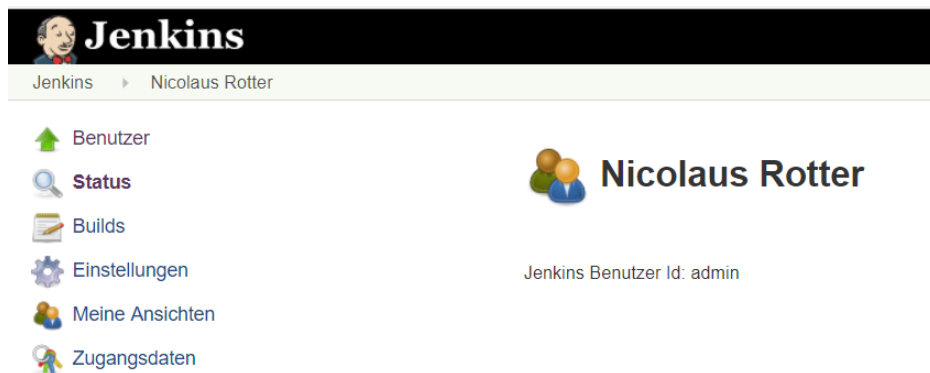
- Konfiguriere und teste eine Git-Hook, sodass Änderungen auf GitHub automatisch einen Build auslösen! Dokumentiere deine Vorgangsweise (mit Screenshots)!
- Recherchiere, wie mithilfe von Jenkins GUI-Tests durchgeführt werden können und baue selbstständig einen Beispiel-GUI-Test ein! Dokumentiere deine Vorgangsweise (mit Screenshots)!
- Lass deine Sphinx-Dokumentation automatisch mitbuilden und veröffentlichen! Dokumentiere deine Vorgangsweise (mit Screenshots)!

## 2 Grundanforderungen

### 2.1 Installation Jenkins

Der erste Punkt verlangt nach der Installation von Jenkins. Dafür wird die Software über die im Kurs abgelegte Verlinkung zu <https://jenkins.io/index.html> heruntergeladen. Nach der erfolgreichen installation sollte es möglich sein über den Browser auf **localhost:8080** Zugriff zu Jenkins zu bekommen.

zunächst wurde ein neuer Benutzer mit Administrator-Rechten hinzugefügt. Dieser kann dann im /user/ - Tab eingesehen werden.



### 2.2 Installation Violations, Corbertura

Zunächst wird die Installation der von uns benötigten Jenkins PlugIns verlangt. Dazu muss man auf **Jenkins verwalten** -> **PlugIns verwalten** gehen und im **Verfügbar**-Tab diese installieren.

Falls dieser Schritt erfolgreich war sollten die PuligIns nach dem Neustart von Jenkins im **Installiert**-Tab erscheinen.

### 2.3 Installation Nose, Coverage, Pylint

Nun müssen folgende Packages über pip installiert werden:

- Nose
- Coverage
- Pylint

Dies wurde im meinem Fall schon gemacht, daher bekomme ich folgende Rückmeldung in der Konsole:

```
C:\Users\nicro>pip install Nose
Requirement already satisfied: Nose in c:\python34\lib\site-packages

C:\Users\nicro>pip install Coverage
Requirement already satisfied: Coverage in c:\python34\lib\site-packages

C:\Users\nicro>pip install Pylint
Requirement already satisfied: Pylint in c:\python34\lib\site-packages
Requirement already satisfied: colorama; sys_platform == "win32" in c:\python34\lib\site-packages (from Pylint)
Requirement already satisfied: six in c:\python34\lib\site-packages (from Pylint)
Requirement already satisfied: isort>=4.2.5 in c:\python34\lib\site-packages (from Pylint)
Requirement already satisfied: astroid>=1.5.1 in c:\python34\lib\site-packages (from Pylint)
Requirement already satisfied: mccabe in c:\python34\lib\site-packages (from Pylint)
Requirement already satisfied: wrapt in c:\python34\lib\site-packages (from astroid>=1.5.1->Pylint)
Requirement already satisfied: lazy-object-proxy in c:\python34\lib\site-packages (from astroid>=1.5.1->Pylint)
```

## Literatur

- [1] A.S. Tanenbaum and M. Van Steen. *Verteilte Systeme: Prinzipien und Paradigmen*. Pearson Studium. Addison Wesley Verlag, 2007.

## Tabellenverzeichnis

## Listings

## Abbildungsverzeichnis