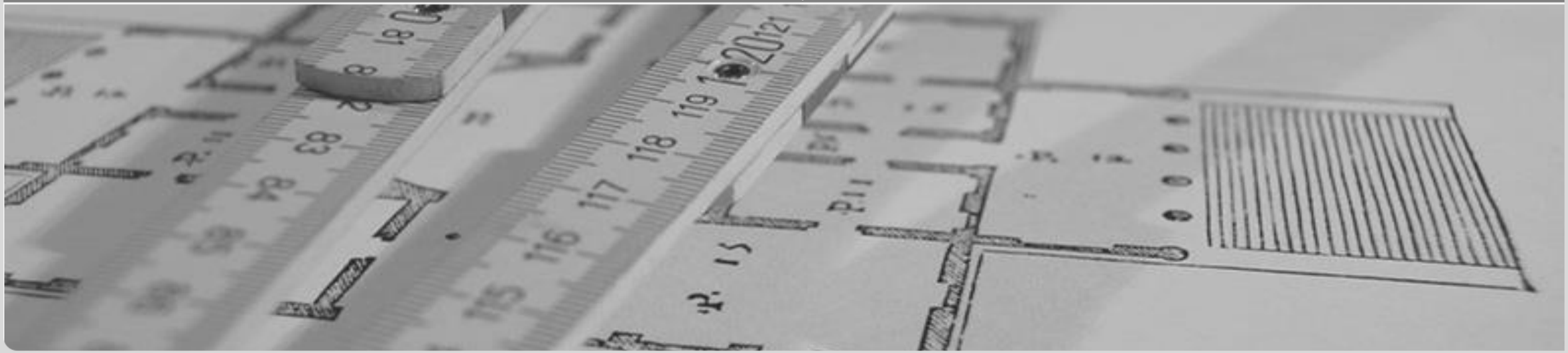
Karlsruher Institut für Technologie

Software Design and Quality

Teaching Programming at Scale

SEUH 2020 – A. Kaplan, J. Keim, Y. R. Schneider, M. Walter, D. Werle, A. Koziolek, R. Reussner

SOFTWARE-ENTWURF UND -QUALITÄT,
INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION, KIT-FAKULTÄT FÜR INFORMATIK



Programmierenlehre - *learning by doing*

Übungsblatt 01

Modellieren Sie ein Auto

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est …

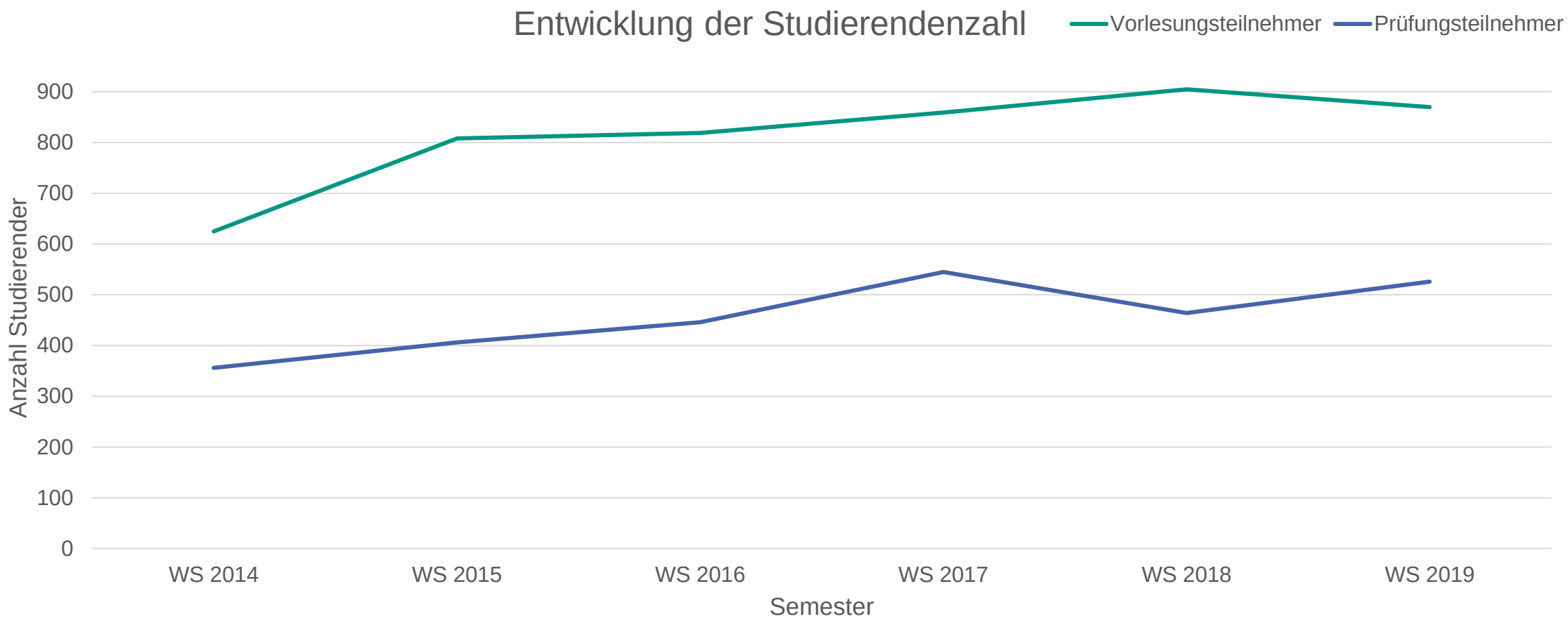
```
class Car {
…..
}
```

```
class Vehicle
{
…..
}
```

