

Handling Requirements Changes in Agile Software Development Projects: Preliminary Results from an Interview-based Industrial Study

Kashumi Madampe*, Rashina Hoda, John Grundy

HumanISE Lab, Monash University, Australia. <https://www.monash.edu/it/humanise-lab>, *kashumi.madampe@monash.edu



HUMANISE



MONASH University

MOTIVATION

“People always fear change. People feared electricity when it was invented, didn't they?” - Bill Gates

- Agile software development is carried out iteratively (in sprints) and incrementally, enabling to create and respond to changes [1].
- Among numerous changes in software development, requirements changes are crucial and frequent.
- Different agile environments treat requirements changes in different ways which reveal the existence of variations in strategies of handling requirements changes.
- Ultimately, agile it easy to understand, but difficult to master [2].

OBJECTIVES

The objectives of this study, in agile software development contexts, are to:

- Understand how software practitioners respond to requirements changes
- Identify how the handling of requirements changes can be improved

OUR APPROACH

Grounded Theory (GT) is used in studies where social interactions and human behaviors are studied [3], making it appropriate for our study. An overview of our approach is given in Fig. 1.

- **Participants:** 10 agile practitioners (n=8 Nz; n=2 Au)
- **Participant recruitment via:** LinkedIn, Twitter, Meetups, Facebook
- **Data collection:** 50-60 min. semi-structured interviews (n=7 face-to-face; n=3 Skype and Zoom) + pre-interview questionnaire
- **Data analysis:** Open coding → Constant Comparison → Theory

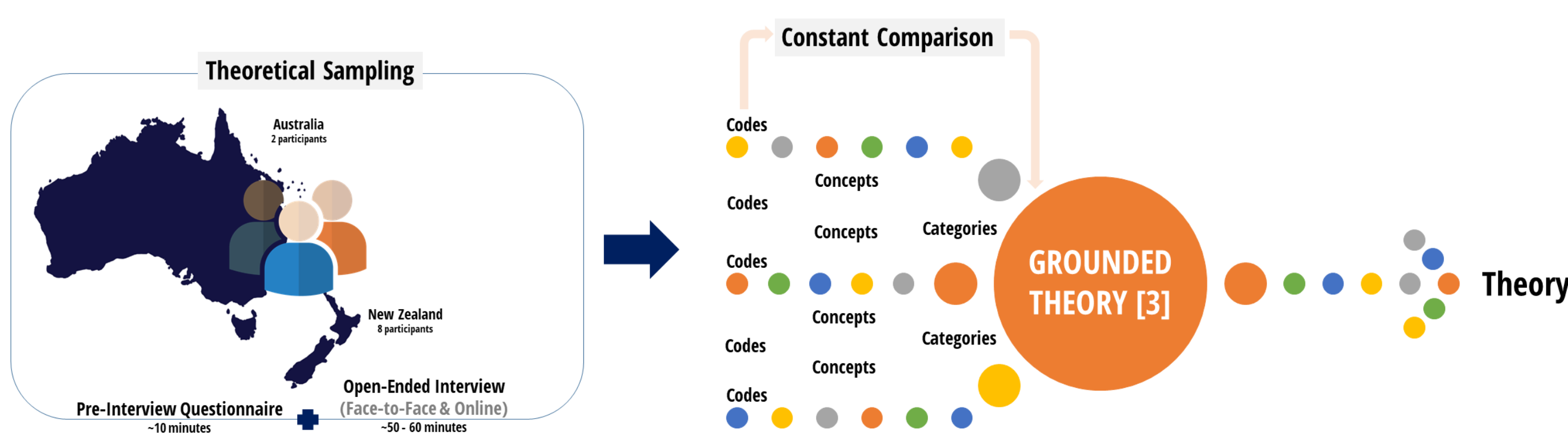


Figure 1:Applying Grounded Theory

FUTURE WORK

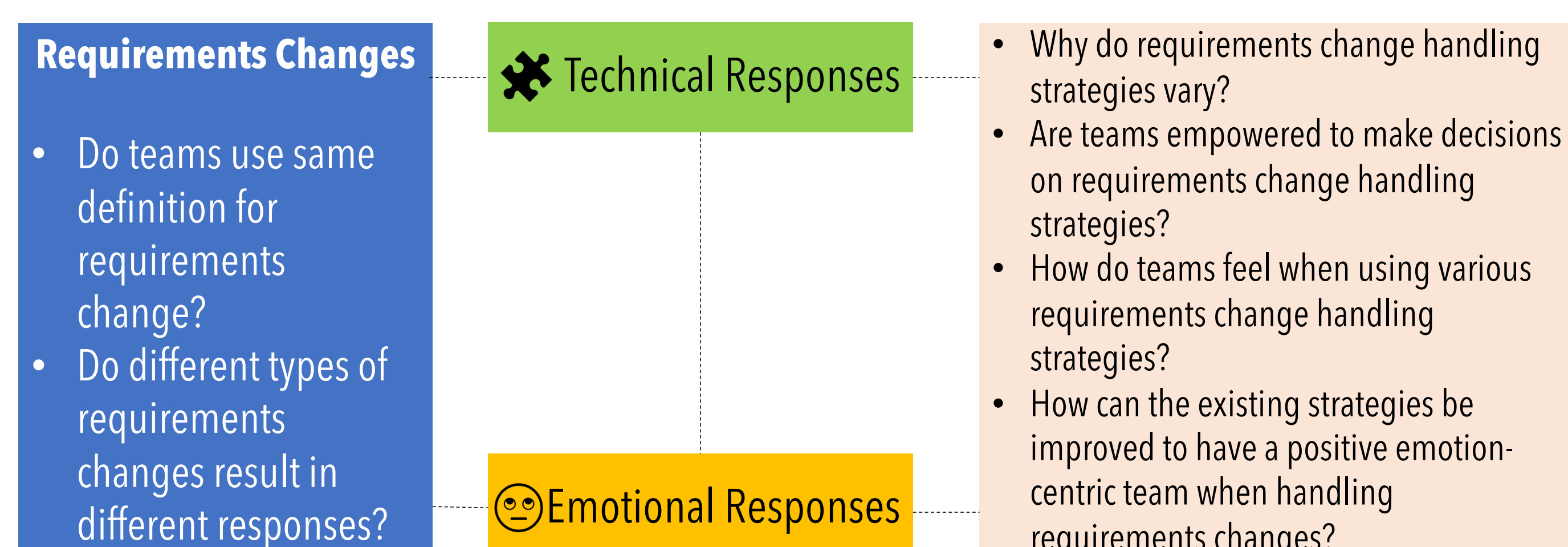


Figure 2:Future Work

OUR FINDINGS



Figure 3:Requirements Change Handling Strategies

Requirements change handling strategies vary depending on contexts and with the requirements changes as they occur as shown in Fig. 3.

Identified Requirements Change Handling Strategies:

- 1 **Business As Usual: Refine Product Backlog » Develop » Deliver (Most common strategy)**
40% of the participants mentioned that they refine the product backlog, develop, and deliver as usual when they receive a requirements change.
- 2 **Delayed Restart: Add to Relevant Source Backlog » Refine Product Backlog » Develop » Deliver**
30% of the participants stated that they maintain different backlogs according to the source (eg: marketing, support) and follow the normal process of product backlog refinement followed by development and delivery of the requirement change.
- 3 **Restart: Cancel Sprint » Refine Product Backlog » Develop » Deliver**
In the case of receiving the requirement change within a running sprint, 30% of the participants reported that they cancel the sprint and move to product backlog refinement to develop and deliver the requirement change.
- 4 **Rehaul: Fill Change Request Form » Refine Product Backlog » Develop » Deliver**
Change request forms are used in traditional software development. As 20% of the participants mentioned that they use change request forms before product backlog refinement when they receive a requirement change, it can be inferred that hybridization of traditional and ASD exists, as preferred by the organization.

References

- [1] Beck et al. Manifesto for Agile Software Development, 2001. URL: <https://agilemanifesto.org/>.
- [2] P. Deemer, G. Benefield, C. Larman, and B. Vodd, "The Scrum Primer", InfoQ, 2021.
- [3] Barney Glaser and Anslem Strauss. Grounded Theory: The Discovery of Grounded Theory. 1967.