ULASSA
Agile Development Tools

Scrum is a lightweight framework, designed to help you overcome complex adaptive tasks – from the idea to the deliverable product.

Scrum encourages communication, enables rapid feedback, boosts creativity and productivity and simplifies priorities so that decisions can be taken quickly.

This produces an organic growth process, resulting in the continuous improvement and maturity of the developed product – and greatly increases the satisfaction and enjoyment of everyone involved.

The Ulassa knowledge cards provide an overview of the most important elements in Scrum and help you to use Scrum in your daily work. We do not claim that the principles explained here are complete or the only ones valid; we are simply passing on information that our customers have found to be pragmatic and useful. We are always happy to receive feedback: **feedback@ulassa.com**

What is Scrum?

ULASSA
Agile Development Tools

- "Scrum" is an English word used in rugby to describe players packing closely together.
- Scrum was first mentioned in 1986 in the article "The New New Product Development Game" by Ikujirō Nonaka and H. Takeuchi.
- During the 1990s, Ken Schwaber, Jeff Sutherland, Mike Cohn and others developed a method on this basis, which is used in software development.
- This method focuses on providing functional software in short iterations so that feedback can be obtained and responses to change undertaken as quickly as possible.

- With a few simple rules and roles, Scrum is a quick starting point for using the method.
- 2001 saw the publication of the "Agile Manifesto", which summarises the most important agile principles and basic values.

A detailed introduction is provided in the Scrum Guide by Jeff Sutherland and Ken Schwaber in the Ulassa Scrum Starter Box. This is also available in more than 30 languages at:

www.scrumguides.org/download.html

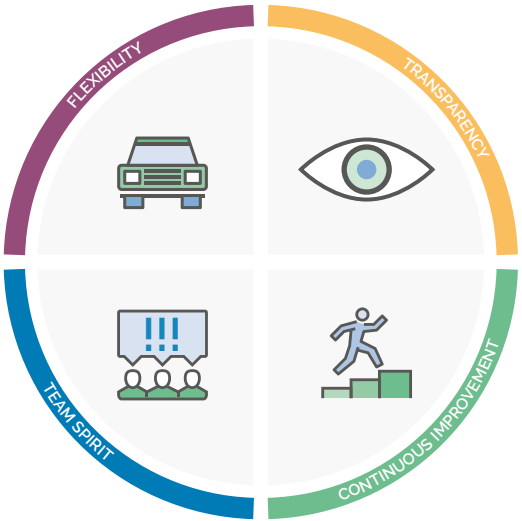
Values, benefits and principles of Scrum

ULASSA
Agile Development Tools

Agile values

- Courage
- Openness
- Respect
- Focus
- Commitment

Benefits of Scrum



Agile principles

- Prioritisation
- Self-organisation
- Empiricism
- Emergent
- Cooperation
- Time-boxed

The Agile Manifesto

ULASSA
Agile Development Tools

➜ ***Individuals and interactions ...***

... are more important to us than ***processes and tools.***

➜ ***Functional software ...***

... is more important to us than ***comprehensive documentation.***

➜ ***Collaboration with the customer ...***

... is more important to us than ***negotiating contracts.***

➜ ***Reacting to change ...***

... is more important to us than ***following a plan.***

Twelve principles are enshrined in the Agile Manifesto (see below). Awareness and embodiment of these principles in an organisation as well as the following agile values are of elementary importance to successful work:

- Communication
- Simplicity
- Commitment
- Courage
- Feedback
- Focus
- Openness
- Respect
- Trust

Agile Manifesto in over 50 languages:
www.agilemanifesto.org/iso/de

Principles behind the Agile Manifesto:
www.agilemanifesto.org/iso/de/principles.html

Product Owner (PO)

- determines **WHAT** is implemented
- is responsible for the product vision
- handles stakeholder management
- sets priorities for the Product Backlog
- ensures that the Development Team has all the information it needs

Scrum Master (SM)

- is responsible for understanding and implementing Scrum
- coaches the Development Team (Servant Leader)
- advises the Product Owner
- represents the agile values within the organisation
- overcomes impediments holding back the Development Team

