Electronics and
Computer Science

University of Southampton

Introduction To Agile Project Management

COMP6204: Software Project Management and Secure Development

Dr A. Rezazadeh (Reza)
Email: ra3@ecs.soton.ac.uk or ar4k06@soton.ac.uk

November 24

University of Southampton

Overview

- What Is Agile?
- Why do we need Agile?
- Agile Characteristics
- Agile Myths
- Agile Manifesto, Values and Principles
- Agile Methodologies
- Scrum & Scrum Pillars
- Scrum Roles, Event & Artifacts
- Kanban

What Is Agile?

- There are several different definitions of agile.
  - The Agile Alliance (2021) defines **agile** as "the ability to create and respond to change."
  - Merriam-Webster dictionary (2021) defines agile as "marked by ready ability to move with quick easy grace," like an agile dancer.
  - PMI's *Agile Practice Guide* (2017) defines agile as "a term used to describe a mindset of values and principles as set forth in the Agile Manifesto."

What Is Agile?

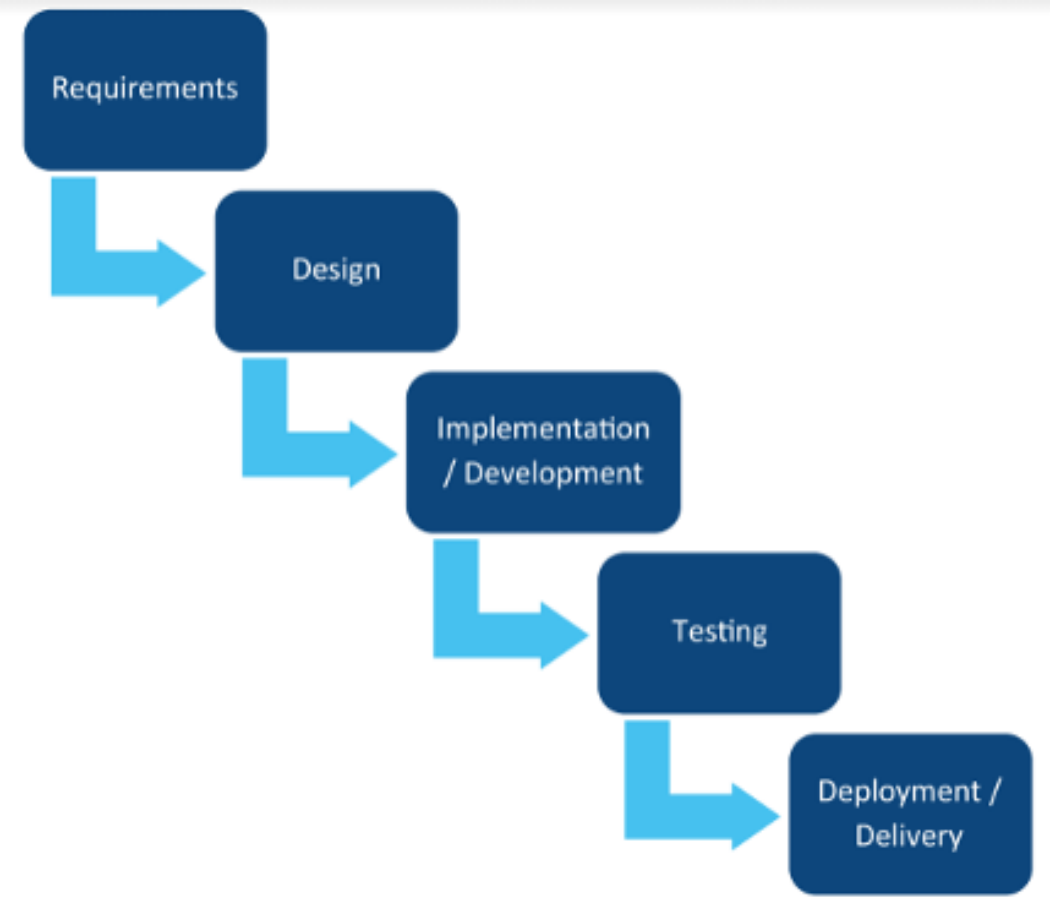
- Agile is simply a concept that states how work should be done.
- This work can be a part of a project or regular operations.
- The concept of Agile has several different implementations, which are called methodologies.
  - These methodologies can either be applied individually to work or together in combination.
  - Most common Agile implementations are a combination of several Agile methodologies.

When to use Agile

- Agile is not recommended for all kind of work.
- There are two pre-requisites that need to be satisfied to get the real benefits of Agile.
- These pre-requisites are:
  1. Work has high uncertainty/flexibility in scope
  2. Work involves "knowledge workers"
  - The term "*knowledge workers*" stands for highly skilled resources.
- One of the key characteristics of Agile is to give a lot of decision-making authority to the team. Hence, the team needs to have the required skills to take proper decisions.

Why do we need Agile?

- The traditional approach to perform work was to gather and document all the requirements in the beginning, plan all of them at once, and then begin execution.
- Customer gets to see the final deliverables only when almost all the stages are complected.
- This approach is sometimes even referred to as the "Big Bang Approach", as all the deliverables are shown to the customer together at a later part of the project.



Waterfall Model

Issues with the Traditional Approaches

- If at the final stage the customer does not like some of the deliverables and requests major changes, then one needs to go back to the previous stages of requirements, design, or development to incorporate them.
  - This could involve a lot of rework, which involves time, effort, and money. Hence, making changes using this approach was quite cumbersome.
- Customer needs to request for changes anywhere during the stages, for example, during development or testing, then once again that could involve a lot of rework by going back to the previous stages.

