

# GitLab Time-Report

## Generierung von Statistiken und Diagrammen über geleistete Arbeitszeit auf GitLab

### Studenten



Jannis Tschan



Nina Grässli

**Einleitung:** Für Semester- und Bachelorarbeiten an der OST müssen Studierende ihre Arbeitszeit erfassen. Sie sind dabei frei in der Wahl ihrer Methodik, doch empfohlen werden Jira und YouTrack. Diese Tools sind jedoch schwerfällig und haben eine steile Lernkurve. Als Alternative kann das Zeiterfassungs-Feature von GitLab verwendet werden, das sich sehr einfach bedienen lässt. GitLab bietet jedoch keinerlei Funktionalitäten zur Auswertung der Zeiteinträge an. Es besteht lediglich die Möglichkeit, die Einträge über eine API auszulesen. Es gibt mehrere Programme, die versuchen, diese Lücke zu schliessen, indem sie die Zeiteinträge von GitLab über die API abrufen und weiterverarbeiten. Diese sind jedoch teils veraltet, fehleranfällig, unflexibel oder erfordern viel manuelle Nacharbeit.

Genau hier setzt unser Projekt "GitLab Time-Report" an: Ein funktionales und wartbares Programm, das die Zeiteinträge von GitLab extrahiert, für die Studierenden aufbereitet und dabei perfekt auf deren Bedürfnisse zugeschnitten ist.

**Vorgehen / Technologien:** Die Business-Logik von "GitLab Time-Report" ist in einer Rust-Library umgesetzt, die über ein CLI-Frontend angesteuert wird. Die Applikation bereitet die von der GitLab GraphQL-API bezogenen Zeiteinträge in verschiedenen Diagramm-Typen und Tabellen auf und unterstützt diverse Filtermöglichkeiten und eine Validierung der Einträge. Um einen Überblick über die geleisteten Arbeitsstunden zu erhalten, werden die Diagramme und Tabellen in ein Dashboard eingebettet, das sich beispielsweise über GitLab Pages veröffentlichen lässt. Zusätzlich können die Diagramme auch über CI/CD in die Projekt-Dokumentation (z.B. LaTeX, Typst, Markdown etc.) übernommen werden.

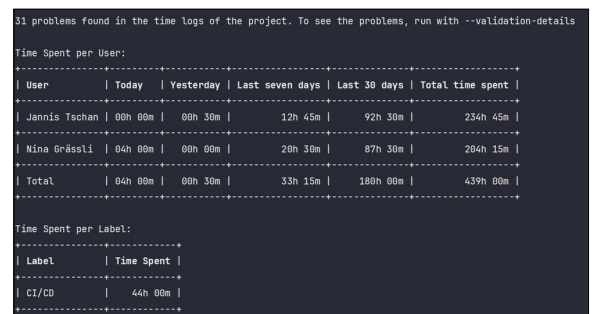
Die Qualität des Programms wird durch Unit- und Integration-Tests mit hoher Testabdeckung und einem strikt konfigurierten Linter sichergestellt. Diese Massnahmen werden automatisiert durch die CI/CD Pipeline ausgeführt, welche auch Builds für Windows und Linux erstellt. Damit unser Programm leicht bedienbar ist, wurden Usability Tests durchgeführt. Für die Sicherstellung der Wartbarkeit haben wir eine umfangreiche Code-Dokumentation geschrieben, die es anderen Personen einfach macht, die Applikation weiterzuentwickeln.

**Fazit:** Mit "GitLab Time-Report" erreichen wir das Ziel, ein praxistaugliches und robustes Werkzeug für die Projekt-Planung via GitLab bereitzustellen. Es erfüllt die am Anfang des Projektes definierten Anforderungen, wird von Usern als intuitiv bedienbar wahrgenommen und lässt sich in bestehende Workflows integrieren. Durch die Wartbarkeit, die wir durch die umfangreiche Dokumentation und die

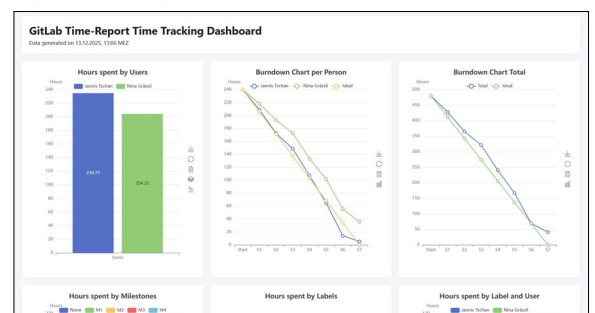
saubere Architektur sichergestellt haben, kann die Applikation in Zukunft ohne Probleme durch weitere Features ergänzt werden.

Wir haben "GitLab Time-Report" unter der GPLv3-Lizenz öffentlich zugänglich gemacht, damit neben den OST-Studierenden auch die breite Öffentlichkeit von diesem Programm profitieren kann.

### Auszug aus dem CLI-Output Eigene Darstellung



### Dashboard mit Zeiterfassungs-Diagrammen im Light Mode Eigene Darstellung



### Dashboard mit Zeiterfassungs-Diagrammen und Tabellen im Dark Mode Eigene Darstellung



Referent  
Prof. Mirko Stocker

Korreferent  
Dominic Klinger

Themengebiet  
Software, Data Science,  
Software Engineering

