

# FreeJDAQ

**Visuelle Programmiersprache zur Datenerfassung auf  
einem Raspberry Pi**

David Gawron, Stefan Geretschlaeger, Leon Huck,  
Jan Kublbeck, Linus Ruhnke

15. September 2019

# Problemstellung

The logo consists of the text "FreeJDAQ" in a bold, black, serif font, centered within a bright yellow rectangular box. The box has a thick black border and is set against a light gray background that features a subtle drop shadow effect.

**FreeJDAQ**

Free Java Data Acquisition

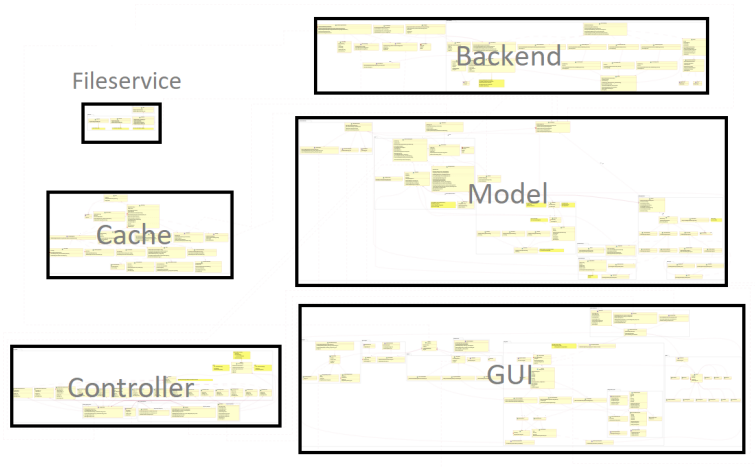
# Abgrenzungen

Was unser Produkt nicht enthält:

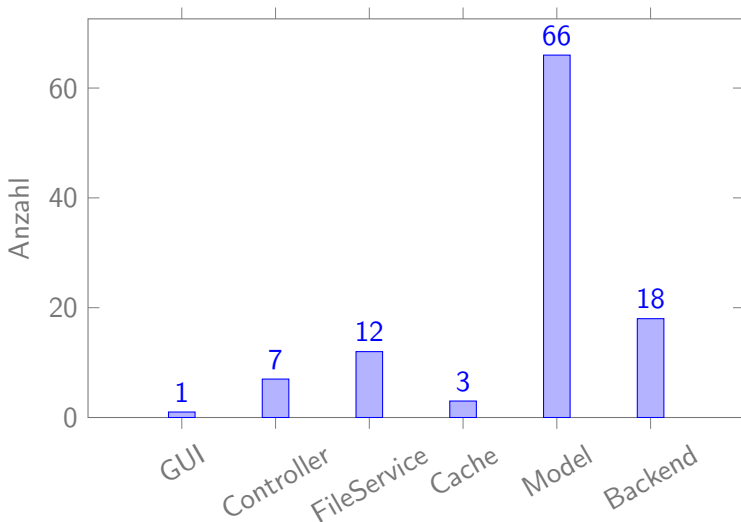
- Direkte Ansprache der Sensoren. (PhyPiDAQ)
- Visuelle Repraesentation der Messkonfiguration