Development Team

- determines **HOW** the product is implemented
- organises itself
- takes responsibility for its own work
- has members from different disciplines and has all the skills needed to produce a Product Increment
- decides how many features are implemented in the Sprint (pull principle)

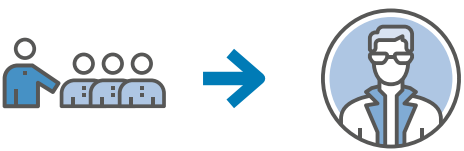
Stakeholders

- are people interested in the product or affected by it (management, end users, service providers)
- are often financial backers of the product
- communicate their product requirements to the PO
- provide the PO and Development Team with feedback

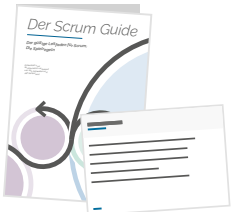
It all starts with a product vision, "What do I want to develop?", "What will make my customers happy?"→ **Product Owner**

Implementation needs an interdisciplinary Development Team, comprising no more than 9 people with the relevant skills.
→ **Development Team**

For continuous improvement the project needs someone who is familiar with the Scrum framework and can support and coach the individuals involved, the team and the organisation. → **Scrum Master**



ULASSA
Agile Development Tools



Read the knowledge cards and Scrum Guide!

I. Break the product idea down into individual features (in Scrum, we call these *User Stories*) using the following structure:

As ...{role} ...
I would like ...{functionality} ...
for ...{business value}.

Meeting: In an initial Refinement Meeting, the PO discusses his or her product vision with the Development Team. The product vision is broken down into small individual features.

The PO prioritises the features by business value and therefore produces the first Product Backlog.

Using the Planning Poker® Cards, each feature is given an estimated value (*see "Estimate" knowledge card*).

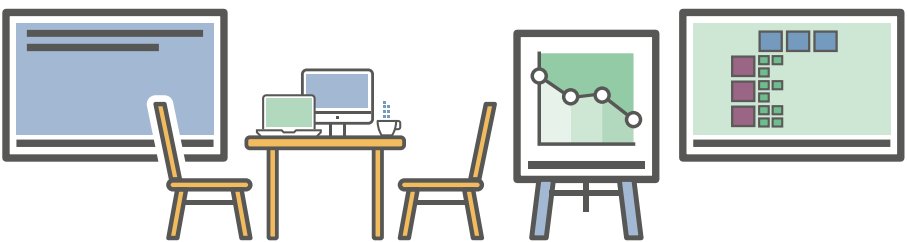
II. Find an empty wall in the team office and produce a task board. Use the tools provided in the Ulassa Agile Starter Box (masking tape, clouds, pens) to produce the following:

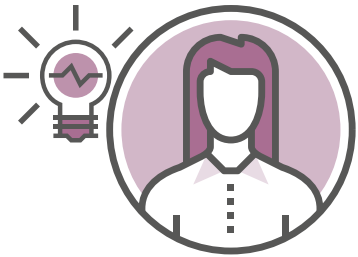
To Do	WIP	Impedi-ments	Done

The Law of Propinquity: *the probability of two people communicating with one another is inversely proportional to the distance between them.*

- One large room for the entire team!
- Space for whiteboards
- Space for the task board
- Flip charts
- Nearby coffee room or another room for discussions

Ensure maximum opportunities for communication. This will allow many problems to be resolved much faster.





The Product Owner (PO) has the vision of WHAT is to be implemented.

Tasks:

- determines **WHAT** is implemented
- is responsible for the product vision
- aims to please the Stakeholders
- handles stakeholder management
- sets priorities for the Product Backlog
- ensures that the Development Team has all the information it needs

Characteristics:

visionary, manager of expectations, decision-maker

PO in a nutshell
www.youtube.com/watch?v=ZUwo8tKoYnQ
In German

PO in a nutshell
www.youtube.com/watch?v=502ILHjX9EE
In English

The Development Team has the skills needed to turn the Product Owner's product vision into reality.

