



OAK INTERLINK

SCRUM FOUNDATION PROFESSIONAL

Certification Course

Services We Offer



OAK INTERLINK



Service 01

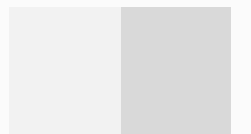
TRAINING.

Service 02

CONSULTING

Service 03

OUTSOURCING.





Course Objectives

What knowledge and skills learners will gain:

01 To give participants basic knowledge of Agile Scrum

02 Acquire skills to adequately plan for scrum implementation

03 Acquires skills to develop products in the workplace

04 Gain a certificate in Scrum foundation



Course Objectives





OAK INTERLINK

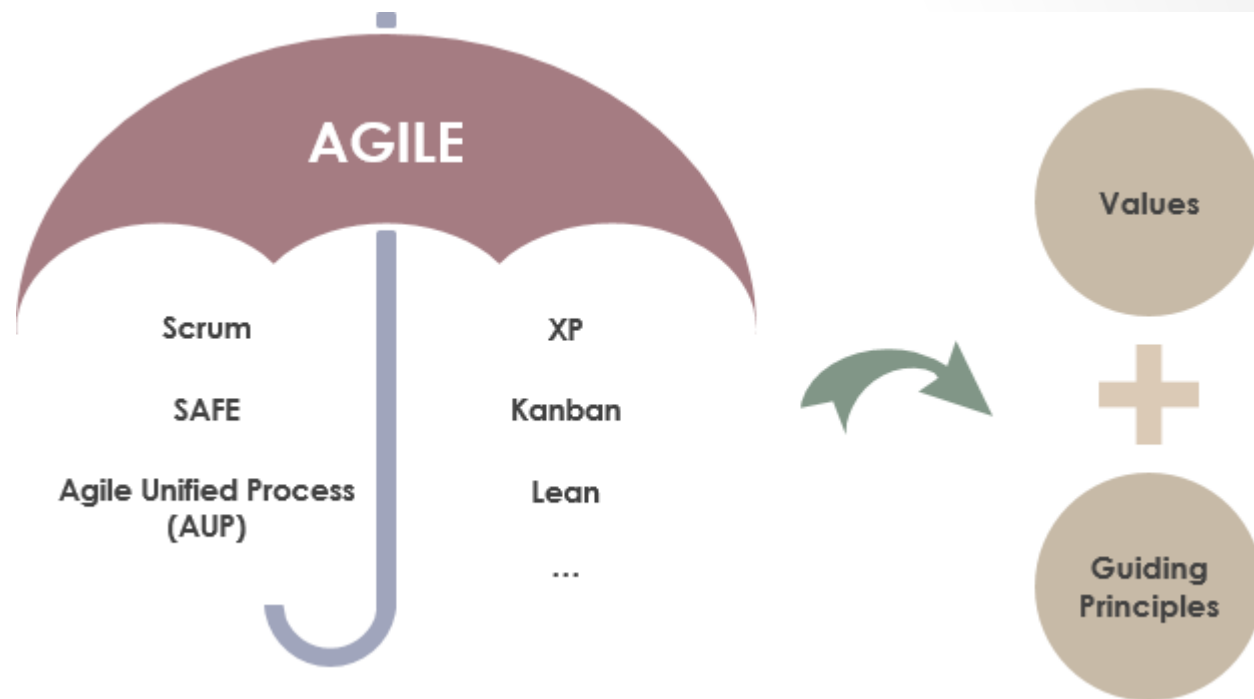
PRE COURSE EXERCISE



OAK INTERLINK

Pre Course Exercise

- 10 questions





What is AGILE?

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

incrementally



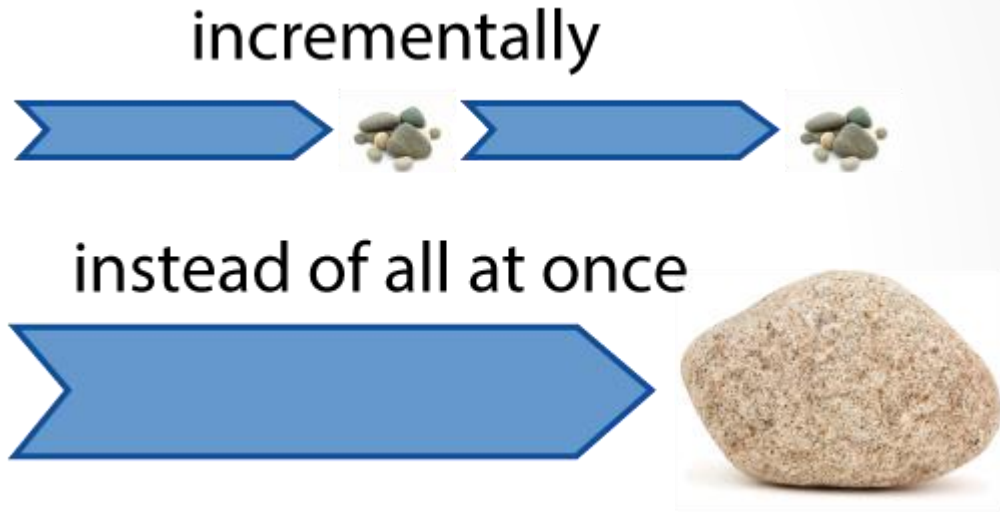
instead of all at once





What is AGILE?

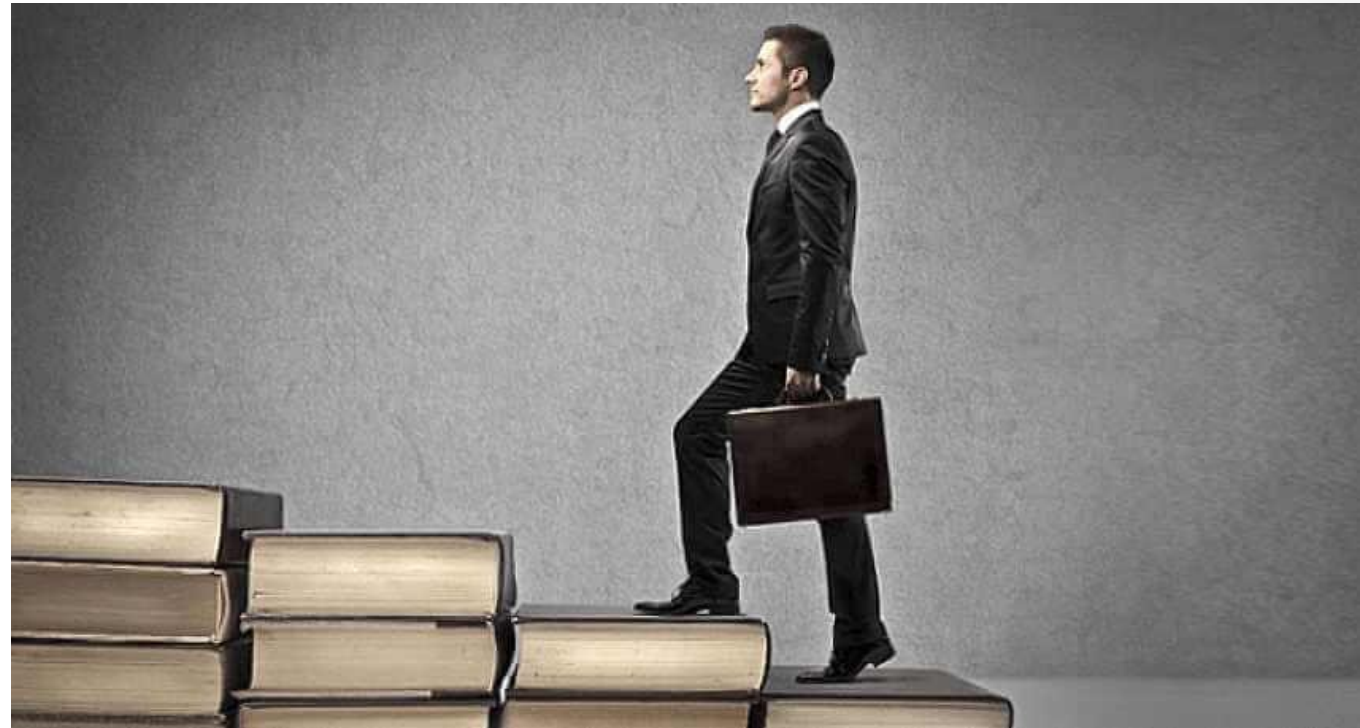
Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster. Instead of relying on a "big bang" launch, an agile team delivers work in small, but consumable, increments. Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly.





