



COGNOTA

Learn About

Agile Facts & Figures

What is Agile?

Models & Methodologies

Common Misconceptions

Agile Philosophy

Agile is a set of methodologies, frameworks, and practices that enable quicker delivery of valuable products and experiences.

The important thing is not your process. The important thing is your process for improving your process

Henrik Kniberg
(Agile trainer and author)

eBook

The Beginner's Guide to Agile Learning

create.



Table of Contents

Introduction	03
What is Agile?	04
Agile Facts & Figures	06
What is Agile Learning?	07
Why Use Agile Learning?	09
Models & Methodologies	10
Is Agile Learning Right for You?	11
Common Misconceptions About Agile	12
Next Steps	17



Introduction

What started out as a framework for software development has quickly taken the entire business world by storm. In CollabNet VersionOne's 2019 **"State of Agile" report**, the findings showed that 97% of companies had adopted Agile in one form or another. Training teams are starting to pay attention to the potential benefits of Agile, but it can be difficult to know where to begin.

Before you dive into new processes and tools, it's essential to understand the core principles of Agile and how they can be applied successfully to learning design and training project management.



"The **important thing is not your process. The important thing is your process for improving your process."**

Henrik Kniberg
(Agile trainer and author)

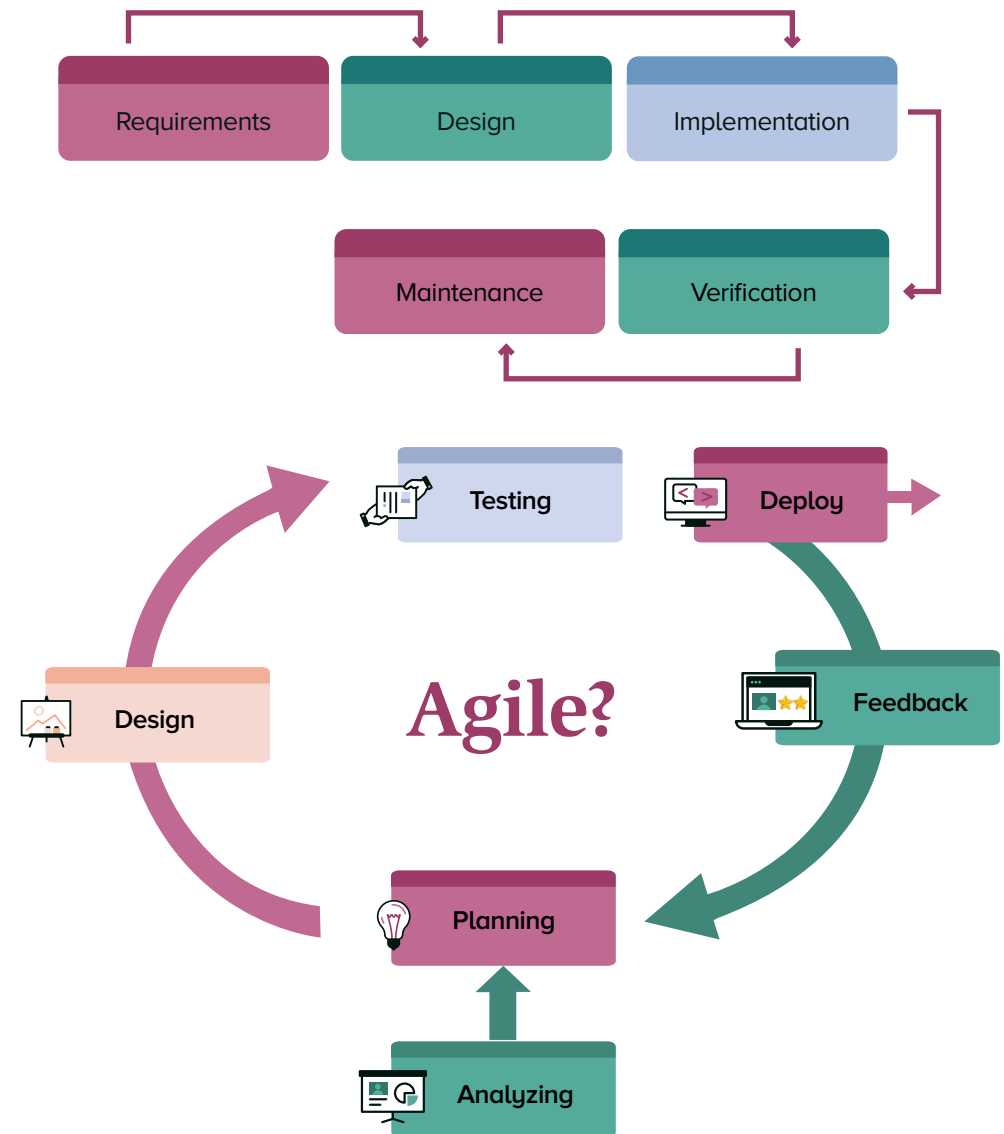


What is Agile?



Agile is a set of methodologies, frameworks, and practices that enable quicker delivery of valuable products and experiences through iterative collaboration between cross-functional teams.

Agile started out as a method for software development back in the early 2000s. Traditionally, software teams had used a Waterfall model, where process steps were sequential and problems were not uncovered until the testing phase at the very end of development.





Waterfall was fine, but it was restrictive. Engineering teams were forced to stick to requirements set out at the beginning of a project, making it difficult to respond to unexpected outcomes or shifting goal posts.

