



Agile Session 1



Did you finish
Agile (What is Agile? & Scrum Overview)
pre-class activity?



Students choose an option

Would you prefer to
work on a project
for a few weeks or
to combine several
projects together?



Students, write your response!



Pear Deck

Pear Deck Interactive Slide

Do not remove this bar

Recap SDLC and Waterfall model

- SDLC is a structured way to create and develop software.

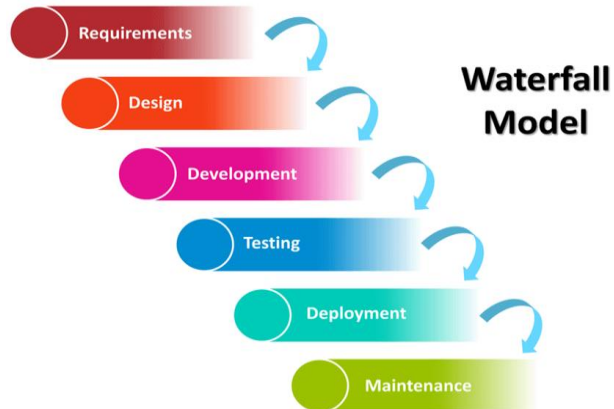
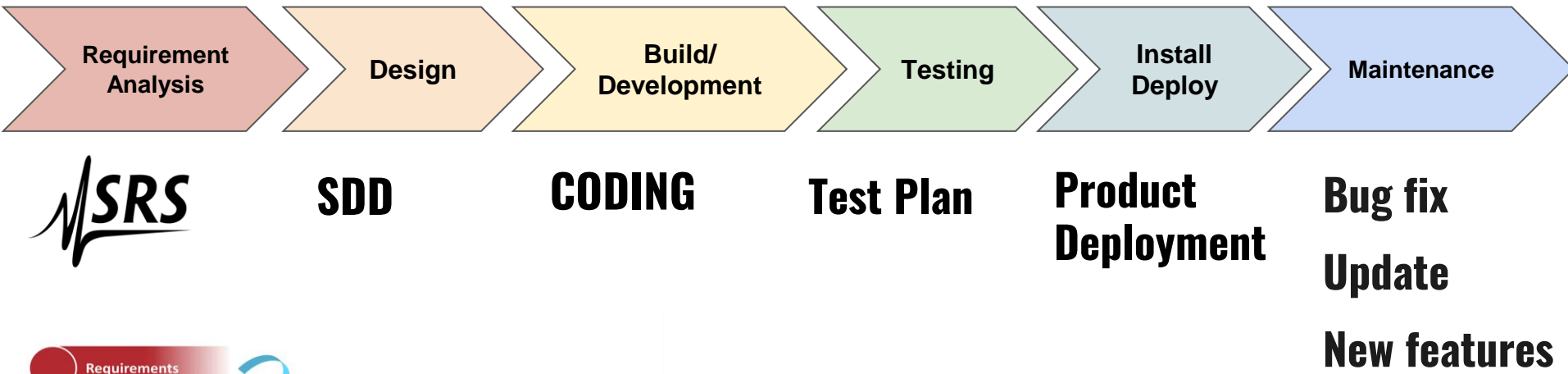


Table of Contents

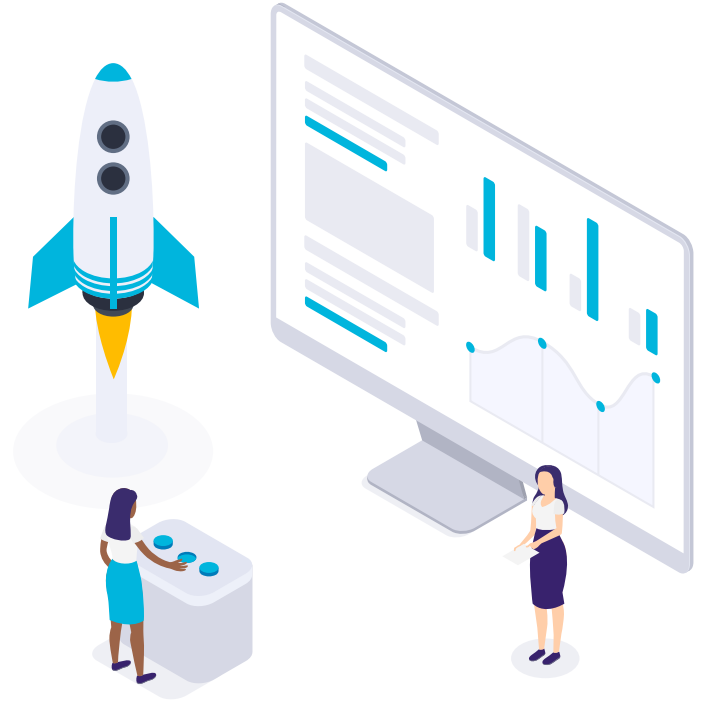


- ▶ Modernizing Project Management
- ▶ Agile Manifesto
- ▶ Agile Principles
- ▶ Overview of Agile Approach
- ▶ Scrum Overview
- ▶ Scrum Roles

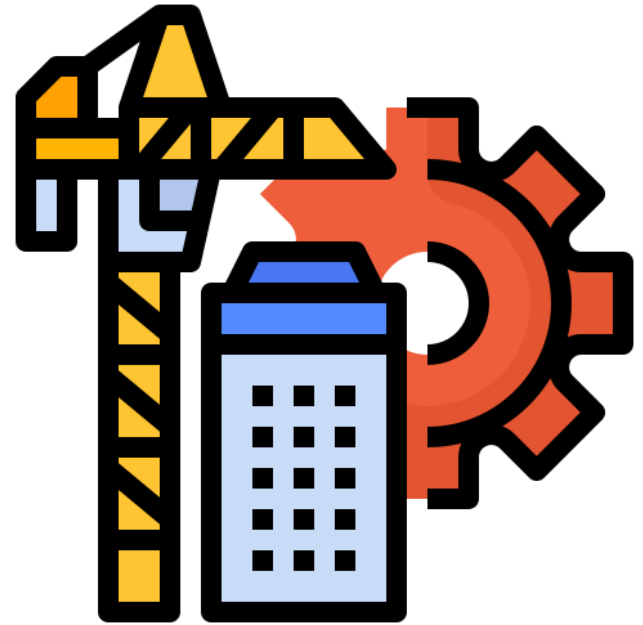


1

Modernizing Project Management

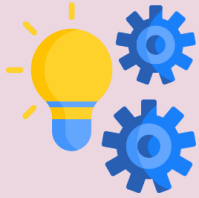


Have you heard the Agile approach before?
What is your history about Agile?



Students choose an option

Modernizing Project Management»



A *project* is a planned program of work that requires a definitive amount of time, effort, and planning to complete.



Goals & Objectives



Fixed period of time



Certain budget

Agile approaches are a response to the need to modernize PM.

The Origins of Modern PM



WWII



step-by-step
manufacturing
models

1940-1950's



manufacturing
process

after 1970's

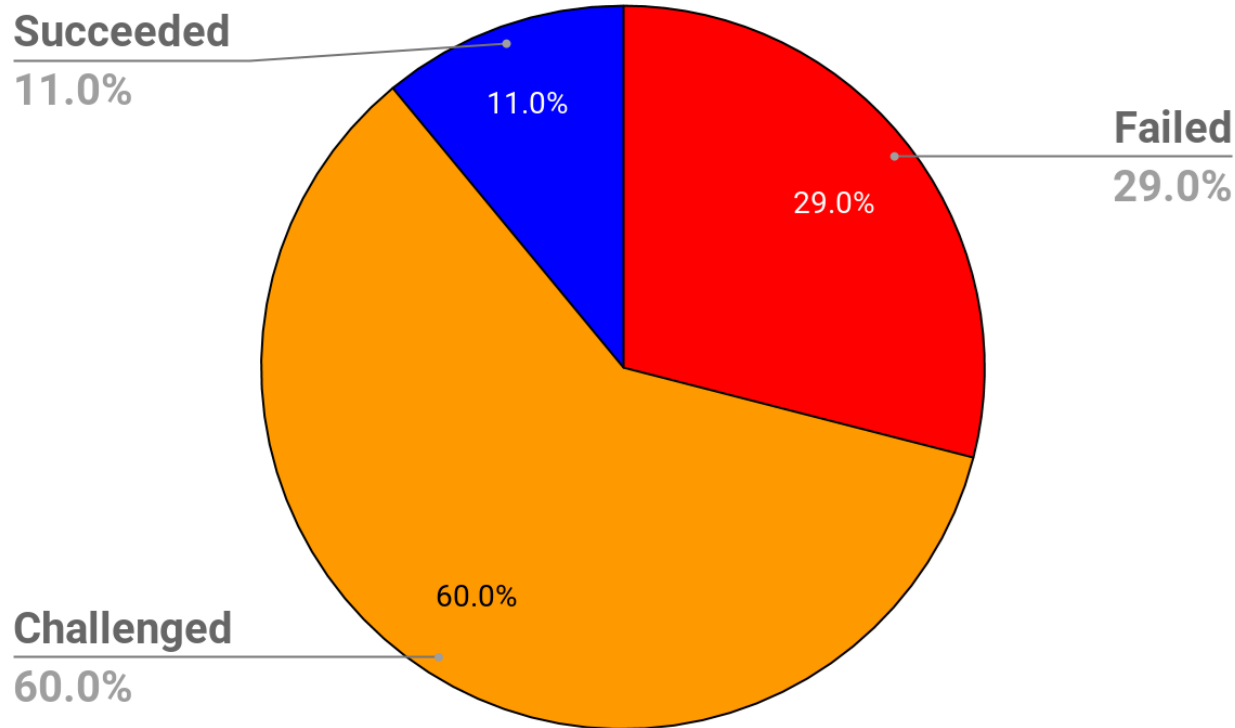


waterfall
model

On waterfall projects, you move to the next phase only when the prior one is complete — hence, the name waterfall.