Issues with the Traditional Approaches

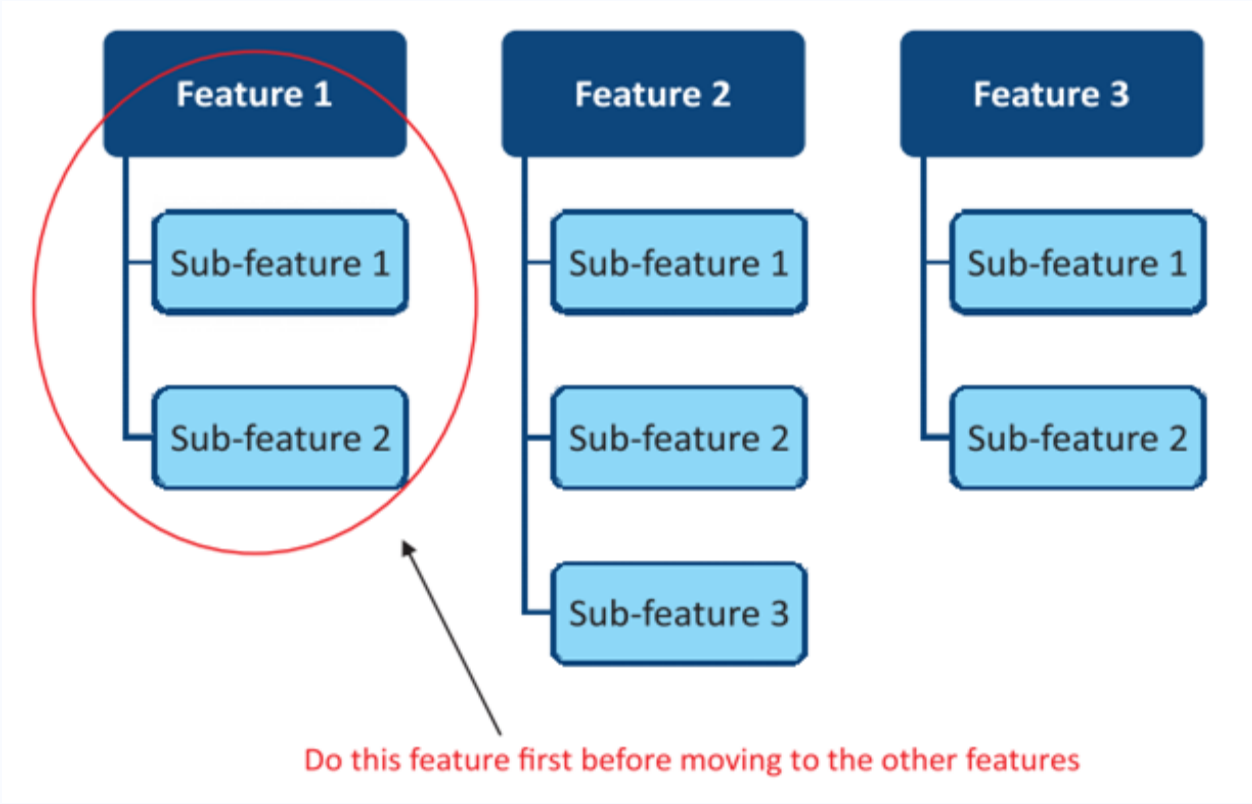
- Handling changes could be quite a challenge in the sequential model.
  - Due to this, the team is most often reluctant to incorporate changes in the work.
  - This means that the team is not customer centric (does not concentrate on the customer's benefit).

University of Southampton

Agile Characteristics – Incremental

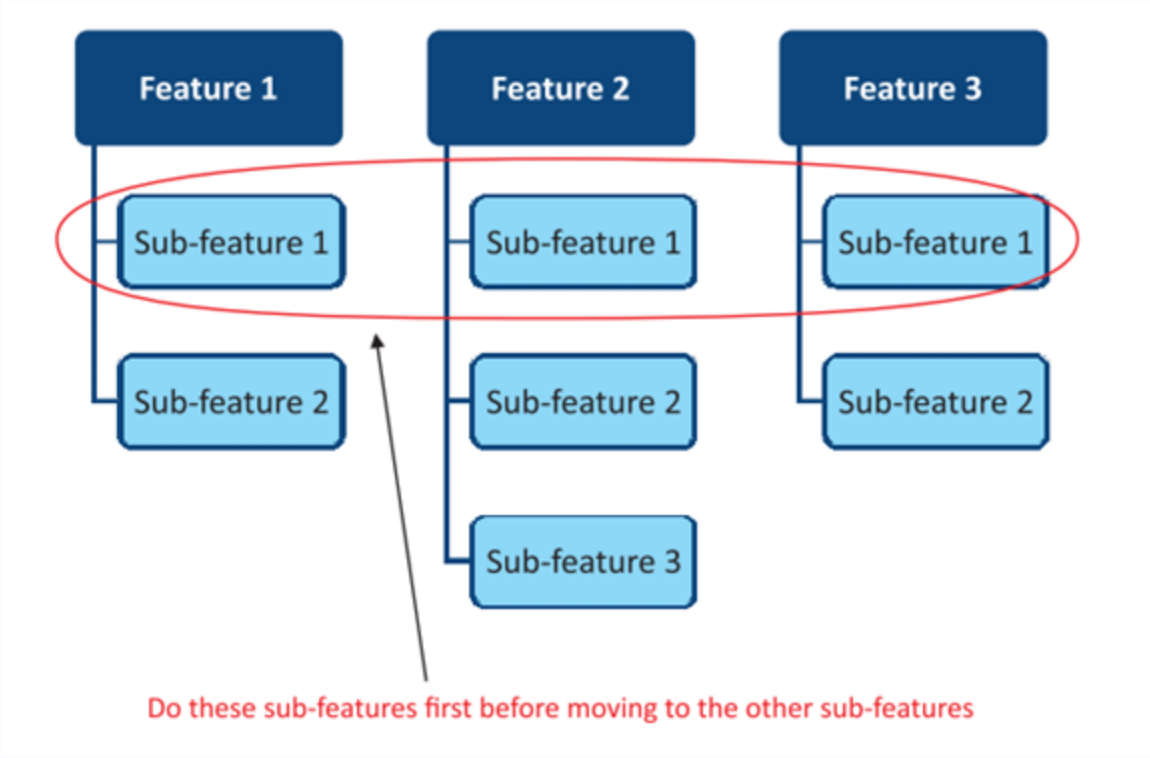
The entire system is divided into pieces and built piece by piece.

- For example, if we have requirements for building various features of a system, and each feature is divided into sub-features, we will build all sub-features of one feature before moving to the next feature.



Agile Characteristics – Iterative

- A few initial parts of the system are built and improved upon as we get customer feedback.
- The entire system is built little by little and continuously improved until it is acceptable.
- For example, if we have requirements for building various features of a system, and each feature is divided into sub-features, we will build a few sub-features of multiple features before moving to the rest of the sub- features.



