Scrum Introduction

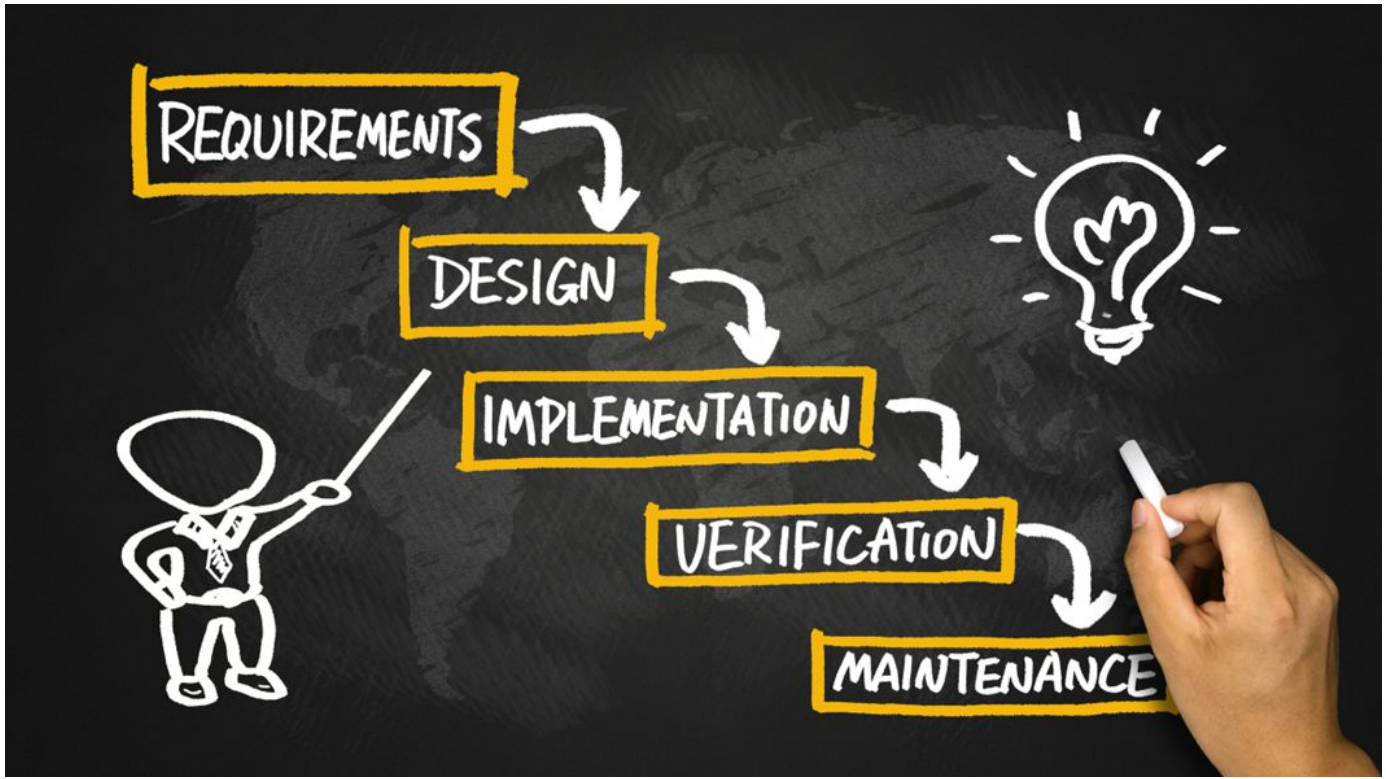
Agile Manifesto

Objectives

At the end of this lesson, learners will be able to:

- Compare the traditional Adaptive Development methodology and approach to solving problems with the Agile methodology.
- Define Adaptive Development.
- Define the Agile Manifesto, including its origins, principles, and values.