Software Project Success and Failure

Standish Group CHAOS Report



Software Project Success and Failure



CHAOS RESOLUTION BY AGILE VERSUS WATERFALL

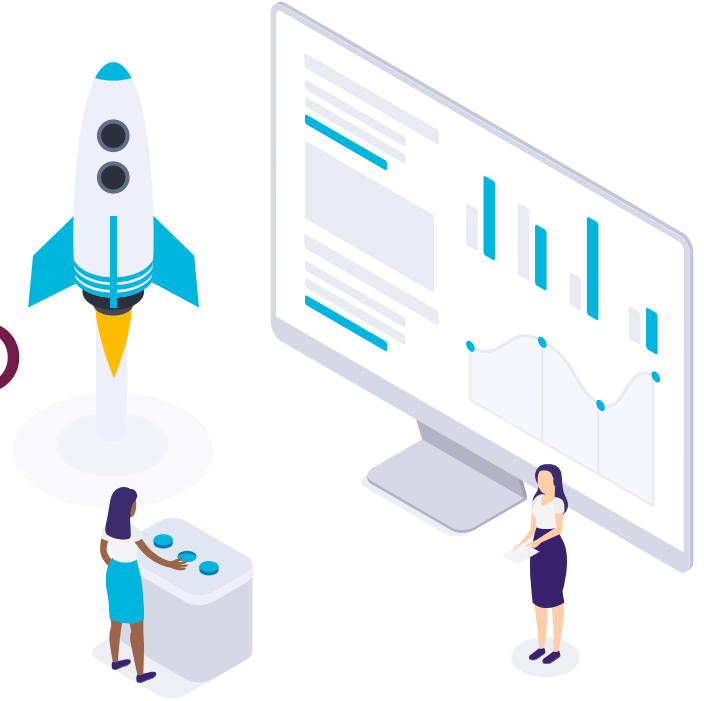
SIZE	METHOD	SUCCESSFUL	CHALLENGED	FAILED
All Size Projects	Agile	39%	52%	9%
	Waterfall	11%	60%	29%
Large Size Projects	Agile	18%	59%	23%
	Waterfall	3%	55%	42%
Medium Size Projects	Agile	27%	62%	11%
	Waterfall	7%	68%	25%
Small Size Projects	Agile	58%	38%	4%
	Waterfall	44%	45%	11%

The resolution of all software projects from FY2011–2015 within the new CHAOS database, segmented by the agile process and waterfall method. The total number of software projects is over 10,000.



2

Agile Manifesto



The Agile Manifesto



A group of 17 people thought:

“We’re all doing these different approaches to developing software. We ought to get together and see where there are commonalities in what we’re thinking about.”

The result was a meeting at a ski resort in Snowbird, Utah in 2001.



The Agile Manifesto



In this video, Ken Schwaber, co-creator of Scrum and original member of the group signing the Agile Manifesto in an interview with the Boston Business Journal discusses how it came about.



The Agile Manifesto



Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

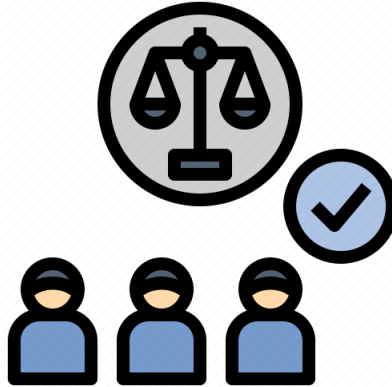
James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

The Group's Works



Agile Manifesto



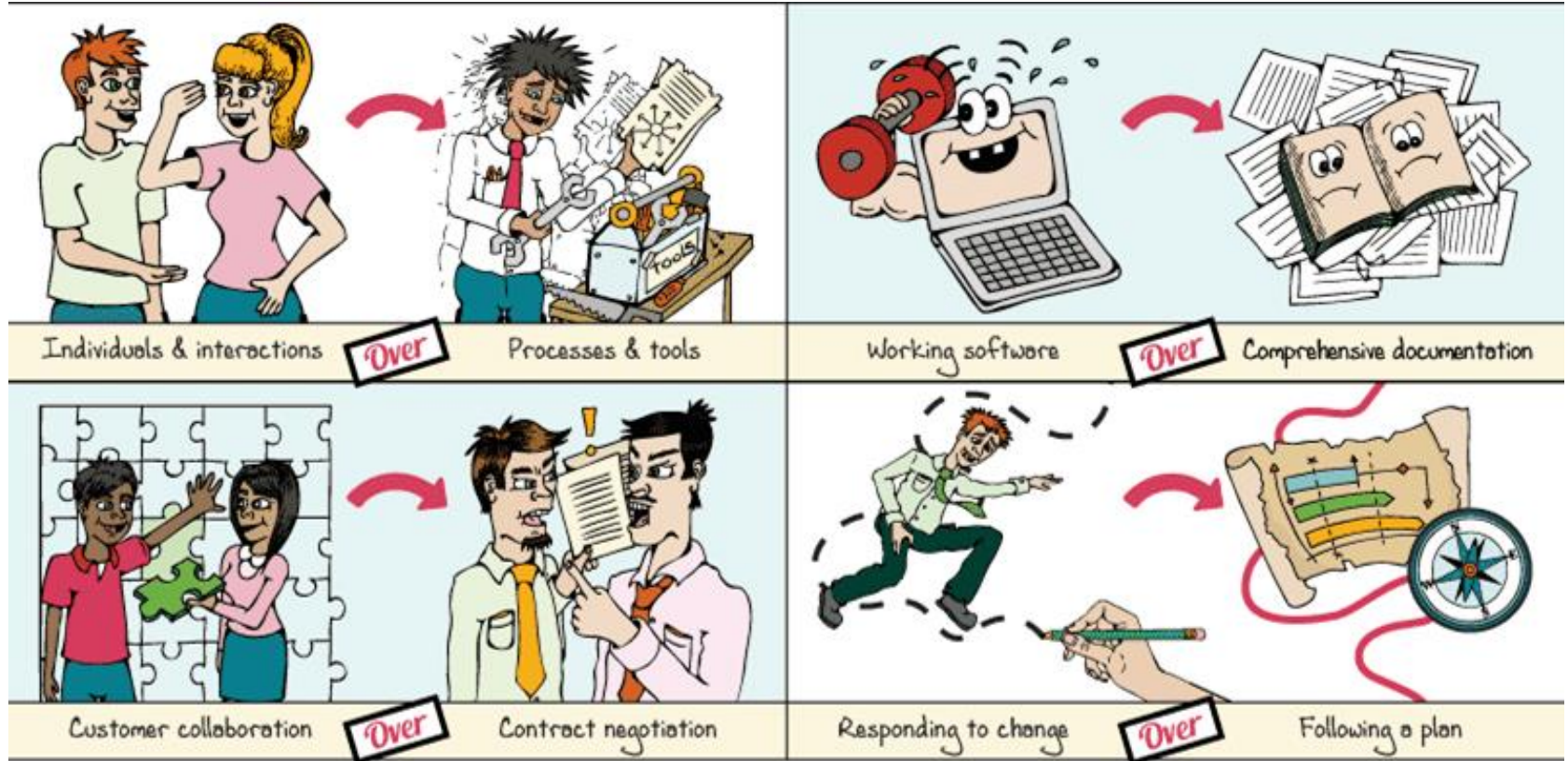
Agile Principles



Agile Alliance



The Four Values of the Manifesto



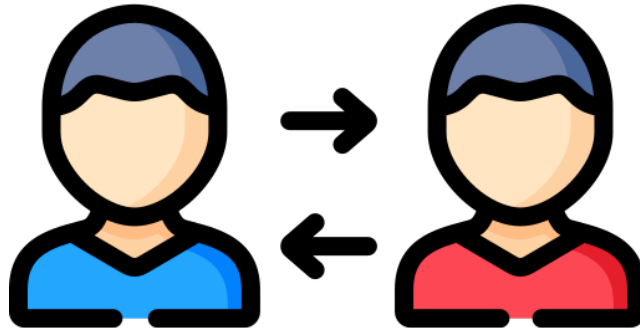
That is, while there is value in the items on the right, we value the items on the left more."

▶ Outlining the Four Values



Individuals and Interactions

over
Processes and Tools



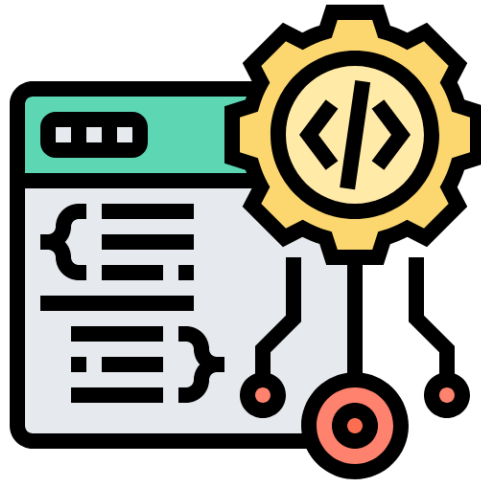
▶ Outlining the Four Values



Working Software

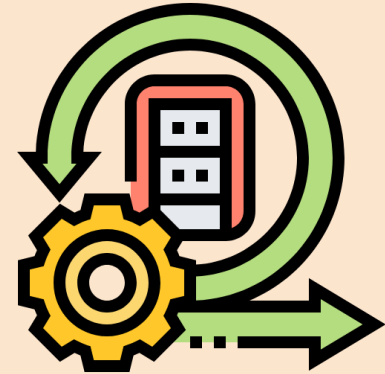
over

Comprehensive Documentation





Agile approaches can only be applied in software development, you can not use them for other types of products.