University of Southampton

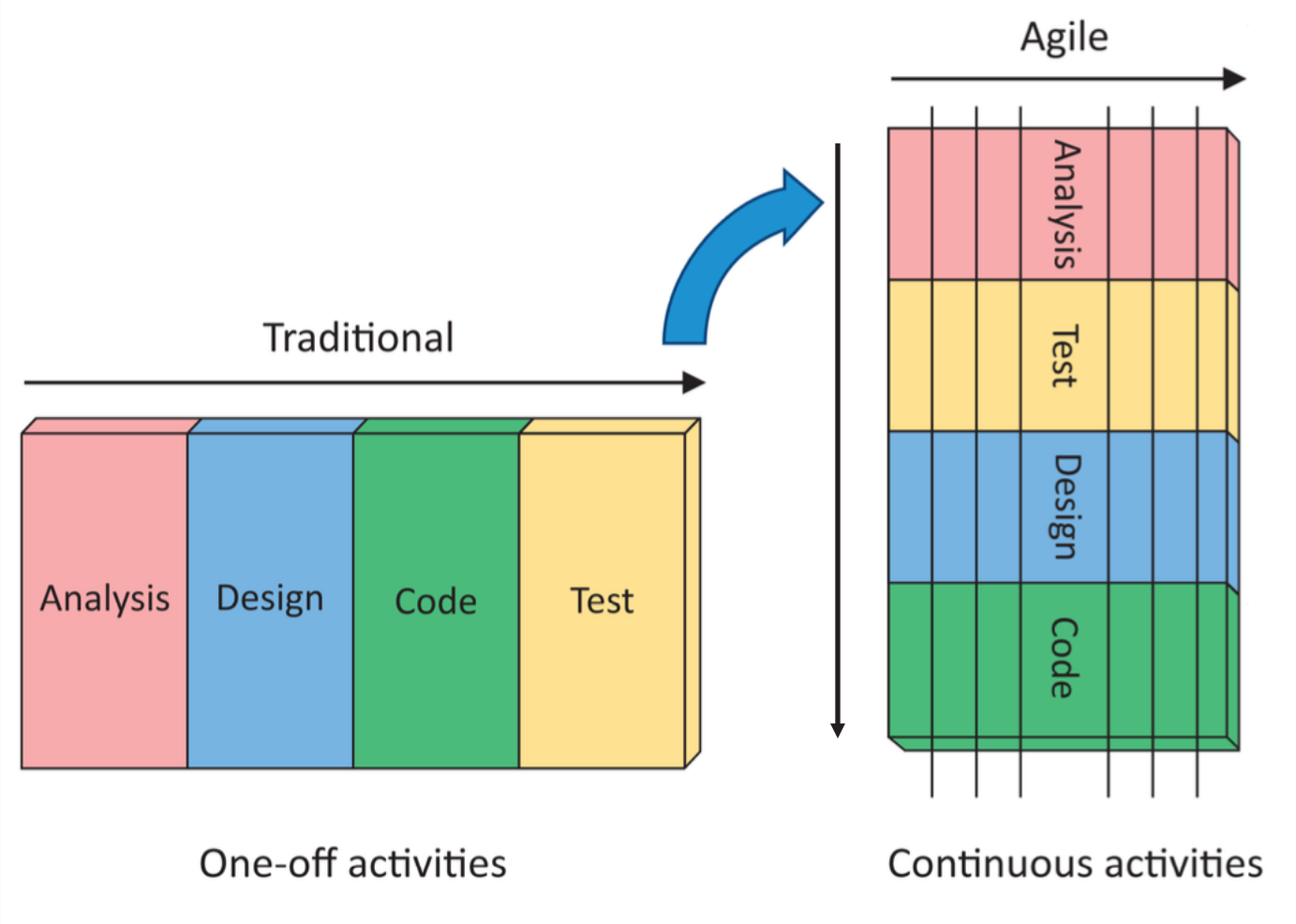
Agile Characteristics – Adaptive

- Being able to change course rapidly and easily is one of the key characteristics and the foundation of Agile.

University of Southampton

Traditional approaches Vs. Agile approaches

- You may have noticed that Test is coming in an earlier stage in Agile. This is not a mistake but by design.
- This is based on a concept called Test Driven Development, that is commonly practiced in Agile.



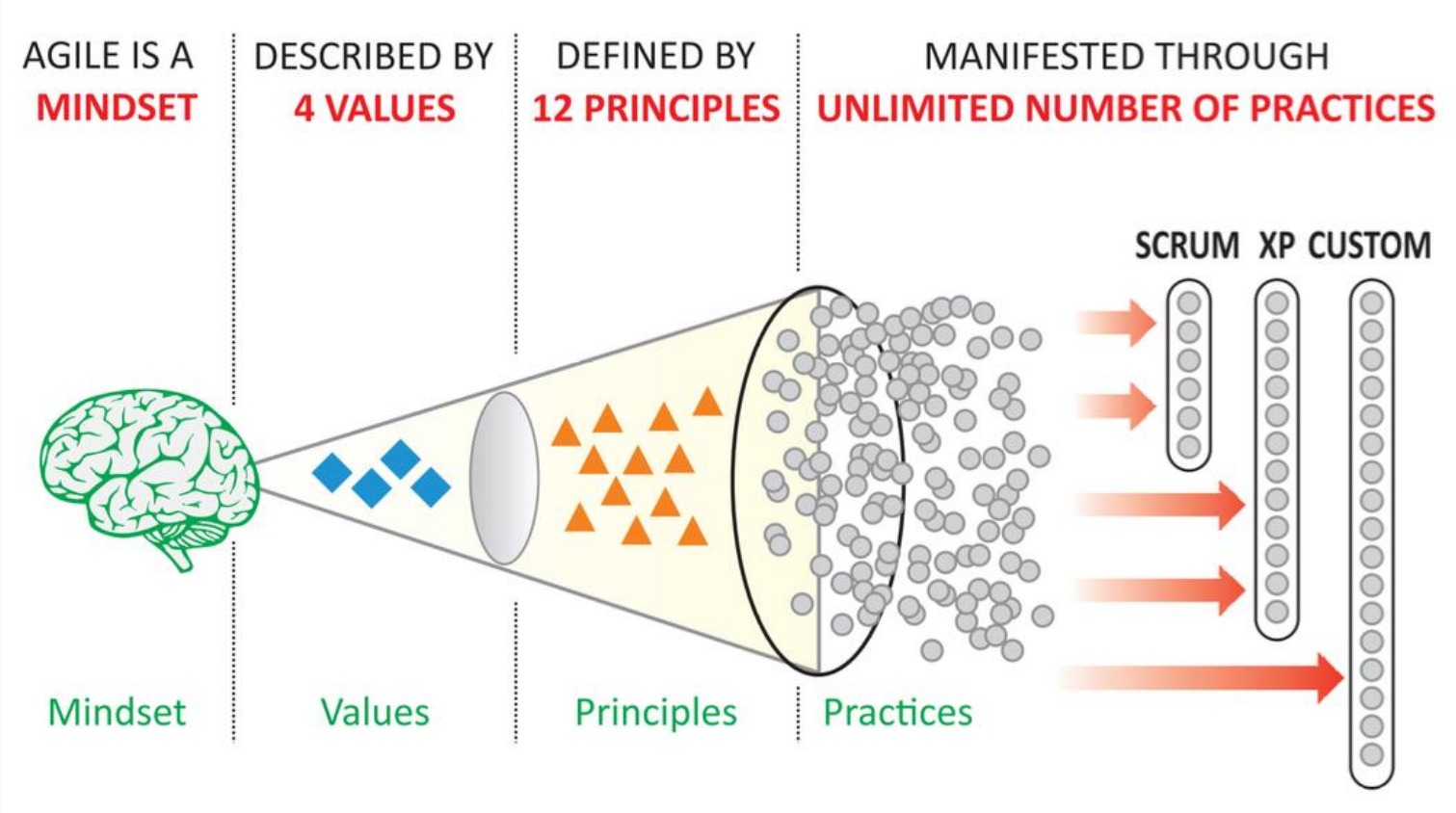
Agile Myths

- Over the years as Agile became popular, there have been several myths that have also crept in about it.