LIST OF WIDELY USED AGILE METHODOLOGIES:

- Agile Scrum Methodology
- Lean Software Development
- Kanban
- Extreme Programming (XP)
- Crystal
- Dynamic Systems Development Method (DSDM)
- Feature Driven Development (FDD)





OAK INTERLINK

WHERE DID AGILE COME FROM?

In 2001, a small group of people, tired of the traditional approach to managing software development projects, designed the agile manifesto. It is a more improved method for managing the progress of software projects.



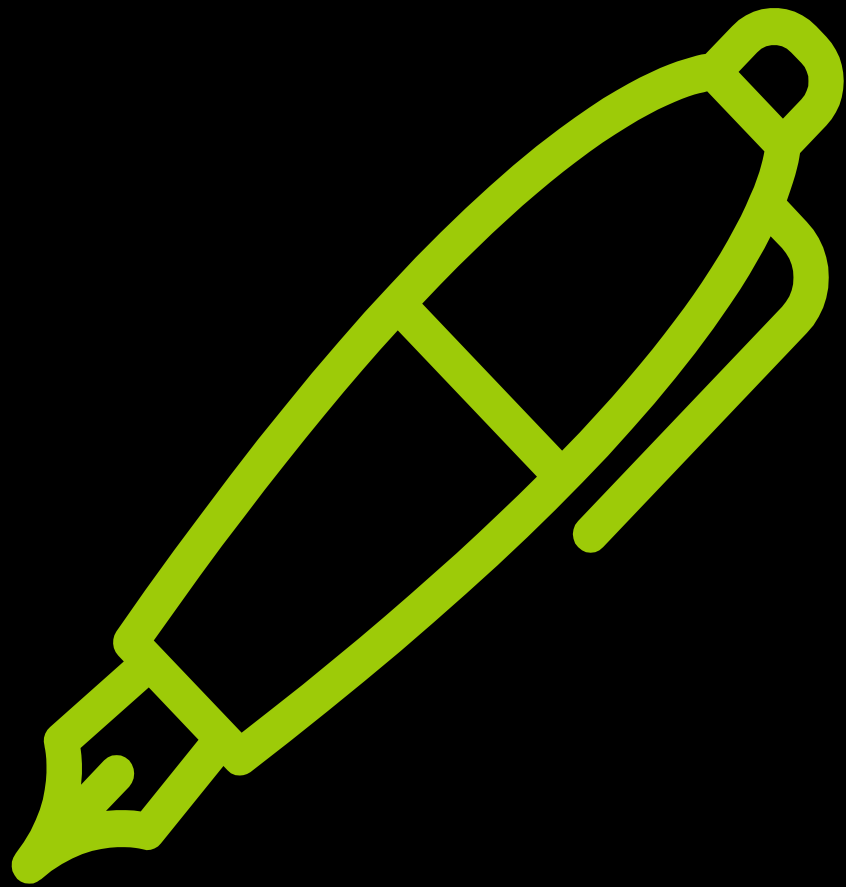


OAK INTERLINK

Module 1

AGILE

Understanding Agile



The Agile Manifesto

ABOUT AGILE



Requirements

Software

Development

AGILE

Debugging

Design

Method

Testing

Maintenance

Deployment



The Four Values of the Agile Manifesto



The Agile Manifesto is comprised of four foundational values and 12 supporting principles which lead the Agile approach to software development. Each Agile methodology applies the four values in different ways, but all of them rely on them to guide the development and delivery of high-quality, working software.

1. Individuals and Interactions Over Processes and Tools



OAK INTERLINK



2. Working Software Over Comprehensive Documentation



OAK INTERLINK





OAK INTERLINK

3. Customer Collaboration Over Contract Negotiation





4. Responding to Change Over Following a Plan



Principles of Agile Manifesto





The Twelve Agile Manifesto Principles

The Twelve Principles are the guiding principles for the methodologies that are included under the title “The Agile Movement.” They describe a culture in which change is welcome, and the customer is the focus of the work. They also demonstrate the movement’s intent as described by Alistair Cockburn, one of the signatories to the Agile Manifesto, which is to bring development into alignment with business needs.

The 12 Principles of Agile Development

1. Customer satisfaction through early and continuous software delivery – Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.

2. Accommodate changing requirements throughout the development process – The ability to avoid delays when a requirement or feature request changes.

3. Frequent delivery of working software – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.

4. Collaboration between the business stakeholders and developers throughout the project – Better decisions are made when the business and technical team are aligned.

5. Support, trust, and motivate the people involved – Motivated teams are more likely to deliver their best work than unhappy teams.

6. Enable face-to-face interactions – Communication is more successful when development teams are co-located.

The 12 Principles of Agile Development (Continues)

7. Working software is the primary measure of progress – Delivering functional software to the customer is the ultimate factor that measures progress.

8. Agile processes to support a consistent development pace – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.

9. Attention to technical detail and design enhances agility – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.

10. Simplicity – Develop just enough to get the job done for right now.

11. Self-organizing teams encourage great architectures, requirements, and designs – Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.

12. Regular reflections on how to become more effective – Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently. The intention of Agile is to align development with business needs, and the success of Agile is apparent. Agile projects are customer focused and encourage customer guidance and participation. As a result, Agile has grown to be an overarching view of software development throughout the software industry and an industry all by itself.

Agile Model	Waterfall Model
•Agile method proposes incremental and iterative approach to software design	•Development of the software flows sequentially from start point to end point.
•The agile process is broken into individual models that designers work on	•The design process is not broken into an individual models
•The customer has early and frequent opportunities to look at the product and make decision and changes to the project	•The customer can only see the product at the end of the project
•Agile model is considered unstructured compared to the waterfall model	•Waterfall model are more secure because they are so plan oriented
•Small projects can be implemented very quickly. For large projects, it is difficult to estimate the development time.	•All sorts of project can be estimated and completed.
•Error can be fixed in the middle of the project.	•Only at the end, the whole product is tested. If the requirement error is found or any changes have to be made, the project has to start from the beginning
•Development process is iterative, and the project is executed in short (2-4) weeks iterations. Planning is very less.	•The development process is phased, and the phase is much bigger than iteration. Every phase ends with the detailed description of the next phase.
•Documentation attends less priority than software development	•Documentation is a top priority and can even use for training staff and upgrade the software with another team
•Every iteration has its own testing phase. It allows implementing regression testing every time new functions or logic are released.	•Only after the development phase, the testing phase is executed because separate parts are not fully functional.
•In agile testing when an iteration end, shippable features of the product is delivered to the customer. New features are usable right after shipment. It is useful when you have good contact with customers.	•All features developed are delivered at once after the long implementation phase.
•Testers and developers work together	•Testers work separately from developers
•At the end of every sprint, user acceptance is performed	•User acceptance is performed at the end of the project.
•It requires close communication with developers and together analyze requirements and planning	•Developer does not involve in requirement and planning process. Usually, time delays between tests and coding





OAK INTERLINK

Module 2

Introduction to Scrum

Understanding Scrum



What is Scrum?

“ Scrum is a subset of Agile. It is a lightweight process framework for agile development, and the most widely-used one.



