

# Selbstgesteuertes Lernen in der Programmierausbildung

Let's build a game!

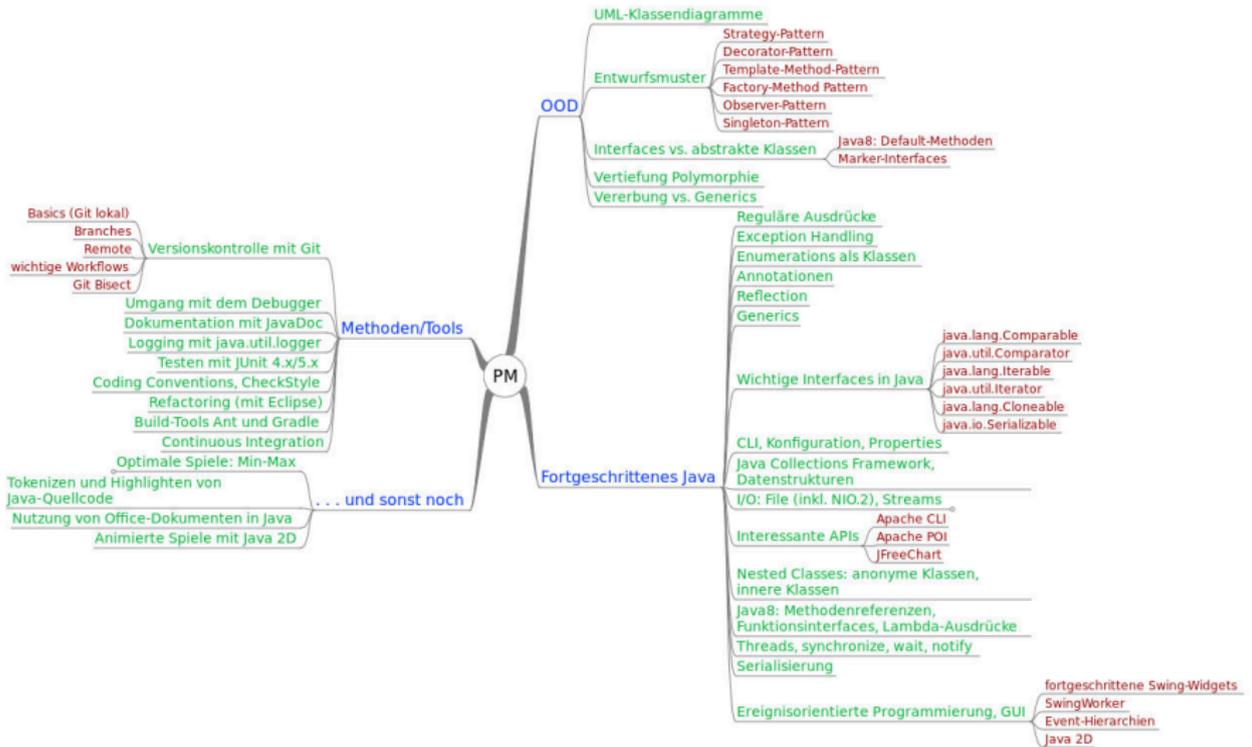
---

Carsten Gips, FH Bielefeld ([carsten.gips@fh-bielefeld.de](mailto:carsten.gips@fh-bielefeld.de))

# **Programmier-Module im Informatikstudium**

---

# Typisches Modul „Programmieren“ in der Informatik



# Bisherige Planung für das Modul

KW	VL (Fr)	Abgabe und Vorstellung im Praktikum (Fr)
15	VL01: Orga, Git Basics	–
16	–	–
17	VL02: Git Branches, Strategy-Pattern	B00 "Galgenmännchen": UML, JavaDoc, Debugger, Git Basics
18	VL03: Git Remote + Workflows, Annotationen	Fragestunde
19	VL04: Generics<T>	B01 "Tic Tac Toe": Strategie-Pattern, UML, JavaDoc, Git Branching
20	VL05: JUnit	Fragestunde
21	VL06: Coding-Rules, Bad Smells, Refactoring	B02a "Listen": JUnit, Generics, Merge-Requests
22	– (VL07: RegExp, CLI/Konfiguration)	<b>B02b</b> "Bundesliga": Generics, JUnit
23	VL08: Logging, Reflection, Factory-Method-Pattern	B03 "Lexer": RegExp, CLI; <b>B02b</b>
24	VL09: CI, Ant/Gradle	Fragestunde
25	VL10: Threads	B04 "Lexer 2.0": Annotationen, Reflection, Logging, Gradle/Ant, CI, Refactoring
26	VL11: Defaultmethoden, Methodenreferenzen, Funktions-Interfaces	Fragestunde
27	Puffer	B05 "Hamsterwelt": Threads, Synchronisation, Apache POI