Bewertung

- Stil
- Funktionalität

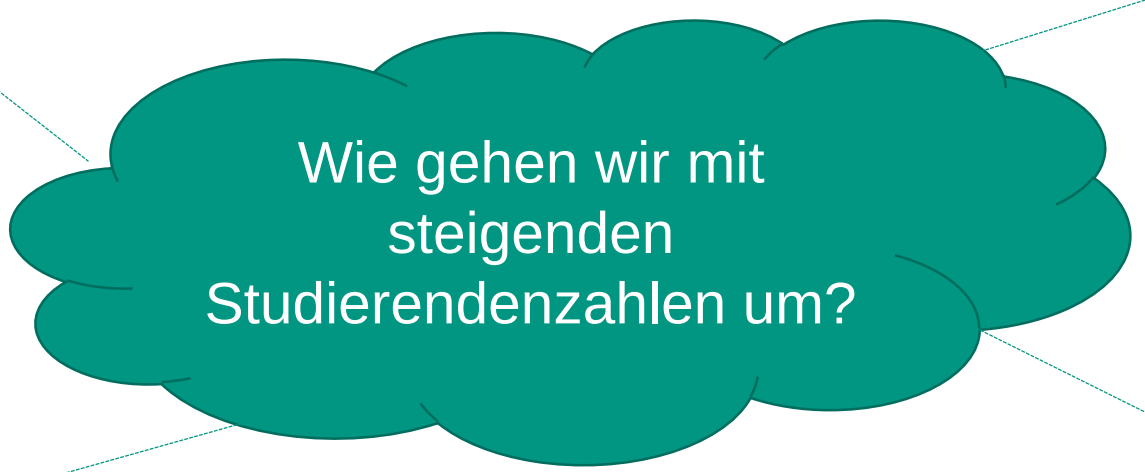
Steigende Studierendenzahlen



Steigende Studierendenzahl

Wie bewerten wir effektiv und nachvollziehbar den Stil der Abgaben?

Wie strukturieren wir den Lehrbetrieb gut?



Wie organisieren wir die Kommunikation mit den Studierenden möglichst effizient und fair?

Wie stellen wir sicher, dass Studierende Ihre Lösung selbständig erstellen?

Veranstaltungsaufbau

Lehrbetrieb:

- Vorlesung – Vorstellung allgemeiner Programmierkonzepte
- Übungsbetrieb - Anwendung der Programmierkonzepte

Leistungskontrolle:

- 2 umfangreiche Programmieraufgaben mit je ca. 1000 Zeilen Code
- Bewertung von Programmierstil und Funktionalität

Lehrkonzept – Erfahrungsbericht

Erfahrung:

- Klare Zeitpläne für Inhalte helfen Studierenden und Lehrenden*
- Aktivierungsaufgaben während der Vorlesung hilfreich*
- Implementierung von Spielen beliebt*

Offene Punkte:

- Vorstellungsreihenfolge von Algorithmen und Objektorientierung
- Trennung von Tutorieninhalt und Vorlesungsinhalt