Purpose of the Scrum

“ Many companies are realizing that the traditional project management principles are no longer applicable in today’s extremely dynamic and demanding environment.

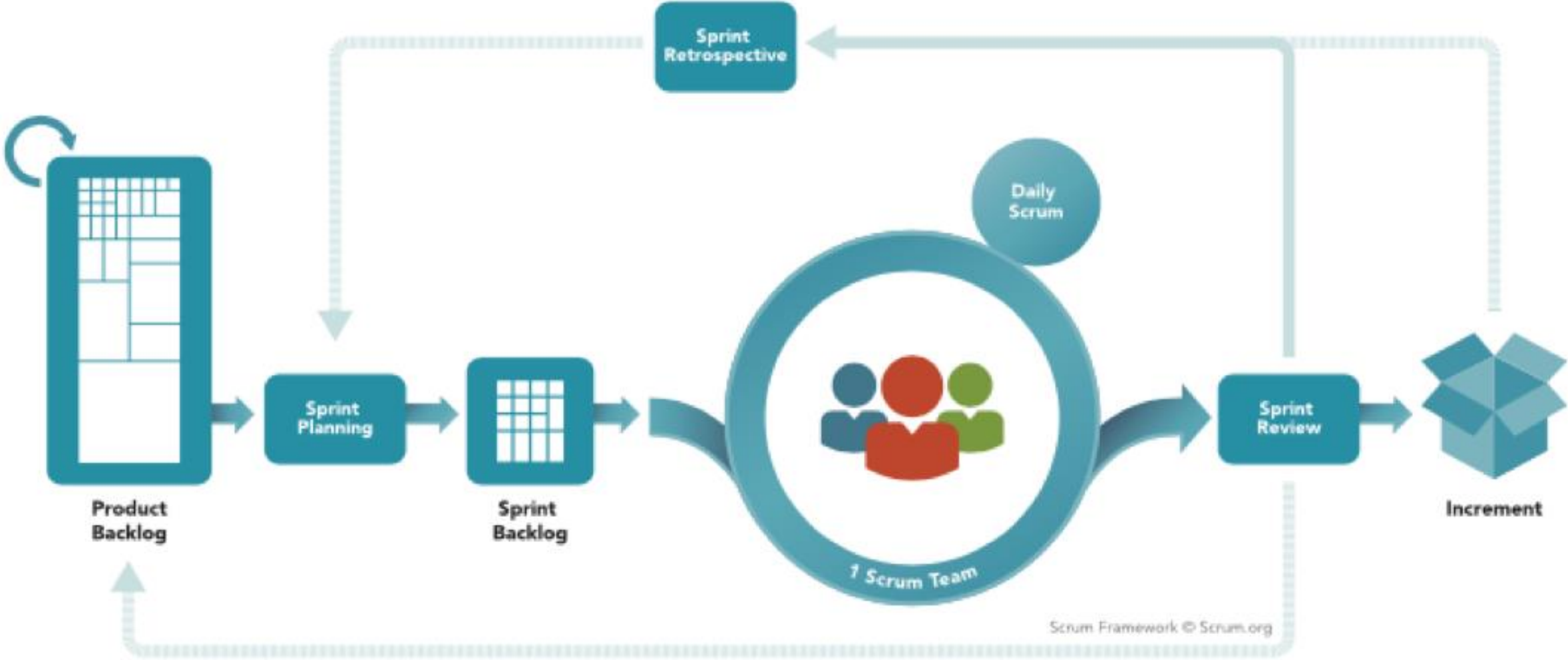
A fixed approach of planning is no longer a good method since it is very difficult to anticipate everything in advance and respond to it in the plan. Instead, more agile planning methods are required to accommodate for changing markets, technology and resources over the course of a typical project.



Purpose of the Scrum

“ Scrum is a framework for developing, delivering, and sustaining complex products.

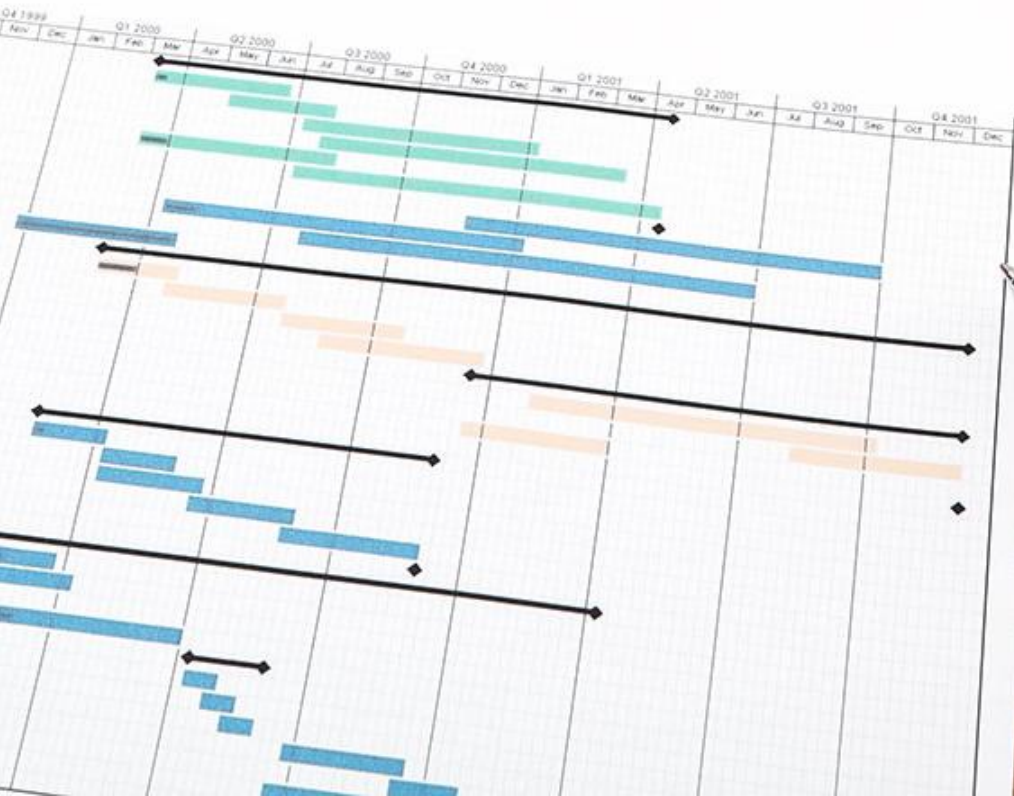
SCRUM FRAMEWORK





Definition of Scrum

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



Scrum is

- Lightweight
- Simple to understand
- Difficult to master



Definition of Scrum

- Scrum is a process framework that has been used to manage work on complex products since the early 1990s.
- Scrum is not a process, technique, or definitive method. Rather, it is a framework within which you can employ various processes and techniques





Definition of Scrum

- Scrum makes clear the relative efficacy of your product management and work techniques so that you can continuously improve the product, the team, and the working environment.