Tasks:

- determines **HOW** the product is implemented
- organises itself
- takes responsibility for its own work
- has members from different disciplines and has all the skills needed to produce a Product Increment
- decides how many features are implemented in the Sprint (pull principle)

Characteristics:

interdisciplinary, get things done, self-organising

The Scrum Master helps the team and organisation to work on the basis of the agile values and to continuously improve.

Tasks:

- is responsible for understanding and implementing Scrum
- coaches the Development Team (*Servant Leader*)
- advises the Product Owner
- is a moderator
- represents the agile values within the organisation
- overcomes impediments holding back the Development Team, which it is not able to remedy itself

Characteristics:

coach, moderator, servant, protector of the Development Team – with the goal of maximising generated value

Stakeholders are people interested in the product or affected by it, such as management, end users or service providers.

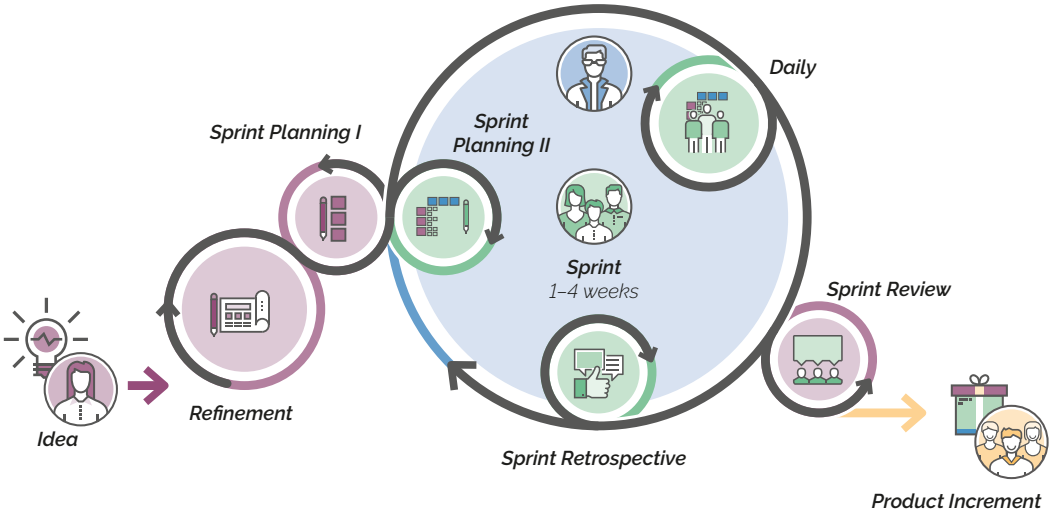
Tasks:

- communicate their product requirements to the Product Owner
- provide the Product Owner and Development Team with feedback and thereby help the Product Owner to assess which features have the greatest business value

Characteristics:

interested in the product, often provide financial backing

Scrum Flow


Idea
Refinement
Sprint Planning I
Sprint Planning II
Daily
Sprint
1–4 weeks
Sprint Review
Sprint Retrospective
Product Increment

Refinement

The Product Backlog is updated, priorities set and estimated values determined.

Time: 60–120 min per week

Participants:

Sprint Planning I

The PO presents his or her most important (i.e. the highest priority) features from the Product Backlog that he or she would like to be implemented in the next Sprint. *The Development Team decides how much of this can be implemented in this Sprint.*

Time: 60 min per Sprint week

Participants:

Sprint Planning II

The Development Team plans the next Sprint and fills the Sprint Backlog with the tasks to be completed in order to successfully implement the features defined in Sprint Planning I.

Time: 60 min per Sprint week

Participants:

Daily Scrum

The Development Team meets at the Sprint Backlog to plan the day. Each member of the team answers 3 questions:

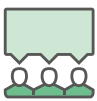
1 *What did I do yesterday for the team?*

2 *What am I planning to do today for the team?*

3 *What is preventing me from creating value for the team in my work?*

Time: 15 min per day

Participants:



Review

The Development Team presents the implemented features to the PO and Stakeholders.