Before Agile (Waterfall)



Problems of Traditional Development

REQUIREMENT ANALYSIS

DESIGN

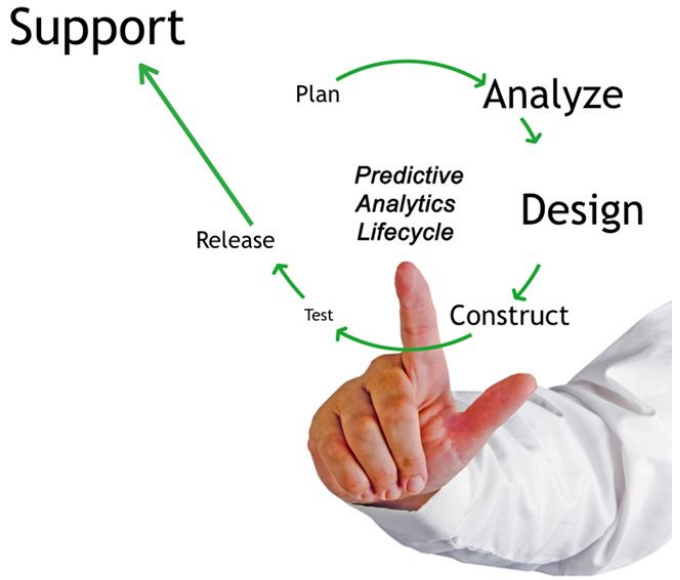
DEVELOPMENT

TESTING

IMPLEMENTATION

MAINTENANCE

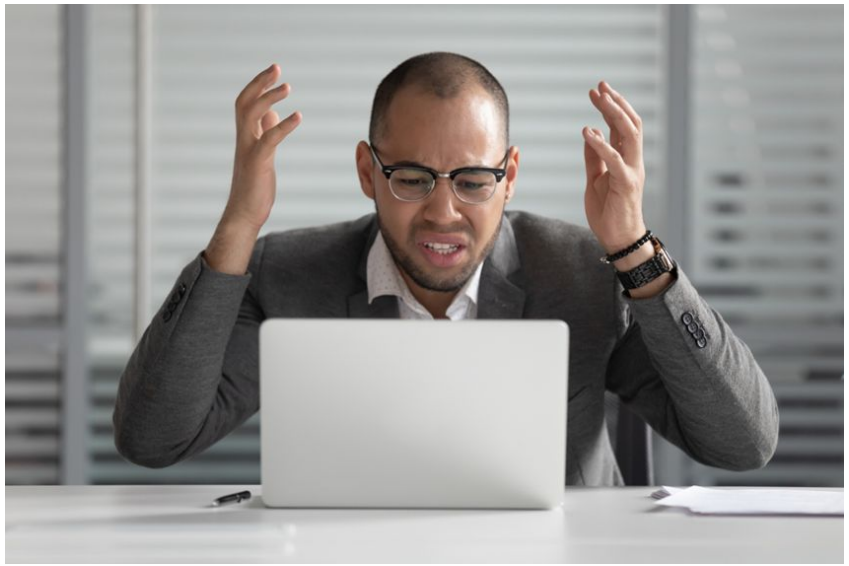
Hard to be Predictive



Traditional Results

You built exactly what I asked for…

But it is not what I need now.



Class Activity

Write down the route you will take to a destination of your choosing that is over five hours away.

Houston, we have a problem!

There is a massive traffic backup on your defined route.