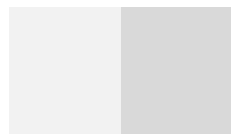
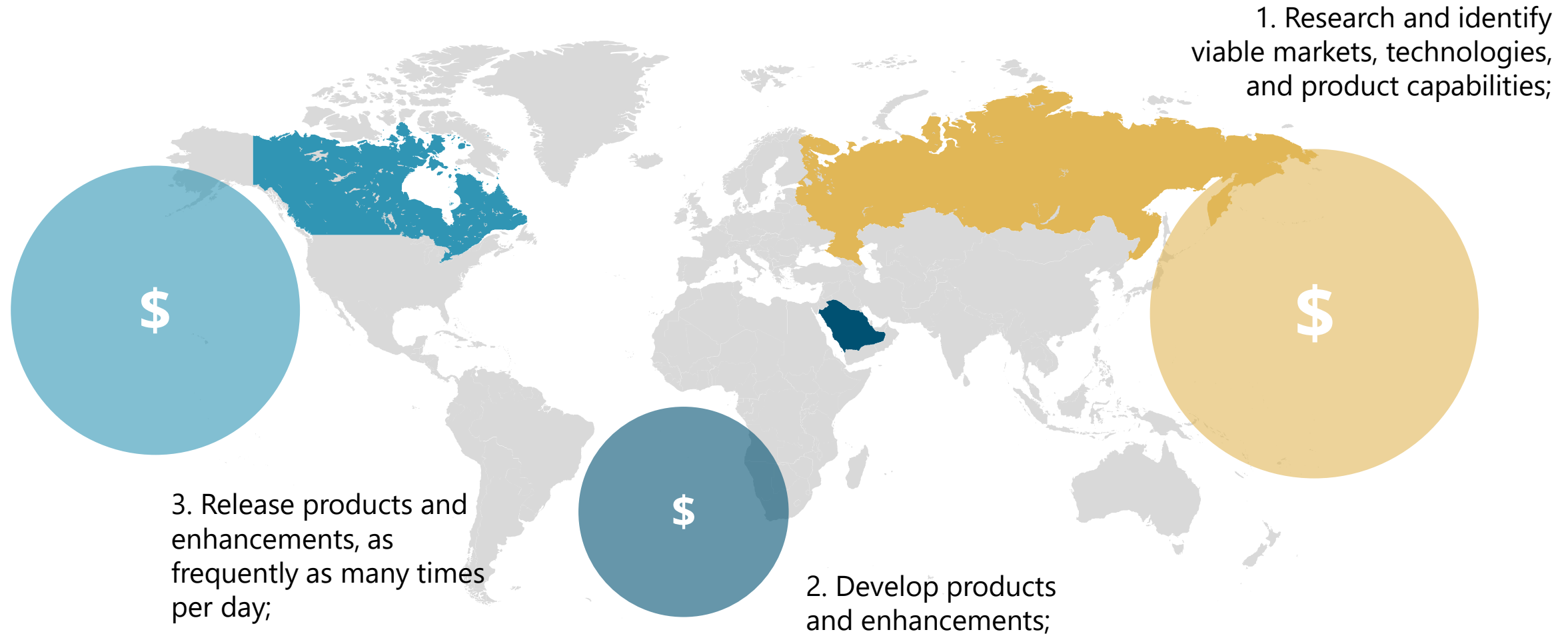
OAK INTERLINK

Uses of Scrum

Scrum was initially developed for managing and developing products.



Scrum has been used extensively, worldwide, to:





Scrum has been used extensively,
worldwide, to (Continues):

4. Develop and sustain Cloud (online, secure, on-demand) and other operational environments for product use; and,
5. Sustain and renew products



Use of Scrum

Scrum has been used to develop software, hardware, embedded software, networks of interacting function, autonomous vehicles, schools, government, marketing, managing the operation of organizations and almost everything we use in our daily lives, as individuals and societies.



Use of Scrum

As technology, market, and environmental complexities and their interactions have rapidly increased, Scrum's utility in dealing with complexity is proven daily. Scrum proved especially effective in iterative and incremental knowledge transfer. Scrum is now widely used for products, services, and the management of the parent organization.





Use of Scrum

The essence of Scrum is a small team of people. The individual team is highly flexible and adaptive. These strengths continue operating in single, several, many, and networks of teams that develop, release, operate and sustain the work and work products of thousands of people.



OAK INTERLINK

Module 3

Scrum Theory

Understanding Scrum

Scrum Theory

Scrum is founded on **empirical process control theory, or empiricism.**

Empiricism asserts that knowledge comes from experience and making decisions based on what is known.

Scrum employs an iterative, incremental approach to optimize predictability and control risk.



Scrum Theory

Three pillars uphold every implementation of empirical process control:

1. **Transparency**
2. **Inspection**
3. **Adaptation.**





Transparency

Significant aspects of the process must be visible to those responsible for the outcome.

Transparency requires those aspects be defined by a common standard so observers share a common understanding of what is being seen.



For Example



A common language referring to the process must be shared by all participants; and,



Those performing the work and those inspecting the resulting increment must share a common definition of "Done"



Inspection

Scrum users must frequently inspect Scrum artifacts and progress toward a Sprint Goal to detect undesirable variances.

Their inspection should not be so frequent that inspection gets in the way of the work. Inspections are most beneficial when diligently performed by skilled inspectors at the point of work





Adaptation

Adaptation If an inspector determines that one or more aspects of a process deviate outside acceptable limits, and that the resulting product will be unacceptable, the process or the material being processed must be adjusted.

An adjustment must be made as soon as possible to minimize further deviation.



ELEMENTS
OF
SCRUM
&
THE RULES THAT
-BIND-
THEM



Framework

Roles

- Product Owner
- Dev Team
- Scrum Master

Artifacts

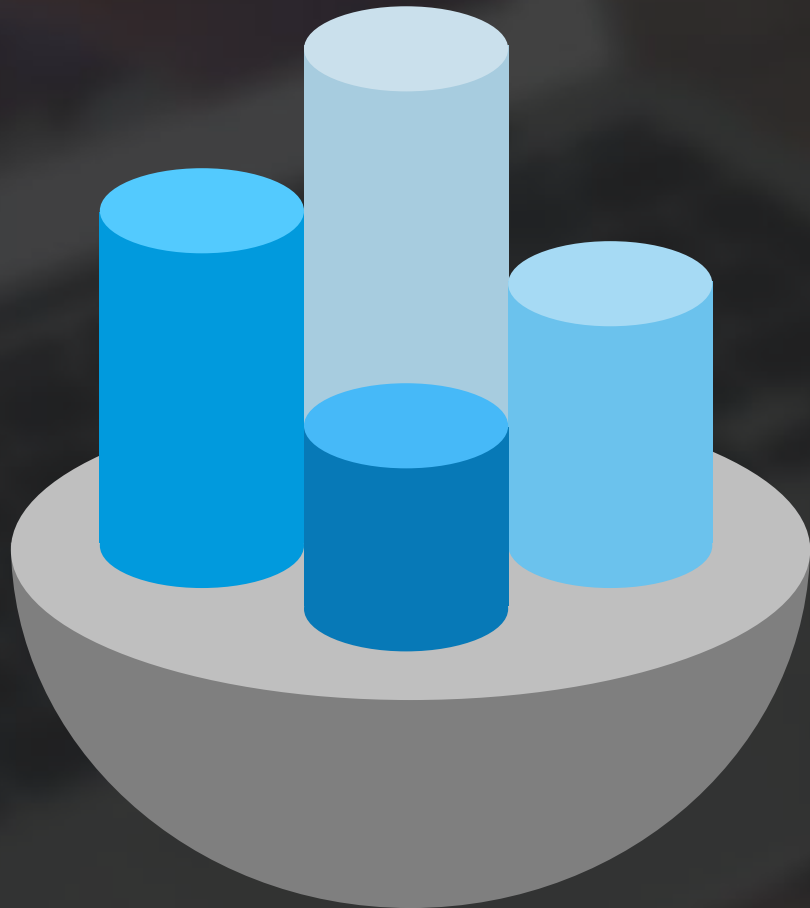
- Increment
- Product Backlog
- Sprint Backlog

Events

- Sprint
- Sprint Planning
- Daily Scrum
- Sprint Review
- Retrospective

[source: ADM]

