

Welche inhaltliche Unterstützung ist für Studierende beim Programmieren in einem Remote-Labor hilfreich?

Anja Hawlitschek¹, André Dietrich² und Sebastian Zug³

Abstract: Der webbasierte Zugriff auf reale Experimente gibt dem Lernenden die Möglichkeit, Aufgaben zu beliebigen Zeitpunkten zu realisieren. Gleichzeitig muss das System aber auch die fehlende unmittelbare Interaktion mit einem Lehrenden kompensieren. Folglich sollte das Aufgabendesign und die Umgebung den individuellen Unterstützungsbedarf adressieren. Das Paper evaluiert hierzu den Einfluss unterschiedliche Detailgrade einer Aufgabenstellung und Unterstützungsmethodik.

Keywords: Remote-Labor, Nutzerspezifische Unterstützung

1 Einleitung

Eine didaktische Herausforderung bei Remote-Laboren besteht darin, Studierenden die notwendige inhaltliche Anleitung und Unterstützung zu bieten. Mit der fehlenden Möglichkeit der synchronen Kommunikation mit Lehrenden können inhaltliche Probleme nicht zeitnah besprochen werden. Die Unterstützung des Lernenden (Anleitungen, Hilfestellungen) sollte deshalb beim Design der Lerninhalte mitgedacht werden, um Frustration und Dropout zu vermeiden. Zu detaillierte Ausführungen können allerdings dazu führen, dass Lernende sich weniger kognitiv engagieren und in geringem Maße eigene Erklärungen für kognitive Herausforderungen generieren [RN13]. Zudem sinkt ggf. die intrinsische Motivation [RJ06]. Um diese didaktische Herausforderung im Kontext des Lernens zu untersuchen, haben wir in zwei experimentellen Studien⁴ zwei Formen von Anleitung für eine Informatikübung implementiert und die Wirkungen auf die Motivation und den Programmier- bzw. Lernerfolg untersucht.

2 Studien

Die Studierenden müssen in den Selbstlernphasen unterschiedliche Aufgaben mit kleinen Robotersystemen im Remote-Labor lösen, die auf Webseiten im System beschrieben werden. In Studie 1 (N = 58) integrierten wir für die Experimentalgruppe (N = 29) im Rahmen einer Aufgabe zusätzlich sogenannte Prompts, d.h. Hinweise welche die

¹ Hochschule Magdeburg-Stendal, Breitscheidstraße 2, 39114 Magdeburg, anja.hawlitschek@hs-magdeburg.de

² Otto-von-Guericke Universität, Universitätsplatz 2, 39106 Magdeburg, andre.dietrich@ovgu.de

³ TU Bergakademie Freiberg, Bernhard-von-Cotta-Straße 2, 09596 Freiberg, sebastian.zug@informatik.tu-freiberg.de

⁴ Im Rahmen des Projekts "Industrial-eLab", gefördert durch das BMBF (Fördernummer 16DHL1034), siehe Projektwebseite <http://www.elab.ovgu.de>

Lernenden auf relevante Fakten, Prozeduren und Ansätze aufmerksam machen sollen [Th09]. So sollen typische Programmierfehler vermieden oder auf effiziente Programmierwege aufmerksam gemacht werden. In der zweiten Studie im Rahmen einer 90 minütigen Lehrveranstaltung (N = 27) fokussierten wir auf den Programmierprozess. Schwer verständliche Compiler-Fehler-Meldungen werden als ein Grund für hohe Dropout-Raten und geringe Erfolge in Programmierkursen angesehen [FZJ04]. Aus diesem Grund implementierten wir ausführlichere Compiler-Fehler-Meldungen, die der Experimentalgruppe (N = 13) zur Verfügung standen.

3 Diskussion der Ergebnisse

ANOVAs ergeben, dass Prompts in Studie 1 keine signifikanten Wirkungen auf die Performanz beim Programmieren ($F(1,54) = 0.18, p = .67, \eta^2 = .00$) und die Abschlussnote ($F(1,26) = .36, p = .55, \eta^2 = .01$) sowie einen negativen Effekt auf die selbsteingeschätzte intrinsische Motivation ($F(1,37) = 4.22, p = .04, \eta^2 = .11$) haben. Dieses Ergebnis deutet an, dass mehr Anleitung nicht immer zu mehr Lernerfolg führt, sondern auch eine motivationsmindernde Einschränkung der Lernenden, im Versuch selbstständig Probleme zu lösen, bedeuten kann. Die ausführlicheren Compiler-Fehler-Meldungen in Studie 2 dagegen führten zu signifikant besseren Programmiererfolgen ($F(1,26) = 6.68, p = .02, \eta^2 = .21$), ohne die intrinsische Motivation zu beeinträchtigen ($F(1,26) = 1.45, p = .24, \eta^2 = .05$), und scheinen daher ein für die Praxis geeignetes Mittel für die Unterstützung von Lernenden in Programmierprozessen – auch in Remote Laboren – zu sein. Die Ergebnisse beider Studien zeigen einmal mehr, wie herausfordernd es ist, das richtige Maß und die richtige Form von instruktionaler Anleitung für Studierende zu finden.

Literaturverzeichnis

- [FZJ04] Flowers, T.; Carver, C. A. & Jackson, J.: Empowering students and building confidence in novice programmers through gauntlet. 34th Annual Frontiers in Education. FIE, 2004.
- [RJ06] Reeve, J. & Jang, H.: What teachers say and do to support students' autonomy during a learning activity. *Journal of Educational Psychology*, 98(1), 209-218, 2006.
- [RN13] Richey, E. & Nokes-Malach, T. J.: How much is too much? Learning and motivation effects of adding instructional explanations to worked examples. *Learning and Instruction*, 25, 104-124, 2013.
- [Th09] Thillmann, H.; Künsting, J.; Wirth, J. & Leutner, D.: Is it merely a question of 'what' to prompt or also 'when' to prompt? - The role of point of presentation time of prompts in self-regulated learning. *Zeitschrift für Pädagogische Psychologie*, 23(2), 105-115, 2009.