▶ Outlining the Four Values

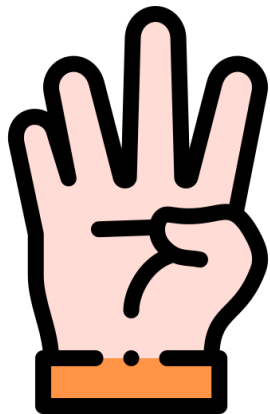


Customer collaboration

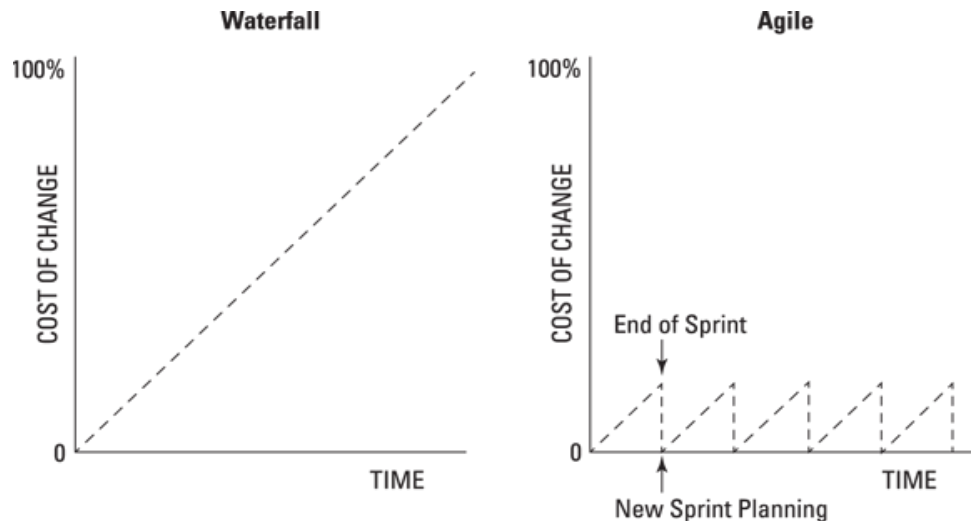
over
Contract Negotiation



▶ Outlining the Four Values



Responding to change over Following a Plan



Manifesto for Agile Software Development

Individuals and interactions over processes and tools

Working software over comprehensive documentation

..... over contract negotiation

Responding to change over following a plan



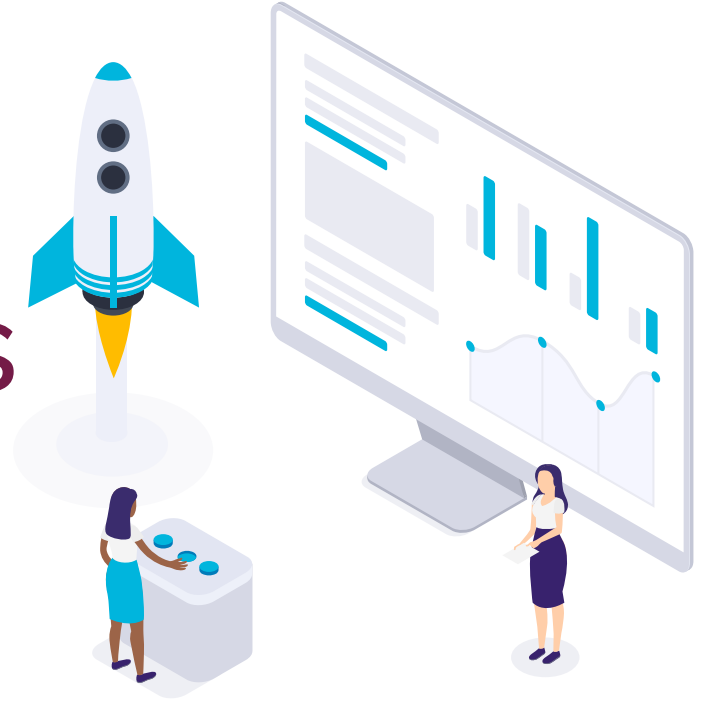
Students choose an option

Pear Deck Interactive Slide
Do not remove this bar



3

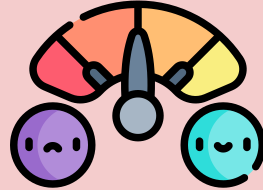
Agile Principles



Agile Principles



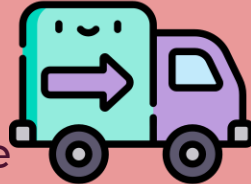
- 1** Our highest priority is to **satisfy the customer** through early and continuous delivery of valuable software.



- 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.



Agile Principles



- 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.



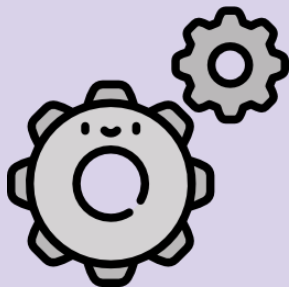
- 8** Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.



Agile Principles



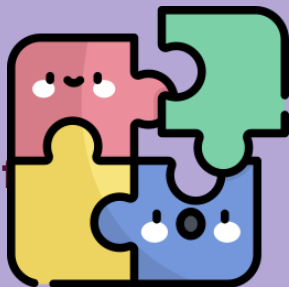
- 9** **Continuous attention** to technical excellence and good design enhances agility.



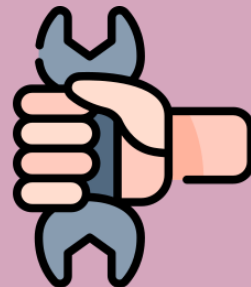
- 10** **Simplicity**--the art of maximizing the amount of work not done--is essential.



- 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.



12 Agile Principles



1 Satisfy the **customer**



2 Welcome **change**



3 Deliver **frequently**

Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5
story	story	story	story	story
story	story	story	story	story
story	story	story	story	story

4 Work **together**



5 Trust and **support**



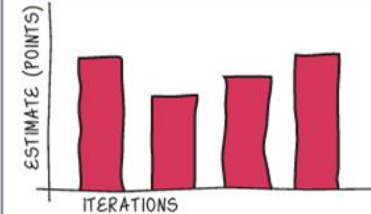
6 Face-to-face **conversation**



7 Working **software**



8 Sustainable **development**



9 Continuous **attention**



10 Maintain **simplicity**



11 Self-organizing **teams**



12 Reflect and **adjust**



Which one is NOT an Agile Principle?

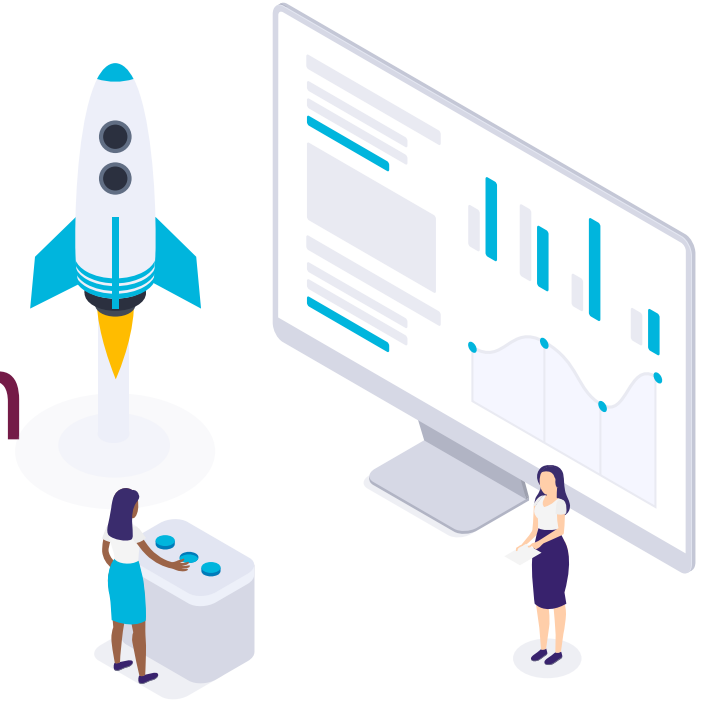
- A. Satisfy the boss
- B. Welcome change
- C. Maintain simplicity
- D. Work together





4

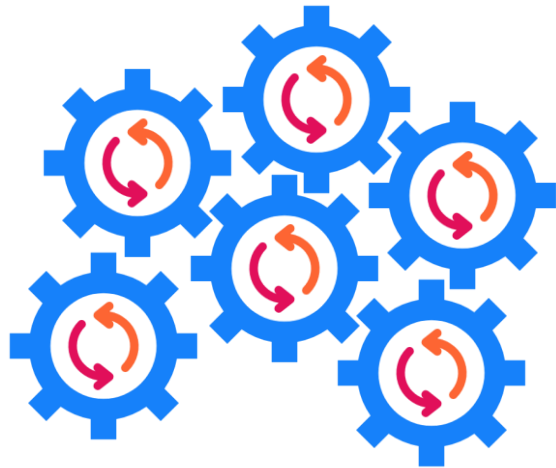
Overview of Agile Approach



Changes as a Result of Agile Values



Agile approaches changed attitudes toward project management processes.

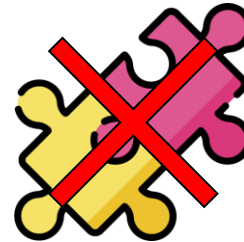


Too much process

=



Problem



Solution

Changes as a Result of Agile Values



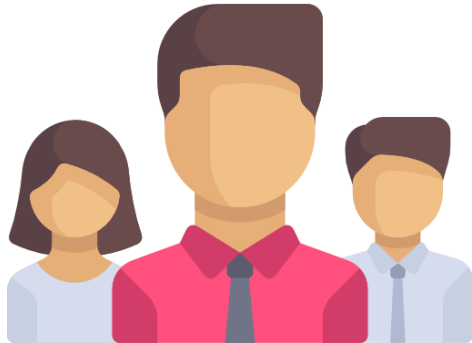
Agile approaches changed attitudes toward knowledge workers.





Changes as a Result of Agile Values

Agile approaches changed the relationship between business and IT groups.



