Is Agile only suitable for software development? I built a playhouse in an Agile manner.

A commonly heard assumption is that Agile is mainly fit for software development. There are plenty of successful examples[1] of use of Agile outside of software development, despite the fact that the Agile Manifesto[2], written in 2001, (to this day) only mentions software development.

I was recently drawn to this misconception again while building a playhouse during the holidays. My in-laws were visiting from the States and the plan was for my wife's stepdad and I to build my kids a playhouse.

This project started with a fixed time, due to the duration of the stay. A (somewhat) fixed budget and a flexible scope.

The goal was defined: Build an awesome playhouse for two kids. The client would be my wife; she would make all executive decisions. The end user was clear, as my 3-year-old son was eagerly jumping with anticipation from the first mention of the project. (Agile principle 4)

For reference, all Agile principles are posted below this blog

Getting to work

We started with aligning expectations. Which (I can tell you), an American's notion of an urban playhouse can vary from an European's idea. We found a design online, roughly meeting both worlds' criteria. Now that we had a global lay-out, we went out into the yard and shaped the outline with tape on a designated spot in between the grass and the shed, adjusting the footprint of the house until all parties agreed.

Finding the desired material was sometimes difficult. The items you would use on a build in the States can look, feel, and cost differently in the Netherlands. Another difference between California (where my in-laws are from) and the Netherlands are the weather conditions. After a day of preparation, we altered the plan and chose to use different, more weather resistant, material. For example, we decided to add a platform/floor because Dutch weather is often cold and wet (even when it is not raining). (Agile principle 11)

Finally, we could begin building. While setting up the frame, we made some minor adjustments. We felt, as building crew, we had the mandate to move the ridge

[1] https://www.techrepublic.com/article/how-to-apply-agile-practices-with-your-non-tech-team-or-business/
[2] https://agilemanifesto.org/

of the roof in order to use some left-over beam rather than cutting into a new beam. Altering the incline of the roof slightly did not matter to the end user but helped us tremendously. (Agile principle 2 and 5)

The original design we used as a starting point for inspiration had a loft inside. Because it had rained, we fell behind on our schedule and decided to leave the loft out of our build. It can be added later and was not a need-to-have for anyone. When our youngest, who is now 8 months old, is old enough to be trusted with a steep ladder we can always reconsider this decision. For now, we left it out. (Agile principle 10)

It happened to be a rainy day when I was hammering shingles on the roof. I noticed what an incredible downpour this roof created, more than I had imagined possible. This brought me to an additional idea of adding gutters and a rain barrel to the house. Not only would I protect the not yet painted walls of the house, but the thought of using the rain water to water the plants in summer in a more eco-friendly way also sparked joy for me. The client (my wife) agreed and we added this feature to the build, ironically introducing the concept of 'waterfall' in this project.

At the end of every build day, when all power saws, sharp nails and other non-kid-friendly stuff had been put away, our customer (the 3-year-old) would come and inspect the product. He loved sticking his head through the window and pretending he was in his own little shop, let's leave this window open so that he can keep using it this way! (Agile principle 1 and 3)

With my family's visit to the Netherlands is coming to an end, it hadn't been dry for days, so we need to paint at a later moment. However, the playhouse has been in use by my son and his friends and cousins for a while (Agile principle 7). We constantly add cool things to it now. A little doormat, the actual door, and a string of lights on the inside are all things that add value to the house. Maybe in spring we'll add some flowerbeds or a mailbox. In the years to come we will probably keep adjusting the add-ons to the likes of our kids. Will it ever be truly 'done'?

Agile or not?

So, is this really an Agile build of a playhouse? Yes, I just explained how it fits the principles. No, I did not consciously set this up as an Agile build. I didn't label it Agile until I decided to write this blog in hindsight. But, does it really matter? The label Agile or not (or Scrum or not) works unintentionally restrictive and excluding.

The Agile principles are not exclusively limited to Agile, but together describe a mindset and approach rather than a framework. These principles can be used anywhere and by everyone, whether you realize it or not.

There are Agile concepts and principles to be recognized in the most Waterfall-project you can think of. Perhaps what makes something Agile is the focus and strive to adhere to these principles. This can, and will, reach far beyond software development.

About the author: Martijn van de Haterd (34) is an Agile Consultant with Improve Quality Services. He lives with his wife and two sons in Amersfoort.

For reference, an overview of the 12 Agile principles:

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.