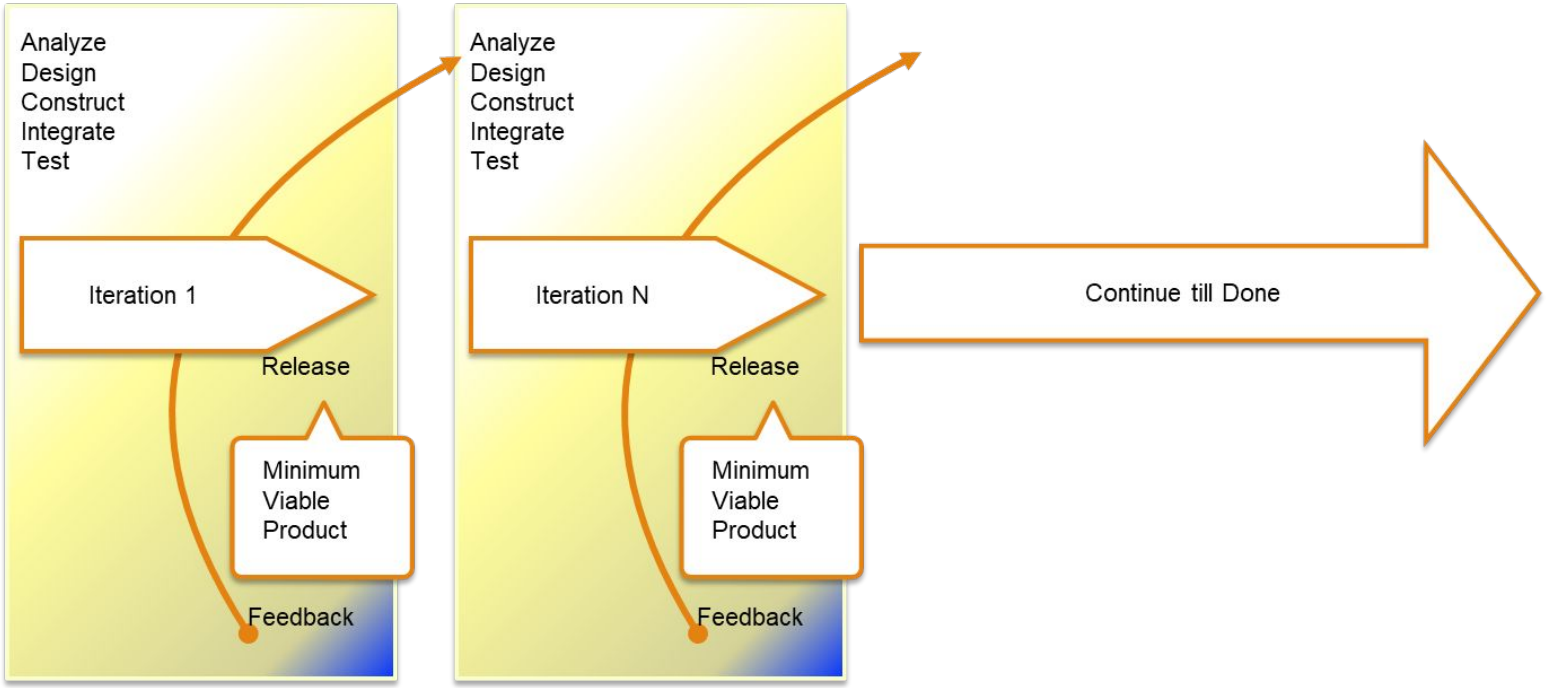
You see full lanes of traffic for a couple of miles ahead and hear on the radio that there is no expected resolution.

What do you do?

1.Wait for the traffic to clear.

2. Attempt a reroute.

Adaptive Development



Source: Per Scholas 2022

Origin of the Agile Manifesto

- 2001: 17 Technologists met.
  - Snowbird Ski Resort in SLC UT.
- Unhappy with state of software development.
- Wanted to delight customers and be more productive.
- Wrote The Agile Manifesto.
  - 12 Principles.
  - 4 Core Values.

Agile Principle #1

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Agile Principle #2

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Agile Principle #3

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Agile Principle #4

Business people and developers must work together daily throughout the project.

Agile Principle #5

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Agile Principle #6

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Agile Principle #7

Working software is the primary measure of progress.

Agile Principle #8

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Agile Principle #9

Continuous attention to technical excellence and good design enhances agility.

Agile Principle #10

Simplicity–the art of maximizing the amount of work not done–is essential.

Agile Principle #11

The best architectures, requirements, and designs emerge from self-organizing teams.

Agile Principle #12

At regular intervals, the team reflects on how to become more effective, and then tunes and adjusts its behavior accordingly.

Statement of Values

Individuals and Interactions		Processes and Tools
	OVER	

Source: Per Scholas 2022

Statement of Values

Working Product	OVER	Comprehensive Documentation

Source: Per Scholas 2022

Statement of Values

	OVER	
Customer Collaboration		Contract Negotiation

Source: Per Scholas 2022

Statement of Values

OVER

Source: Per Scholas 2022

Statement of Values

Individuals and Interactions	OVER	Processes and Tools
Working Product	OVER	Comprehensive Documentation
Customer Collaboration	OVER	Contract Negotiation
Responding to Change	OVER	Following a Plan

Source: Per Scholas 2022

Introducing Scrum

Focus on People

Expect professionalism be used to produce high quality.

Working Software

Agile aims to deliver high-quality working
(functioning with needed functionality) software products.