Realizing that they needed to find a way to speed up the development process and respond faster to feedback, a group of engineers decided to switch things up. When creating a new approach, the objectives were to achieve a framework that:





Agile Facts and Figures

Agile has been growing in popularity in project management, marketing, and a vast array of other business functions. In response to various surveys, organizations that have adopted Agile reported the following:



60%

Growth Experienced in **revenue and profit**



98%

of **companies** say Agile has helped



78%

of scrum teams use **Scrum** in conjunction with other **frameworks**



63%

of **failed Agile implementations** are due to a clash between existing culture and **Agile philosophy**



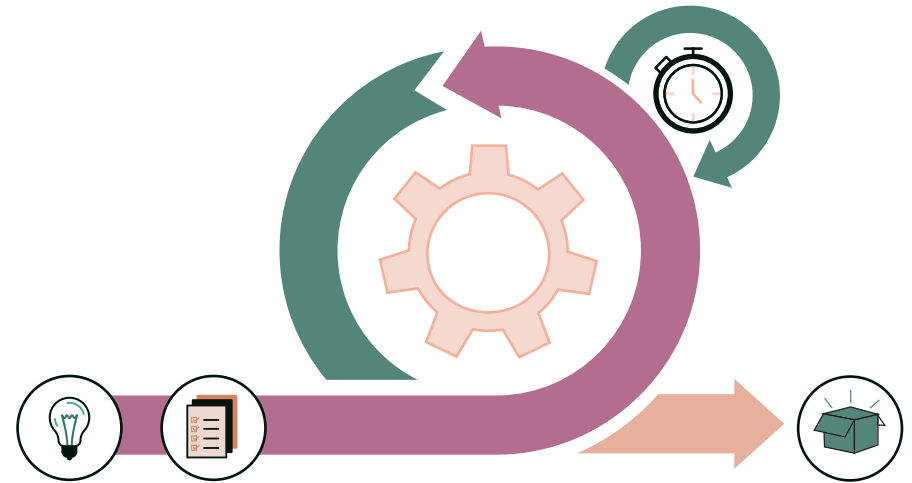
What is **Agile Learning**?



Agile Learning, or Agile instructional design, refers to any approach to training development that focuses on speed, flexibility, and collaboration.

Agile is a term usually associated with software development. But as the speed of business increases, other business functions have adopted Agile techniques to help them pivot quickly with market changes.

L&D leaders observed that the process of developing learning experiences in today's digital age could greatly benefit from the Agile methodology. As such, Agile Learning was born.





Agile Learning Vs. Learning Agility



Agile learning is the application of Agile philosophies to the learning design process.

Agile learning focuses on speed, flexibility, and collaboration.

Some of the key concepts that influence the philosophy of agile learning include:

- Minimal focus on procedures or rules
- Reactive to market feedback
- Flat team structures and continuous collaboration
- Iterative “fail fast” approach to development
- Highly adaptable and responsive

Learning agility is an ability held by an individual to adapt to unfamiliar situations, learn quickly, and in a mostly self-managed way.