1. It is a silver bullet that ensures success of a project
2. It does not require documentation.
3. There is no need to plan
4. It does not need a disciplined approach
5. It involves a lot of rework

Agile Manifesto

- The work we do in Agile must adhere to the Agile Manifesto



University of Southampton

Agile Manifesto Values

The Agile Manifesto

Individuals And Interactions	over	Processes and Tools
Working Product	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.

www.agilemanifesto.org

University of Southampton

Agile Manifesto Principles

1	Our highest priority is to satisfy the customer through the early and continuous delivery of valuable software.	7	Working software is the primary measure of progress.
2	Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.	8	Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
3	Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.	9	Continuous attention to technical excellence and good design enhances agility.
4	Business people and developers must work together daily throughout the project.	10	Simplicity–the art of maximizing the amount of work not done–is essential.
5	Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.	11	The best architectures, requirements, and designs emerge from self-organizing teams.
6	The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	12	At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/

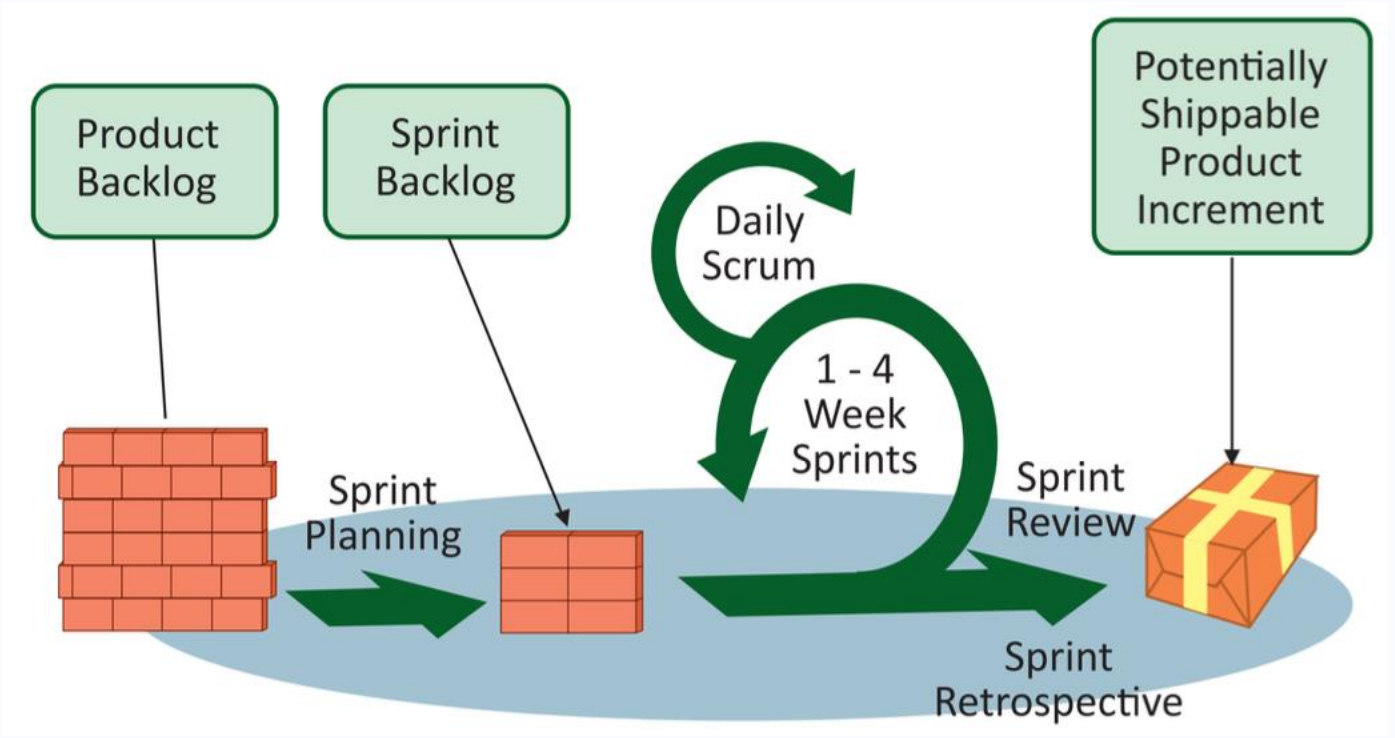
Agile Methodologies

- Values and Principles of Agile are implemented through several Practices. These practices are defined as part of several Agile Methodologies.
  - Scrum
  - XP (eXtreme Programming)
  - Lean (aligned closely to Agile, but not really an Agile methodology)
  - Kanban (implementation of Lean)
  - FDD (Feature Driven Development)
  - Crystal
  - DSDM (Dynamic Systems Development Method)

Scrum

- Scrum is based on short iterations called Sprints that generally go for 2 weeks.
- In Scrum, daily collaboration between all stakeholders happens through Scrum meetings or Scrum calls.
- Retrospective meetings are held at the end of each iteration to incorporate continuous learning.

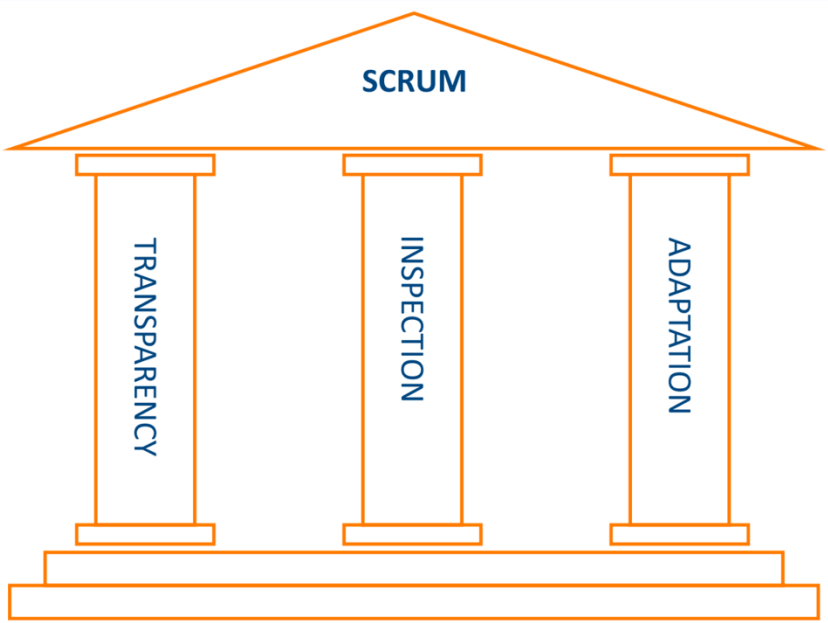
planning stage



University of Southampton

Scrum Pillars

- Transparency - Transparency means that all stakeholders have a visibility to what is being done and where we stand.
- It also encompasses the fact that all parties are involved in deciding what will be the acceptance criteria for each deliverable. This is referred to as the definition of "Done".
- Inspection - Inspection is done in Agile on a continuous basis to check the work progress towards achieving its goals and correcting any deviations early on. This is done daily using the Scrum meetings.
- Adaptation - Adaptation is somewhat related to inspection to ensure that any deviations are taken care of to avoid failure.



Scrum Roles – Product Owner

- Product Owner represents the customer and is responsible and accountable for:
  - The requirements,
  - Managing the product backlog,
  - Prioritising items in the backlog,
  - Accepting deliverables,
  - Providing support required by the development team from the business front.
- Product Owner is solely responsible to maximise value of the product being delivered. He also creates a product vision and shares it with all other members of the scrum team.