Flexibility

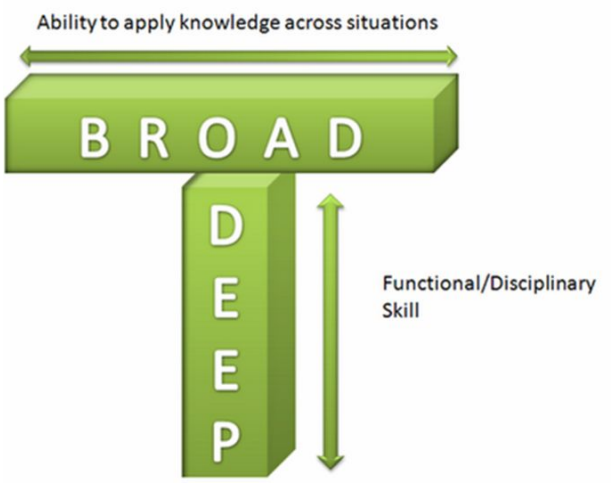
Agility: The ability to adjust to changing conditions.

Customer Involvement

Customers or User representatives are expected to be involved, give opinions and feedback, and help to prioritize next steps.

Multidisciplinary, Cooperating Teams

Scrum teams consist of different professionals; each with different specialty skills, working together to deliver the requested product.



Trust

This is the basic ingredient to be able to deliver the requested quality.

Agile Frameworks

1. Scrum.
2. Kanban.
3. Extreme Programming (XP).
4. Feature Driven Development (FDD).
5. Crystal.
6. Dynamic System Development Method (DSDM).
7. Rapid Application Development (RAD).
8. Adaptive Software Development (ASD).
9. Disciplined Agile (DA).
10. Scaled Agile Framework (SAFe).
11. Large-Scale Scrum (LeSS).
12. Lean Software Development (LSD).

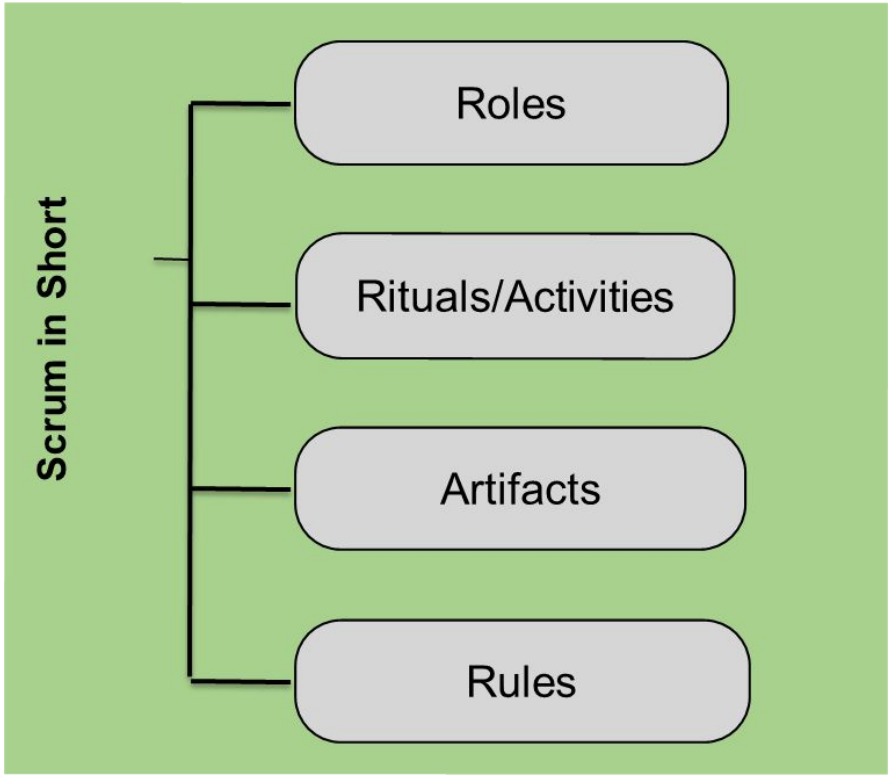
What Scrum is Not

- Scrum has no exhaustive and formal prescriptions on how to design and plan the work, actions, and behaviors of all players involved in product development against time.
- Scrum is also not a methodology.