Dr. W. Warner Burke of Columbia University credits agile learners with:

- Flexibility (Open to New Concepts)
- Speed (Taking Quick Action)
- Willingness to Experiment
- Performance Risk-Taking
- Interpersonal Risk-Taking
- High Collaboration
- Gathering (Seeks Information)
- Feedback Seeking
- Reflecting



Why Use Agile Learning?

So, why should you care about Agile Learning? What are the benefits to your team, your learners, and your organization?

Flexible Response to the Changing Needs/Feedback of Learners

Agile learning allows you to respond to quickly changing training needs and become part of the success story, rather than struggling to keep up.

Achieve Rapid Course Development

The iterative nature of the Agile methodology allows for quick feedback and quick updates on site with your end-user, rather than going through lengthy testing and feedback reviews.

Improve Team Collaboration

Agile promotes effective collaboration for project teams. And with enhanced collaboration comes fewer misunderstandings, less frustration, and higher quality outcomes.

Increased Accountability & Transparency

Team structure is an important component of Agile. Different methodologies provide specific team frameworks designed to promote accountability through specific project roles and responsibilities.

Improve Effectiveness of Training

As you gather feedback through testing at every stage of development with Agile, the effectiveness of your learning experiences grows and improve continually along the way.

Improve Learner Engagement

Feedback from learners and continuous iteration on course structures and content creates experiences that are more engaging. Learners will be far more willing to take up training that is designed with their needs and wants in mind.





Agile Learning Models and Methodologies

You may have heard of different Agile methodologies such as Scrum, Kanban, or Lean. However, Agile philosophies have recently been reshaped into several different frameworks that directly address the application of Agile philosophies to learning design processes.



Successive Approximation Mode (SAM)

The Successive Approximation Model, developed by Allen Interactions, provides an agile version of traditional models like ADDIE. It emphasizes repetition, collaboration, and efficiency to help overcome common pain points for training teams.



A.G.I.L.E Learning Design

Unlike Agile Learning, AGILE is an acronym for a **step-by-step process designed by Conrad Gottfredson**, a learning strategist and industry leader. The five steps of the AGILE instructional design model are: Align, Get set, Iterate & implement, Leverage, Evaluate



Rapid Content Development (RCD)

Also known as rapid e-learning, this model consists of a preparation phase, iterative design, reusable templates, and tools to support quick execution of course design and deployment.



LLAMA (Lot Like Agile Management Approach)

Developed by Megan Torrance, this approach focuses on quick iterations to increase speed to market and quality. It's also about managing all the other aspects of your training project, from requirements gathering to SME collaboration.



Is Agile Learning Right for You?

Before you start changing processes or settling on an Agile Learning methodology, it's important to make sure Agile Learning is beneficial for your organization and your training projects. Here are some questions to ask yourself first:

1. What Type of Learning Needs to be Produced?

If your team is tasked with complicated, multi-dimensional training projects, a 100% Agile Learning solution may present a challenge.

3. Is There Previous Organizational Experience With Agile?

Previous success with Agile in other departments can indicate that your organizational culture is a good fit for a smooth shift to Agile Learning.

2. Do You Produce a High Volume of Training?

Agile Learning solutions are ideal for teams who must produce a high volume of courses or frequent updates of existing materials.

4. Can You Identify Early Adopters on Your Team?

You'll need Agile champions within the training team to help achieve the transition and get others on board.