Time: 60 min per Sprint week

Participants:

Retrospective

The team undergoes the last Sprint Review and evaluates how it can work together even better in the future (continuous improvement).

Time: 45 min per Sprint week

Participants:

The focus in the **Refinement Meeting** is on the product vision and the linked Product Backlog. The Product Backlog is updated, priorities set and estimated values determined.

How do you produce a User Story (feature) for the Product Backlog?
The **INVEST rule** is useful here:

Time:

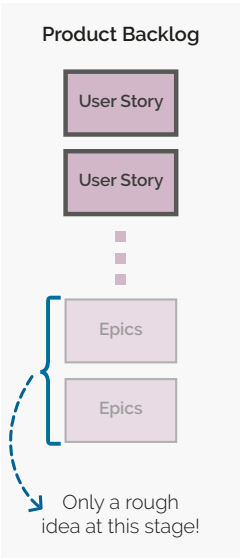
60–120 min per week

Participants:

When:

1–2 times per week

I	**Independent:** A story is independent of other features.
N	**Negotiable:** User Stories are not cast in iron. PO and Development Team discuss and define them together.
V	**Valuable:** The stories should deliver recognisable added value.
E	**Estimatable:** A story must be clear enough for implementation of the requirement to be estimated.
S	**Small:** Features should be small enough to be implemented in one Sprint. If this is not the case, the User Story is split.
T	**Testable:** It must be possible for the feature to be tested to determine whether the acceptance criteria have been successfully completed.



Using the DEEP rule helps to produce a good Product Backlog.

DEEP rule:

D **Detailed appropriately:** The features, which have a high business value and are therefore right at the top of the Product Backlog, are already described in detail. Features, which are only roughly outlined, are further down the Product Backlog (they are also known as Epics).

E **Estimated:** The features are estimated by the Development Team with the help of the Planning Poker® Cards and expressed in Story Points.

E **Emergent:** The Product Backlog changes all the time as new knowledge and new features may be added and priorities may change.

P **Prioritised:** The features are prioritised – the higher the business value, the higher the priority.

Structure of the User Story

A User Story always also includes the acceptance criteria – i.e. things which need to be satisfied for the feature to be successfully implemented.

As ...{role} ...
I would like ...{functionality} ...
for ...{business value}.

The relative complexity of a feature is jointly estimated by the team and numbered using Story Points.

To undertake Planning Poker®, a simplified Fibonacci sequence is used: **0, 1, 2, 3, 5, 8, 13, 20, 40, 100.** The numbers, known as *Story Points*, do not indicate the amount of work in days. Instead they are relative variables, which reflect the *complexity* of a feature, i.e. the "intellectual outlay" involved in its implementation.

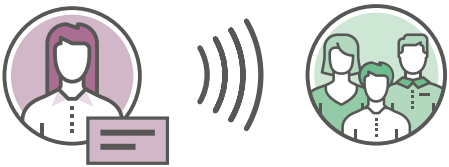
Why do we use this sequence of numbers? An estimate is hard to produce for very complex tasks, so the spacing between the numbers increases with complexity (13, 20, 40, 100). The smaller the task, the better and more accurately it can be estimated.

Velocity: Velocity describes the number of User Story Points the Development Team implements per Sprint.

Features estimated to a value of 20 or higher should not usually be implemented in one Sprint. It is a good idea to break such features down into several features (User Stories) in the next Refinement Meeting.

The outcome of the estimates is especially interesting for the PO. Using this and the velocity of the team, the Product Owner is able to carry out release planning for all features.

I The PO presents the feature (**What**) and explains the **business value** and **acceptance criteria**.



II If the Development Team has no further questions, the process of estimating the feature begins:

- Each member of the Development Team is given one set of poker cards, comprising the cards *0, 1, 2, 3, 5, 8, 13, 20, 40, 100, ∞, Question, Pause*.
- Individually and without discussion, each person assesses **how complex the User Story is** (expressed in Story Points), and selects a card, which they lay face down in front of them.
- As soon as all members have put down their cards, they are turned over. If there are differences in the estimations, **the Scrum Master initiates a discussion**. This is done by asking the two people with the highest and lowest numbers to discuss their estimates.
- After these views have been heard, either the Development Team agrees on a number or a second round of discussion takes place.