Business Team



IT Team

Changes as a Result of Agile Values

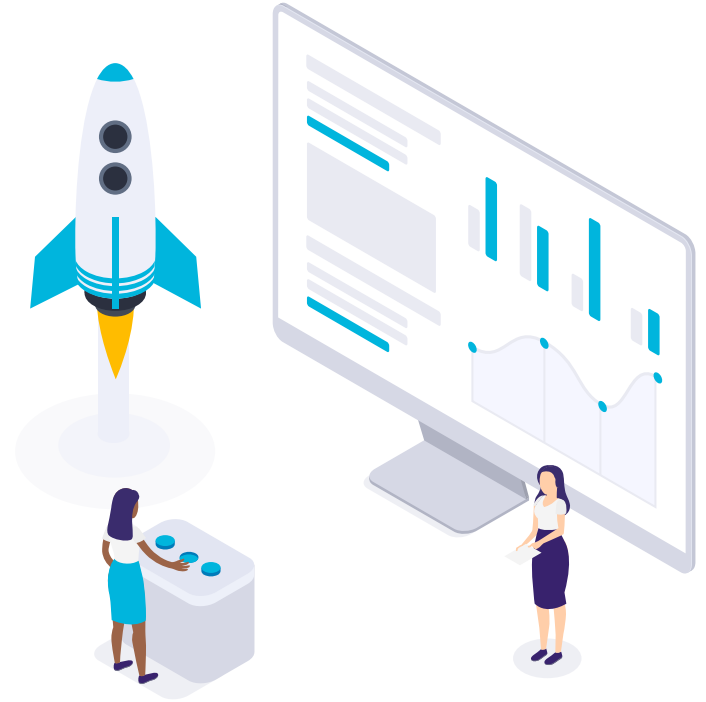


Agile approaches corrected attitudes toward change.





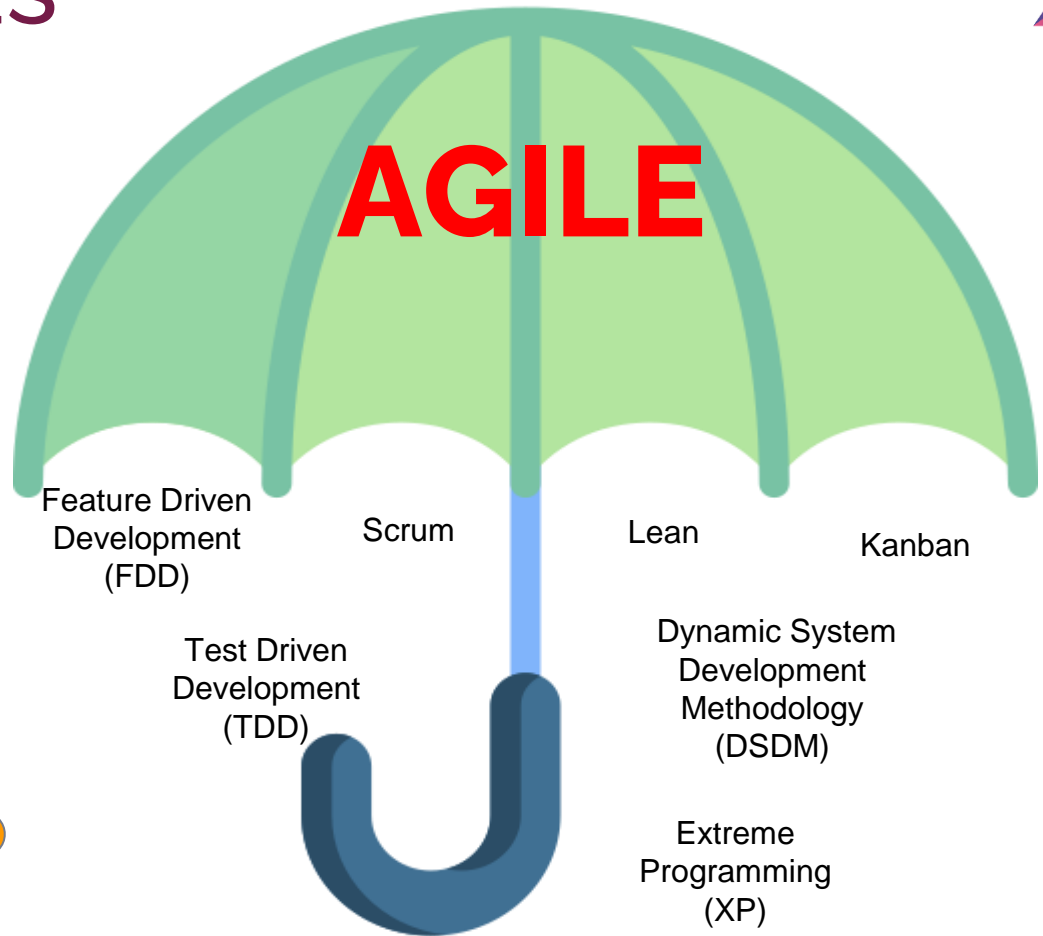
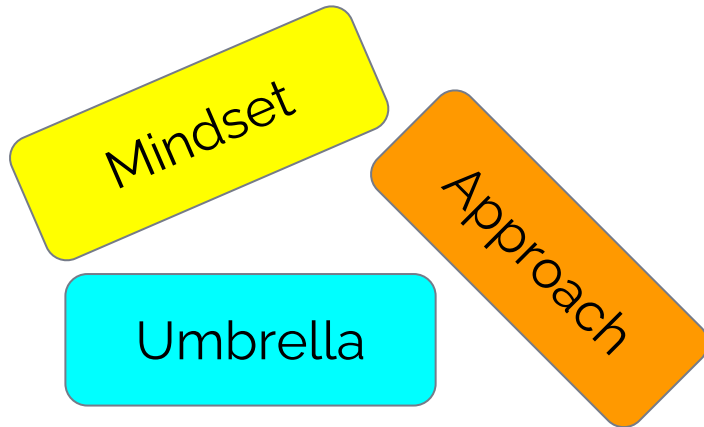
Agile Methods



Agile Methods



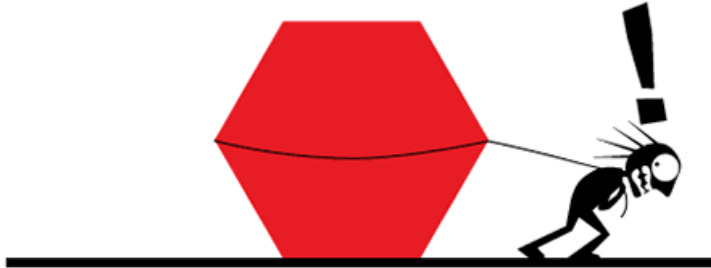
Agile is an umbrella under which many specific methodologies have been developed and are thriving.



Let's Wrap it Up



THE WATERFALL PROCESS



*'This project has got so big,
I'm not sure I'll be able to deliver it!'*

THE AGILE PROCESS



*'It's so much better delivering this
project in bite-sized sections'*

The product can be easily and rapidly developed and tested by dividing it into small pieces.

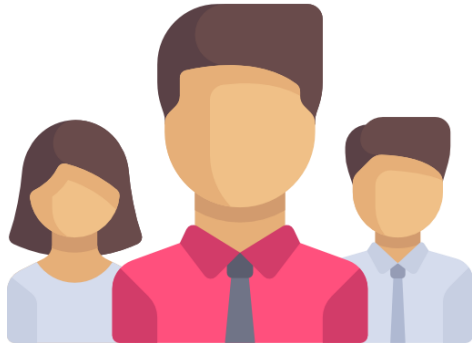
Small pieces are produced in iteration cycles.

By using iterations, changes can be made without having to wait for the final product.



Changes as a Result of Agile Values

Agile approaches changed the relationship between business and IT groups.



Business Team



IT Team

In one minute,
explain the most
important thing
about Agile.



Kahoot!





5

Scrum Overview



What is Scrum?



A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.



Lightweight



Simple to understand

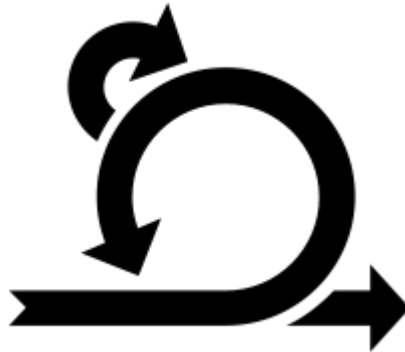


Difficult to master

What is Scrum?



Scrum emphasizes delivering business value frequently through short iterations known as sprints.



This gives visibility to the work that's being done and creates opportunities for feedback.