- Aufwand für die Abnahme/Bewertung im Praktikum
- Motivation der Teilnehmer: (Zu) kleine und (zu) künstliche Probleme
- Relativ hohe Prüfungsbelastung
- Förderung stärkerer Teilnehmer, schwächere Teilnehmer nicht überfordern
- Bzgl. Vorwissen sehr heterogenes Teilnehmerfeld

**Idee: Entwickeln wir ein Spiel**

---

# Game-Based Learning



*... versucht DGBL (Digital Game-based Learning) das Lern- und Motivationpotential von digitalen Spielen, zum Erlernen von „realem“ Wissen zu verwenden, welches auch über das Spiel hinaus genutzt werden kann, ...*

*... soll der Leistungsdruck im digitalen Spiel positive und damit lernfördernde Effekte haben, im Gegensatz zum negativ wahrgenommen / lernhemmenden Leistungsdruck bei herkömmlichen Lernmethoden ...*

– Quelle: Wikipedia

# Umbau der Lehrveranstaltung: Let's build a game!

## Basis

- Game-Loop
- GUI

## Fortgeschrittenes Java

Generics

Reflection

Threads

...

## Objektorientiertes Design

UML

Pattern

Polymorphie

...

## Methoden und Tools

Git

Test

Refactoring

...

## Verschiedenes

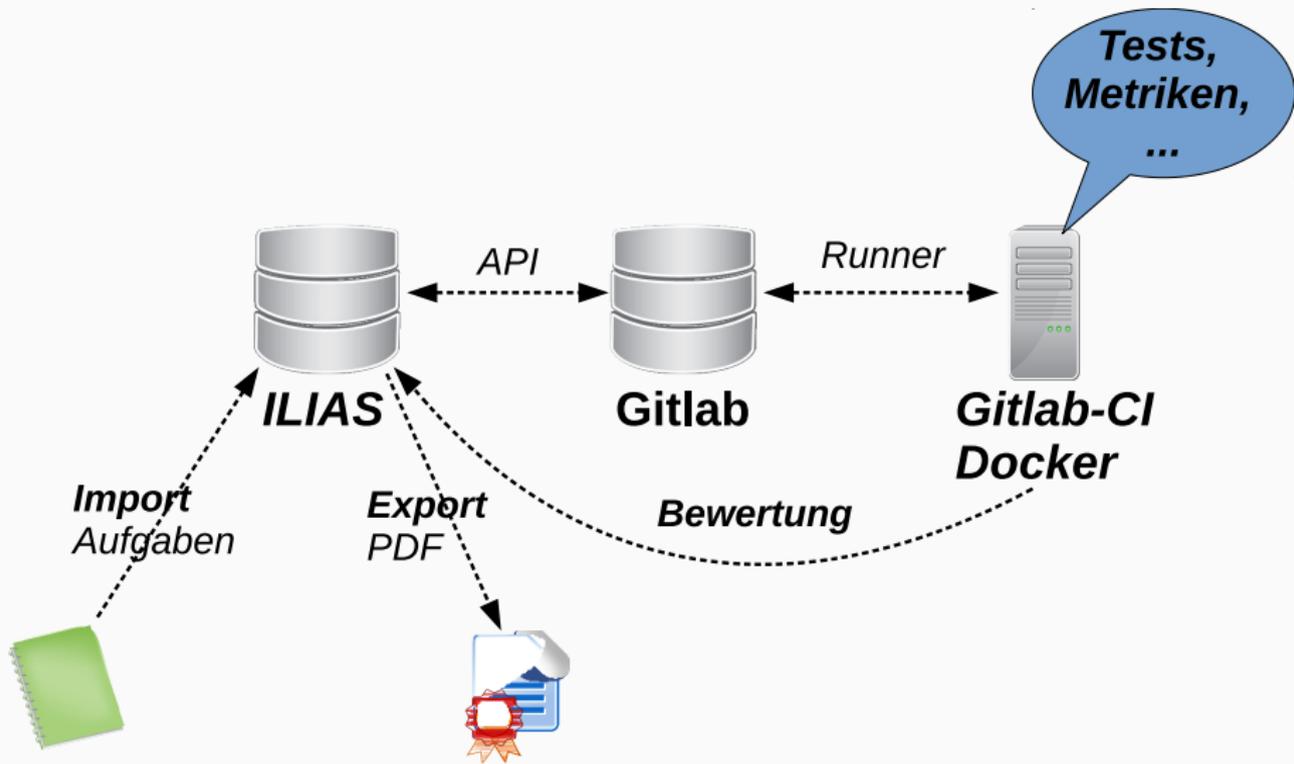
APIs

Logging

KI

...

# Detail: Abgabesystem für Praktikum



# Offene Fragen

- Qualität trotz „nur“ Testat im Praktikum?
- Qualität bei „nur“ (teil-) automatisierter Bewertung?
- Studierende überfordert mit selbstgesteuertem Bearbeiten?
- Messen und Beurteilen des Erfolgs der gesamten Maßnahme?

**Vielen Dank! Fragen?**

---