Before the estimation process starts, the Development Team selects a User Story of medium relative complexity (e.g. 5 Story Points). This "original Story" serves as the reference parameter for other User Stories in the future.

Time:

60 min per Sprint week

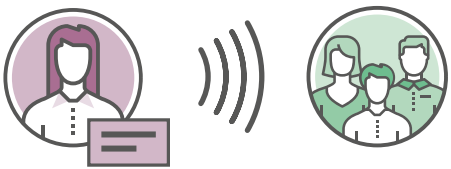
Participants:

When:

Start of Sprint

The Product Owner presents his or her most important (i.e. the highest priority) features from the Product Backlog that he or she would like to be implemented in the next Sprint.

→ **WHAT** is to be done



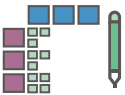
Product Backlog

User Stories with highest business value

The Development Team decides how much can be implemented in this Sprint!

Sprint Planning II

ULASSA
Agile Development Tools



Time:

60 min per Sprint week

Participants:

When:

At start of Sprint, directly after Sprint Planning I

The Development Team plans the Sprint. → **HOW** do we want to implement these features?

User Stories sorted by business value.
→ Use of square Post-It® notes

All **To Do Tasks after** Sprint Planning II, i.e. all written tasks.
→ Use of square Post-It® notes Super Sticky

WIP (Work in Progress): tasks undertaken between the Daily Scrums.

Post-It® notes detailing **impediments** hampering the Development Team from carrying out its work are stuck here.

All **completed tasks.**

To produce a workflow, we have found that it is useful to divide several larger tasks into lots of small tasks.

Time: 15 min per day

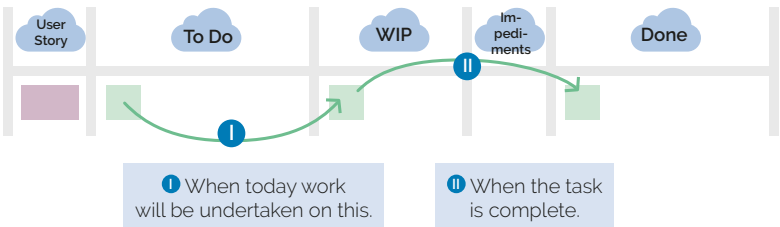
(Time-boxed! After 15 min, the meeting comes to an end – this helps the Development Team to focus.)

Participants:

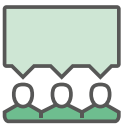
When: Ideally every day in the morning at the same time (the Development Team is self-organising and decides this itself)

The Development Team meets at the task board to plan the day. Each member of the team answers 3 questions:

1. *What did I do yesterday for the team?*
2. *What am I planning to do today for the team?*
3. *What is preventing me from creating value for the team in my work?*



If impediments arise which no-one in the Development Team can overcome, they are stuck to the task board under *Impediments* so that the Scrum Master can look into them.



Time:

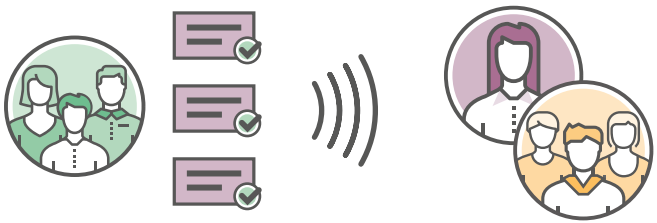
60 min per Sprint week

Participants:

When:

At the end of the Sprint

The Review focuses on **feedback from the Product Owner** and **Stakeholders**. The Development Team presents all features (User Stories) completed in this Sprint to the Product Owner and Stakeholders.



The Development Team is **self-organising** and therefore decides itself which team member makes the presentation and how the presentation is given.

The meeting is an **open meeting**. The Product Owner can invite Stakeholders who are interested in the features implemented in this Sprint.

Time:

45 min per Sprint week

Participants:

(The Development Team decides whether the PO attends the meeting.)