What is Scrum?

- A lightweight framework that helps teams and organizations generate value through adaptive solutions.
- It is ideal for complex product development.
- It is Iterative and incremental.

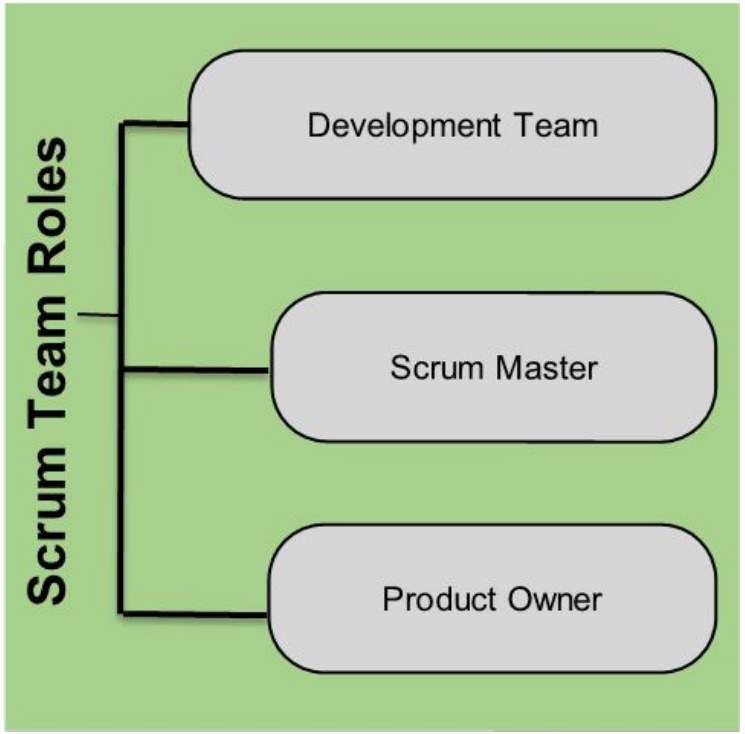
Scrum in Short



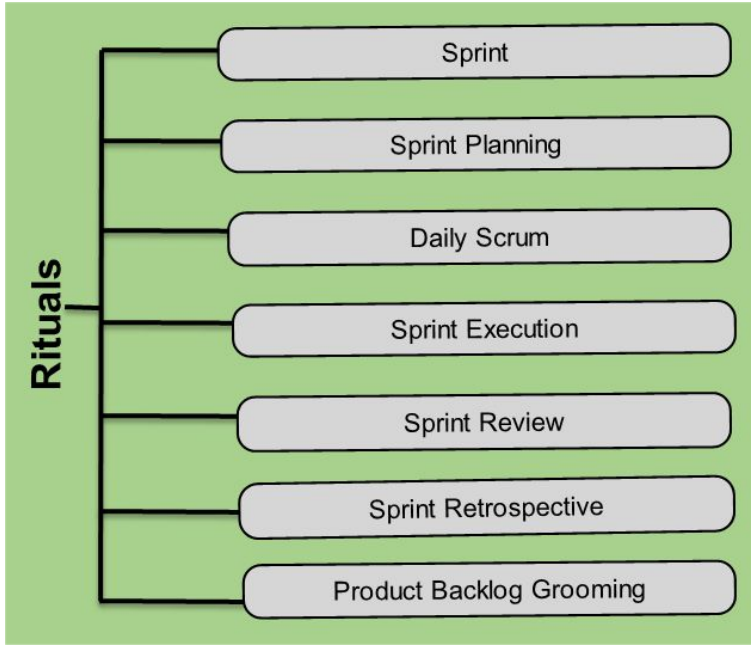
Source: Per Scholas 2022

Role

A Role is a set of **activities, responsibilities, and/or authorities** granted to/performed by a person, group or team.



Rituals/Activities



Rituals/Activities represent actions to be taken or a specific act of work to be performed.

Source: Per Scholas 2022

Rules

Rules define prescribed behavior, or a control mechanism that is intended to be followed.

Questions?

Summary

This lesson introduced the Agile Manifesto and what it means to be agile. Being agile, and working in an Agile environment enables an organization to become high-performing and efficient. In the next lessons, we will discuss other concepts that come with being a Scrum Master and performing in an Agile environment.