

# Quality Experience: A Grounded Theory of Successful Agile Projects Without Dedicated Testers

Lutz Prechelt,<sup>1</sup> Holger Schmeisky,<sup>2</sup> Franz Zieris<sup>3</sup>

This is an extended abstract of the paper with the same title [PSZ16] which was presented at the 38th International Conference on Software Engineering (2016).

**Keywords:** agile software development; quality assurance; industrial case study; grounded theory

## 1 Background, Context, and Research Method

Some agile development teams have team members or partner teams who are exclusively considered testers, whereas other teams do not even a separate tester *role*. Agile development methods emphasize personal communication and shun departments in favor of multidisciplinary teams. They have no uniform attitude towards an explicit tester role, let alone to the need for separate testing personnel: Scrum only talks about *Developers* [SS13], XP defines a *Tester* role (which is not necessarily to be assumed by non-developers) [BA04], and Kanban is agnostic on the issue of separate testers [An10].

In our exploratory, holistic, multiple-case study [Yi03], we wanted to find out how quality is assured in agile teams that do not employ separate testers and what the advantages and disadvantages of not employing separate testers are. We selected three agile teams (one with separate testers, two without) from two companies, all of which have a similar context: In-house development of large web portal that has millions of users, which are partially paying for particular services; any individual customer accounts for only a small part of the revenue stream; new versions of the software can (in principle) be deployed immediately.

We collected data through direct observation and semi-structured interviews with developers, product owners, scrum masters, testers, and higher-level managers in multiple rounds. We analyzed the transcripts using the Grounded Theory Methodology [SC90].

---

<sup>1</sup> Freie Universität Berlin, Institut für Informatik, Takustr. 9, 14195 Berlin, Deutschland prechelt@inf.fu-berlin.de

<sup>2</sup> Zalando SE, Berlin, Deutschland holger.schmeisky@zalando.de

<sup>3</sup> Freie Universität Berlin, Institut für Informatik, Takustr. 9, 14195 Berlin, Deutschland zieris@inf.fu-berlin.de

## 2 Results

All three agile teams were able to produce high-quality software – but the ways of working differ drastically along a dimension we call **Quality Experience**. A team with a strong Quality Experience (1) *feels fully responsible* for the general quality of their software, (2) has a *tight feedback loop* concerning this quality, and (3) *rapidly repairs deficiencies* when they occur. The two teams without separate testers have a strong Quality Experience, whereas the third one has a comparatively weak one.

There are several factors that influence the elements of the *Quality Experience* of a team, some of which relate directly to software engineering, others are sociological or psychological in nature: Having a *modularized architecture* that sufficiently decouples the work of one team from that of another is a fundamental precondition for having a strong Quality Experience. Given that precondition, management may then decide to hand over complete *control over deployment* to the development team. From here, it is only a small step to the developers both being *held responsible* and actually *feeling responsible*. *Automated tests* and *automated deployment* then facilitate *frequent deployments*. The tight feedback loop, which is (due to the lack of separate testers) characterized by *direct* (not intermediated), *quick* (available early on), and *realistic* (from a non-artificial setting) feedback, leads to a *high motivation* and ultimately *rapid repairs of defects*.

## 3 Conclusions

Agile teams without separate testers can achieve high quality through a strong Quality Experience. If the preconditions are fulfilled by the existing architecture, suitable domain, and willing management, separate testers would only get in the way of the developers learning what actually works from the end-users in the field.

## References

- [An10] Anderson, David J.: Kanban: Successful Evolutionary Change for Your Technology Business. Blue Hole Press, 2010.
- [BA04] Beck, Kent; Andres, Cynthia: Extreme Programming Explained: Embrace Change, Second Edition. Addison-Wesley Professional, 2004.
- [PSZ16] Prechelt, Lutz; Schmeisky, Holger; Zieris, Franz: Quality Experience: A Grounded Theory of Successful Agile Projects Without Dedicated Testers. In: Proc. 38th Int'l. Conf. on Software Engineering. ICSE '16, ACM, New York, NY, USA, pp. 1017–1027, 2016.
- [SC90] Strauss, Anselm L.; Corbin, Juliet M.: Basics of Qualitative Research: Grounded Theory Procedures and Techniques. SAGE, 1990.
- [SS13] Schwaber, Ken; Sutherland, Jeff: The Scrum Guide. Technical report, scrum.org, July 2013.
- [Yi03] Yin, Robert K.: Case Study Research: Design and Methods. Sage, 2003.