# Grundaufbau

# Paketdiagramm

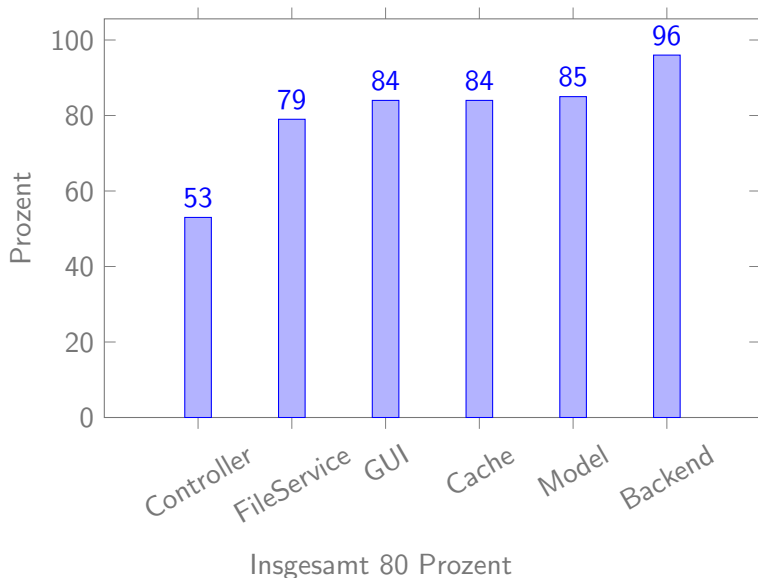


# Unit-Tests

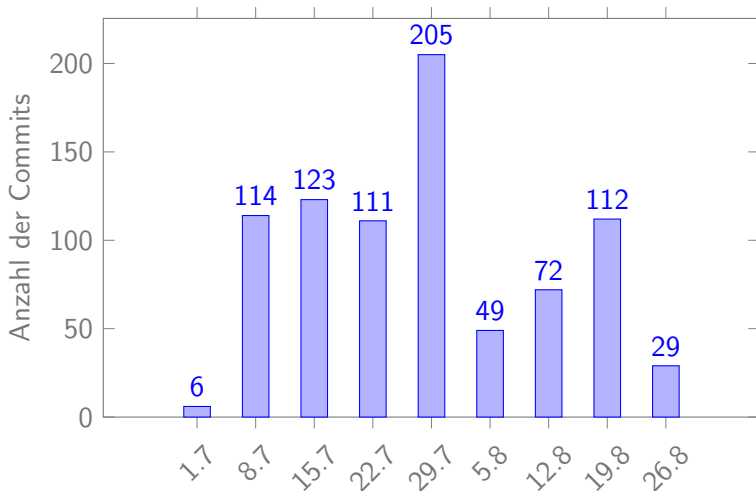


Insgesamt 107 Testcases, zzgl. 33 GUI - Klickstrecken

# Testabdeckung



# GitHub - FreeJDaq - Commits



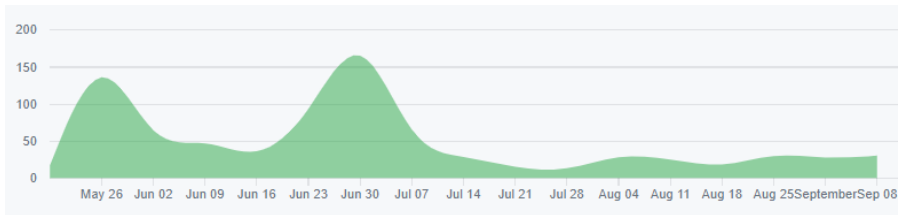
Insgesamt 824 Commits, 51/64 Issues closed



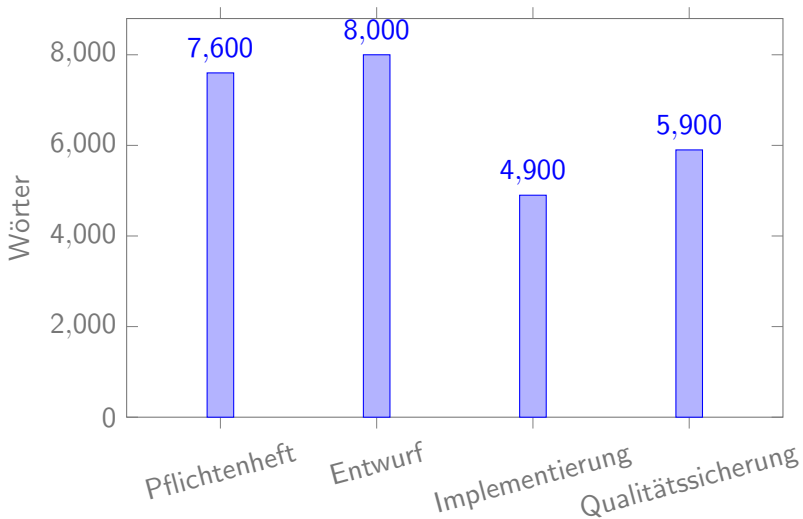
# GitHub - FreeJDaq - Lines of Code

Datei	Anzahl Zeilen
Java	5255
JavaScript	todo
XML	todo
Properties	todo
CSS	todo
Gesamt (inklusive Kommentar- und Leerzeilen)	12776

# GitHub - DAQDocuments



# GitHub - DAQDocuments



Insgesamt ca. 26400 Wörter über 975 Commits

# Allgemein

## UML



## Unit-Testing

JUnit

## IDE



## Yaml-Editor

SnakeYAML

## SSH

SSHJ

## Build Management

Maven™

## Statische Codeanalyse



# Probleme

- Teamkommunikation in den ersten Phasen
- Nacharbeiten von Fehlern oder Vervollständigung
- Technologiewahl → Technologiewechsel

# Was haben wir gelernt

- Phasen planen → Meilensteine, Deadlines setzen und Zuständigkeiten zuteilen
- Meilensteine überprüfen und ggf. Ressourcen verschieben
- Vor der Implementierung die nötigen Tools aussuchen und in diese einlesen