

FreeJDAQ

Visuelle Programmiersprache zur Datenerfassung auf einem Raspberry Pi

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

23. September 2019

Problemstellung

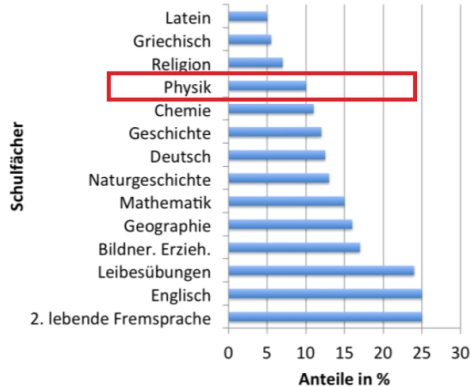


Abbildung 1: Lieblingsfächer auf Grund des Interesses

Problemstellung

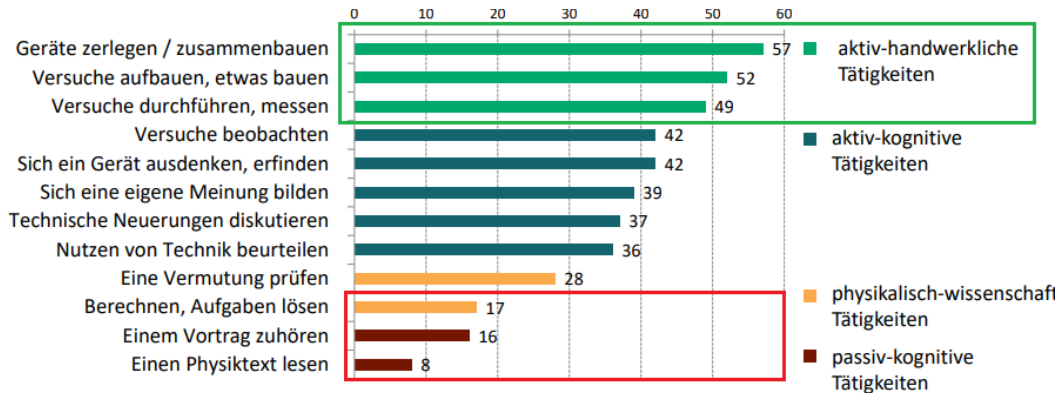
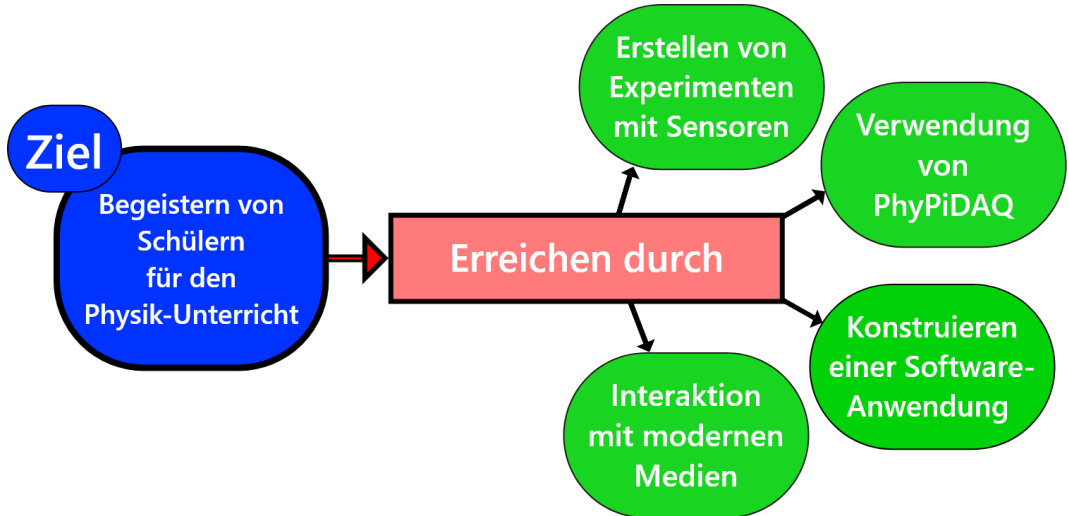


