

# **Konzeption und Implementierung einer REST-basierten Schnittstelle zum Kopieren von Jira-Projekten**

**Praxisbericht 2 - T2000**

des Studiengangs Informatik

an der Dualen Hochschule Baden-Württemberg Stuttgart

von

**Luca Stanger**

Februar 2020

**Bearbeitungszeitraum**  
**Matrikelnummer, Kurs**  
**Ausbildungsfirma**  
**Betreuer**

12 Wochen  
7474265, TINF-18B  
camos Software und Beratung GmbH, Stuttgart  
M.Sc. Markus Riegger

# Inhaltsverzeichnis

Abkürzungsverzeichnis	II
Abbildungsverzeichnis	III
Tabellenverzeichnis	IV
Listings	V
1 Deployment	1
2 Git/Github	2
3 Continuous Integration	3
4 Chapter2	4
Anhang	6

# Abkürzungsverzeichnis

# Abbildungsverzeichnis

# Tabellenverzeichnis

# Listings

# 1 Deployment

## 2 Git/Github



## 3 Continuous Integration

Als CI wurde am Anfang AWS CodePipeline verwendet. Dies bot eine einfache Einbindung zwischen Github und AWS ElasticBeanstalk. Sobald der Master des Github Repositories einen neuen Push erhalten hat, wurde automatisch der neue Code auf den Server geladen. Jedoch als später Tests zum Projekt hinzugefügt wurden, wollten wir, dass automatisch die Tests vor dem Deploy ausgeführt werden. Dazu wäre AWS CodeBuild eine Möglichkeit gewesen. Doch persönlich wurde bereits mit Travis-CI gearbeitet, was eine kostenlose und einfach implementierte Version darstellt. Für Travis CI muss man nur dem Programm Berechtigungen für das Github Repository geben, sowie eine `.travis.yml` Datei im Root des Repositories vorhanden sein. In dieser yml-Datei spezifiziert man Befehle die in einer bestimmten Reihenfolge ablaufen. Außerdem wurde AWS CodePipeline nicht mehr benötigt, da Travis CI eine eigene kostenlose Deploy Möglichkeit bereitstellt. Dadurch wird nach jedem Push auf den Master des Github Repositories bei Travis CI eine virtuelle Umgebung aufgebaut die die definierten Befehle ausführt. Schlägt ein Build fehl wird es gestoppt. Den Status des Builds erkennt man an der build Badge in der ReadMe Datei. Sehen Sie hier den letzten Build des Projektes: <https://travis-ci.com/github/Drinkler/Planning-Poker>.

# 4 Chapter2

Chapter2



# Anhang

A.