

Classroom 50: Klassenzimmer und Aufgaben verwalten

Diese Seite beschreibt den **laufenden Betrieb**. Ein Klassenzimmer neu aufsetzen: [Einrichtung](#). Von GitHub Classroom umsteigen: [Migration](#).

Werkzeug

Classroom 50 hat keine Web-Oberfläche zum Verwalten. Alles läuft über die GitHub-CLI-Erweiterung `gh teacher`:

```
gh extension install foundation50/gh-teacher
gh auth refresh -h github.com -s admin:org,read:org,repo,workflow
```

Der zweite Befehl ist keine Formalie: Ohne `admin:org` bricht **jeder** `gh teacher`-Aufruf mit einer Scope-Meldung ab. Er öffnet den Browser und braucht deshalb ein interaktives Terminal.

```
gh teacher classroom list <ORG>
```

Klassenliste (Roster)

Die Klassenliste liegt als `<CLASSROOM>/roster.csv` im Config-Repo, mit diesem Header:

```
username,first_name,last_name,email,section,github_id,role
```

Ältere Klassenzimmer haben stattdessen eine `students.csv`. Damit scheitert jeder `gh teacher roster`-Befehl, und die Namen in `scores.json` bleiben leer. Datei umbenennen und die Spalte `role` ergänzen.

Das Roster steuert die Bewertung **nicht** — es liefert nur Anzeigedaten. Eingesammelt wird nach der Union aus Studenten-Team und **allen** Staff-Teams (`teacher`, `hta`, `ta`). Wer als Lehrperson eine Aufgabe annimmt, um den Ablauf zu testen, wird wie eine lernende Person bewertet und landet in `scores.json` — und damit im Moodle-Übertrag, wo der Account dann fehlt. Testrepos nach dem Test löschen **und** den Eintrag aus `scores.json` entfernen.

Aufgaben

Alle Aufgaben einer Klasse stehen in `<CLASSROOM>/assignments.json` im Config-Repo. Ein Eintrag verweist auf das Template-Repo, den Slug, den Abgabetermin und den zu verwendenden Autograder.

An der BZZ steht bei allen Aufgaben „autograder“: „default“ und „feedback_pr“: true, und **kein** tests-Block. Damit greift überall der Klassen-Default, also pygrader50 mit pytest und pylint.

Aufgaben auflisten:

```
gh api repos/<ORG>/classroom50/contents/<CLASSROOM>/assignments.json \
-H 'Accept: application/vnd.github.raw' | jq -r '.assignments[].slug'
```

Zu Templates siehe [Template-Repositories](#), zum Aufbau des Config-Repo [Das classroom50-Repository](#).

Resultate einsehen

- **Pro Abgabe** — Release und Commit-Status im Studi-Repo.
- **Pro Klasse** — <CLASSROOM>/scores.json im Config-Repo, nachts von collect-scores aktualisiert.
- **In Moodle** — nachts um 04:57 durch moodle-sync, siehe [Classroom 50: Noten nach Moodle übertragen](#).

```
gh api repos/<ORG>/classroom50/contents/<CLASSROOM>/scores.json \
-H 'Accept: application/vnd.github.raw' | jq '.'
```

Nachbewerten

Eine Abgabe neu bewerten lassen: im Studi-Repo unter **Actions** den Bewertungslauf erneut starten (*Re-run all jobs*). Das erzeugt ein neues Release; der nächste Moodle-Lauf überträgt die geänderte Punktzahl automatisch, weil sich die Punktzahl unterscheidet.

Nur die Note in Moodle nachtragen, ohne neuen Lauf:

```
gh workflow run moodle-sync.yaml --repo <ORG>/classroom50 \
-f classroom=<CLASSROOM> -f assignment=<SLUG> -f force=true
```

Betriebsfallen

scores.json wird nie aufgeräumt

Das Einsammeln ist ein reiner Upsert, geschlüsselt auf den Repo-Owner. Ein Eintrag bleibt stehen, auch wenn das Repository gelöscht oder die Person ausgetragen wurde. Weil der Moodle-Übertrag mit Exit 1 endet, sobald **eine** Übertragung scheitert, färbt ein einziger solcher Karteileichen-Eintrag den Nachlauf dauerhaft rot.

Nach einer Abmeldung den Eintrag von Hand aus <CLASSROOM>/scores.json entfernen.

Der Service-Token braucht Administration-Rechte

Fehlt dem CLASSROOM50_SERVICE_TOKEN das Recht *Repository → Administration: Read and write*, scheitert collect-scores mit staff-team access grant failed with HTTP 403. Die Punkte werden trotzdem gesammelt, der Job endet aber rot und die TA-Teams bekommen keinen Zugriff.

```
gh teacher rotate-service-token <ORG>
```

Python-Version

Ohne Angabe von runtime.python wählt der Runner **Python 3.14**. Dort installieren sich pytest 9.x und pylint 4.x — die Lint-Punkte fallen strenger aus als mit den alten Template-Pins (pytest==8.3.3, pylint==3.2.7).

Weiter

- [Wie bewertet wird](#)
- [Noten nach Moodle übertragen](#)
- [Problembehandlung](#)

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/howto/git/classroom50/lehrpersonen>

Last update: **2026/08/14 11:13**