Scrum Roles – ScrumMaster

- ScrumMaster is a servant leader who is responsible for removing impediments to progress and facilitating various scrum events.
  - The term "servant leader" means that the ScrumMaster does not manage the team.
  - He rather performs administrative tasks for them to ensure that they concentrate on delivering items from the backlog
  - He also acts as a bridge between product owner and development team for communicating vision, goals, and backlog items.
- ScrumMaster should not be mapped to a traditional Project Manager role as the latter manages the project whereas the prior facilitates things on the project for the self-organising development team.

Development Team

- Development Team consists of resources with skills that are adequate to deliver all the work. It is a self-organising cross-functional team.
  - It is recommended to keep this team small, ranging from 5 to 9 people.
  - It is also recommended that the development team should not change during the Sprint but may change across Sprints.
- All the three roles - Product Owner, ScrumMaster, and Development Team, put together is referred to as the Scrum Team.
- Apart from the Scrum Team, there could be other stakeholders, like Project Manager, Organisation, Users etc.
  - These roles are not specifically defined in Scrum, but they may exist.

Scrum Events

- Several **events** are used in Scrum, however not all of them are defined as part of the Scrum methodology. They are described below.

1. **Strategy Meeting** (often used but not defined in Scrum)
2. **Release Planning Meeting** (often used but not defined in Scrum)
3. **Sprints** (defined in Scrum)
4. **Sprint Planning Meeting** (defined in Scrum)
5. **Daily Scrum Meeting** (defined in Scrum)
6. **Sprint Review Meeting** (defined in Scrum)
7. **Sprint Retrospective** Meeting (defined in Scrum)

University of Southampton

Strategy Meeting (often used but not defined in Scrum)

- Strategy Meeting is a kick-off meeting held:
  - to discuss product strategy,
  - to come up with a product vision, define epic user stories,
  - to define detailed user stories,
  - to make product backlog,
  - and to create product roadmap.

Strategy Meeting in Agile – Notes on Outputs

- Epic user stories are requirements at a high-level.
- Requirements on Agile are generally in the form of user stories.
- Product backlog contains all items that need to be delivered for the product.
- Product roadmap is a pictorial view of the requirements included in each release.
- Work is broken down into releases which are further broken down into iterations.
- A Strategy Meeting should happen at least once in 12 months and the duration is generally 4-16 hours.
- All the stakeholders are invited to this meeting.

Release Planning Meeting (often used but not defined in Scrum)

- Release Planning Meeting is held for each release to decide:
  - Prioritisation of requirements in the product backlog,
  - Decide what should be included in the release,
  - And put high-level estimates against each item in the product backlog.
- Product Owner is in- charge of this meeting and all stakeholders are invited to attend it.
- It generally runs for 4-8 hours.

Sprints (defined in Scrum)

- Sprints are time boxed iterations of 1 to 4 weeks, most often done for 2 weeks.
- They build a potentially releasable product increment.
- Several Sprints make up a Release and several Releases make up a Project.

Sprint Planning Meeting (defined in Scrum)

- Sprint Planning Meetings are held by the Product Owner to do detailed estimation of stories in the product backlog and agree on which ones can be taken up in the sprint and put in the sprint backlog.
  - The decision on which stories to include depends upon the priority set by the Product Owner and the risks perceived by the Development team.
  - We try to deliver those stories that provide the most business value first.
- Sprint Planning meeting generally includes only the Scrum Team consisting of Product Owner, ScrumMaster, and Development Team.
  - Its duration is about less than 2 hours per week of length of the Sprint.
  - For example, if the Sprint is for 2 weeks, then this meeting goes for about 4 hours.

Daily Scrum Meeting (defined in Scrum)

- Daily Scrum is a meeting that is organized on a daily basis and is time boxed to a maximum of 15minutes.
  - Irrespective of anything, the duration should never cross 15 minutes.
  - Invitees generally include the Development Team, Product Owner, and ScrumMaster.
- Only the Development Team is allowed to speak.
- They only provide an answer to three questions:

1. What has been achieved yesterday
2. What will be achieved today
3. What are the obstacles in the work

No other points are allowed to be discussed. This is ensured by the ScrumMaster. Any other issues or risks are put in a “Parking lot” to be taken up later offline with only the required stakeholders.

Sprint Review Meeting (defined in Scrum)

- Sprint Review Meeting is held at the end of the Sprint to present what work has been completed.
- Development Team shows the "Done" items to the Product Owner and other stakeholders.
- This is about getting acceptance of work done, also referred to as Scope Validation in traditional project management.
- Future plans for next Sprint may also be discussed in this meeting that runs from 1-2 hours.

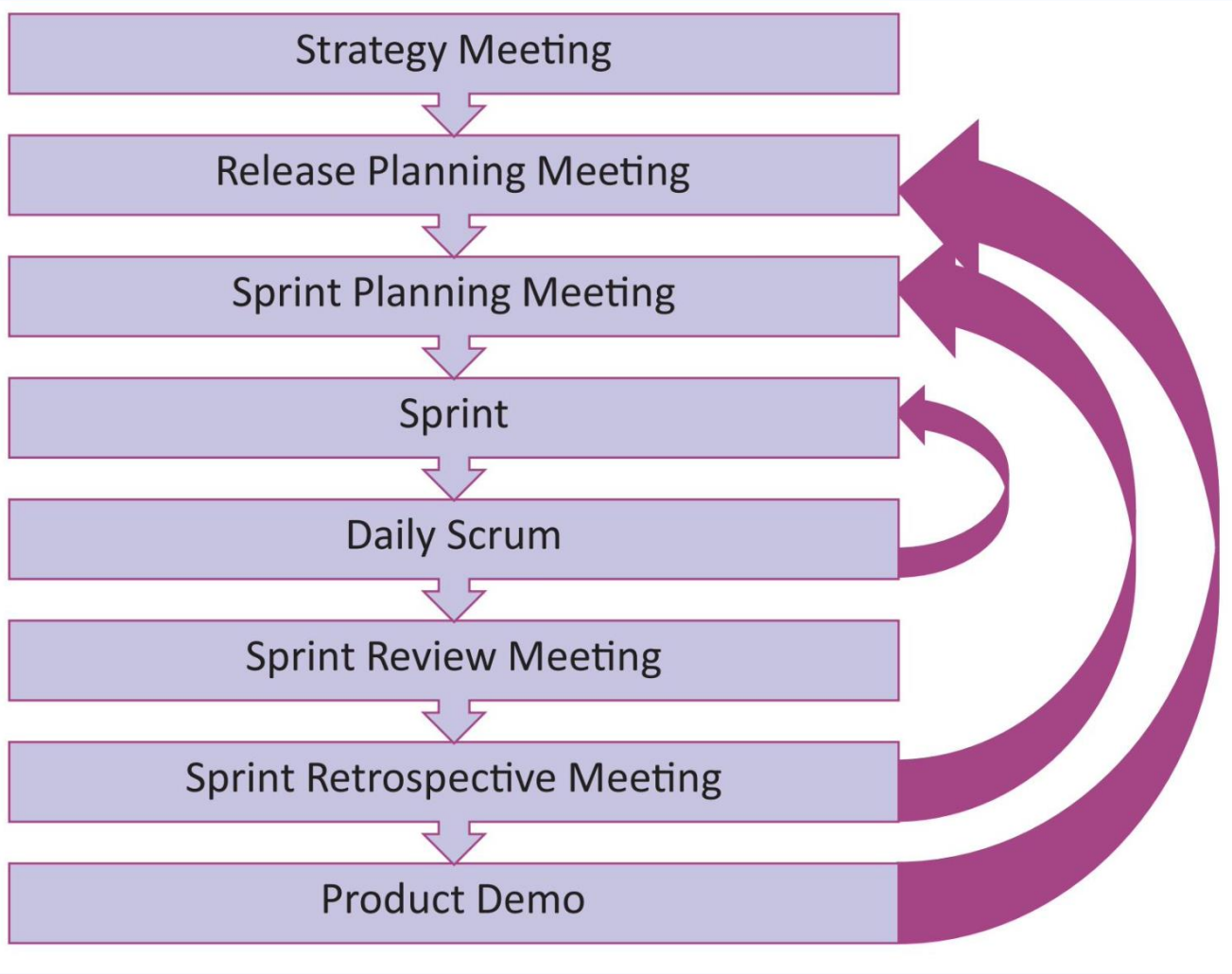
Sprint Retrospective Meeting (defined in Scrum)

