

Agile @ HR



# Agile @ HR

Thomas Pensel

## **Imprint**

Copyright: Thomas PenseL

Year: 2023

ISBN: 9789403654140

Editing/ Proofreading: Junaid Sanwal

Illustrations: Latif Hubaeri

Cover Design: Thomas PenseL, Latif Hubaeri

Publishing Portal: Bookmundo Direct

Printed in Germany

The German National Library lists this publication in the German National Bibliography.

The work including all its parts is protected by copyright. Any use is not permitted without the author's consent.

## Table of contents

<b>About the Author</b> .....	<b>7</b>
<b>Chapter 1: Introduction</b> .....	<b>8</b>
What Exactly Is Agile Work? .....	8
The evolution of agility .....	9
Agile in HR .....	15
Benefits of an agile Human resource strategy.....	22
Importance of agile manifesto in HR .....	23
A Guide to Agile Team Practices.....	24
Critical steps in Agile HR .....	30
How to use agility in HR.....	31
How can managers create an agile team? .....	38
The various forms of Agile meetings .....	43
Agile HR teams' methods for handling employee feedback.....	52
<b>Chapter 2: Agile frameworks &amp; methods</b> .....	<b>61</b>
Facilitation .....	63
Liberating Structures .....	71
Lessons learned (root cause analysis).....	77
OKRS.....	85
KANBAN.....	93
Working out Loud.....	102
Design Thinking.....	109
Lean / Lean six sigma.....	116
Management 3.0 (empowerment) .....	124

**SCRUM..... 131**  
**How are agile frameworks and methods used in human resources? ..138**  
***Conclusion* ..... 151**

## **About the Author**

**Thomas Pense** is an HR and IT Professional with more than 25 years of experience in different industries. He has completed the Power MBA from The Power Business School.

In addition to his formal education, **Thomas Pense** is certified as Agile Coach, Professional Scrum Master, Six Sigma Green Belt, IAPM Agile Project Manager, and many others.

**Thomas Pense** currently resides in Germany with his wife and two children and enjoys Western horseback riding in his spare time. He can be found on LinkedIn, where he shares his work and life information.

**Thomas Pense** is dedicated to changing the culture of work. He strives to make the way we work together more transparent and agile through his work and inspire others to do the same.

## Chapter 1: Introduction

### *What Exactly Is Agile Work?*

Agile working uses decentralized work processes in which companies give employees the freedom and accountability to succeed with the least restraint possible. Agile work is a more freeform style of working that is not constrained by physical spaces or the traditional idea of the "workplace." It is not constrained by time, place, cooperation, or role.

In all honesty, this is a conceptual term. So, to define it more precisely, we must break it down into its measurable parts:

- ✓ **Time:** The number of hours that workers put in.
- ✓ **Space:** The area in which workers are located.
- ✓ **Collaboration** refers to how coworkers cooperate.
- ✓ **Role:** The primary task that employees perform.

Thus, none of these traits are strictly required for agile work. It's more of a mash-up, which gives it a completely different kind of work—one whose competitive edge is supported by technology advancements, creative solutions, and strategic position.

## ***The evolution of agility***

Today's development teams frequently choose Agile methods, especially those attempting to establish a continuous delivery environment. Agile development is commonly associated with high degrees of flexibility and collaboration and an iterative setting where requirements change with changing demands. As a result, we also frequently think of Agile as a methodology that enables development teams from various sectors to release new features more quickly.

How did we get here, however? What does Agile's history include? And how can understanding the origins of Agile help us better comprehend the approach and the beneficial effects it has had on the modern development industry? Let's look at it.

### **Getting Things Done: The Early History of Agile**

Development teams have concentrated on accelerating the time to market for new goods and features to address needs in real-time during the past few years. Development teams have generally accomplished this goal in incredibly unique ways. And Agile is primarily to thank for that pace. Imagine being forced to wait years for a solution to the main issues your company is currently experiencing. That kind of wait time was typically less than 30 years ago. And we can go back in Agile's history to the first instances of this issue.

