

Feedback in der Softwaretechnikausbildung: Automatische Analyse von studentischen Softwareprojekten für Lehrkräfte

Thema:

Feedback in der Softwaretechnikausbildung: Automatische Analyse von studentischen Softwareprojekten für Lehrkräfte

Art:

MA

BetreuerIn:

Alexander Bazo

BearbeiterIn:

Felix Riedl

ErstgutachterIn:

N.N.

ZweitgutachterIn:

N.N.

Status:

abgeschlossen

Stichworte:

Software Engineering Education

angelegt:

2019-11-28

Antrittsvortrag:

2020-02-18

Abschlussvortrag:

2020-10-20

Hintergrund

Um die Qualität von Team- und Projektarbeiten in der Softwaretechnik-Ausbildung adäquat bewerten zu können reicht es oft nicht, ausschließlich das eingereichte Artefakt (z.B. Projekt Quellcode) zu begutachten. Neben der Qualität des finalen Artefakts spielen zum Beispiel auch durchgängige Prozessqualität, Koordination und Zeitmanagement eine wichtige Rolle (Kay et al., 2010). Zudem ist es notwendig Einzelleistungen zu identifizieren und deren Qualität hinreichend exakt einschätzen zu können (Jones, 2010).

Die mittlerweile weitverbreitete Verwendung von Systemen zur Versionskontrolle in studentischen Projekten kann dabei helfen, dieser Problematik beizukommen. Sie gewährt Lehrpersonen kontinuierlichen und unmittelbaren Einblick in Planungs- und Entwicklungsprozesse. Auch ermöglicht sie im Zweifel eine eindeutige Zuordnung von Contributions im Quellcode und eine frühzeitige Identifikation von Problemen während der Bearbeitung. Eine Einsicht der Rohdaten aus Versionskontrolle und PM-Tools kann hierfür bereits einen Überblick über die laufende Arbeit an einem Projekt bieten (Jones, 2010), ist jedoch auch aufwendig und bietet nur wenig detaillierte Einblicke.

Liu, Y., & Stroulia, E. (2003, November). Reverse Engineering the Process of Small Novice Software Teams. In WCRE (Vol. 3, p. 102).

Sprint, G., & Conci, J. (2019). Mining GitHub Classroom Commit Behavior in Elective and Introductory Computer Science Courses. The Journal of Computing Sciences in Colleges, 76.

Rastogi, A., Gupta, A., & Sureka, A. (2013, February). Samiksha: mining issue tracking system for contribution and performance assessment. In Proceedings of the 6th India Software Engineering Conference (pp. 13-22). ACM.

Kazerouni, A. M., Edwards, S. H., Hall, T. S., & Shaffer, C. A. (2017, June). DevEventTracker: Tracking Development Events to Assess Incremental Development and Procrastination. In Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education (pp. 104-109). ACM.

Eyolfson, J., Tan, L., & Lam, P. (2011, May). Do time of day and developer experience affect commit bugginess?. In Proceedings of the 8th Working Conference on Mining Software Repositories (pp. 153-162). ACM.

Wittmann, M. R. A., Bower, M., & Kavakli-Thorne, M. (2011, June). Using the SCORE software package to analyse novice computer graphics programming. In Proceedings of the 16th annual joint conference on Innovation and technology in computer science education (pp. 118-122). ACM.

Poncin, W., Serebrenik, A., & van den Brand, M. (2011, October). Mining student capstone projects with FRASR and ProM. In Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion (pp. 87-96). ACM.

Mittal, M., & Sureka, A. (2014, May). Process mining software repositories from student projects in an undergraduate software engineering course. In Companion Proceedings of the 36th International Conference on Software Engineering (pp. 344-353). ACM.

From:
<https://wiki.mi.ur.de/> - MI Wiki

Permanent link:
https://wiki.mi.ur.de/arbeiten/feedback_in_der_softwaretechnikausbildung_automatische_analyse_von_studentischen_softwareprojekten_fuer_lehrkraefte

Last update: 15.10.2020 10:53