- Sprint Retrospective Meeting is the last event to be held after Sprint Review.
- In that meeting we invite only the Scrum Team to discuss what went right and what went wrong during the just completed Sprint.
- This learning is documented so we can improve our processes or the way of working in later Sprints. It goes for 1-3 hours.
- As a rule of thumb, it goes for 45 minutes per week of length of Sprint.
  - For example, if the Sprint was for 2 weeks, then this meeting goes for about 1.5 hours.

University of Southampton

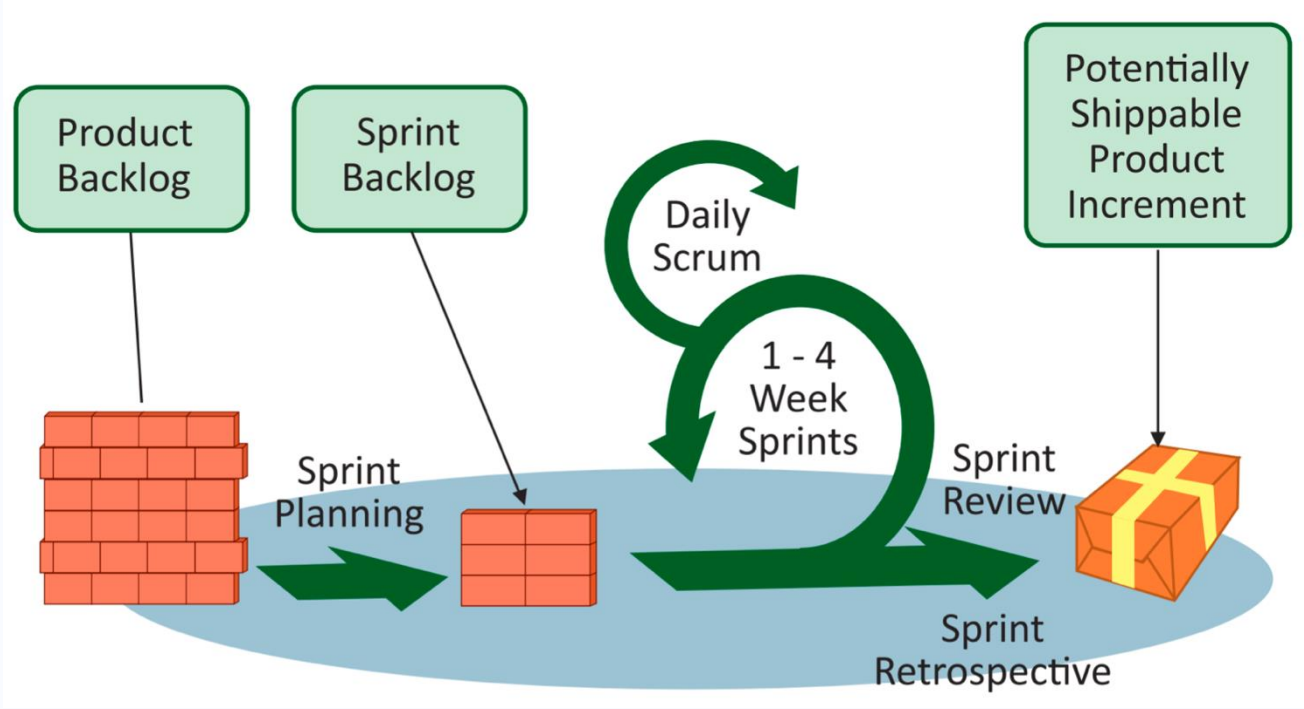
Scrum Chronology of Events

- First the Strategy Meeting is held to make a Product Backlog.
- The work is then broken down into Releases, starting with the Release Planning Meeting and ending with the Product Demo.
- Within a Release there are several Sprints that start with the Sprint Planning Meeting and end with the Sprint Review and Sprint Retrospective Meeting.
- Within a Sprint, we have Daily Scrum Meeting or Call.



Scrum Chronology of Events

- Releases are not specifically defined in Scrum. Only Sprints are defined.
- However, it is a common Agile practice that is used on Scrum to break the work into Release and then into Sprints.



Scrum Artifacts

- Various artifacts are created in Scrum.
- All of these are seen as adding value to the customer; hence they are all needed.
- Apart from these, the team may decide if any other artifacts are needed, and if so, make them.
- However, we should always uphold the thinking of only making those artifacts that are deemed necessary to add value to the customer.