Common Agile Misconceptions

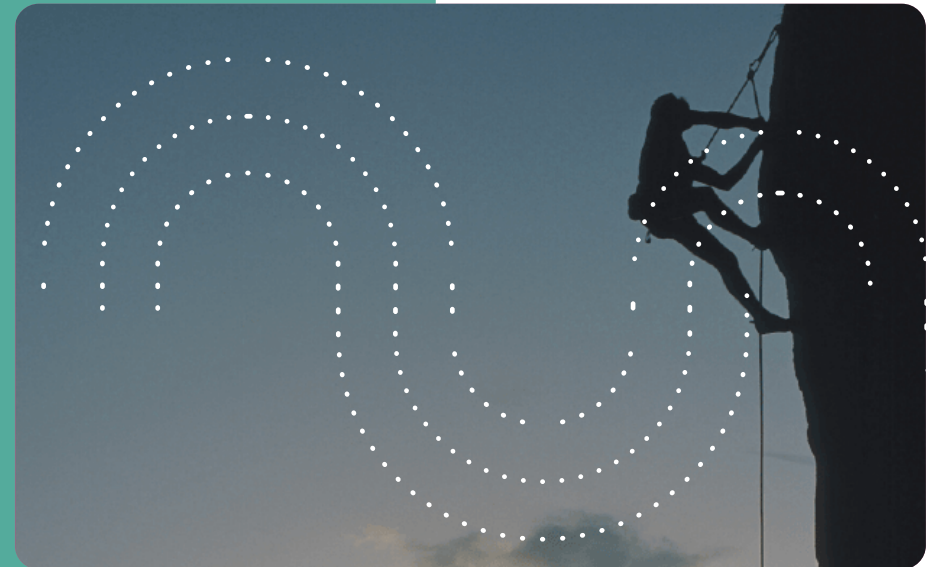
L&D leaders are starting to see that the process of developing learning experiences in today's digital age could greatly benefit from the agile methodology.

As training teams must continually roll out training, the ability to rapidly develop and distribute a large volume of learning experiences is paramount.

However, with all this talk of Agile, people are misunderstanding what it's all about and how best to leverage it. Here are some common Agile misconceptions and how you can explain them better to a reluctant Agile Learning team:

**“Do The Planning, But
Throw Out The Plans.”**

Mary Poppendieck
(Lean trainer and author)





1. Agile Means Just Going With The Flow

A common misconception about Agile is that it does not require a formulated plan or structured process. And while Agile promotes flexibility and the ability to respond quickly to a changing goal, there are structures and processes in the framework that help the whole Agile team to stay on track.

The idea is to keep iterating and improving throughout the project. And these iterations allow the team to achieve an MVP (Minimum Viable Product) much quicker than more traditional approaches. So, while Agile promotes flexibility, there is a lot more to Agile than simply going with the flow.



**“Be prepared to cut your losses
– Canceling bad projects early is
success because you save time,
money and resources that can be
applied to better opportunities.”**

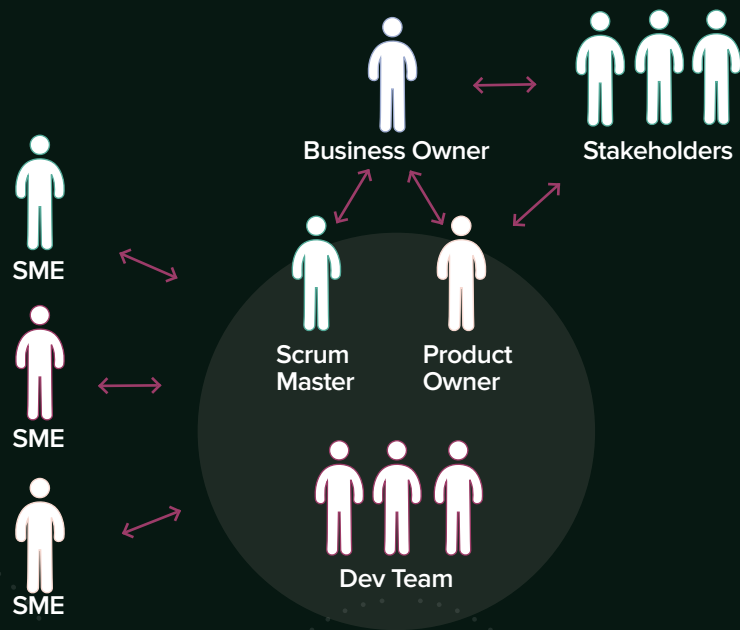
**Ian Spence and Kurt Bittner
(Agile authors)**



2. There Are no Project Managers in Agile

Another common Agile misconception is that there is no team structure or project managers.

Although a flat team structure is an essential concept in Agile, every person on an Agile team has a clearly defined role and transparent responsibilities. This helps keep everyone accountable and driving forward.



