

# Eben Tobler

## Agile Software Development and Pair Programming Department of Computer Science, University of New Hampshire, Durham, NH 03824

### Introduction

Agile project management has become the most common development methodology in the software industry. The 17th Annual State of Agile Report found that out of 788 respondents, 71% use Agile methodologies in their software development cycle. Of those that use Agile, 44% said that Agile works very well or somewhat well, while 26% said that Agile does not work that well or not well at all.

Agile is generally effective according to the report, but there are many anecdotal opinions about Agile among software developers. My thesis aims to research Agile methodologies, gain more experience with Agile, and learn about the strengths and weaknesses of Agile.

### What is Agile?

Agile was created in 2001 at a ski resort in Utah by a group of seventeen software engineers. The Agile Manifesto is a short document, consisting of four values and twelve principles that further specify the values. The values are:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Some common Agile methodologies are Scrum, Kanban, and Extreme Programming, though there are many more frameworks that could be considered Agile.

### Methodology

- Pair programming with Menlo developers for 4 hours a week (one half day)
- Individual work on specified cards
- Time estimate and breakdown of cards
- Reflection on time estimate and work done
- Online research
- Interviews

Menlo uses pair programming almost exclusively. They organize their work in a kanban-style board, and sort them by priority. Once pairs have completed a card, they have their work reviewed by another pair, who then approve it and flag it as ready for QA.

Scrum
Kanban
Extrem Progran ming (X





Data from McDowell et al. (2006). Paired programming improved student retention in an intro-level computer science course.

### Agile Frameworks

### Agile Frameworks

	Iteration time	Approach to change	Main focus	Potential challenges
	Sprints (2 weeks to a month)	Keep work during sprints static	Organization, structure, empiricism, improvement	Can feel rigid meetings may lack substance
	Continuous	Respond to change as it happens	Visualization of work, prioritization, flexibility	Undefined deadlines, unclear responsibility
e n- (P)	Frequent releases (1-2 weeks)	Respond to change as it happens	Interacting with customers, teamwork, code quality	Customer participation, locality



Data from Digital.ai (2022). Each respondent could mark multiple frameworks used, resulting in a summed percentage greater than 100%.

### Agile Workflows

# Menlo's Methodology Client Developers

Pair Programming

### Pair Programming

Product

### Benefits of Pair Programming

- Higher code quality
- ➢ Greater confidence in solutions
- Greater learning / more spread of knowledge
- Higher satisfaction and enjoyment
- Stronger team dynamic



### Results



- □ Agile (Scrum in particular) in larger companies tends to go through the motions – meetings and retrospectives not being used for what they were intended for
- □ Agile is not always a good fit for a team if the team is not entirely committed to Agile, it often costs more than it is worth
- Combining Agile methodologies to suit project or team needs is common and worthwhile if done with intent
- Pair programming is often very effective, but can be misunderstood as twice the cost for half the output
- Pairing with Menlo
- □ Multiple times working individually on cards when I made a small error that would have been caught by a teammate
- □ When one of us was unfamiliar with the code files we were working in, the other was often able to explain them
- □ Frequent customer interaction allowed developers to focus on what the customer really wanted

### Conclusions

Creating software involves more than writing software. Agile frameworks tend to work, but they do not automatically solve problems. The frameworks exist to support teams and make their jobs easier, but ultimately, the success of a team relies on the team itself.

### Acknowledgements

Thank you to Dr. Jon Niehof for providing me with this opportunity and for guiding me in this project.

Thank you to all the developers at Menlo Innovations for sharing their expertise and for their continued kindness.

### References

Digital.ai. (2022). 16th Annual State of Agile Report. Digital.ai. (2023). 17th Annual State of Agile Report.

McDowell, C., Werner, L., Bullock, H. E., & Fernald, J. (2006). Pair programming improves student retention, confidence, and program quality. Communications of the ACM, 49(8), 90-95. https://doi.org/10.1145/1145287.1145293

Shore, J., & Warden, S. (2021). The Art of Agile Development (2nd ed.). O'Reilly Media. Verwijs, C. (2023, March 6). In-depth: The costs and benefits of pair programming. The Liberators. https://medium.com/the-liberators/in-depth-the-costs-and-benefits-ofpair-programming-b4b54b27c6ff







