



AGILE WORKSHOP

Art Nooijen



OCTOBER 9, 2022
FONTYS

Contents

Introduction	2
Manifesto.....	2
Values Examples.....	3
Principles Examples	3
Bibliography	6

Introduction

This Document Contains information about my preparation for a workshop about Agile. The workshop was given by five people:

- Lavrijsen, Ward W.J.J.
- Todea, Alexandra A.K.
- Enden, Dirk D. van der
- Grootel, Marc M.H. van
- **Nooijen, Art A.T.G.**

My contribution to the Workshop was explaining the Agile Manifesto. With the purpose of giving examples for my fellow students.

Manifesto

Background Info from: (history, 2001)

On February 11-13, 2001, at The Lodge at Snowbird ski resort in the Wasatch mountains of Utah, seventeen people met to talk, ski, relax, and try to find common ground—and of course, to eat. What emerged was the Agile ‘Software Development’ Manifesto. Representatives from Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming, and others sympathetic to the need for an alternative to documentation driven, heavyweight software development processes convened.

Now, a bigger gathering of organizational anarchists would be hard to find, so what emerged from this meeting was symbolic—a Manifesto for Agile Software Development—signed by all participants. The only concern with the term agile came from Martin Fowler (a Brit for those who don’t know him) who allowed that most Americans didn’t know how to pronounce the word ‘agile’.

Alistair Cockburn’s initial concerns reflected the early thoughts of many participants. "I personally didn't expect that this particular group of agilites to ever agree on anything substantive." But his post-meeting feelings were also shared, "Speaking for myself, I am delighted by the final phrasing [of the Manifesto]. I was surprised that the others appeared equally delighted by the final phrasing. So we did agree on something substantive."

Naming ourselves "The Agile Alliance," this group of independent thinkers about software development, and sometimes competitors to each other, agreed on the Manifesto for Agile Software Development displayed on the title page of this web site.

Values Examples

1. Individuals and Interactions over Processes and Tools

Ask yourself, are you and your team having face-to-face conversations? Or are you relying on emails, documents and certain much-loved messaging tools (naming no names here)? Even if you're working remotely, having a face-to-face conversation can really enable things to happen faster. It encourages trust, openness and relationship-building.

2. Working Software Over Comprehensive Documentation

It's all very well writing streams of requirements documentation, mapping out all the needs of the business, client and customer—all great. But ultimately seeing something working, behaving and reacting is much more useful. Consider some of the following ways to push your project forward quickly.

3. Customer Collaboration over Contract Negotiation

Rather than a weekly status call, and then delivering to your client and stakeholders at the end of a certain period, try and get your client or stakeholders closely involved in the project and working with you on a day-to-day basis. If you're running more of a Scrum process, make the client the Product Owner, leading on bringing the business, technical and customer requirements together. Getting their buy-in and collaboration early on and frequently is critical to avoiding delays, and ultimately getting to a better product or service.

4. Responding To Change Over Following A Plan

Everybody here knows that following the plan as strictly as you can never works out there are always complications and other factors you can't foresee happening. So, you always must be prepared to change the plan when necessary.

Principles Examples

1. Satisfy Customers Through Early & Continuous Delivery

By applying this concept, you will increase your process's agility and respond to changes in a timely fashion. On the other hand, your customers will be happier because they will get the value they are paying for more frequently. Also, they will be able to provide you with feedback early on, so you will be able to decrease the likelihood of making significant changes later in the process. So in the end everyone has a better project.

2. Welcome Changing Requirements Even Late in the Project

Changes late in any project are always hard. Because you worked on something for a certain time and now it must change. However, making changes late in the process should be possible if everyone sets their mind to it.

3. Deliver Value Frequently

If u can deliver small batches off work more frequently. You can also ask feedback and change the direction of the project more easily.

4. Businesspeople and developers must work

together daily throughout the project

The goal is to create a synchronization between the people who create value and those who plan or sell it. This way, you can make internal collaboration seamless and improve your process performance.

5. Build Projects Around Motivated Individuals

The second sentence of this principle is especially important. If you don't trust your team and keep even the tiniest decisions in your company centralized, you will only hinder your team's engagement. As a result, individuals will never feel a sense of belonging to the purpose that a given project is trying to fulfil, and you won't get the most of their potential.

6. The Most Effective Way of Communication is Face-to-face

Speaking to your colleagues directly makes sure he/she know exactly what you want out of them instead of having to make a guess based on an email or passed along information.

7 Working Software is the Primary Measure of Progress

If you can't show the client what is happening, they don't know if they like where the project is going.

8. Maintain a Sustainable Working Pace

A comfortable work environment is always better than a place where you want to leave. So, create a place where everyone can get a vote in, and everyone is comfortable to say what they mean.

9. Continuous Excellence Enhances Agility

Don't have an example yet.

10. Simplicity is Essential

Why do something the hard way if it can be done in a simple way. This also applies to the ideas of the project keep it simple so everyone on the project has the same ideas.

11. Self-organizing Teams Generate Most Value

If you must motivate your team, you first must work on motivation before you try agile.

12. Regularly Reflect and Adjust Your Way of Work to Boost Effectiveness

A team that reflects knows the way forward from looking behind. When you regularly communicate with your teammates and look back on pros/cons you know what to do what is necessary for the project

(agile-manifesto, n.d.) (agile/project-management/principles, n.d.)

Bibliography

agile/project-management/principles. (n.d.). Retrieved from kanbanize:
<https://kanbanize.com/agile/project-management/principles>

agile-manifesto. (n.d.). Retrieved from thedigitalprojectmanager:
<https://thedigitalprojectmanager.com/agile-manifesto>

history. (2001). Retrieved from agilemanifesto: <https://agilemanifesto.org/history.html>