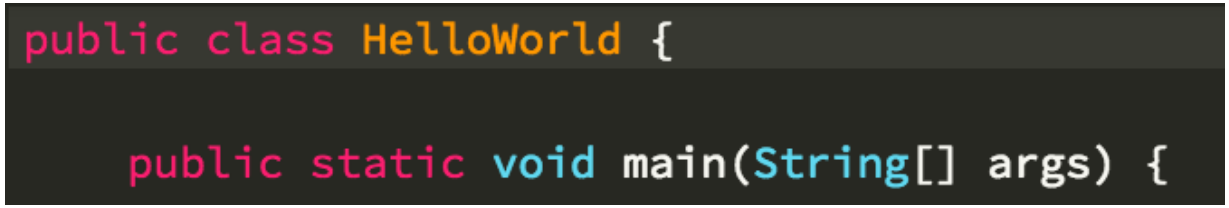
eCULT+
eCompetence and Utilities for Learners and Teachers

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

# Stud.IP lehrt Programmieren – Der neue Aufgabentyp „Programmieraufgabe“

Dr. Elmar Ludwig, Dominik Feldschnieders, Universität Osnabrück

Oliver Rod, TU Braunschweig




Oliver Rod
ol.rod@ostfalia.de
Dominik Feldschnieders
dofeldsc@uos.de
Dr. Elmar Ludwig
elmar.ludwig@uos.de

# Agenda

- Vorstellung
- Demonstration
- Motivation / Vorteile
- Programmieraufgaben in Vips
- Didaktische Einsatzszenarien
- Tools und Funktionen
- Austauschformat
- Grenzen der Werkzeuge
- Ausblick


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Vorstellung

# Vips



Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Wieso ProFormA?

- Programming
- Formative
- Assessment
  - Programming for Formative Assessment
  - Methoden zur Verbesserung der Qualität in der Programmierausbildung durch geeignete Verfahren zur automatisierten Programmbewertung
- Austausch Programmieraufgaben **und** LMS


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

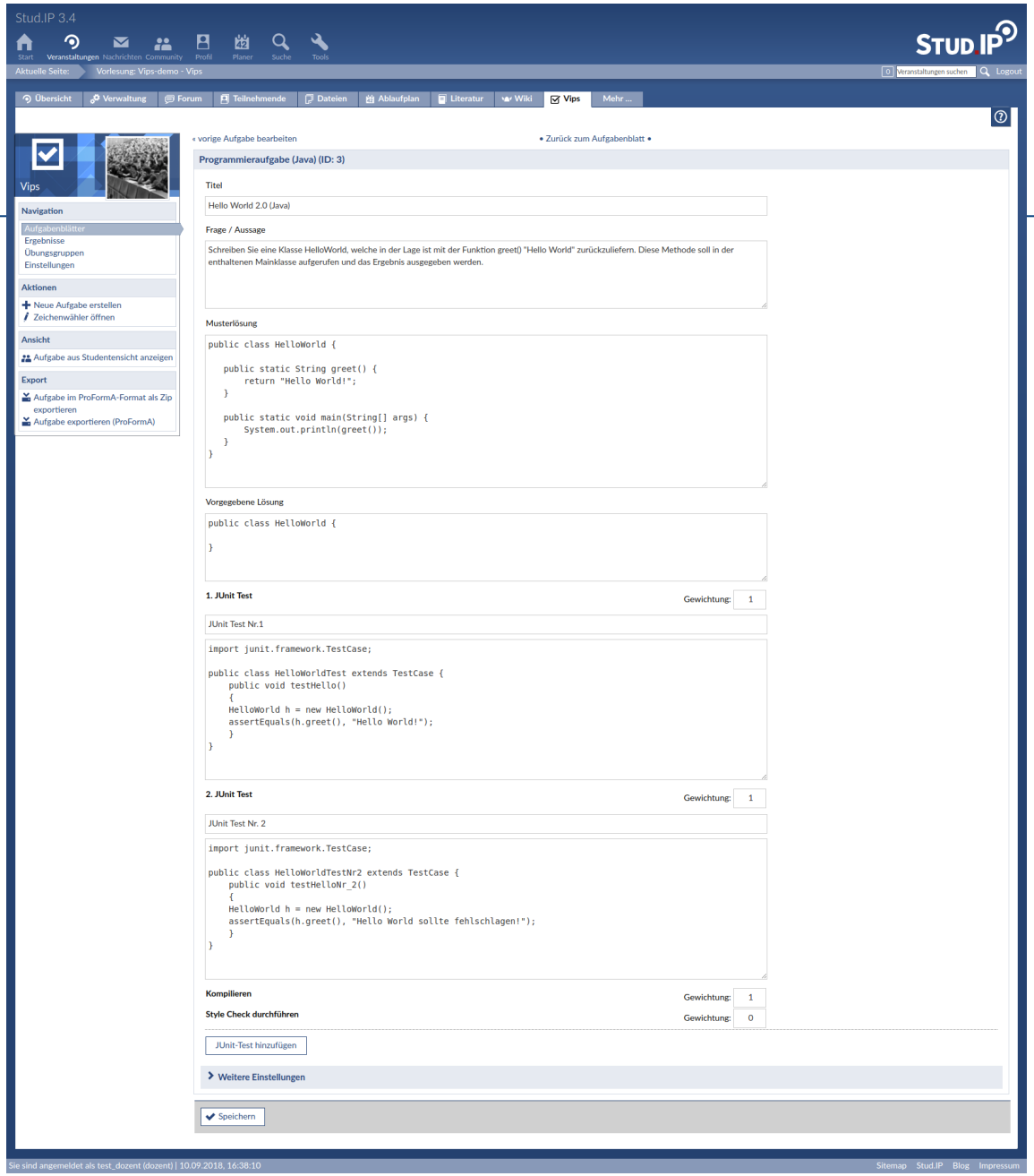
# ABP Workshop 2017

**3. Workshop "Automatische Bewertung von Programmieraufgaben“ (2017)**

# Demonstration

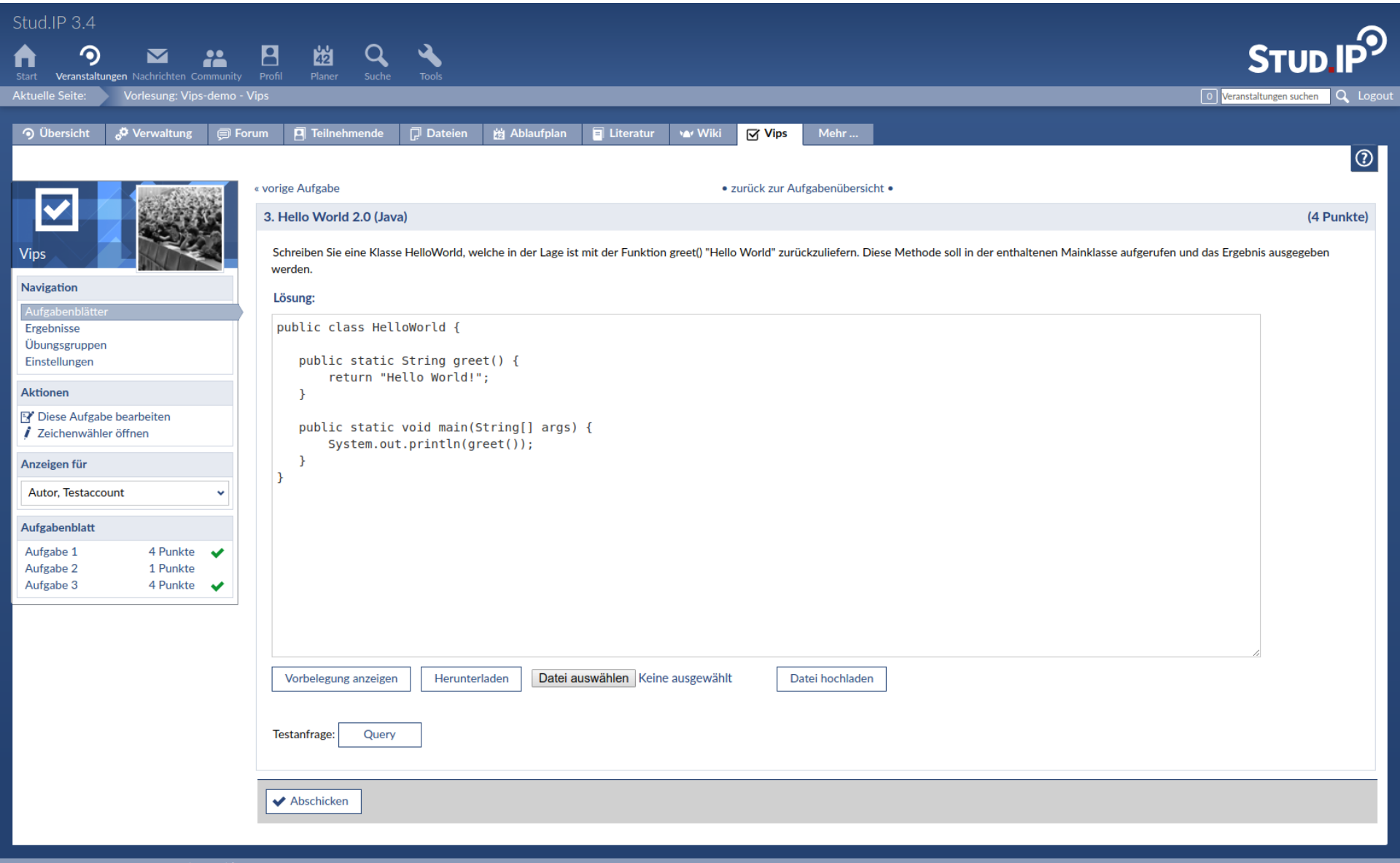


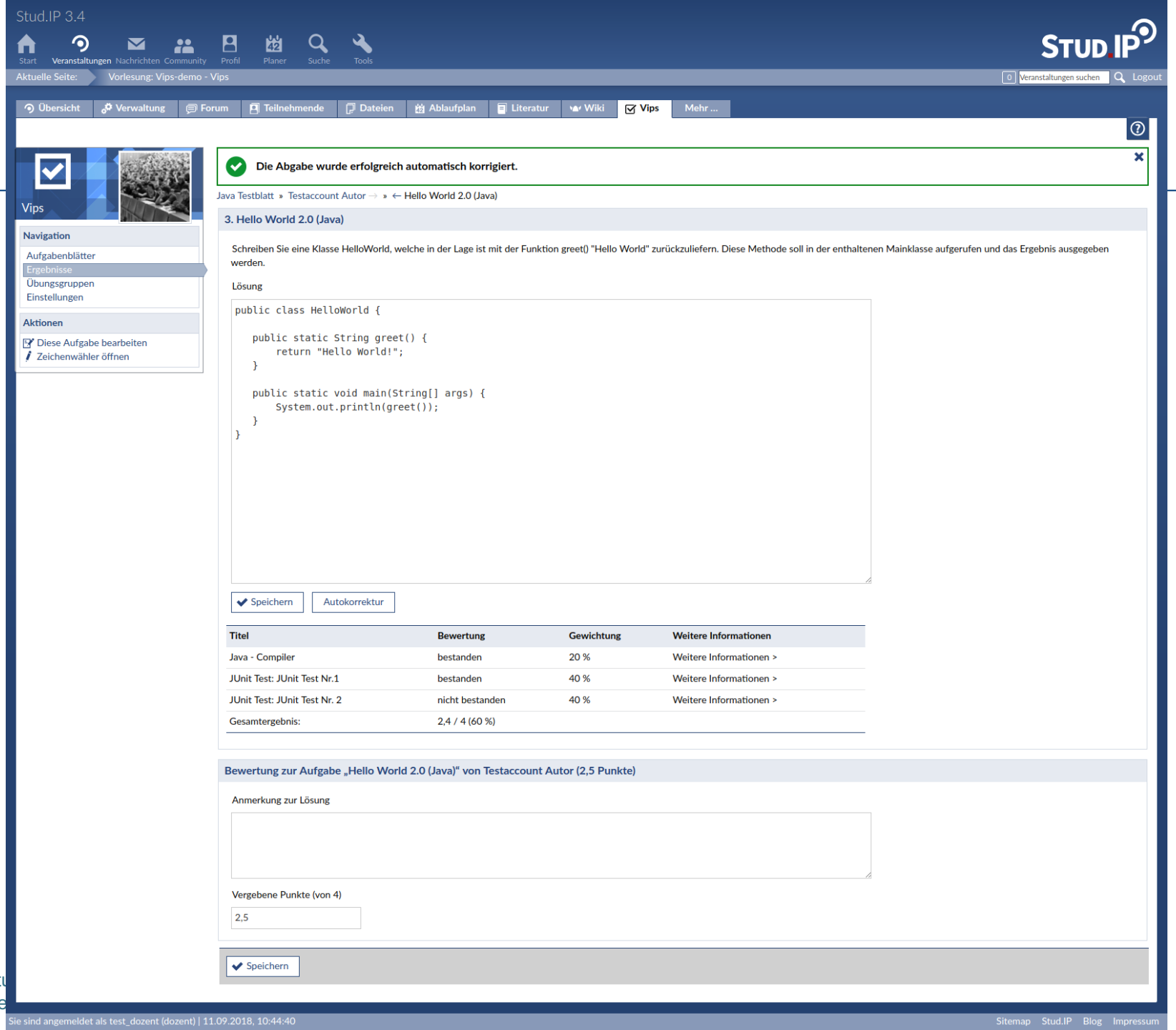
Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de






Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de


Stud.IP 3.4
Start
Veranstaltungen
Nachrichten
Community
Profil
Planer
Suche
Tools
STUD.IP
Aktuelle Seite:
Vorlesung: Vips-demo - Vips
Veranstaltungen suchen
Logout
Übersicht
Verwaltung
Forum
Teilnehmende
Dateien
Ablaufplan
Literatur
Wiki
Vips
Mehr ...
Vips
Navigation
Aufgabenblätter
Ergebnisse
Übungsgruppen
Einstellungen
Aktionen
Diese Aufgabe bearbeiten
Zeichenwähler öffnen
Anzeigen für
Autor, Testaccount
Aufgabenblatt
Aufgabe 1 4 Punkte
Aufgabe 2 1 Punkte
Aufgabe 3 4 Punkte
« vorige Aufgabe
• zurück zur Aufgabenübersicht •
3. Hello World 2.0 (Java)
(4 Punkte)
Schreiben Sie eine Klasse HelloWorld, welche in der Lage ist mit der Funktion greet() "Hello World" zurückzuliefern. Diese Methode soll in der enthaltenen Mainklasse aufgerufen und das Ergebnis ausgegeben werden.
Lösung:
public class HelloWorld {
public static String greet() {
return "Hello World!";
}
public static void main(String[] args) {
System.out.println(greet());
}
}
Vorbelegung anzeigen
Herunterladen
Datei auswählen
Keine ausgewählt
Datei hochladen
Testanfrage:
Query
Abschicken
Sie sind angemeldet als test_dozent (dozent) | 10.09.2018, 16:39:24
Sitemap
Stud.IP
Blog
Impressum


Stud.IP 3.4
Start
Veranstaltungen
Nachrichten
Community
Profil
Planer
Suche
Tools
STUD.IP
Aktuelle Seite:
Vorlesung: Vips-demo - Vips
Veranstaltungen suchen
Logout
Übersicht
Verwaltung
Forum
Teilnehmende
Dateien
Ablaufplan
Literatur
Wiki
Vips
Mehr ...
Vips
Navigation
Aufgabenblätter
Ergebnisse
Übungsgruppen
Einstellungen
Aktionen
Diese Aufgabe bearbeiten
Zeichenwähler öffnen
Die Abgabe wurde erfolgreich automatisch korrigiert.
Java Testblatt » Testaccount Autor → » ← Hello World 2.0 (Java)
3. Hello World 2.0 (Java)
Schreiben Sie eine Klasse HelloWorld, welche in der Lage ist mit der Funktion greet() "Hello World" zurückzuliefern. Diese Methode soll in der enthaltenen Mainklasse aufgerufen und das Ergebnis ausgegeben werden.
Lösung
public class HelloWorld {
public static String greet() {
return "Hello World!";
}
public static void main(String[] args) {
System.out.println(greet());
}
}
Speichern
Autokorrektur
Titel
Bewertung
Gewichtung
Weitere Informationen
Java - Compiler
bestanden
20 %
Weitere Informationen >
JUnit Test: JUnit Test Nr.1
bestanden
40 %
Weitere Informationen >
JUnit Test: JUnit Test Nr. 2
nicht bestanden
40 %
Weitere Informationen >
Gesamtergebnis:
2,4 / 4 (60 %)
Bewertung zur Aufgabe „Hello World 2.0 (Java)“ von Testaccount Autor (2,5 Punkte)
Anmerkung zur Lösung
Vergebene Punkte (von 4)
2,5
Speichern
Sie sind angemeldet als test_dozent (dozent) | 11.09.2018, 10:44:40
Sitemap
Stud.IP
Blog
Impressum

# Motivation


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Motivation

- Programmierausbildung lebt von praktischen Übungen
- Erreichbarkeit
- Sofortiges Feedback
- Anpassbarkeit
- Feedback für Lehrende
- Zeitersparnis
- Aufgabenpool


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Programmieraufgaben in Vips



Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Programmieraufgaben in Vips - Eigenschaften

- Fileupload oder Editor für die Einreichung
- Template im Editor oder als Download
- Bibliotheken und zusätzliche Anhänge zur Aufgabenbeschreibung können dem Lernenden zum Download angeboten werden
- Gewichtung


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Programmieraufgaben in Vips - Einschränkungen

- geplanter Import noch nicht Angeboten
- derzeit nur Programmiersprachen Java


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Programmieraufgaben in Vips - Ausblick

- Ausbau der Editorfunktion (Grading Hints)
- Evaluation weiterer Programmiersprachen (Python, C, ..)
- Anbindung an externe Repositories


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Didaktische Einsatzszenarien



Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Didaktische Einsatzszenarien

- Didaktischer Wert hängt auch immer von der Qualität des Feedbacks ab (John Hattie 2007)
  - Hinweise für den Lösungsweg
  - größter Erfolg, wenn die Aufgabe anspruchsvoll aber nicht komplex ist

- Herausforderungen für die Studierenden
  - Objektorientierung verstehen
  - zusätzlich eine Programmiersprache lernen
  - grundlegende Programmierfähigkeiten erlernen


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Didaktische Einsatzszenarien

Einführende Aufgaben

- automatische Systeme Code-Schnipsel zum Vervollständigen
- keine kompletten Programme notwendig
- Lückentexte
- Exceptions können als Statements für Lernende in Feedback umgewandelt werden

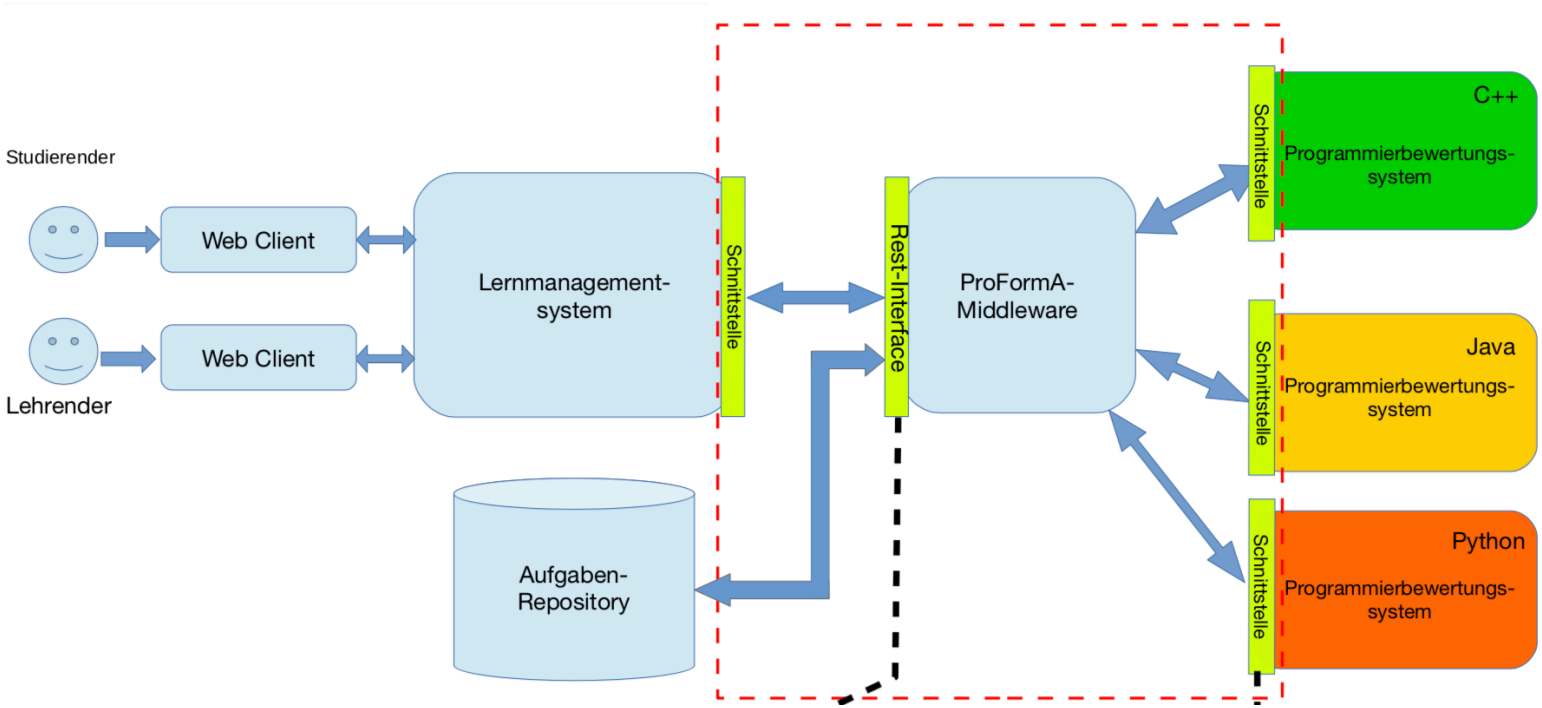
Fortgeschrittene Übungsaufgaben:

- Studierende motivieren → nicht nur die Tests zu erfüllen
- teilweise Testfälle veröffentlichen
- Fehlermeldungen von Compiler, Unittest ungefiltert zurückgeben


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Tools und Funktionen

# Tools und Funktionen






Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Lernmanagementsysteme

- Kursverwaltung (Inhaltsverwaltung, Kurse)
- verschiedene Kommunikationsmöglichkeiten zwischen Lehrenden und Studierenden (Forum / Chat / Nachrichten / Kalender)
- Unterstützung gängiger Aufgabentypen (Multiple Choice, Lückentext)
- Verwaltung von Lernobjekten, Aufgaben, Tests

- Beispiele sind Systeme wie Moodle, Stud.IP, ILIAS und LON-CAPA

# Programmierbewertungssysteme

- automatisierte Bewertung von Einreichungen
- Sicherheit in der Ausführung der Einreichung für das ausführende System
- angepasste Tests an die jeweiligen Aufgaben bzw. Sprachen
- Unterstützung von Werkzeugen in der Softwareentwicklung wie Kompiler, Unittests, Versionierung und Style-Tester

Beispiele sind Systeme wie JACK, Praktomat, Web-CAT oder Graja

# Middleware

- Komplexität und Implementierungsaufwand der Schnittstellen für das LMS und die jeweiligen Bewertungssysteme werden durch die Middleware so gering wie möglich gehalten
- zentrale Aufgaben sind die Verwaltung der angeschlossenen Systeme
- Übersetzung der jeweiligen Antwortformate ineinander

# Austauschformat



Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Austauschformat

Warum Aufgabenaustausch?

- Wiederverwendung/Verbreitung „guter“ Aufgaben
- weniger Arbeitsaufwand beim Erstellen von Aufgaben
- Förderung einer stärkeren universitätsübergreifenden Vernetzung


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Grenzen der Werkzeuge

# Grenzen der Werkzeuge

- nicht alles kann (vollständig) automatisiert getestet/überprüft bzw. bewertet werden
  - sehr offene Aufgaben (z. B. Freitextaufgaben)
  - Stil → sinnvolle Variablennamen
- zusätzliche manuelle Kontrolle durch Tutor*innen / Dozierende bei manchen Aufgabetypen und bei Plagiatstests weiterhin notwendig / sinnvoll


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Ausblick

# ProFormA - Ausblick

- Repository
- Stärkere Integration der Tools in Learning Management Systeme
- Parametrisierung / Randomisierung von Aufgaben
- Peer Review
- Feedbackverbesserung


Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Buchempfehlung

Automatisierte Bewertung in der Programmierausbildung

(Herausgeber)
Oliver J. Bott , Peter Fricke, Uta Priss, Michael Striewe

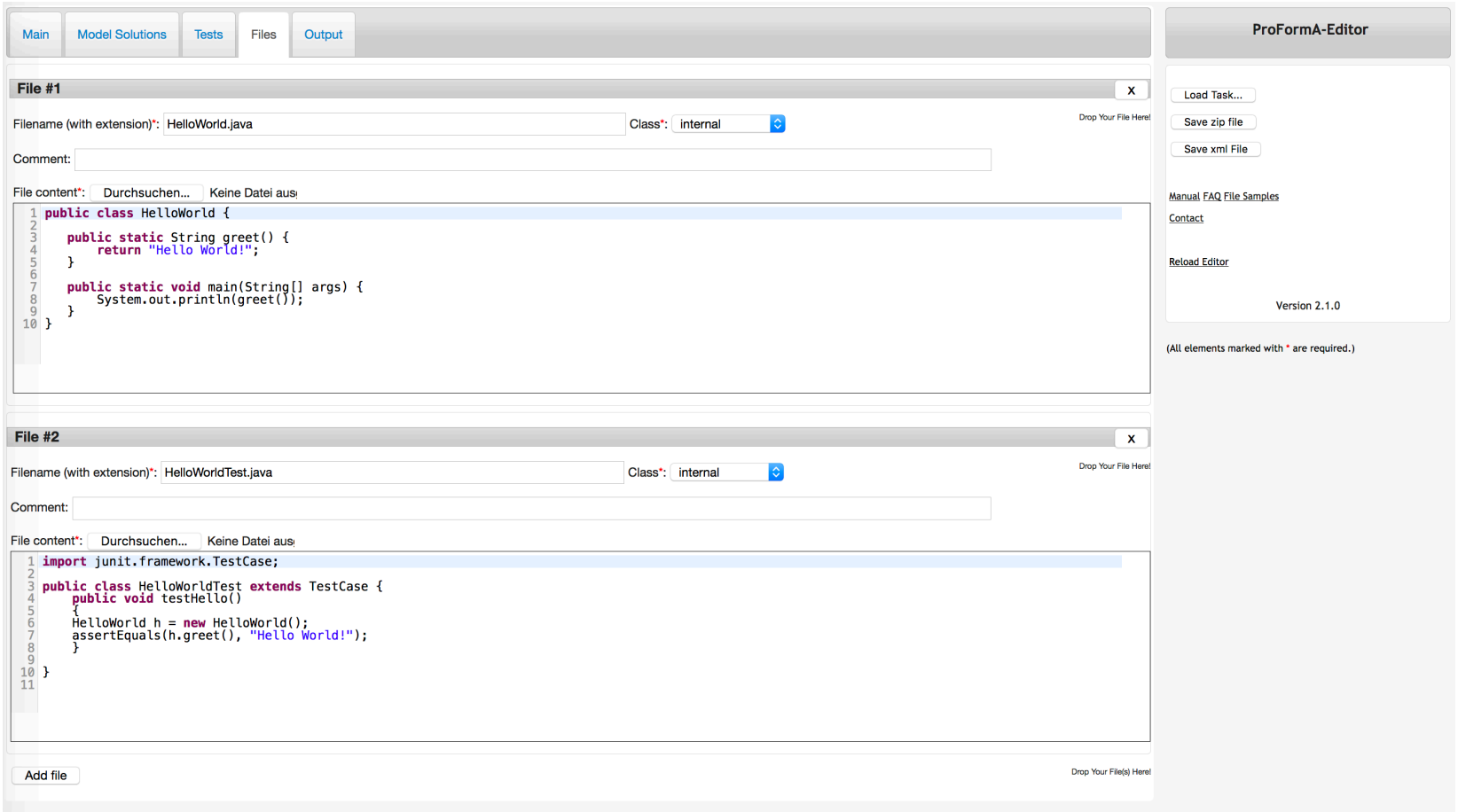
Open Access

https://www.waxmann.com/automatisiertebewertung/

# Vielen Dank für die Aufmerksamkeit!



Oliver Rod ol.rod@ostfalia.de
Dominik Feldschnieder dofeldsc@uos.de

# Appendex

# Austauschformat Editor