Before Agile, development teams would identify issues and plan solutions, especially in the software, manufacturing, aerospace, and defense sectors. Then they would begin creating that solution and bringing it to the market. The Waterfall approach, a development process that follows a predetermined course, was explicitly employed by most teams.

- ✓ Specify the project's requirements and its task scope.
- ✓ Create a product based on those established specifications.
- ✓ Create the product.
- ✓ Test the item.
- ✓ Address any issues that were found during the testing.
- ✓ Launch a completed project

This method may appear OK, but Waterfall demanded that teams adhere to the specifications and work scope established at the very beginning of the project and refrain from making any changes or additions later. Following that predetermined plan might be challenging because Waterfall stressed bringing a finished product to market, meaning it might be years before teams completed the task.

The problem's nature frequently changed during that time, but the project requirements did not, making the planned solution obsolete when it finally hit the market. On the customer's end, this delay resulted in significant issues going unresolved for extended periods. Furthermore, even when a remedy was found, the point it was meant to address probably had evolved.

On the development end, this struggle meant releasing fresh products that no longer had a good fit with the market. Teams frequently abandoned projects mid-stream rather than providing an outdated product, resulting in a development graveyard of unfinished goods. Because of these difficulties, we might think of the Waterfall methodology as Agile's arch-enemy.

## **A Shift is Happening: The Agile Past Forms**

Throughout the 1990s, several software development teams started to alter how they planned and delivered new products because they were dissatisfied with the current situation.

This time, development approaches were introduced, including Scrum, Rapid Application Development, Extreme Programming, DSDM, Feature-Driven Development, and Pragmatic Programming. Although methods are different, they all share a more adaptable, lightweight methodology that was required. The original Agile practices that eventually gave rise to what we know today were these methodologies for software development.

A group of 17 software developers, including Martin Fowler, Jim Highsmith, Jon Kern, Jeff Sutherland, Ken Schwaber, and Bob Martin, got the ball rolling in the spring of 2000 when they got together in Oregon to talk about how they could accelerate the development process so they could release new software more quickly. They identified two significant opportunities that would be made attainable by fulfilling this goal:

- ✓ Reducing the time it takes for users to receive advantages to address the issues with product-market fit and development graveyard
  
- ✓ Receiving user feedback soon allows developers to continue refining new software while also confirming its usefulness.

Although this meeting did not lead to the development of the Agile approach as we know it today, it was still a significant turning point in the development of Agile because of its emphasis on speed to market, quick feedback, and continual improvement.

## **Agile Gains Emphasis as a Manifesto is Born**

The history of Agile was finally revealed when the same 17 engineers reunited less than a year later, this time at a ski resort in Snowbird, Utah. They wanted to discuss their efforts at this meeting in further detail and find a more tangible solution to the pressing development issues of the day. The team generated the "Manifesto for Agile Software Development" in about three days (known more commonly as the Agile Manifesto). This manifesto, which marked a real turning point in agile history, outlined four essential values:

- ✓ Persons and interactions over procedures and tools
- ✓ Functional software over thorough documentation
- ✓ Consumer involvement in contract negotiations
- ✓ Adapting to change over sticking to a plan

In the history of Agile, these values are a significant advancement, but the organization didn't stop there. They also outlined 12 principles supporting these beliefs to enliven their vision further. These guidelines consist of the following:

- ✓ delivering quality software on time and continuously to consumers to satisfy their needs.
- ✓ accepting alterations to the requirements at any stage of the delivery cycle
- ✓ providing software more frequently by reducing development times
- ✓ Working software is used as the primary metric for progress

- ✓ Regularly reflecting on oneself to find areas where one might improve

Today's teams still apply the original Agile framework's 12 principles and four guiding ideals.

### **The Following Chapter in Agile History: Presenting a New Approach to the World**

During the February 2001 summit in Snowbird, Utah, Agile's history advanced significantly, but its future trajectory had just started. After that three-day gathering, the 17 leaders were prepared to begin the next phase of Agile's development: persuading the rest of the world of the importance of everything they had outlined in the Agile Manifesto.