Abbildung 2: Interessen im Physikunterricht

Projektvorstellung



Erweiterungsmöglichkeiten von PhyPiDAQ

Von Prof. Dr. Günter Quast

Vorhandene Funktionen

Verarbeiten von Sensordaten durch
einen **Raspberry Pi**

Erstellen von Messkonfigurationen
durch **Python Code** und **Yaml**

Volle Funktionalität über das
Terminal/Commandozeile

Ausführliche Dokumentation
in der **Softwarebeschreibung**

Erweiterungs Potenzial

Erweiterung auf
Linux und **Windows** Computer

Vereinfachung durch
Schülergerechten Konfigurationsbau

Kreieren einer
Grafischen Benutzeroberfläche

Integrieren einer
leicht verständlichen **Hilfe**

Projektvorstellung

The logo consists of a bright yellow rectangle with a thick black border. Inside the rectangle, the text "FreeJDAQ" is written in a bold, black, serif font. The rectangle is centered on the slide and has a subtle gray drop shadow beneath it.

FreeJDAQ

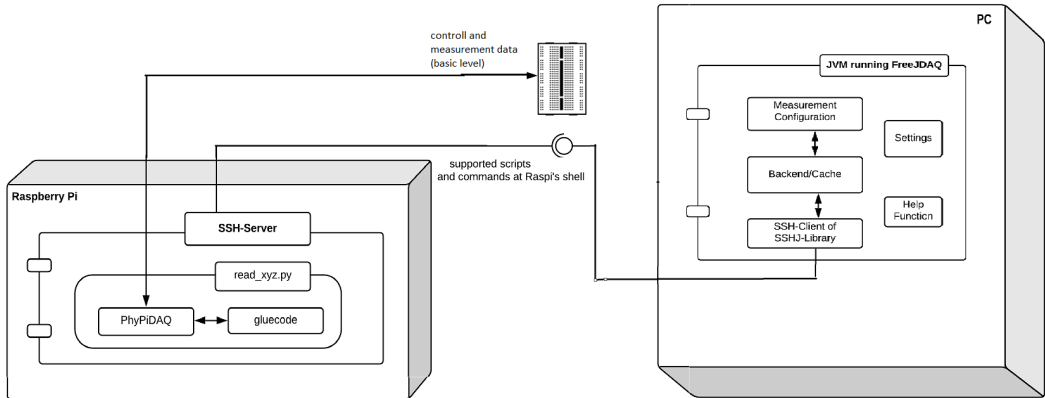
Free Java Data Acquisition

Abgrenzungen

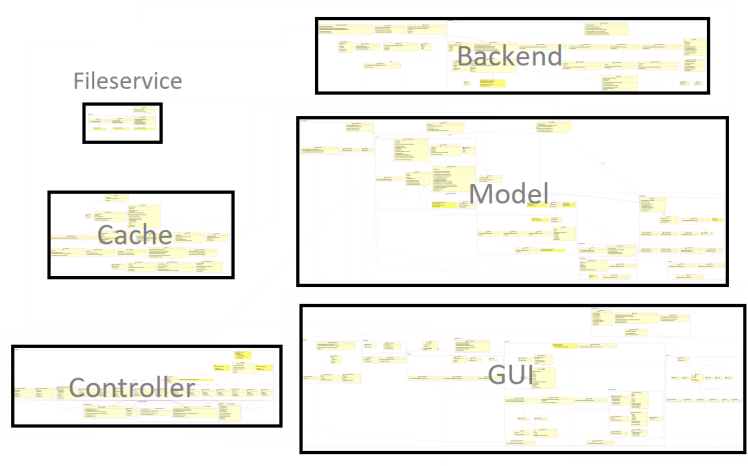
Was unser Produkt nicht enthält:

- Direkte Ansprache der Sensoren (PhyPiDAQ)
- Visuelle Repräsentation der Messkonfiguration
- Abfangen von Fehlern beim Anschließen der Messtechnik
- Erklärungen auf physikalischer Ebene