Scrum Events



Inspection & Adaptation

Scrum prescribes four formal events for inspection and adaptation



Sprint Planning



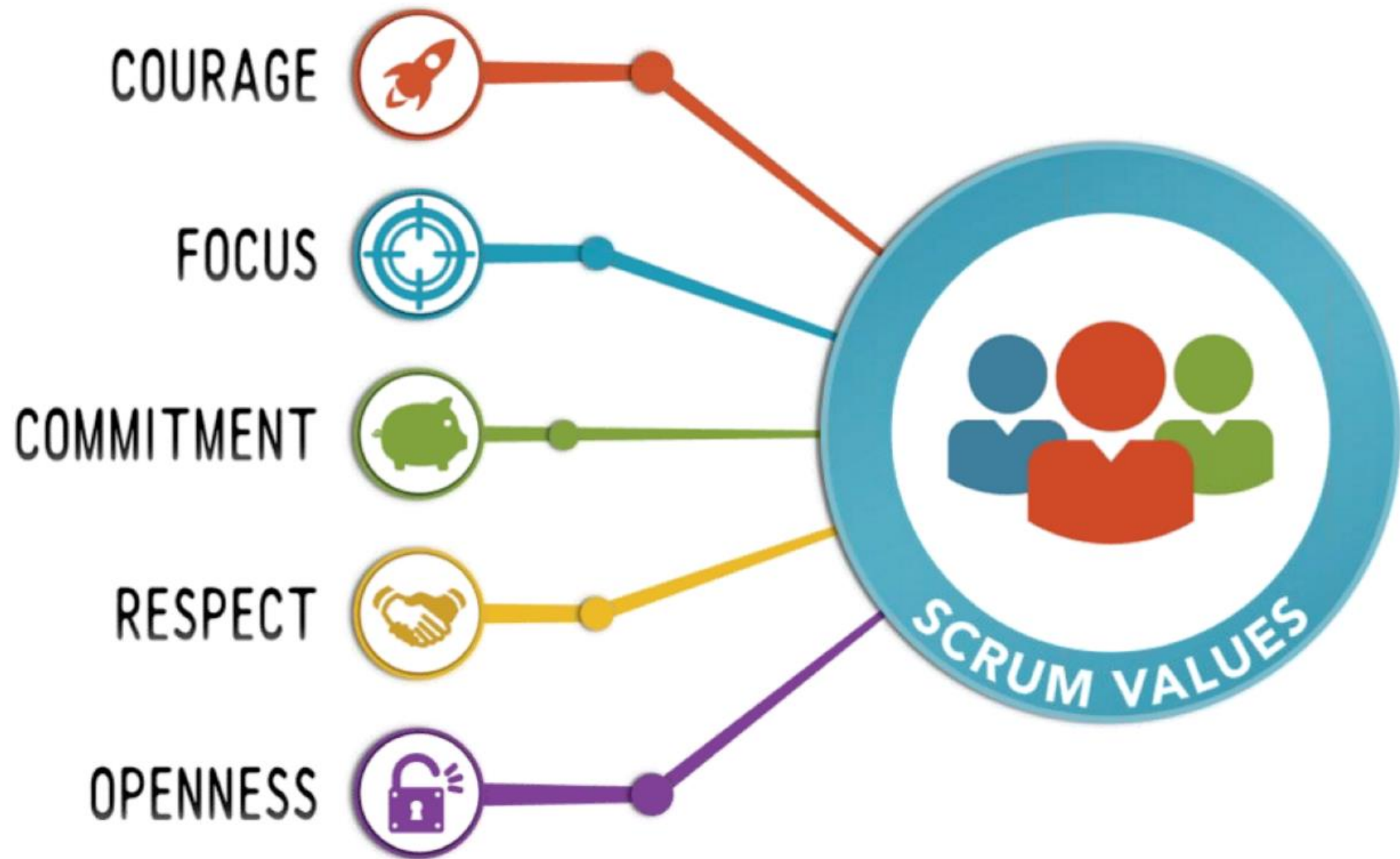
Daily Scrum



Sprint Review



Sprint Retrospective



Scrum Values

When the values of commitment, courage, focus, openness and respect are embodied and lived by the Scrum Team, the Scrum pillars of transparency, inspection, and adaptation come to life and build trust for everyone.

The Scrum Team members learn and explore those values as they work with the Scrum roles, events, and artifacts.



Scrum Values

Successful use of Scrum depends on people becoming more proficient in living these five values.



Scrum Value



The Scrum Team members have courage to do the right thing and work on tough problems.



Everyone focuses on the work of the Sprint and the goals of the Scrum Team.



The Scrum Team and its stakeholders agree to be open about all the work and the challenges with performing the work.



Scrum Team members respect each other to be capable, independent people.



People personally commit to achieving the goals of the Scrum Team.



OAK INTERLINK

Module 4

The Scrum Team

Understanding the Scrum Team



THREE MAIN SCRUM ROLES



The Scrum Team

The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master.

Scrum Teams are self-organizing and cross-functional.

Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team.



The Scrum Team

Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team.

The team model in Scrum is designed to optimize flexibility, creativity, and productivity.

The Scrum Team has proven itself to be increasingly effective for all the earlier stated uses, and any complex work.



The Scrum Team

Scrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback. Incremental deliveries of “Done” product ensure a potentially useful version of working product is always available.



The Product Owner

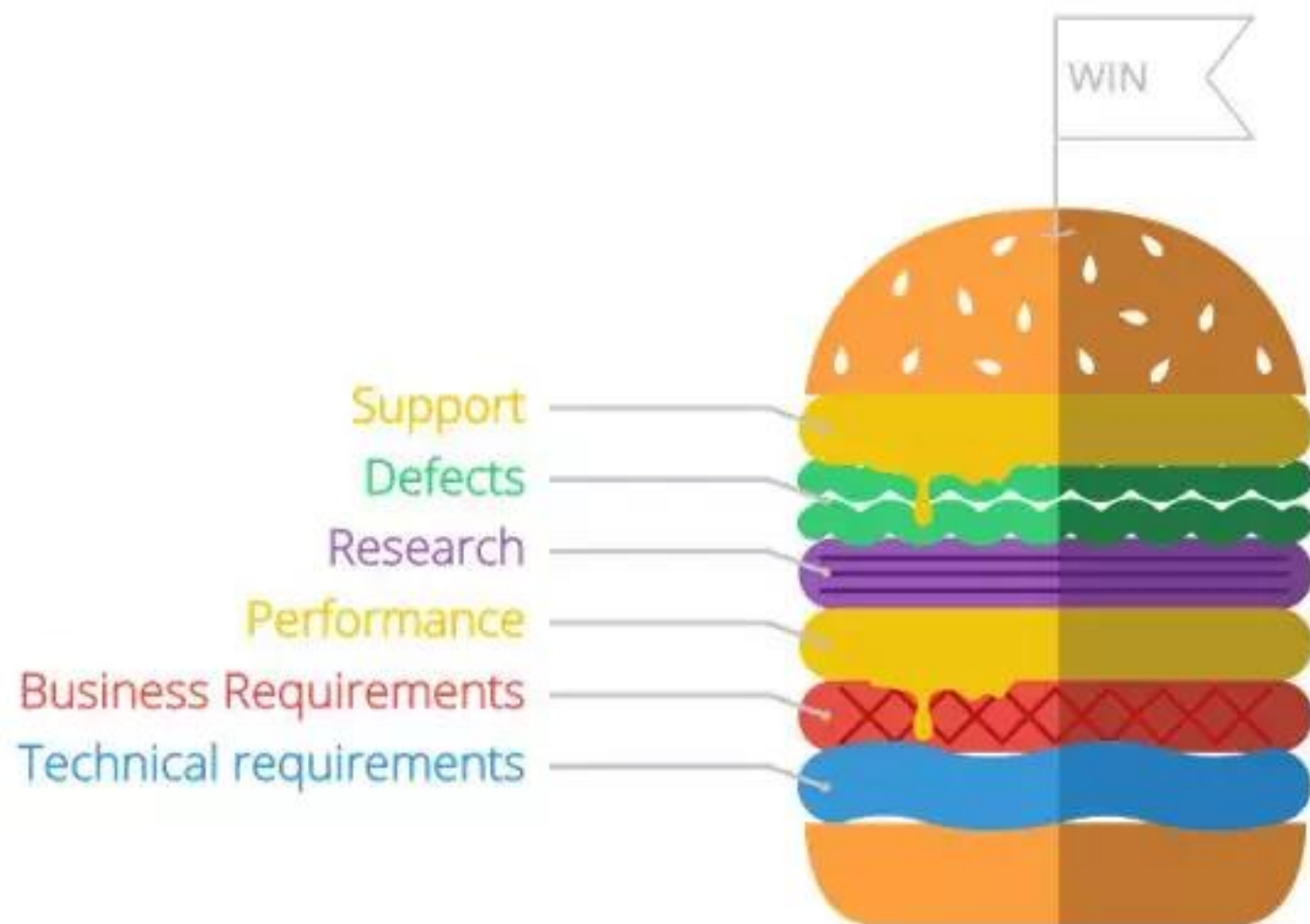


“ The Product Owner is responsible for maximizing the value of the product resulting from work of the Development Team.