### **Agile becomes widely used**

The Agile Alliance was founded due to the founding fathers in the history of Agile deciding to establish a more lasting organization to aid in spreading awareness of the Agile Manifesto. There is still a nonprofit organization called The Agile Alliance. The organization's mission is to disseminate knowledge about Agile, offer tools to teams wishing to embrace the technique, and keep improving the strategy in response to evolving client demands.

As the partnership was established, the history of Agile took off, gaining popularity among software development teams throughout the early 2000s. The techniques created by those teams along the way, such as quick judgment calls, the "role-feature-reason" pattern for user stories, retrospectives, and

daily meetings, frequently led to the growth of Agile as we know it today (often referred to as standups).

The Agile Alliance's function grew as Agile gained popularity. When the now-official Agile Alliance visited Utah for the first annual Agile conference in 2003, it represented a critical turning point in the history of Agile. The group stated that it was "committed to fostering Agile principles and providing a space for individuals and ideas to flourish" at this meeting. The Agile 20XX annual conference is still being held today. The Agile Alliance has expanded geographically over time as well. The Agile Alliance now supports member groups worldwide that assist area businesses in implementing the Agile Manifesto and spreading the word about Agile in local markets.

### **Agile gains momentum**

Agile began to gain popularity in the early 2000s, but the Agile Manifesto gained momentum again in the 2010s. By this point, development teams routinely discussed the history of Agile, but between 2012 and 2015, actual success metrics started to go along with that history. The advantages of adopting the lightweight methodology became indisputable at that point because it was possible to show success in Agile. That Agile adoption crossed the 50% mark over these three years and entirely revolutionized the development business should not be surprising.

Agile soon started to take off, expanding beyond development this time. In 2017, we saw the first clear explanation of Agile Testing. This description described cooperative testing procedures emphasizing the timely delivery of high-quality products and defect avoidance more than defect detection.

## ***Agile in HR***

Agile @ HR examines how HR may use different working methods and an Agile mentality within our teams and projects. Agile @ HR has the potential to transform our current operational paradigm and advance our profession. Agile for HR can be done without ever working in an agile organization, and you can immediately reap its rewards by using a few fundamental procedures in your routine tasks.

### **Accepting the Mentality**

Agile is all about providing value to your clients and business. In HR, we frequently discuss the need to "bring value," but we often struggle to describe and quantify this value precisely. Most HR plans cover important, intricate subjects like creating a customized employee experience for a varied, multigenerational workforce. Or training up-and-coming leaders for positions that don't even exist yet. The value supplied at any given time is frequently left undefined or tied directly to big bang releases, like a new employee benefits package or leadership development program, even though these complex goals are admirable objectives. Agile provides a solution by assisting HR in dividing these vast, complex topics into manageable chunks of value. It helps us clearly express what we're delivering to the business and why we're doing it and prioritize our work based on weight.

### **Value centered on people**

According to the agile HR mindset, the employee experience is a product; our people are the users and customers. When seen in this manner, the value becomes to stand for something that pleases the client and aids the employee

in performing their work successfully. Additionally, it makes it necessary to develop people's practices that are human-centric and validated by the users themselves. Designing HR services and procedures based on past best practices or what management believes the organization needs is a different approach from doing so.

### **Continuous improvement**

Agile HR then tries to deliver this value to the client gradually. The key term here is "incremental," which signifies a considerable shift in HR's perspective. Incremental refers to providing value by gradual, modest changes verified using a test-and-learn methodology. The Agile feedback loop represents an intentional departure from rigorous upfront planning that cannot be modified to accommodate changing client needs, a shift in the project's scope, or a better concept.

### **Getting past the tools**

This indicates that the Agile HR attitude is more than just a few stand-up meetings, Post-it notes, and Kanban or Scrum boards. It's about genuinely departing from the norm of adhering to HR best practices and adopting a test-and-learn strategy to offer value that our employees and their work experiences confirm. HR won't be able to use Agile tools successfully and methods in our teams and projects until it adopts this approach. Also, all the frameworks and practices will be adapted to what works best for you and your company. For example, Scrum describes itself as "purposely incomplete, " meaning you are welcome to adjust it how you need it to be run. This also