Origin of Scrum

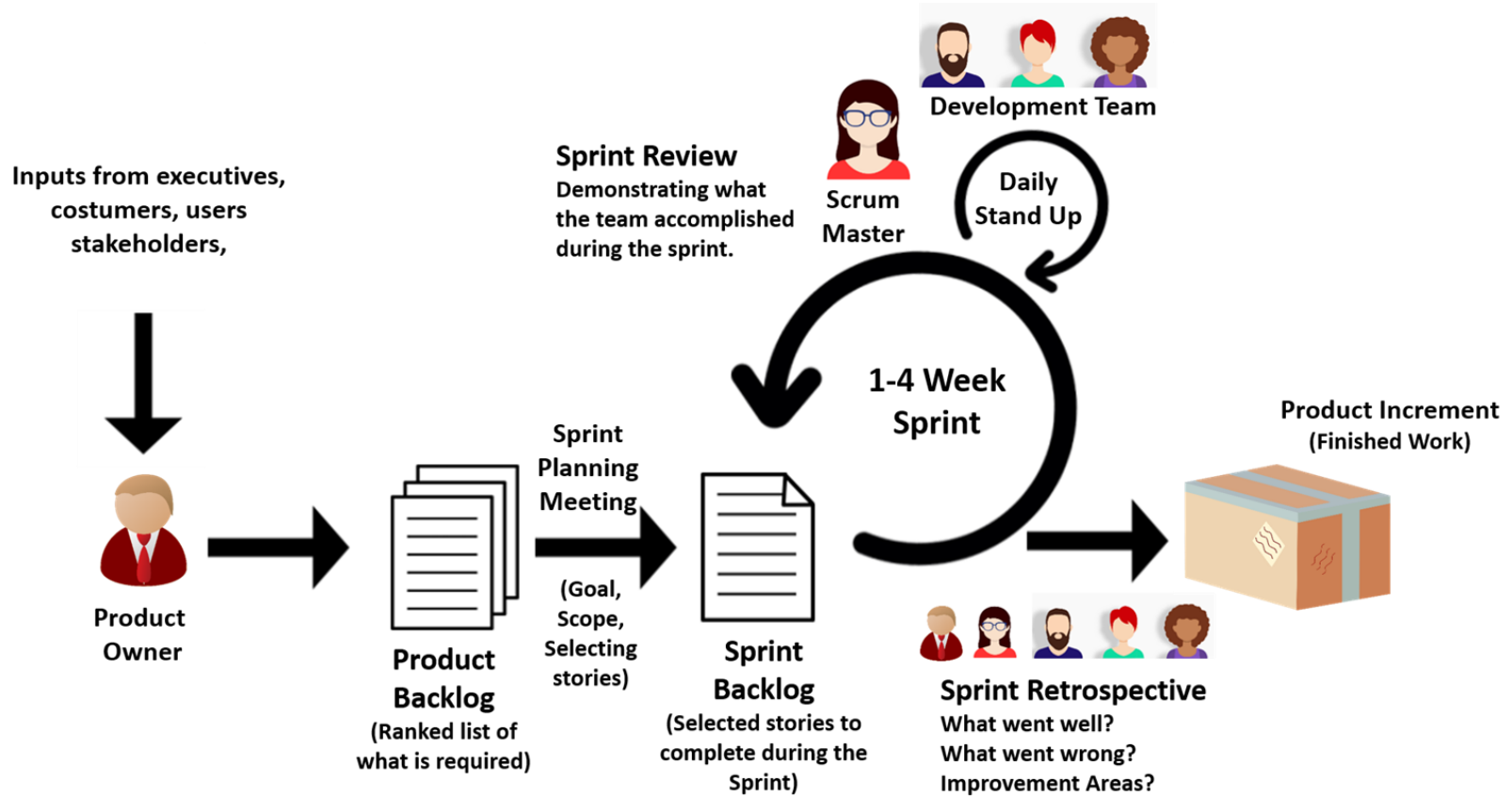


Scrum is one of the leading agile techniques developed in the 1990s by Ken Schwaber and Jeff Sutherland.

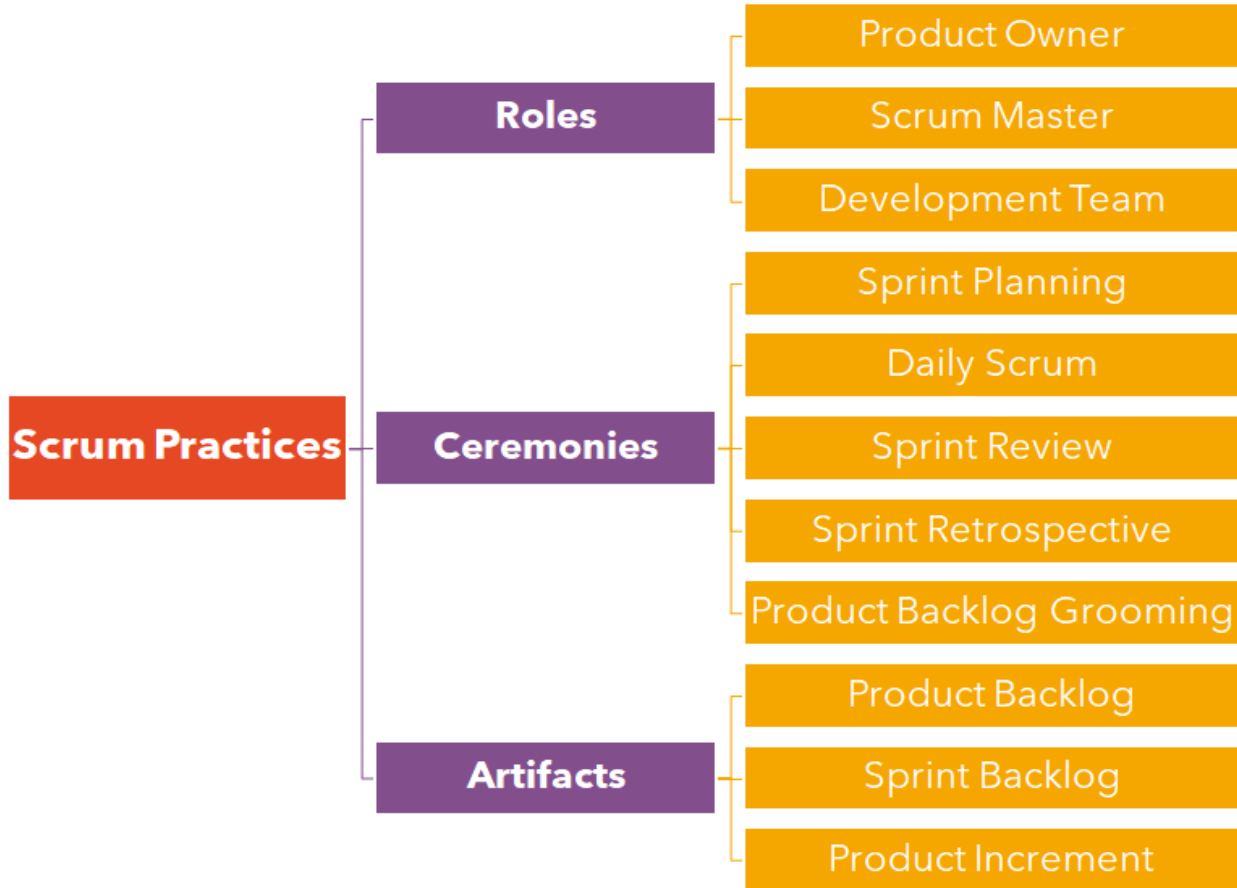


More than 50% of all projects use this methodology.