* In Lehrevaluation durch Studierende genannt

Effektive und verständliche Bewertung

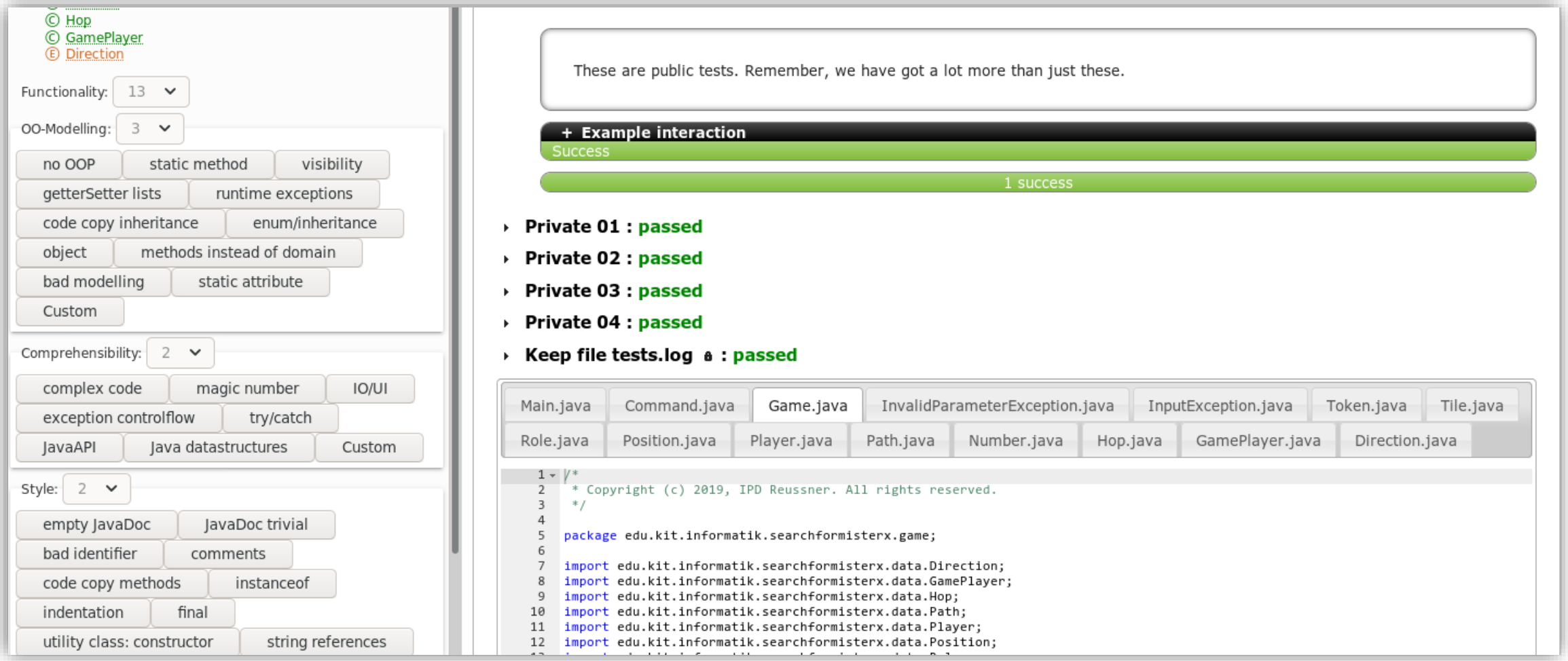
Herausforderung:

- Manuelles Bewerten von Programmierstil zeitaufwendig
- Korrekturanmerkungen teilweise schwer nachvollziehbar
- Bewertung von Stil teilweise sehr subjektiv

Lösungsansatz:

- Erweiterung des elektronischen Abgabesystems um Textbausteine
- Vorgegebene Liste der Bewertungskriterien
- Automatische Punkteberechnung auf Basis der Kategorien

Bewertung – Korrektursicht



Bewertung – Studierendensicht

Textbausteine mit Verweis auf Dateiname und Zeile des Punktabzugs

Erreichte Punktzahl je nach Kategorie

Erreichte Gesamtpunktzahl

Ergebnis funktionaler Tests

Bewertung – Erfahrungsbericht

Erfahrung:

- Liste mit Stilkorrekturen verringert die Anzahl der Fehlkorrekturen
- Verweise auf Fehler helfen Studierenden die Korrektur nachvollzuziehen
- Semi-automatische Werkzeuge reduzieren die Korrekturdauer

Offene Punkte:

- Stilkorrektur weiterhin aufwendig
- Aufwendige Wartung und Abhängigkeit von selbstentwickelten Werkzeugen
- Zu allgemeine Fehlerkategorien

Kommunikation mit Studierenden

Herausforderung:

- Individuelle Kommunikation mit Studierenden zeitaufwendig
- Für alle Studierende die gleichen Informationen
- Reduzierung von Wiederholungen

Lösungsansatz:

- Einsatz von Online-Lernplattformen

Kommunikation - Online-Lernplattform ILIAS

Wiki-Funktionalität:

- Dokumentation der Bewertungsgrundlage für Programmierstil
- Einstiegshilfen zur Programmierung mit Java (IDE, Installation Java)

Foren:

- Inhaltliche und organisatorische Fragen + Ankündigungen
- Aktuelles WS ca. 700 Beiträge
- Hauptsächlich Fragen zum Übungsbetrieb (ca. 85%)

Kommunikation – Erfahrungsbericht

Erfahrung:

- Vereinfachung der Kommunikation durch Online-Lernplattformen
- Online-Lernplattformen unterstützen sowohl Lehrende als auch Studierende*

Offene Punkte:

- Stärkere Beteiligung von Studierenden beim gegenseitigen Beantworten von Fragen
- Reduzierung von organisatorischen E-Mails

* In Lehrevaluation durch Studierende genannt

Zusammenfassung

Wie bewerten wir effektiv und nachvollziehbar den Stil der Abgaben?
- *Automatische Textbausteine*
- *Ausführliche Korrekturrichtlinien*

Wie strukturieren wir den Lehrbetrieb gut?
- *Zeitpläne erstellen*
- *Spiele sind beliebt*

Wie gehen wir mit steigenden Studierendenzahlen um?

Wie organisieren wir die Kommunikation mit den Studierenden möglichst effizient und fair?
- *Nutzen von Online-Lernplattformen*

Wie stellen wir sicher das Studierende Ihre Lösung selbständig erstellen?
- *Nutzen von Plagiaterkennungs-programmen*
- *Schriftliche Zwischenprüfung*