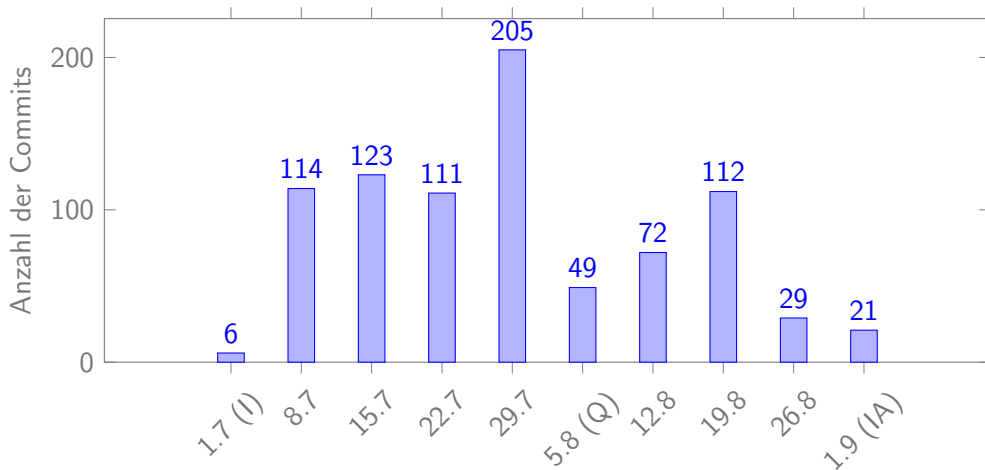
Grundaufbau



Paketdiagramm



GitHub - FreeJDaq - Commits



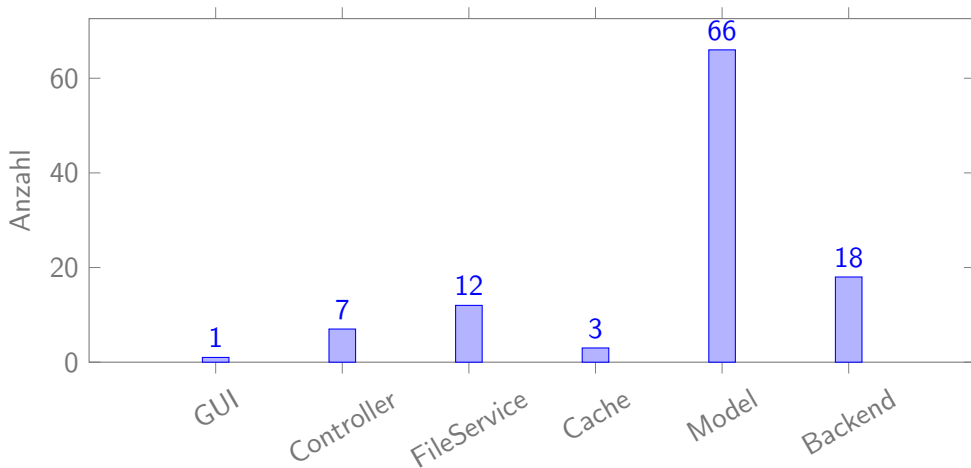
Insgesamt 842 Commits, 54/64 Issues closed, (15.09, 18:00 Uhr)

GitHub - FreeJDaq - Lines of Code

| Datei | Anzahl Zeilen |
|--|---------------|
| Anwendung | 4013 |
| Test | 1539 |
| Gesamt | 5552 |
| Gesamt (inklusive Kommentar- und Leerzeilen) | 12776 |

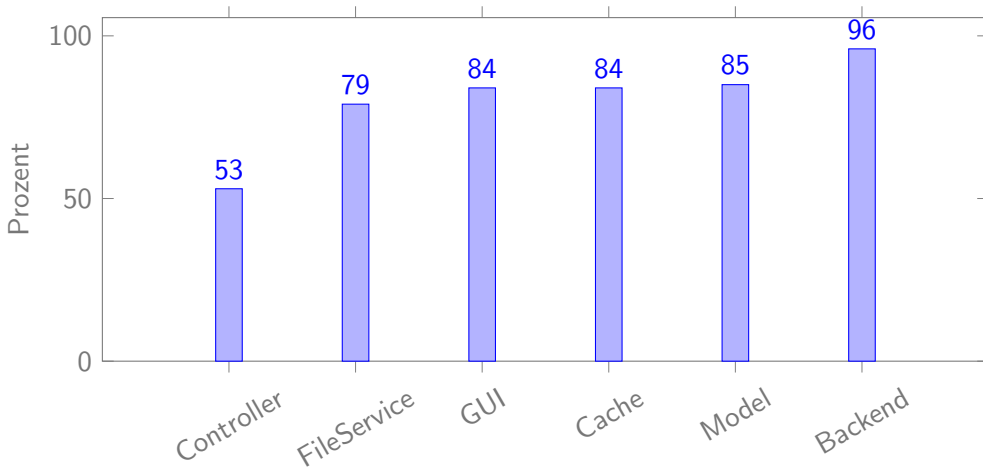
Verteilt über 122 Mainklassen und 23 Testklassen

Unit-Tests



Insgesamt 107 Testcases, zzgl. 33 GUI - Klickstrecken

Testabdeckung



Insgesamt 80 Prozent Bedingungsüberdeckung.