How this is done may vary widely across organizations, Scrum Teams, and individuals. The Product Owner is the sole person responsible for managing the Product Backlog

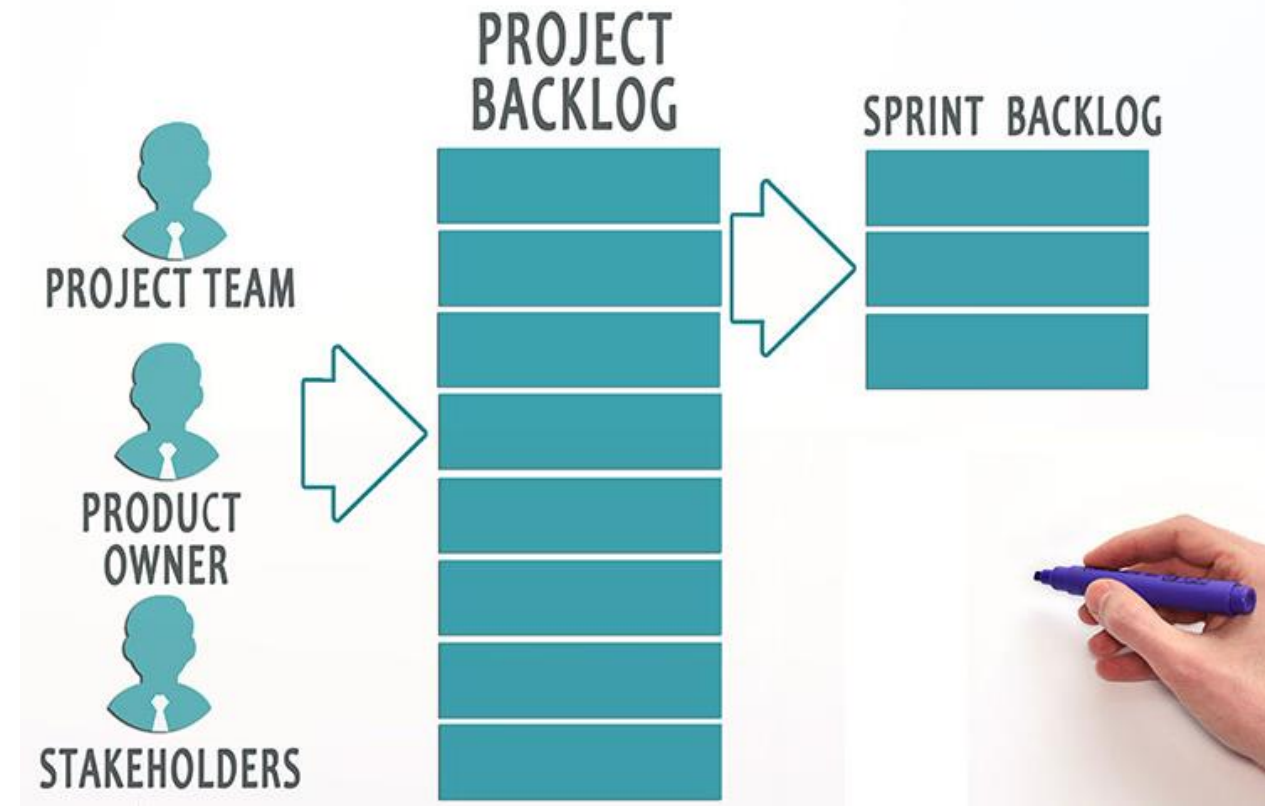
John Smith

Product Backlog Structure



Product Backlog Management Includes:

- 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; and,
- Ensuring the Development Team understands items in the Product Backlog to the level needed.



The Product Owner

The Product Owner may do the above work, or have the Development Team do it. However, the Product Owner remains accountable.

The Product Owner is one person, not a committee. The Product Owner may represent the desires of a committee in the Product Backlog, but those wanting to change a Product Backlog item's priority must address the Product Owner.



The Product Owner

For the Product Owner to succeed, the entire organization must respect his or her decisions.

The Product Owner's decisions are visible in the content and ordering of the Product Backlog. No one can force the Development Team to work from a different set of requirements.



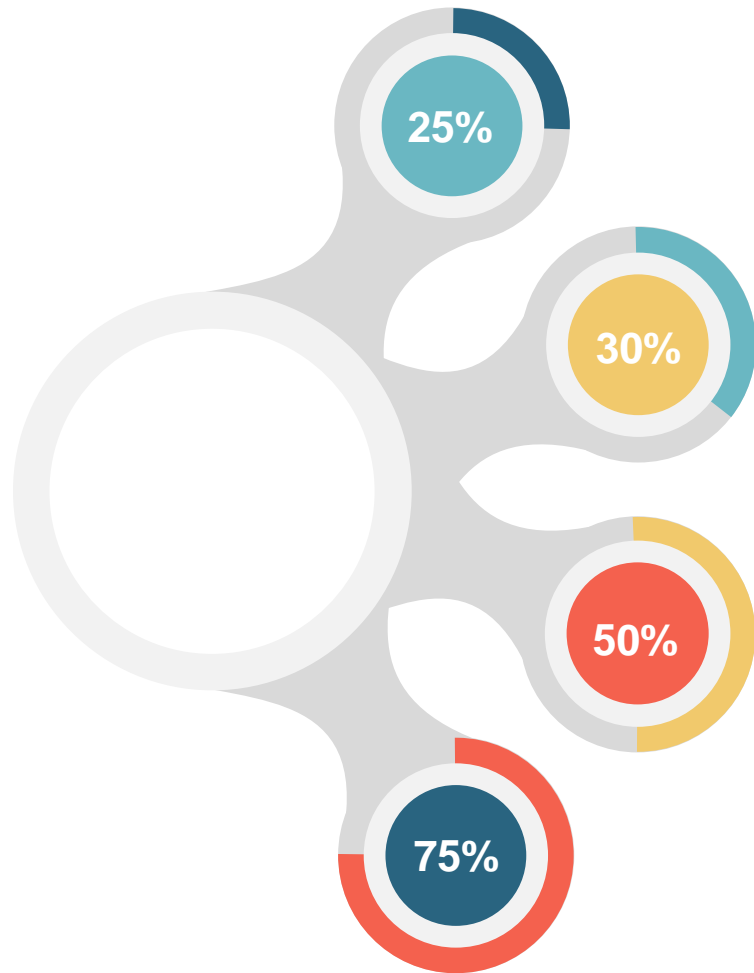


The Development Team

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.

A “Done” increment is required at the Sprint Review. Only members of the Development Team create the Increment. Development Teams are structured and empowered by the organization to organize and manage their own work. The resulting synergy optimizes the Development Team’s overall efficiency and effectiveness.

Development Teams Have The Following Characteristics



They are self-organizing. No one (not even the Scrum Master) tells the Development Team how to turn Product Backlog into Increments of potentially releasable functionality;



Development Teams are cross-functional, with all the skills as a team necessary to create a product Increment;



Scrum recognizes no titles for Development Team members, regardless of the work being performed by the person;



Scrum recognizes no sub-teams in the Development Team, regardless of domains that need to be addressed like testing, architecture, operations, or business analysis; and,
Individual Development Team members may have specialized skills and areas of focus, but accountability belongs to the Development Team as a whole.