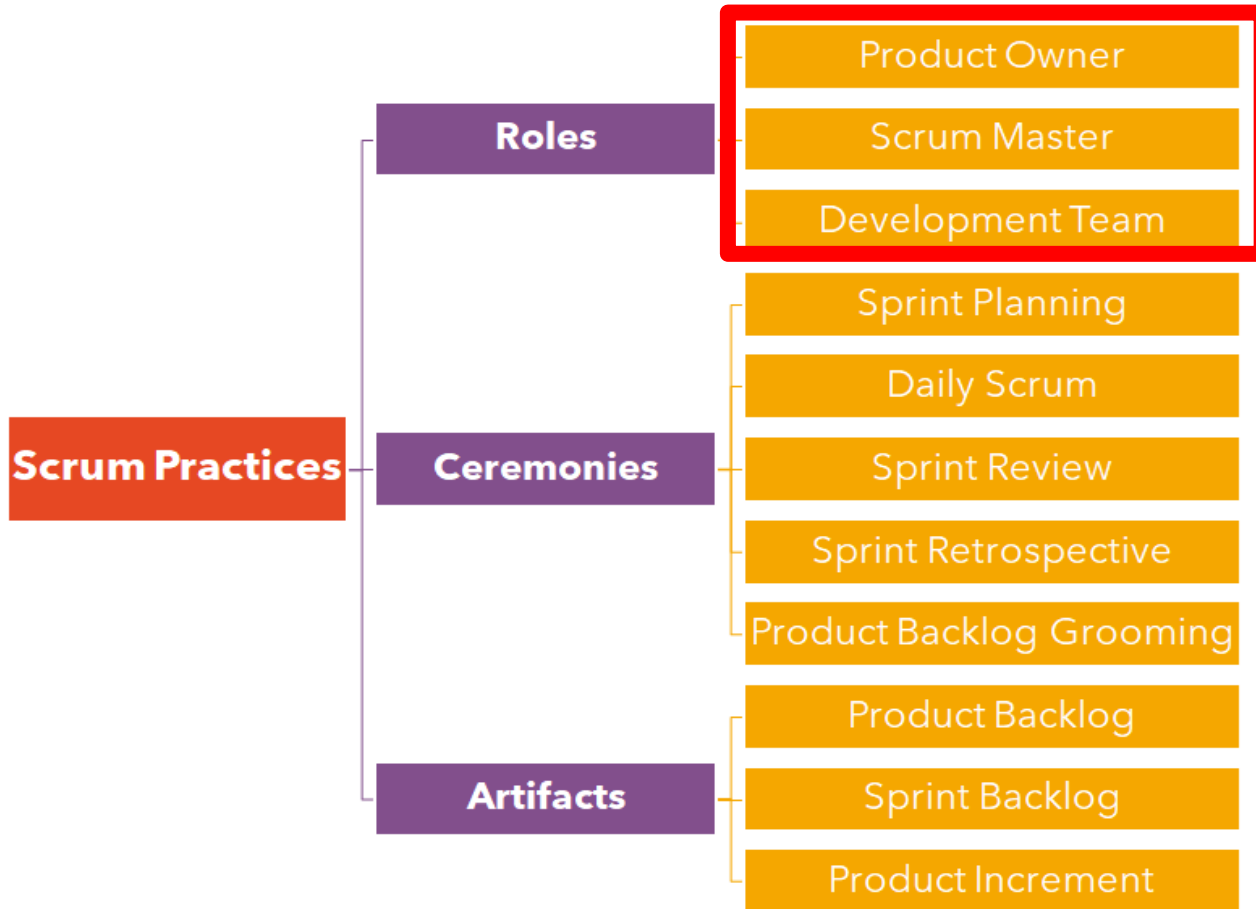
Scrum Framework



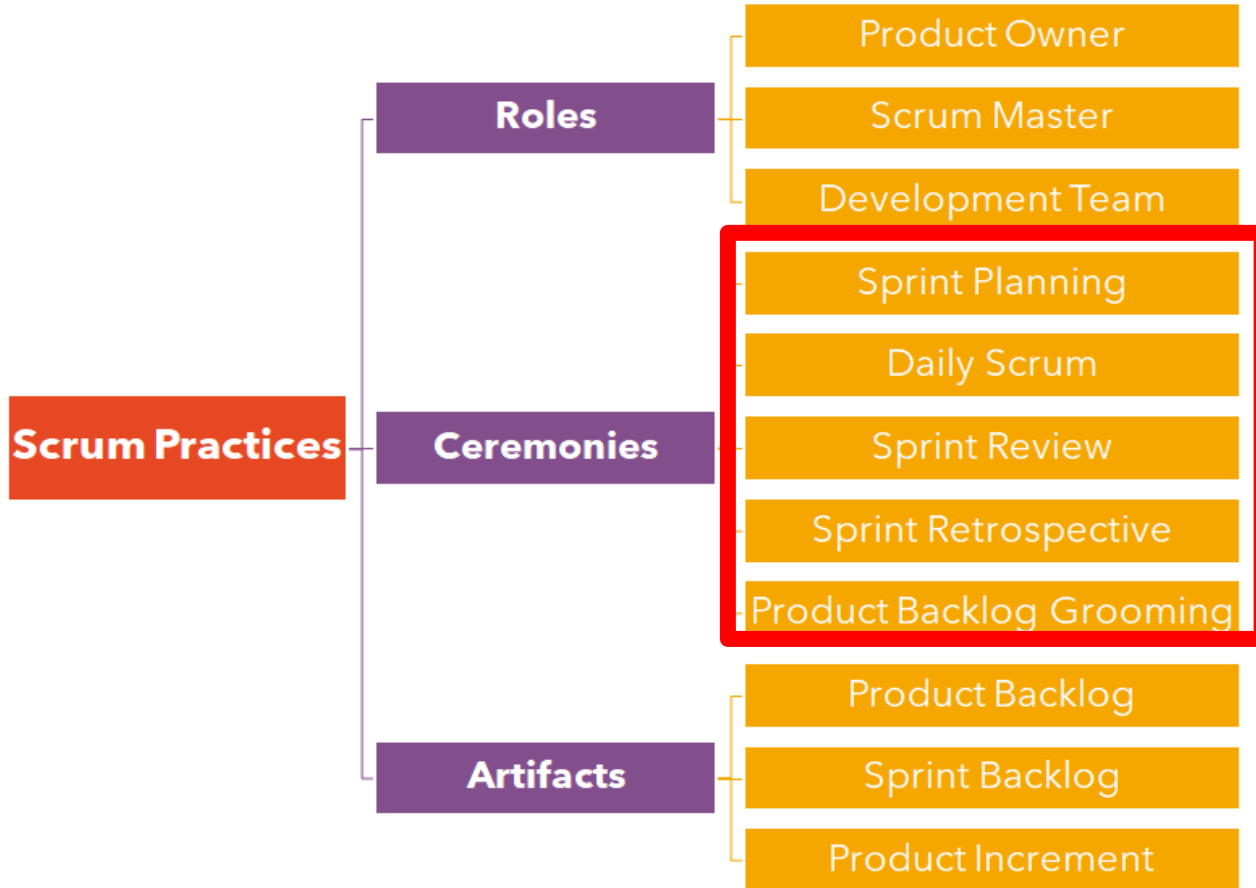
Scrum Practices



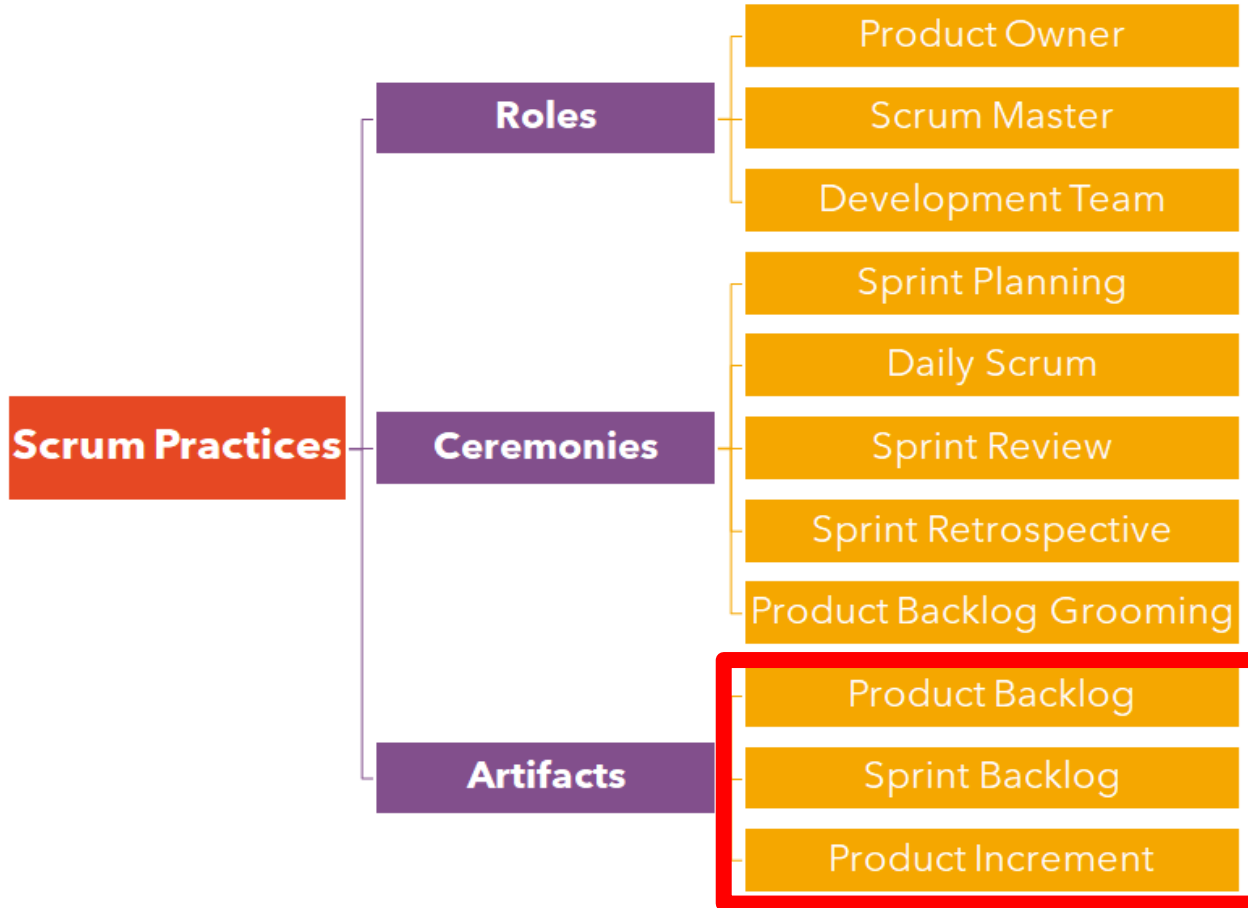
Scrum Practices

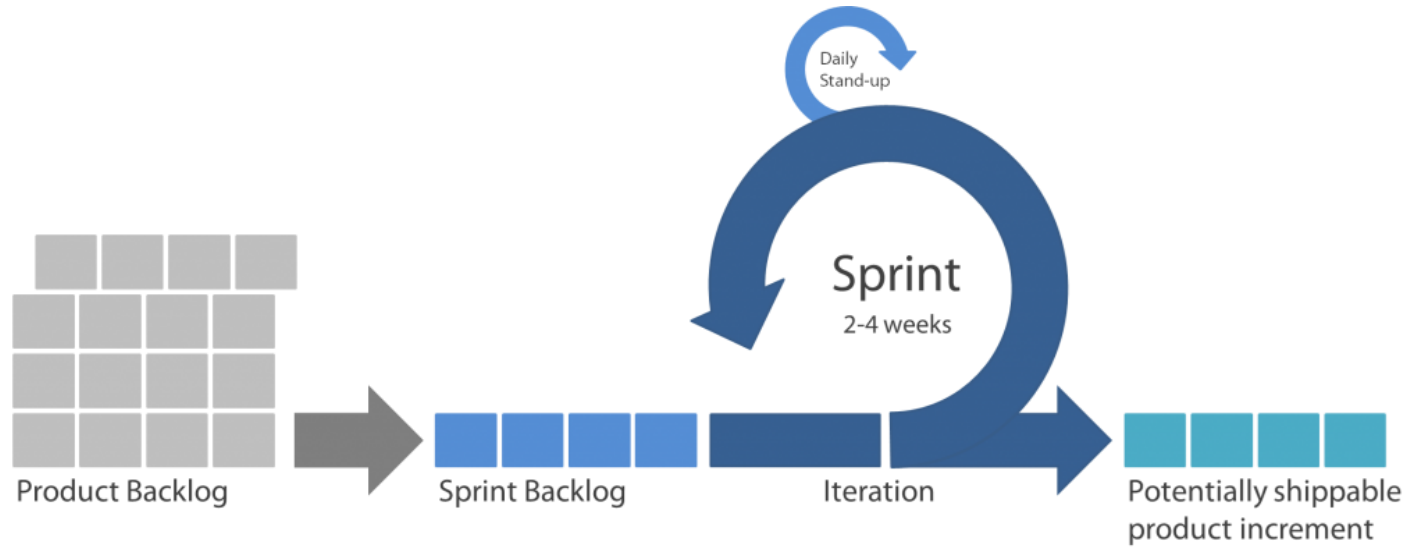


Scrum Practices



Scrum Practices





What is Scrum?