Allgemein

UML



Unit-Testing



IDE



Java-Bibliotheken



Build Management



Statische Codeanalyse

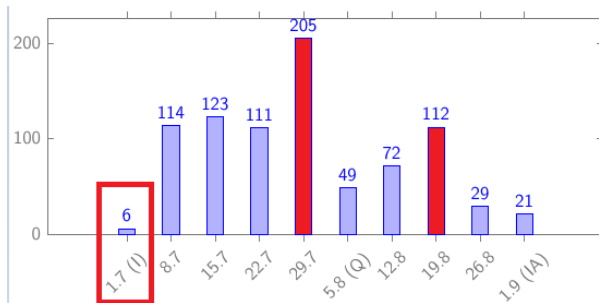


Probleme

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

Was haben wir gelernt

- Phasen planen → Meilensteine, Deadlines setzen und Zuständigkeiten zuteilen
- Arbeitsverteilung gleichmäßig über den Zeitraum verteilen
- Meilensteine überprüfen und ggf. Ressourcen verschieben
- Vor der Implementierung die nötigen Tools aussuchen und in diese einarbeiten



Livedemo

Begriffserklärung

Messkonfiguration:

Virtueller Versuchsaufbau im Yaml-Format bestehend aus:

- Bausteinen (Ein- und Ausgänge)
- Verbindungen

Bausteine:

- Sensoren
- Transformationen (Funktion zur Veränderung der Messdaten)
- Repräsentationen (Darstellung der Messdaten)

Verbindungen:

- Verknüpfungen zwischen den Ein- und Ausgänge der Bausteine.

Zusammenfassung

Zur Anwendung:

- Es wurde mit FreeJDAQ eine Basis geschaffen, welche Schülern und Physikinteressierten Menschen eine Plattform bietet, Messläufe einfach und schnell durchzuführen
- Weitere Produkteigenschaften und Erweiterungen können dieser Basis hinzugefügt werden

Zur Gruppenarbeit:

- Trotz Schwierigkeiten während jeder Phase hat sich unsere Gruppendynamik positiv entwickelt
- Gewinnung wichtiger Erfahrung in der Projektplanung und Softwareentwicklung

Vielen Dank für Ihre Aufmerksamkeit

The logo consists of the text "FreeJDAQ" in a bold, black, serif font. It is centered within a bright yellow rectangular box that has a thick black border. This box is positioned on a light gray rectangular base, which is slightly offset to the right and bottom, creating a 3D effect.

FreeJDAQ

Free Java Data Acquisition

Quellen

- <https://github.com/osl2/DAQ-Documents>
- <https://github.com/osl2/PhyPiDAQ>
- <https://github.com/GuenterQuast/PhyPiDAQ>
- <http://plantuml.com/de/>
- <https://junit.org/junit5/>
- <https://www.eclipse.org/ide/>
- <https://bitbucket.org/asomov/snakeyaml/src>
- <https://github.com/hierynomus/sshj>
- <https://maven.apache.org/>
- <https://www.eclemma.org/>
- <https://www.eclemma.org/jacoco/>
- <https://www.sonarlint.org/>
- <http://www.jfree.org/jfreechart/>

Quellen

- Abbildung 1: https://ag4physik.files.wordpress.com/2017/03/interessensforschung_strahl.pdf
- Abbildung 2: [http://www.physikdidaktik.info/data/_uploaded/Delta_Phi_B/2015/Caglar-Oeztuerk\(2015\)Interessenforschung_DeltaPhiB.pdf](http://www.physikdidaktik.info/data/_uploaded/Delta_Phi_B/2015/Caglar-Oeztuerk(2015)Interessenforschung_DeltaPhiB.pdf)