Development Team Size: 9 -3

Optimal Development Team size is small enough to remain nimble and large enough to complete significant work within a Sprint.

Fewer than three Development Team members decrease interaction and results in smaller productivity gains. Smaller Development Teams may encounter skill constraints during the Sprint, causing the Development Team to be unable to deliver a potentially releasable Increment. Having more than nine members requires too much coordination. Large Development Teams generate too much complexity for an empirical process to be useful. The Product Owner and Scrum Master roles are not included in this count unless they are also executing the work of the Sprint Backlog.

The Scrum Master



“ The Scrum Master is responsible for promoting and supporting Scrum as defined in the Scrum Guide. Scrum Masters do this by helping everyone understand Scrum theory, practices, rules, and values. The Scrum Master is a servant-leader for the Scrum Team. The Scrum Master helps those outside the Scrum Team understand which of their interactions with the Scrum Team are helpful and which aren't. The Scrum Master helps everyone change these interactions to maximize the value created by the Scrum Team.

Jane Lucas

The Scrum Master Service To The Product Owner



- Scrum Master
Jane Lucas

The Scrum Master serves the Product Owner in several ways, including:

- Ensuring that goals, scope, and product domain are understood by everyone on the Scrum Team as well as possible;
- 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; and,
- Facilitating Scrum events as requested or needed.

The Scrum Master Service To The Development Team



- Scrum Master
Jane Lucas



The Scrum Master serves the Development Team in several ways, including:

- 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; and,
- Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood.

The Scrum Master Service To The Organization



- Scrum Master
Jane Lucas



The Scrum Master serves the organization in several ways, including:

- Leading and coaching the organization in its Scrum adoption;
- Planning Scrum implementations within the organization;
- Helping employees and stakeholders understand and enact Scrum and empirical product development;
- Causing change that increases the productivity of the Scrum Team; and,
- Working with other Scrum Masters to increase the effectiveness of the application of Scrum in the organization.



OAK INTERLINK

Module 5

The Scrum Events

Understanding the Scrum Events



Scrum Events

Prescribed events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum.

All events are time-boxed events, such that every event has a maximum duration.

Once a Sprint begins, its duration is fixed and cannot be shortened or lengthened. The remaining events may end whenever the purpose of the event is achieved, ensuring an appropriate amount of time is spent without allowing waste in the process



Scrum Events

Each event in Scrum is a formal opportunity to inspect and adapt something. These events are specifically designed to enable critical transparency and inspection. Failure to include any of these events results in reduced transparency and is a lost opportunity to inspect and adapt.



The Sprint

The heart of Scrum is a Sprint, a time-box of one month or less during which a “Done”, useable, and potentially releasable product Increment is created. Sprints have consistent durations throughout a development effort. A new Sprint starts immediately after the conclusion of the previous Sprint.



The Sprint

Sprints contain and consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.



The Sprint

Sprints contain and consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.



The Sprint

Sprints contain and consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.



The Sprint

Each Sprint may be considered a project with no more than a one-month horizon.

Like projects, Sprints are used to accomplish something. Each Sprint has a goal of what is to be built, a design and flexible plan that will guide building it, the work, and the resultant product increment.

Sprints are limited to one calendar month. When a Sprint's horizon is too long the definition of what is being built may change, complexity may rise, and risk may increase.

Sprints enable predictability by ensuring inspection and adaptation of progress toward a Sprint Goal at least every calendar month. Sprints also limit risk to one calendar month of cost.

Cancelling A Sprint





Cancelling A Sprint

A Sprint can be cancelled before the Sprint time-box is over. Only the Product Owner has the authority to cancel the Sprint, although he or she may do so under influence from the stakeholders, the Development Team, or the Scrum Master.

A Sprint would be cancelled if the Sprint Goal becomes obsolete. This might occur if the company changes direction or if market or technology conditions change. In general, a Sprint should be cancelled if it no longer makes sense given the circumstances.

But, due to the short duration of Sprints, cancellation rarely makes sense.



Cancelling A Sprint

When a Sprint is cancelled, any completed and “Done” Product Backlog items are reviewed. If part of the work is potentially releasable, the Product Owner typically accepts it.

All incomplete Product Backlog Items are re-estimated and put back on the Product Backlog. The work done on them depreciates quickly and must be frequently re-estimated.

Sprint cancellations consume resources, since everyone regroups in another Sprint Planning to start another Sprint. Sprint cancellations are often traumatic to the Scrum Team, and are very uncommon.

Sprint Planning





Sprint Planning

The work to be performed in the Sprint is planned at the Sprint Planning. This plan is created by the collaborative work of the entire Scrum Team.

Sprint Planning is time-boxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendants understand its purpose. The Scrum Master teaches the Scrum Team to keep it within the time-box.



Sprint Planning Answers The Following:

- What can be delivered in the Increment resulting from the upcoming Sprint?
- How will the work needed to deliver the Increment be achieved?

Sprint Goal





Sprint Goal:

The Sprint Goal is an objective set for the Sprint that can be met through the implementation of Product Backlog. It provides guidance to the Development Team on why it is building the Increment.

It is created during the Sprint Planning meeting. The Sprint Goal gives the Development Team some flexibility regarding the functionality implemented within the Sprint. The selected Product Backlog items deliver one coherent function, which can be the Sprint Goal.

The Sprint Goal can be any other coherence that causes the Development Team to work together rather than on separate initiatives.



Sprint Goal:

As the Development Team works, it keeps the Sprint Goal in mind. In order to satisfy the Sprint Goal, it implements functionality and technology.

If the work turns out to be different than the Development Team expected, they collaborate with the Product Owner to negotiate the scope of Sprint Backlog within the Sprint.

Daily Scrum





Daily Scrum:

The Daily Scrum is a **15-minute time-boxed** event for the Development Team. The Daily Scrum is held every day of the Sprint. At it, the Development Team plans work for the next 24 hours. This optimizes team collaboration and performance by inspecting the work since the last Daily Scrum and forecasting upcoming Sprint work. The Daily Scrum is held at the same time and place each day to reduce complexity



Daily Scrum:

The Development Team uses the Daily Scrum to inspect progress toward the Sprint Goal and to inspect how progress is trending toward completing the work in the Sprint Backlog. The Daily Scrum optimizes the probability that the Development Team will meet the Sprint Goal.

Every day, the Development Team should understand how it intends to work together as a self-organizing team to accomplish the Sprint Goal and create the anticipated Increment by the end of the Sprint.



Daily Scrum:

The structure of the meeting is set by the Development Team and can be conducted in different ways if it focuses on progress toward the Sprint Goal.

Some Development Teams will use questions, some will be more discussion based. Here is an example of what might be used:

- What did I do yesterday that helped the Development Team meet the Sprint Goal?
- What will I do today to help the Development Team meet the Sprint Goal?
- Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?



Daily Scrum:

The Development Team or team members often meet immediately after the Daily Scrum for detailed discussions, or to adapt, or re-plan, the rest of the Sprint's work. The Scrum Master ensures that the Development Team has the meeting, but the Development Team is responsible for conducting the Daily Scrum. The Scrum Master teaches the Development Team to keep the Daily Scrum within the 15-minute time-box.

The Daily Scrum is an internal meeting for the Development Team. If others are present, the Scrum Master ensures that they do not disrupt the meeting. Daily Scrums improve communications, eliminate other meetings, identify impediments to development for removal, highlight and promote quick decision-making, and improve the Development Team's level of knowledge. This is a key inspect and adapt meeting.

Sprint Review





A Sprint Review:

A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed. During the Sprint Review, the Scrum Team and stakeholders collaborate about what was done in the Sprint.