Scrum Team

Sourced from [Mendix](#)

Common Agile Team Roles

Product Owner

- Keeps the core vision front of mind for the team
- Directs collaborative efforts and keeps the team updated on significant developments
- Focuses on the high-level and steers the ship

Scrum Master

- Role most closely associated with a traditional project manager
- Focuses on fine-tuning processes, providing feedback, and maintaining cohesion
- Oversees day-to-day operations of the team

Team Members

- Responsible for the technical development
- Instructional designers and content creators

Team Members

- The people who will use the end product
- Subject Matter Experts and learners



3. It's Only **for Software**

The need to keep up with increasingly dynamic market forces means organizations must find ways to kickstart a higher pace and keep up with change. That's why practices such as Agile have been seeping into multiple business functions in recent years.

Having started out in software, Agile was soon picked up by project managers everywhere and today, it's being continuously adapted for the needs of many different business functions. That's because, regardless of its software origins, the key principles of the Agile manifesto culminate in a philosophy that benefits business in general, not just software or product development:

Individuals and interactions over processes and tools

Professional software/product/experience over comprehensive documentation

Collaboration over negotiation when it comes to contracts and agreements

Responding to change over following a plan

So, Agile was developed as a way to respond more quickly to customer needs, market forces, changing project goals, and a generally more sensitive and volatile market. It enables teams to quickly bring products to market and respond to customer feedback. It's not difficult to see why the same principles are appealing when applied to other business functions such as learning and development.



5. Agile is All or Nothing

Agile is not necessarily better than all other frameworks, depending on the project and the team's goals and objectives. It's also not the case that it cannot work in tandem with other more traditional models. But it's a common Agile misconception that you must fully embrace Agile to the exclusion of all other frameworks, models, and processes.

Other options such as the Successive Approximation Model (SAM), AGILE Learning, and Rapid Content Development (RCD) are all learning specific project frameworks that are based on Agile principles. The most important thing is that you embrace the underlying philosophies of Agile and select a model that works best for your team and your organization.

ADDIE Vs Agile

ADDIE has been the almost universally accepted model for course design for a long time. However, it's a linear process. And although newer versions of ADDIE are more iterative, Agile models are much more suitable for learning teams who are under increasing pressure to keep up with training demand.

But many learning teams who work Agile choose to either 1) divide their projects into Agile or ADDIE frameworks depending on the project requirements, 2) incorporate only the elements of Agile that help to speed up existing ADDIE processes, or 3) create a hybrid of both methodologies that best suits the structure, culture, and workload of the team.





Next Steps

Now that you're better acquainted with some of the fundamental concepts related to Agile Learning and the Agile philosophy, you may want to think about the next steps towards implementing an Agile framework and processes that work for you and your team:

1. Assess the Need

First, perform a careful assessment of your current approach to identify bottlenecks and room for improvement. It may also be a good idea to identify an upcoming project that will be ideal for testing out new Agile Learning methodologies.

3. Consider Outside Help

Professional help is available (and often advised) for Agile implementation. Consider hiring a consultant or Agile trainer. This also has the added benefit of formalizing the culture shift for team members who are more hesitant about adopting Agile.

2. Choose your approach

Do some more research into Agile and Agile Learning methodologies. Which one sounds like it best addresses the needs uncovered during your assessment?

4. Agile Implementation Should be... Agile

Don't be afraid to iterate continuously on your new Agile approach. It may take a lot of experimentation to find the right framework or perfect blend of processes for your learning projects.



Summary

There's no getting away from the fact that a shift to Agile is a huge undertaking. It's not just about changing processes, it's a change in philosophy and way of thinking when it comes to the way you and your team approach training development.

However, the benefits of Agile Learning far outweigh the initial investment, especially if you take the time to properly implement and develop your unique Agile solution.

The First and Only Learning Operations Platform for L&D Teams

Increase effectiveness by bringing the entire workflow, from intake to design, into the Cognota LearnOps platform. You'll save time, improve collaboration, and get data about team performance and learning needs across the organization, which disparate or stitched-together tools can't provide.

[Try It Free](#)

[Speak With Sales](#)





COGNOTA