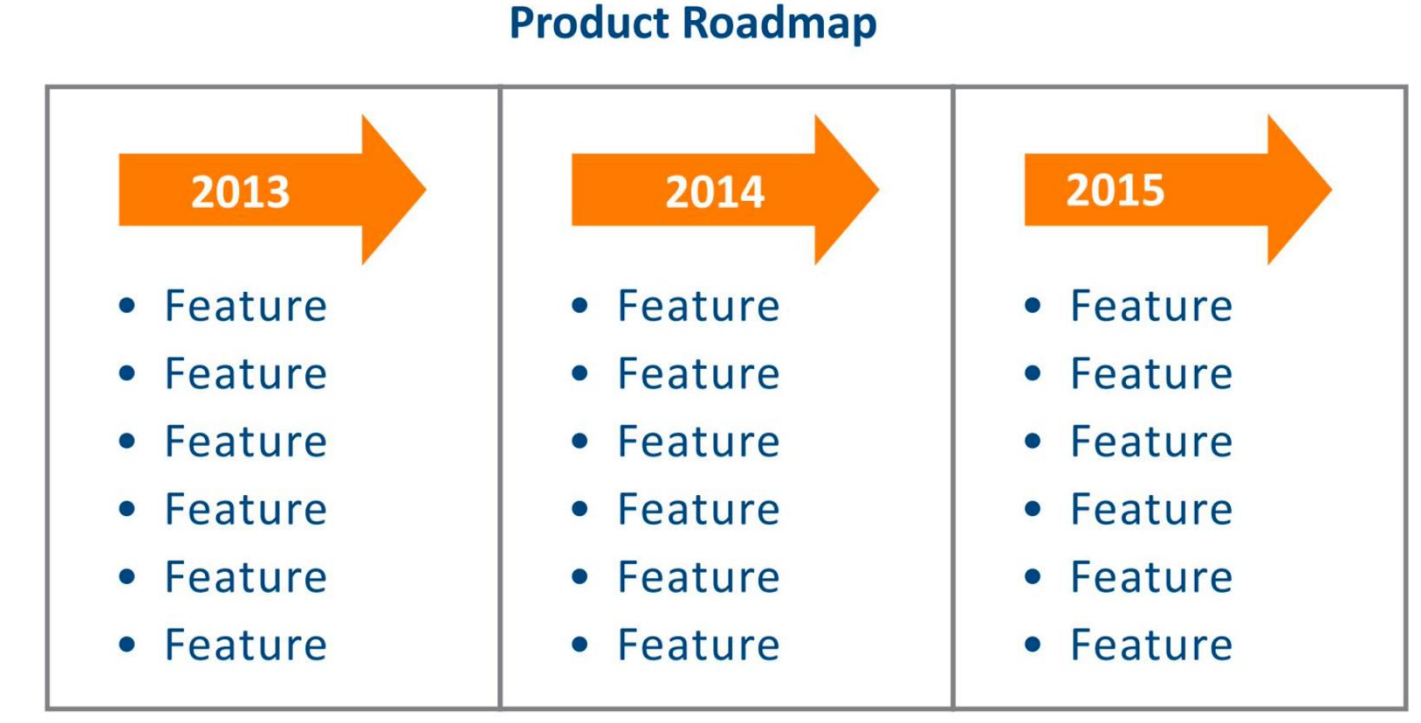
Product Vision (often used but not defined in Scrum)

- A Product Vision Statement is a statement that is created during the Strategy Meeting that describes the business vision.
- It gives a view to all involved about why the work is being undertaken and what to expect from it.

University of Southampton

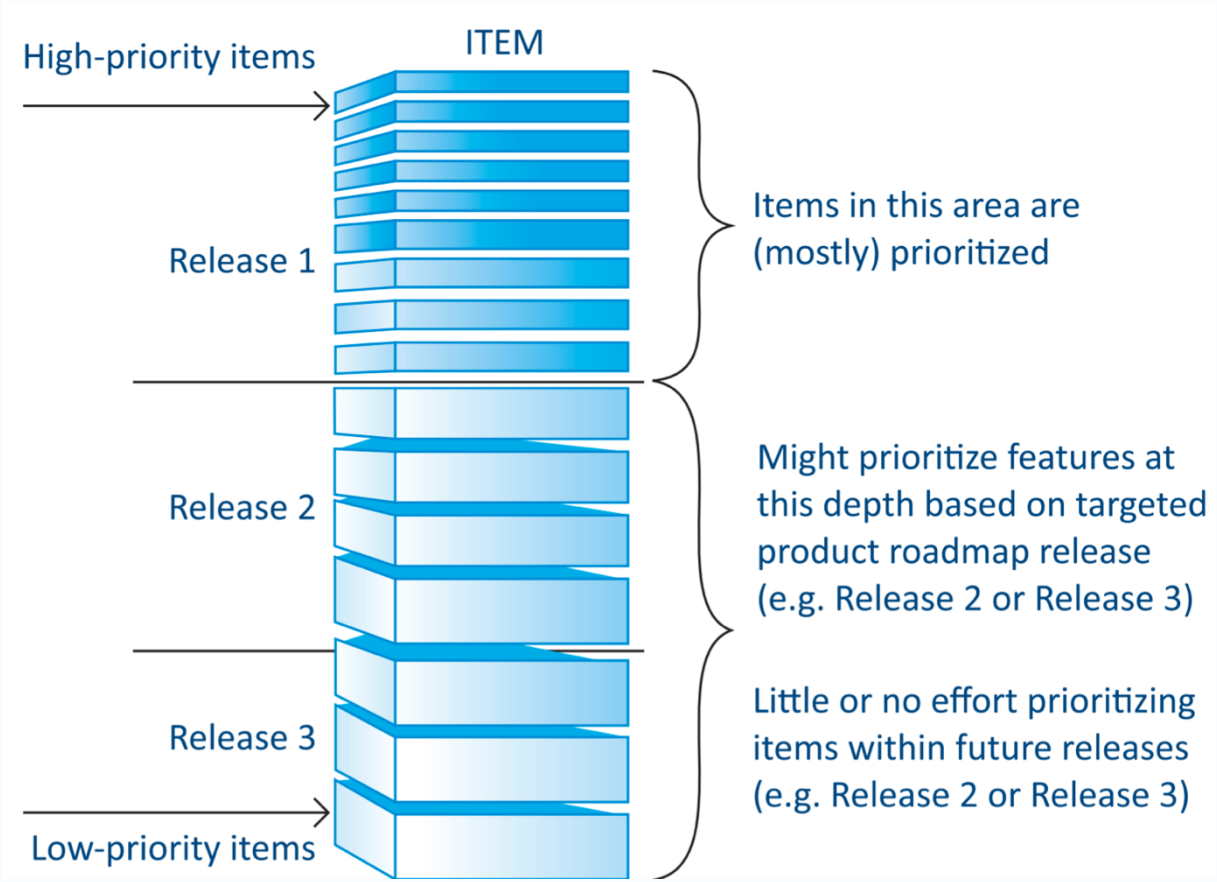
Product Roadmap (often used but not defined in Scrum)

- Product Roadmap is a pictorial representation of which features would be delivered in which release.



Product Backlog (defined in Scrum)

- Product Backlog contains a list of requirements generally in the form of user stories.
- It is also created during the Strategy Meeting. It is arranged using priority from **Product Owner** based on which ones add how much business value.
- The prioritisation process may also involve inputs on risks involved with the user stories.
- A Product Backlog may be changed to add more details, or to refine estimates. "Grooming" or "Refinement"



Release Backlog (often used but not defined in Scrum)

- Release Backlog is created during the Release Planning Meeting and includes only those user stories that are to be delivered in that release.
- It is a subset of the Product Backlog.

Sprint Backlog (defined in Scrum)

