

Novizen-zentriertes Re-Design eines Code-Review-Prozesses

Thema:

Novizen-zentriertes Re-Design eines Code-Review-Prozesses

Art:

BA

BetreuerIn:

Alexander Bazo

BearbeiterIn:

Fabian Zeiher

Status:

in Bearbeitung

angelegt:

2020-04-03

Antrittsvortrag:

2020-09-28

Hintergrund

Peer-Code-Review(oder einfach Code-Review) ist der Prozess einer Code-Analyse in der festgestellt werden soll, ob ein Code-Artefakt qualitativ ausreichend ist, um in den Produktionscode integriert zu werden (Bavota & Russo, 2015, S. 81). Code Reviews sind eine in der Industrie übliche und in der Wissenschaft anerkannte Methode zur Sicherung von Code Qualität(MacLeod et al., 2018, S. 34). Darüber hinaus helfen Code Reviews in einem Team dabei, alle Mitglieder über den aktuellen Entwicklungsstand zu informieren. Außerdem profitieren Junior-Entwickler vom Wissen der erfahreneren Entwickler, indem sie sich mit deren Code auseinandersetzen(Bacchelli & Bird, 2013, S. 717). Damit können Code-Reviews auch als didaktisches Hilfsmittel für den Erwerb von sowohl „Hard-Skills“ als auch „Soft-Skills“ in der Lehre eingesetzt werden(Rivera et al., 2020, S. 103). Programmernovizen lernen vom gegenseitigen Austausch und der gezielten Auseinandersetzung mit Programmcode den sie nicht selbst geschrieben haben, sie entdecken schlechte Funktionen und Features und heben besonders gute Eigenschaften hervor(Hyyrynen et al., 2010, S. 94). Zudem erfahren sie direkt wie sich Code-Formatierung und Sauberkeit auf die Lesbarkeit und Verständlichkeit des Programmcodes auswirken(Rivera et al., 2020, S. 103).

Allerdings benötigt die Methode, für den Einsatz als effektives Lehrmittel, viel Kompetenz bei den Studierenden. Nach Trytten (2005, S. 456) ist ein ordentlich durchgeführtes Code-Review eine „Evaluation“ und damit eines der höchsten und anspruchsvollsten Lernziele nach „Bloom’s Taxonomy“ (Krathwohl, 2002). Umalso in vollem Umfang von der Methodik zu profitieren, sollten die Studierenden durch die richtigen Mittel unterstützt werden.

In den letzten Jahren wurde der Einsatz von Online Tools, speziell für den Code-Review-Prozess im universitären Kontext, viel diskutiert (Sondergaard, 2009, S. 31). Auch am Lehrstuhl werden Code-Reviews bereits im Kurs Multimedia Engineering versuchsweise eingesetzt. Allerdings hat sich gezeigt, dass die für die Industrie konzipierten Plattformen wenig Funktionen bieten die Programmernovizen an das Thema Code-Review heranzuführen können und sich teilweise nur

aufwendig in die Lehre integrieren lassen. Deshalb wurde bereits 2019 von Alicia Fritsche prototypisch eine Code-Review-Plattform gezielt für Studierende entwickelt. Allerdings konnte dabei auf die Erstellung der Reviews an sich nur untergeordnet Wert gelegt werden und der Fokus lag eher auf der Implementierung der Plattform als solche.

Zielsetzung der Arbeit

Ziel der Arbeit ist es, im Kontext der bestehenden Code-Review-Plattform, einen Code-Review-Editor zu implementieren, der Programmiernovizen - also die Studierenden - bei der Erstellung von Reviews für ihre Kommilitonen unterstützt und begleitet. Die spezifischen Anforderungen der Programmiernovizen und der Nutzungskontext, als didaktisches Hilfsmittel innerhalb der Programmierausbildung, sollen berücksichtigt werden. Dabei steht die Frage „Wie sollte ein Novizen zentriertes Tool für die Erstellung von Code-Reviews gestaltet sein?“ im Mittelpunkt.

Konkrete Aufgaben

- Anforderungserhebung bei den Studierenden, die im aktuellen MME Kurs Reviews für ihre Kommilitonen erstellen (1 Woche)
- Konzeption des Tools nach den Ergebnissen der Anforderungserhebung (1 Woche)
- Implementierung des Code-Review-Tools innerhalb der Plattform (3 Wochen)
- Test des Code Review Editors, beispielsweise im Vergleich zur Ursprünglichen Plattform (2 Wochen)
- Finalisierung der schriftlichen Arbeit (1 Woche)

Erwartete Vorkenntnisse

- Vue.js, javascript, Firebase, HTML, CSS, User Centered Design und Developement

Weiterführende Quellen

Bacchelli, A., & Bird, C. (2013). Expectations, Outcomes, and Challenges of Modern Code Review. In D. Notkin (Ed.), 35th International Conference on Software Engineering (ICSE), 2013: 18 - 26 May 2013, San Francisco, California, USA ; proceedings (pp. 712–721). Piscataway, NJ: IEEE. Retrieved from <https://dl.acm.org/doi/10.5555/2486788.2486882>

Fagan, M. E. (2002). A History of Software Inspections. In M. Broy & E. Denert (Eds.), Software Pioneers: Contributions to Software Engineering (pp. 562–573). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-59412-0_34

Fritsche, A. (2019). Design und Implementierung einer prototypischen Code-Review-Plattform für peer-based Feedback im Rahmen der Softwaretechnikausbildung (Bachelorarbeit). Universität Regensburg, Regensburg.

Kononenko, O., Baysal, O., & Godfrey, M. W. (2016). Code Review Quality: How Developers See It. In 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE).

MacLeod, L., Greiler, M., Storey, M.-A., Bird, C., & Czerwonka, J. (2018). Code Reviewing in the Trenches: Challenges and Best Practices. IEEE Software, 35(4), 34–42.

<https://doi.org/10.1109/MS.2017.265100500>

Rigby, P. C., Cleary, B., Painchaud, F., Storey, M.-A., & German, D. (2012). Contemporary Peer Review in Action: Lessons from Open Source Development. IEEE Software, 29(6), 56–61.

<https://doi.org/10.1109/MS.2012.24>

Rivera, V., Aslam, H., Naumchev, A., Carvalho, D. de, Khazeev, M., & Mazzara, M. (2020). Towards Code Review Guideline in a Classroom. In J.-M. Bruel, A. Capozucca, M. Mazzara, B. Meyer, A. Naumchev, & A. Sadovykh (Eds.), *Frontiers in Software Engineering Education* (pp. 88–105). Cham: Springer International Publishing.

Sondergaard, H. (2009). Learning from and with peers. ACM SIGCSE Bulletin, 41(3), 31.

<https://doi.org/10.1145/1595496.1562893>

From:

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

Permanent link:

https://wiki.mi.ur.de/arbeiten/novizen-zentriertes_re-design_eines_code-review-prozesses?rev=1603716632

Last update: **26.10.2020 12:50**