6

Scrum Roles



Scrum Roles



Product Owner



Responsible for the project's success by defining the project vision, requirements, and priorities

Scrum Master



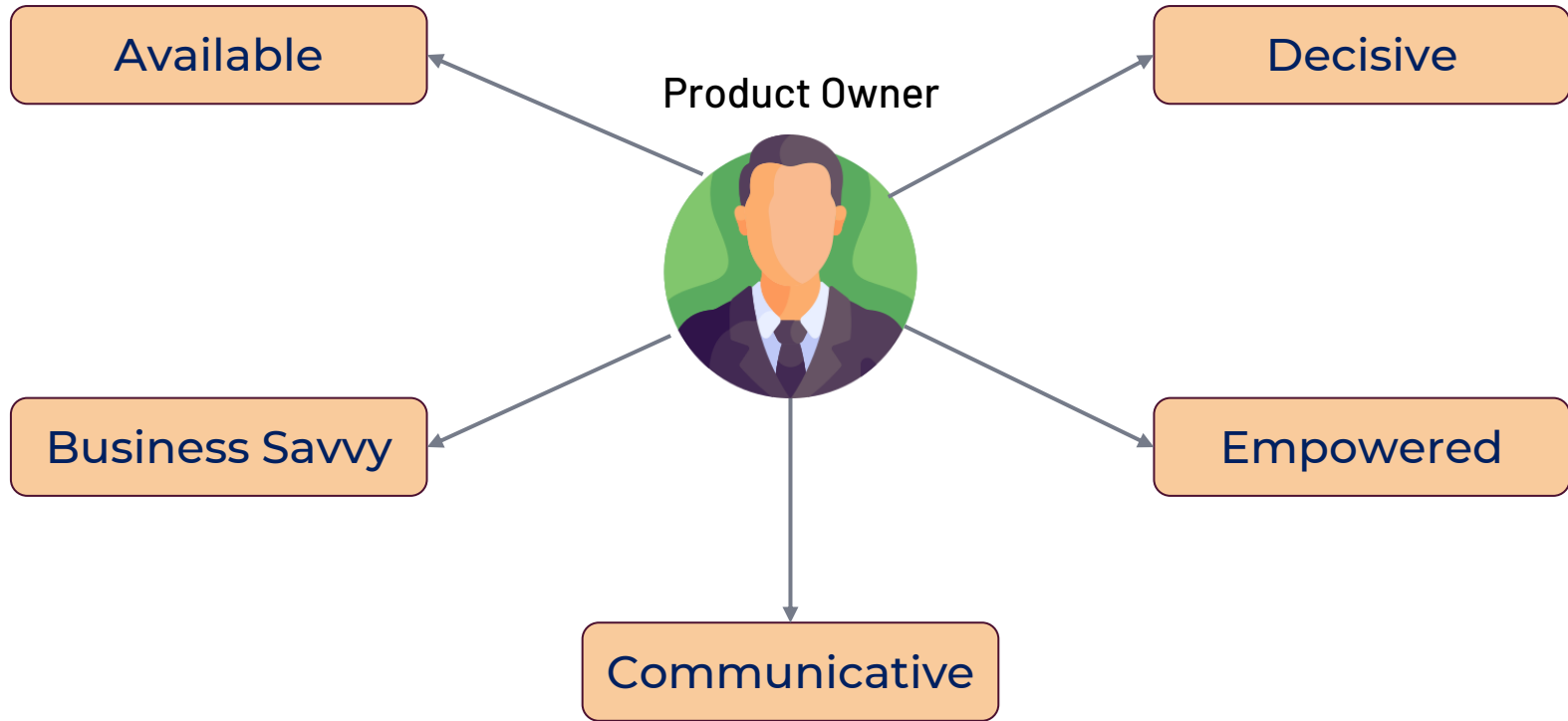
Accountable to the team to remove impediments that will prevent them from achieving the goals of the Product Owner

Development Team



Team comprises 3-9 people, with a mix of roles, and self-organizes to determine how to best meet the goals of the Product Owner

Product Owner



Product Owner

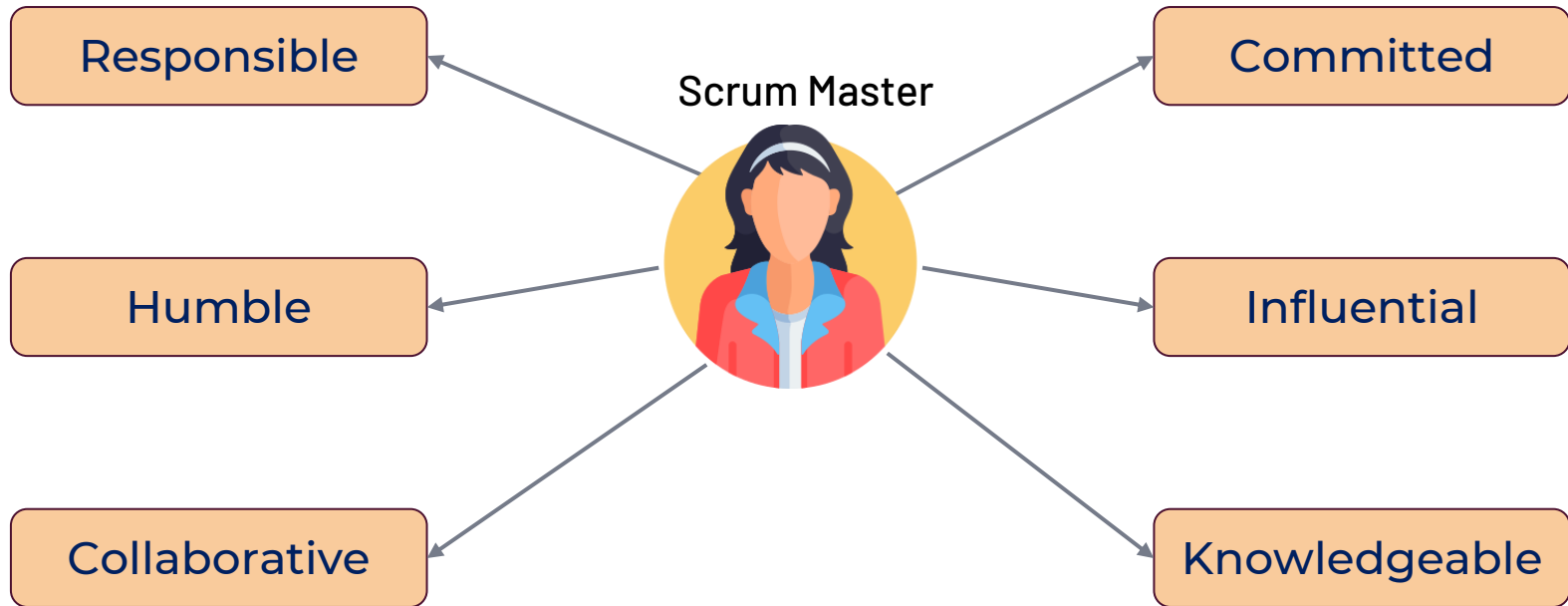


- ☒ Clearly expressing Product Backlog items
- ☒ Ordering the items in the Product Backlog to best achieve goals and missions
- ☒ Optimizing the value of the work the Development Team performs
- ☒ Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next
- ☒ Ensuring the Development Team understands items in the Product Backlog to the level needed

Scrum Master



The Scrum Master is responsible for promoting and supporting Scrum



Scrum Master



Scrum Master Service to the Product Owner

- ✓ Ensuring that goals, scope, and product domain are understood
- ✓ Finding techniques for effective Product Backlog management
- ✓ Helping the Scrum Team understand the need for clear and concise Product Backlog items
- ✓ Understanding product planning in an empirical environment
- ✓ Ensuring the Product Owner knows how to arrange the Product Backlog to maximize value
- ✓ Understanding and practicing agility
- ✓ Facilitating Scrum events as requested or needed

Scrum Master



Scrum Master Service to the Development Team

- ☒ Coaching the Development Team in self-organization and cross-functionality
- ☒ Helping the Development Team to create high-value products
- ☒ Removing impediments to the Development Team's progress
- ☒ Facilitating Scrum events as requested or needed
- ☒ Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood



Which one looks at the project from the customer's perspective?



Students choose an option



What is the role of the Scrum Master?



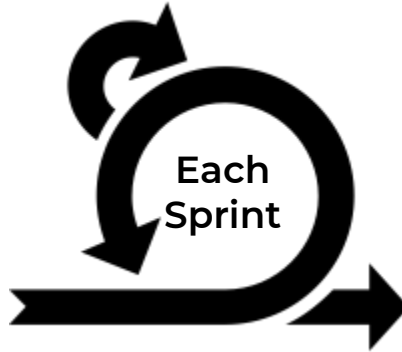
Students choose an option

Development Team



The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of “Done” product at the end of each Sprint.

Development Team



Product Increment



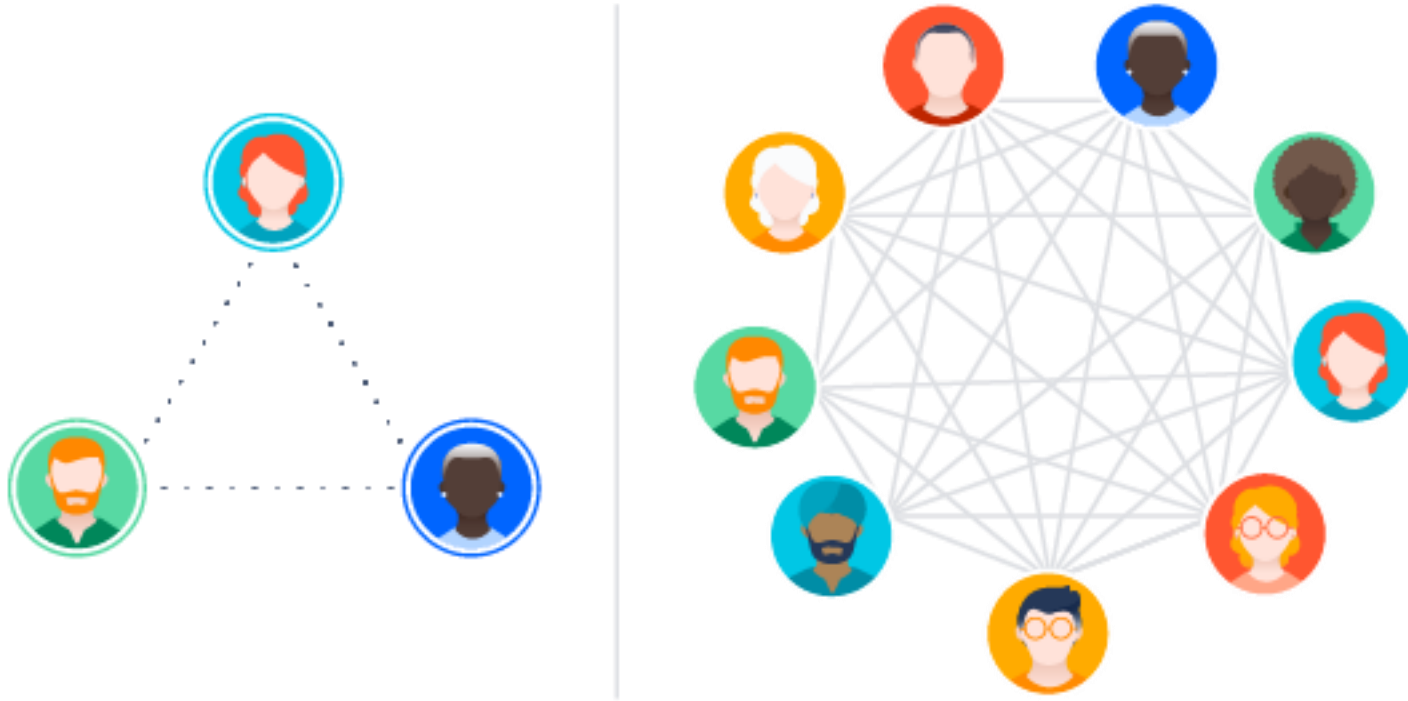
Development Team



- ☒ They are self-organizing.
- ☒ Development Teams are cross-functional.
- ☒ Scrum recognizes no titles for Development Team members.
- ☒ Scrum recognizes no sub-teams in the Development Team.
- ☒ Accountability belongs to the Development Team as a whole



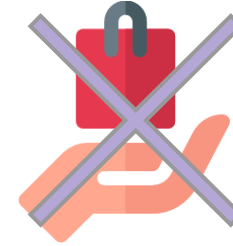
Team Size vs. Coordination



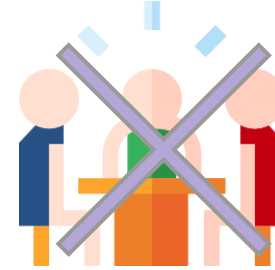
Development Team Size



Small Team



Large Team



3-9 Members



Coordination
Product Increment

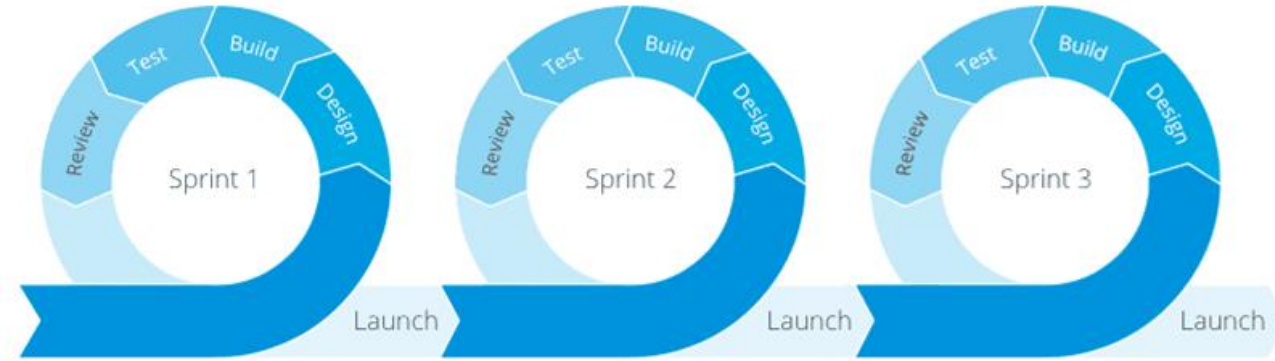


What do we mean by a cross-functional Development Team?



Students choose an option

Agile Model



Each iteration lasts from 1 - 3 weeks

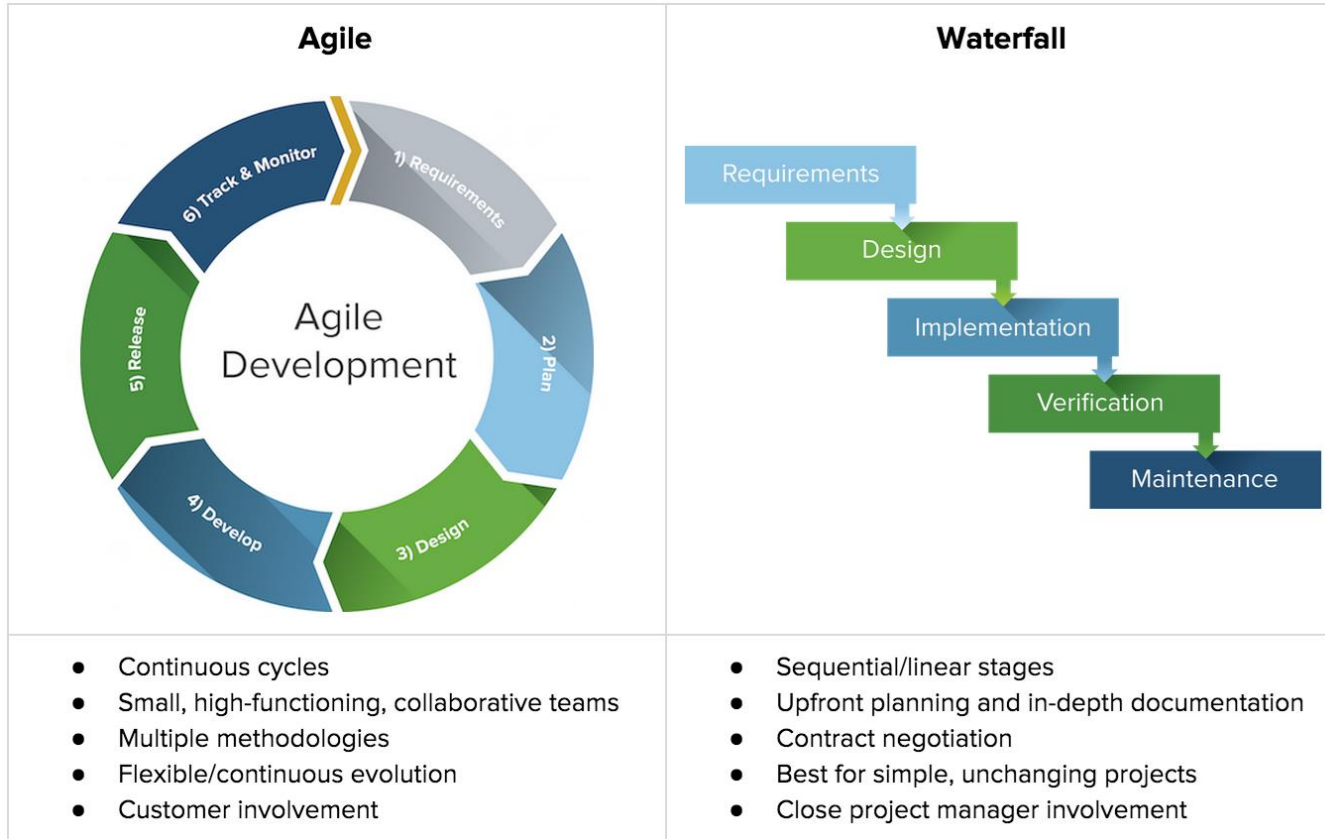


Agile Model

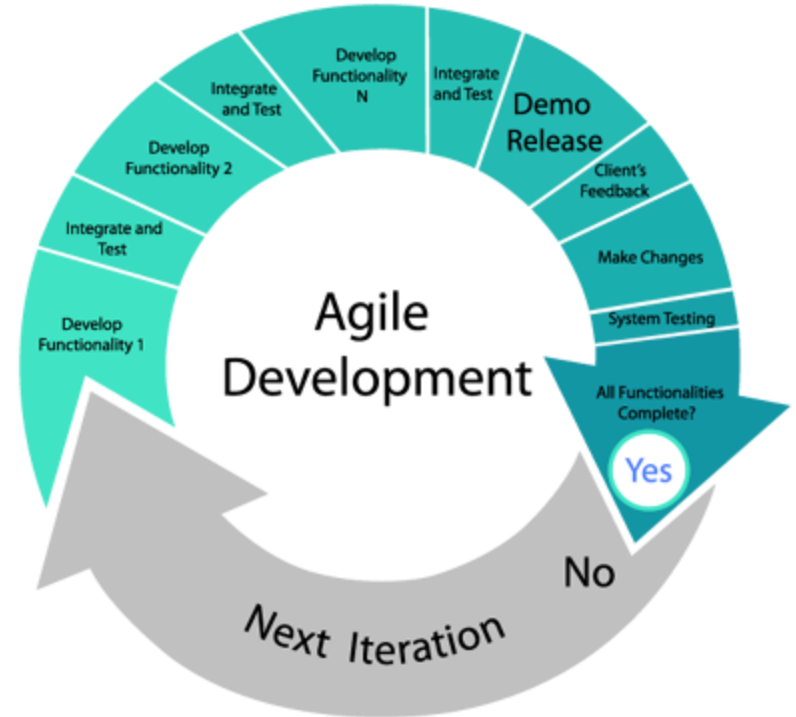


AGILE SOFTWARE DEVELOPMENT

Agile Vs Traditional SDLC Models



Agile Vs Traditional SDLC Models





THANKS!

Any questions?