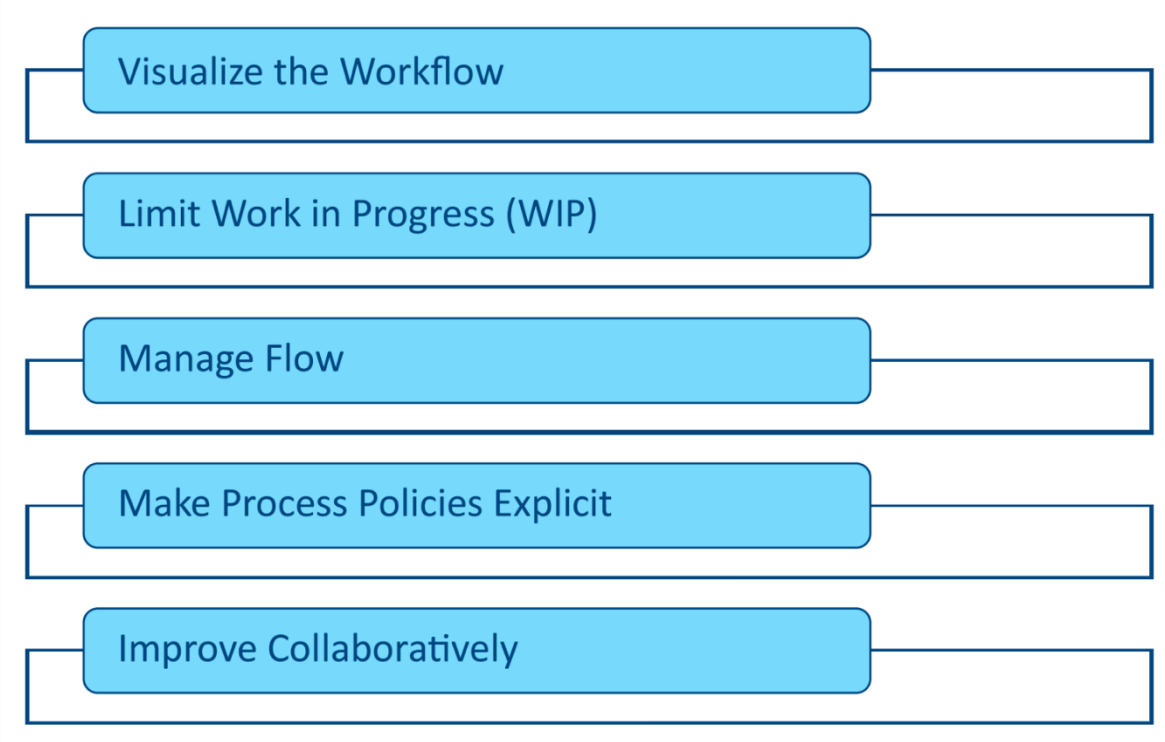
- Sprint Backlog is a subset of Release Backlog and contains only those user stories that are targeted to be delivered in the particular Sprint.
- It is also accompanied by a plan for the Sprint. It is created during Sprint Planning Meeting.

Definition of “Done” — DoD (defined in Scrum)

- Definition of “Done” consists of completion actions required for deliverables in a sprint.
- For example, peer review, unit testing, and meeting acceptance criteria can be a part of this.
- Deliverables cannot be considered completed until these actions have been successfully completed.

Kanban

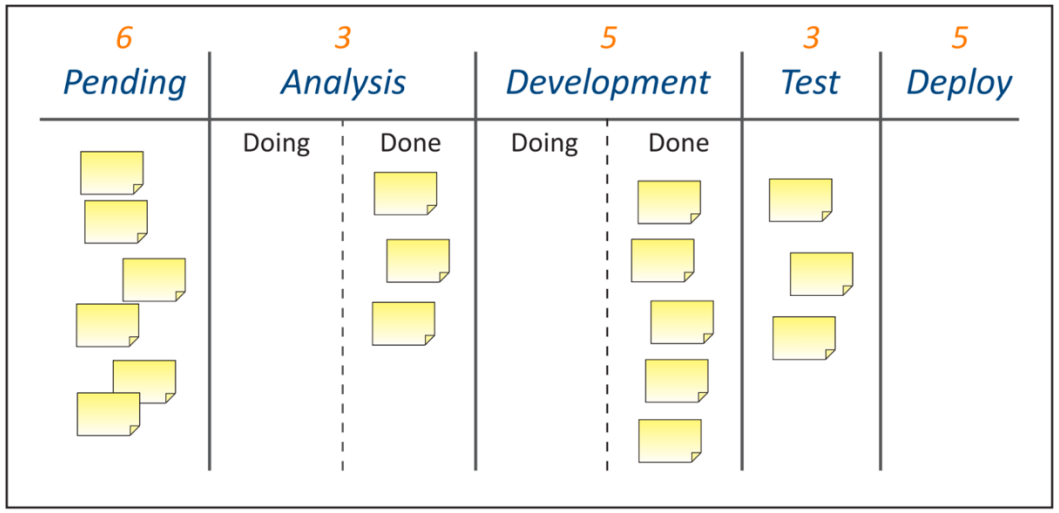
- Kanban is based on "lean" production system used at Toyota.
- Kanban in Japanese means "billboard" or "signboard".
- A Kanban system concentrates on the process's value stream.
- It is an implementation of a lean system.
- There are 5 principles that govern a Kanban system.



University of Southampton

Kanban Board

- The key tool provided by Kanban is a *Kanban Board /Task Board*, which is almost universally used in all Agile work.



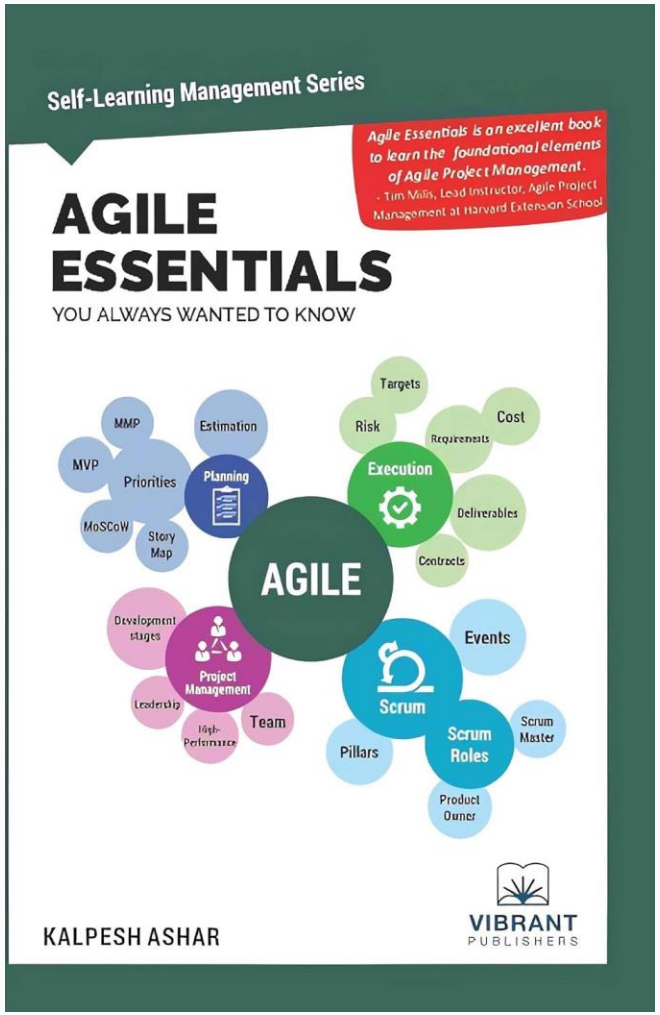
- Each *sticky note consists* of a requirement, generally in the form of a user story.
- These notes are placed in the various columns to show where each of the work items are in the entire process.
- WIP (Work in Progress) limits are put at each stage, written on the top of the stage. For example, we cannot have more than 5 requirements in Development stage at any time and no more than 3 can be in Analysis.

University of Southampton

References

- Chapter 1 & 2 of

  Agile Essentials You Always Wanted To Know



University of Southampton

YOUR QUESTIONS