Based on that and any changes to the Product Backlog during the Sprint, attendees collaborate on the next things that could be done to optimize value.

This is an informal meeting, not a status meeting, and the presentation of the Increment is intended to elicit feedback and foster collaboration.



A Sprint Review:

This is at most a four-hour meeting for one-month Sprints. For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendees understand its purpose.

The Scrum Master teaches everyone involved to keep it within the time-box.



A Sprint Review:

The Sprint Review includes the following elements:

- Attendees include the Scrum Team and key stakeholders invited by the Product Owner;
- The Product Owner explains what Product Backlog items have been “Done” and what has not been “Done”;
- The Development Team discusses what went well during the Sprint, what problems it ran into, and how those problems were solved;
- The Development Team demonstrates the work that it has “Done” and answers questions about the Increment;



A Sprint Review:

The Sprint Review includes the following elements:

- The Product Owner discusses the Product Backlog as it stands. He or she projects likely target and delivery dates based on progress to date (if needed);
- The entire group collaborates on what to do next, so that the Sprint Review provides valuable input to subsequent Sprint Planning;
- Review of how the marketplace or potential use of the product might have changed what is the most valuable thing to do next; and,
- Review of the timeline, budget, potential capabilities, and marketplace for the next anticipated releases of functionality or capability of the product.



Sprint Review.

“ The result of the Sprint Review is a revised Product Backlog that defines the probable Product Backlog items for the next Sprint. The Product Backlog may also be adjusted overall to meet new opportunities.

Daily Scrum





A Sprint Retrospective:

Sprint Retrospective The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint. The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning.

This is at most a three-hour meeting for one-month Sprints. For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendants understand its purpose.



A Sprint Retrospective:

The Scrum Master ensures that the meeting is positive and productive. The Scrum Master teaches all to keep it within the time-box.

The Scrum Master participates as a peer team member in the meeting from the accountability over the Scrum process.



The Purpose of the Sprint

Retrospective:

- Inspect how the last Sprint went with regards to people, relationships, process, and tools;
- Identify and order the major items that went well and potential improvements; and,
- Create a plan for implementing improvements to the way the Scrum Team does its work.



Sprint Retrospective:

The Scrum Master encourages the Scrum Team to improve, within the Scrum process framework, its development process and practices to make it more effective and enjoyable for the next Sprint. During each Sprint Retrospective, the Scrum Team plans ways to increase product quality by improving work processes or adapting the definition of “Done”, if appropriate and not in conflict with product or organizational standards.

By the end of the Sprint Retrospective, the Scrum Team should have identified improvements that it will implement in the next Sprint. Implementing these improvements in the next Sprint is the adaptation to the inspection of the Scrum Team itself. Although improvements may be implemented at any time, the Sprint Retrospective provides a formal opportunity to focus on inspection and adaptation.



OAK INTERLINK

Module 6

The Scrum Artifacts

Understanding the Scrum Artifacts



Scrum Artifacts:

Scrum's artifacts represent work or value to provide transparency and opportunities for inspection and adaptation. Artifacts defined by Scrum are specifically designed to maximize transparency of key information so that everybody has the same understanding of the artifact.

Product Backlog



Scrum Artifact



Scrum Artifacts:

The Product Backlog is an ordered list of everything that is known to be needed in the product. It is the single source of requirements for any changes to be made to the product. The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering. A Product Backlog is never complete.

The earliest development of it lays out the initially known and best-understood requirements. The Product Backlog evolves as the product and the environment in which it will be used evolves. The Product Backlog is dynamic; it constantly changes to identify what the product needs to be appropriate, competitive, and useful. If a product exists, its Product Backlog also exists..



Product Backlog:

The Product Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases. Product Backlog items have the attributes of a description, order, estimate, and value. Product Backlog items often include test descriptions that will prove its completeness when “Done.”

As a product is used and gains value, and the marketplace provides feedback, the Product Backlog becomes a larger and more exhaustive list. Requirements never stop changing, so a Product Backlog is a living artifact. Changes in business requirements, market conditions, or technology may cause changes in the Product Backlog.



Product Backlog:

The Product Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases. Product Backlog items have the attributes of a description, order, estimate, and value. Product Backlog items often include test descriptions that will prove its completeness when “Done.”

As a product is used and gains value, and the marketplace provides feedback, the Product Backlog becomes a larger and more exhaustive list. Requirements never stop changing, so a Product Backlog is a living artifact. Changes in business requirements, market conditions, or technology may cause changes in the Product Backlog.

Sprint Backlog



Scrum Artifact



Sprint Backlog:

The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality into a “Done” Increment.

The Sprint Backlog makes visible all the work that the Development Team identifies as necessary to meet the Sprint Goal. To ensure continuous improvement, it includes at least one high priority process improvement identified in the previous Retrospective meeting.



Sprint Backlog:

The Sprint Backlog is a plan with enough detail that changes in progress can be understood in the Daily Scrum. The Development Team modifies the Sprint Backlog throughout the Sprint, and the Sprint Backlog emerges during the Sprint.

This emergence occurs as the Development Team works through the plan and learns more about the work needed to achieve the Sprint Goal.

Increment



Scrum Artifact

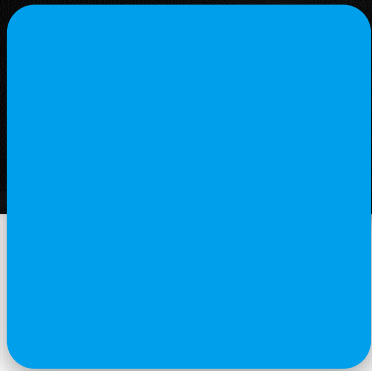


Increment:

The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints. At the end of a Sprint, the new Increment must be “Done,” which means it must be in useable condition and meet the Scrum Team’s definition of “Done.”

An increment is a body of inspectable, done work that supports empiricism at the end of the Sprint. The increment is a step toward a vision or goal. The increment must be in useable condition regardless of whether the Product Owner decides to release it.

Definition of Done





Definition of Done:

Definition of “Done” When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means. Although this may vary significantly per Scrum Team, members must have a shared understanding of what it means for work to be complete, to ensure transparency. This is the definition of “Done” for the Scrum Team and is used to assess when work is complete on the product Increment.

The same definition guides the Development Team in knowing how many Product Backlog items it can select during a Sprint Planning. The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the Scrum Team’s current definition of “Done.”

Summary

In summary we have covered.

- ✓ Details of THE AGILE MANIFESTOR.
- ✓ The use of SCRUM
- ✓ The Elements of the Scrum Frame Work.



Congratulations!

Congratulates, you have successful completed the course. You can now proceed to take the SCRUM MASTER OR PRODUCT OWNER certification course.



THE END



OAK INTERLINK

Presenter
**SHAKA OVIE
ERNEST**



Shaka Ovie Ernest, M.sc, B.sc., CSM

Shaka is the Head of Training at OAK Interlink Company whose primary focus is on human resource development, He has a special executive master's degree in Entrepreneurship from the Metropolitan school of business & management, United Kingdom and a first degree in Geography and regional planning from Ambrose Alli university, Ekpoma.

Shaka is a professionally certified scrum master, project manager professional, health and safety certified manager, lean six sigma green belt professional, warehouse and stores management professional, supply chain and logistics management professional, procurement and contract management professional, facility and asset management professional, document controller(ISO 9001/303010),ITIL service management professional and first aid treatment provider.

He is also a certified member of the national association of first aid treatment providers, international association of safety professionals, document management institute as well as the project management institute, USA. He has led several projects and has consulted for many organisations both in and outside the country.





OAK INTERLINK

Our Contact



+01-234-090-91020049, 090-91020048



info@oakinterlink.com



OAK Interlink company –
70b, Olorunlogbon street,
Anthony, Lagos.



www.oakinterlink.com



OAK INTERLINK

Thank You