When:

At the end of the Sprint after the Review Meeting

With the support of the Scrum Master, the team reflects on the last Sprint and works out how it can collaborate even better in the future.

Start to ...	Stop to ...	Continue to ...
Things we want to do in the future	Things we no longer want to do in the future	Things we want to continue doing

A lot of important things happen in the team in this meeting: The team improves with every Sprint, so it gradually develops into a high-performance team!

Examples and ideas associated with the Retrospective: **www.plans-for-retrospectives.com**

Product Backlog, Sprint Backlog and Product Increment are three key artefacts in Scrum. Other artefacts such as User Story, Definition of Done and Burn Down Charts support the Scrum Flow.

Product Backlog

A list of features (User Stories) sorted by business value. Using the **DEEP rule** helps to produce a good Product Backlog.

User Story

The description of a product feature. The best way for the Product Owner to create the User Story with acceptance criteria is using the **INVEST rule** and the model:

As ...{role} ...
I would like ...{functionality} ...
for ...{business value}.

Sprint Backlog

The Sprint Backlog contains those **User Stories**, which are being implemented in the current Sprint by the Development Team. To visualise the Sprint Backlog, it is a good idea to create a task board containing all the tasks which must be implemented to complete the User Story.

Tip! Anyone who takes on a task under **WIP** puts his or her initials next to it. This reinforces the person's commitment to deal with the task.

Definition of Done (DoD)

A checklist, containing all those activities which must be implemented to complete a User Story. The DoD is produced at the start of the project but can be adapted by the Development Team at any time.

Burn Down Charts

Transparency is key to Scrum. A Sprint Burn Down Chart shows the User Story Points still to be implemented for the current Sprint. A Release Burn Down Chart shows the work still outstanding (in User Story Points) for the next product release.

Product Increment

At the end of a Sprint, the Development Team always has a product which can potentially be delivered. The product may comprise between 1 and n features from this and previous Sprints.

Ideas of how to adapt the DoD often emerge during a Retrospective.

Every day we achieve incredible things as teams – but it's often hard to find a moment to acknowledge our colleagues' hard work with positive feedback.

That's where the Awesome Cards come in. Use them for a Retrospective or simply leave them out in your team room, explain their purpose (acknowledgement and positive feedback) – and see what happens!

TED Talk Video
http://go.ted.com/bXwU
"Everyday Leadership"

Important agile laws – Part 1

ULASSA
Agile Development Tools

1 **Parkinson's law:**

Work expands to fill the time available for its completion.

2 **Goodhart's law:**

When a measure becomes a target, it ceases to be a good measure.
Measures can be influenced.

3 **Humphrey's law:**

Users don't know what they want until they see the end result (and sometimes not even then!). **So short Sprints and Review Meetings.**

4 **Law of Propinquity:**

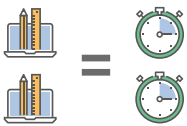
The probability of two people communicating with one another is inversely proportional to the distance between them.
Ideally the team is based in one room.

5 **Law of the instrument:**

If all you have is a hammer, everything looks like a nail.

6 **Conway's law:**

The interface structure of a software system will show a congruence with the social structure of the organisation that produced it. **Communication and power structures are evident in the company's products.**



7 **Little's law:** Processing time = $\frac{\text{number of parallel tasks}}{\text{output}}$

If you work on twice as many things in parallel, the processing time is doubled too.

8 **Dude's law**: Value = $\frac{\text{Why}}{\text{How}}$

If you don't have a good reason for a product, it doesn't matter how well you produce it. **Always start with the question why!**

ULASSA
Agile Development Tools

Everything starts with the question why ...

WHY?

We believe in ...

... the ability to have a positive influence in the world

... continuous improvement

... growth of individuals and organisations

HOW?

We develop products with passion and taking account of agile values so that they are easy to use and you can instantly get into agile software development with Scrum.

WHAT?

We provide a wealth of agile development tools – including the Agile Starter Box, Planning Poker® Cards, Awesome Cards, Scrum knowledge cards and much more.

Space for notes: