

# Integrating Agile development and UX design: What does it look like in practice?

Jennifer Ferreira  
The Open University  
j.ferreira@open.ac.uk

The literature contains numerous accounts that talk of *integrating* Agile development with something that is considered *not* Agile, or external to Agile, for example, Plan-Driven Methods [1], security considerations [2], and stage-gate managed product development [3]. Integrating Agile development and user experience (UX) design<sup>1</sup> is one such combination that enjoys ongoing debate and discussion about how it can be achieved. In the literature, accounts of combining Agile development with UX design present a number of strategies for practice: from adding specific techniques into the development effort (such as personas [4]), to merging two separate processes into one [5]. All strategies are concerned with coordinating the work of developers and designers such that each can adequately contribute to the software. At face value, these strategies present an unproblematic, harmonious back-and-forth between design and implementation throughout the development effort. However, the accounts hint at practice in realistic settings being more complicated, as having complicating factors, and that there is something more to their intergation than process (e.g. [6]). What does integration mean in terms of the everyday work that developers and designers have to do in order to create software?

Our observations of day-to-day practice in organisational settings uncovered work arrangements, dependencies and mechanisms that made the integration and coordination of work possible. We found that the interactions between Agile developers and UX designers were shaped and sustained by the values and assumptions promoted by the organisation in which the developers and designers were embedded --- values and assumptions about how best to combine the skills of Agile developers and UX designers to create quality software. According to these values and assumptions teams were set up in various ways, such as either having developers and designers sitting together in one room, or apart on separate floors within their separate developer and designer teams. We observed that in the interests of making progress, developers and designers elicit the kind of cooperation they require from one another by means of articulation work. This ranged from an explicit form of articulation work, such as walking around, to more implicit mechanisms that are unobservable but evident in the mutual awareness they have of one another's work. Agile developers and UX designers carried out work independently from each other, but maintaining progress also required focusing on the same part of the software together, synchronously. Their synchronous activities were vital for achieving harmony in their work and deciding what further action was meaningful with respect to the software. Progress, and therefore, the emergent solution, was negotiated between the developers and designers, via decisions incorporating input from both sides.

Integrating Agile development and UX design in practice is neither random, nor a matter of process decided outside of the immediate work situation. Instead, the work of integration is localised and contingent and involves making qualitative judgments about role boundaries and expertise. Our research promotes an understanding of integration that hinges on mutual awareness, self-organisation, negotiating progress, and how achieving these depend in important ways on the work cultures in the organisation --- the very details that *enable* integration, but are glossed over by accounts in the literature.

---

1. UX design can also be replaced with related terms such as Interaction Design, User-Centered Design, Usability Engineering, and others that are not equivalent, but have related aims with respect to understanding the users for which the software is intended, and the design and evaluation of the users' experience with the software.

**References:**

- [1] B. Boehm and R. Turner. Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods. In ICSE '04: Proceedings of the 26th International Conference on Software Engineering, 2004. IEEE Computer Society, Washington, DC, USA, 718-719.
- [2] M. Siponen, R. Baskerville and T. Kuivalainen. Integrating Security into Agile Development Methods. In HICSS '05: Proceedings of the 38th Annual Hawaii International Conference on System Science. doi: 10.1109/HICSS.2005.329
- [3] D. Karlström and P. Runeson. Integrating agile software development into stage-gate managed product development. *Empirical Softw. Engg.* 11, 2 (June 2006), 203-225. DOI=10.1007/s10664-006-6402-8 <http://dx.doi.org/10.1007/s10664-006-6402-8>
- [4] J. Haikara. Usability in Agile software development: Extending the interaction design process with personas approach. In G. Concas, E. Damiani, M. Scotto, and G. Succi, editors, *Agile Processes in Software Engineering and Extreme Programming*, Lecture Notes in Computer Science, pages 153–156. Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007.
- [5] L. Miller. Case study of customer input for a successful product. In *Agile '05: Proceedings of the 2005 Agile Conference*, Denver, CO, USA, July 24–29, pages 225–234, Los Alamitos, CA, USA, 2005. IEEE Computer Society. Doi: 10.1109/ADC.2005.16.
- [6] M. Najafi and L. Toyoshiba. Two case studies of user experience design and Agile development. In *Agile '08: Proceedings of the Agile 2008 Conference*, Toronto, Ontario, Canada, August 4–8, Los Alamitos, CA, USA, 2008. IEEE Computer Society.