



Scrum

This document is created as per Scrum guide 2020, which will help people to get some better understanding of Scrum framework and cover areas for PSM 1 or CSM exams.

Author:

Muhammad Zahid Hussain

Date:

August 21, 2022



Agile:

- Agile is the ability to create and respond to change. It is a way of dealing with, and ultimately succeeding in, an uncertain and turbulent environment. It is an approach to software development that seeks the continuous delivery of working software created in rapid iterations.

Agile Mindset:

- Being Agile is not simply a matter of using a certain set of tools or practices or following a specific methodology.
- Agility involves a “new mindset” and “way of thinking” – which is based on agile manifesto including 4 values and 12 principles.

Doing Agile Vs Being Agile:

- Doing Agile refers to Agile Practices like Iterative and incremental development including events like Daily Stand ups etc. Also have “No agile mindset”.
- Being Agile refers to “Agile Mindset”, “Embracing Agile Practices” and “Tailoring Process”.

Agile Project Manager/ Scrum Master, Must:

- Embrace Agile Mindset.
- Understand Agile Principles.
- Sell the idea of Agile / Educate others.
- Expect skeptics / Coach other.

Agile Team Members, Must:

- See the value of agile approach. (Why we are using it?)
- Experience benefits agile provides.
- Have quick victories/ Wins.
- Main Concept in agile is that there is prioritize backlog.

Project Stakeholders, Must:

- Be willing to try agile approach.
- Be convinced on agile values.
- Experienced quick ROI (Return of Investment).
- Eat dessert (good stuff) first.

Agile Principles and Mindset Tasks:

- Advocate for agile principles and values in the organization.
- Ensure common understanding of agile principles.
- Transparency equates to trust.

- Share knowledge and new collaboration, (Daily Stand ups and retrospective sessions).
- Emergent leadership, mean anyone in team can become a leader.
- Practice servant leadership.

Agile Values and Principles:

- There are two strategies to fulfill Agile values and principles: -
 1. Adopt a formal agile approach, designed and proven to achieve desired results.
 2. Implement **changes** to project. Practices in a manner that fits the project context to achieve progress on a core values and principles.

Agile Manifesto:

- Agile manifesto includes 4 values statements and 12 guiding principles. These were written by software development experts.
 1. Individuals and interactions over processes and tools
 2. Working software over comprehensive documentation
 3. Customer collaboration over contract negotiation
 4. Responding to change over following a plan

Principles behind the Agile Manifesto:

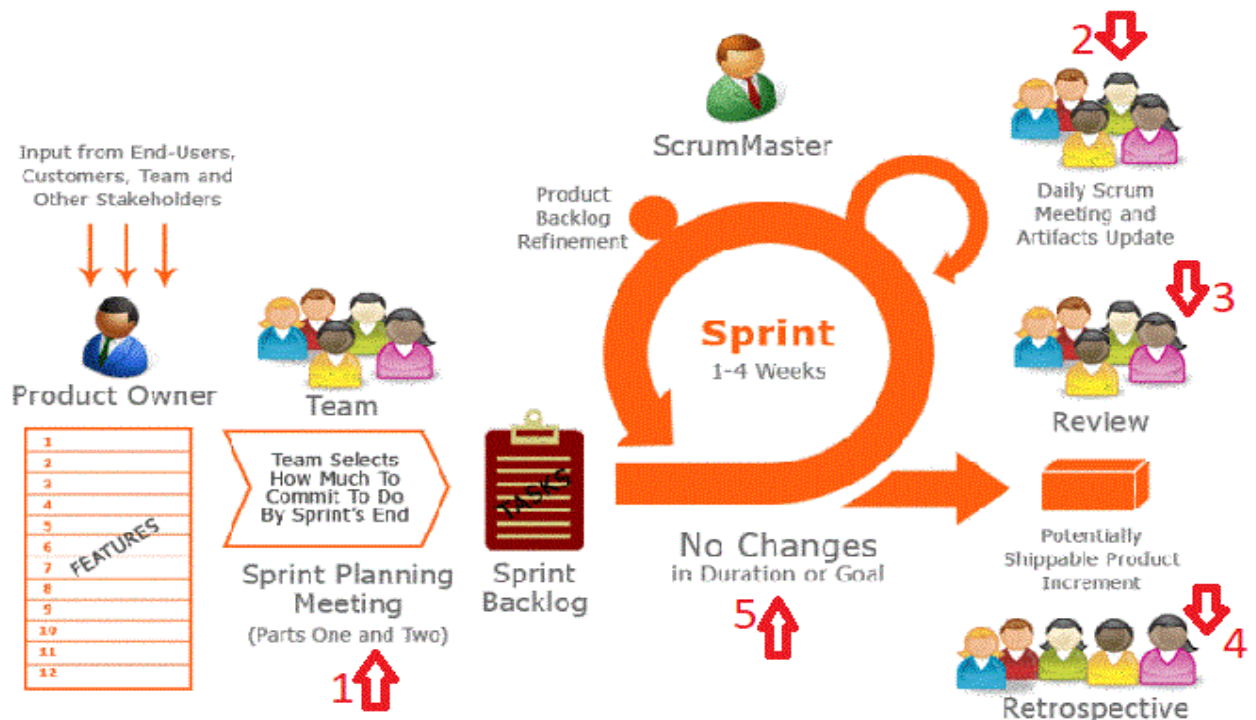
1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of 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.
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.
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.

Scrum in Short:

- In Scrum teams take a problem and divide into smaller chunks and run incremental and iterative method:

For Example:

- Listing the features of website like (Signup, Login, CMS and Payment) etc.
- Team will list all these features in (Product Backlog) and order the list to work on upper items on priority.
- After prioritizing the items team collaborate in (Sprint Planning) event and select few top items and added them in (Sprint Backlog).
- Team work on sprint backlog items during (sprint) and at the end product (increment) was delivered.
- (Increment) for any service or product should be up & running and new functionalities were ready to go OR something customer can see and use.

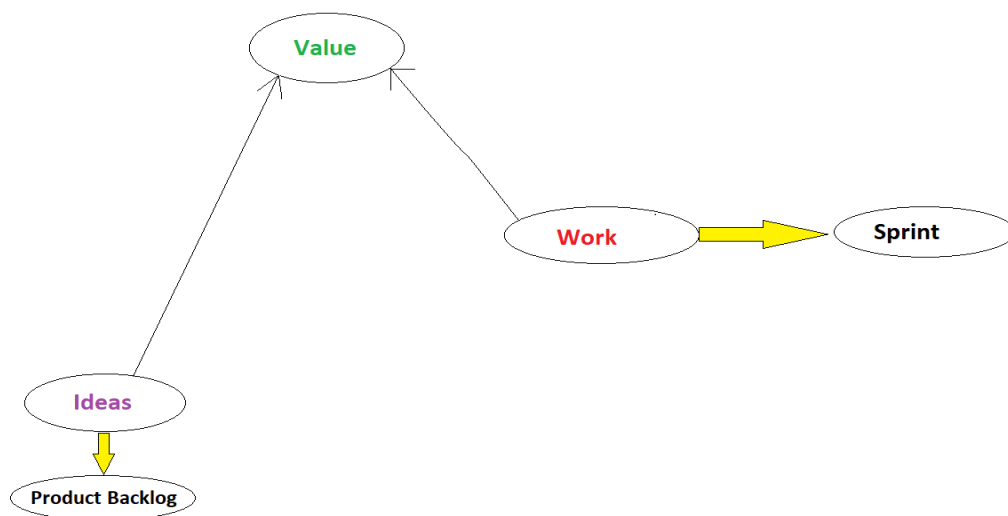


The Scrum Flow:

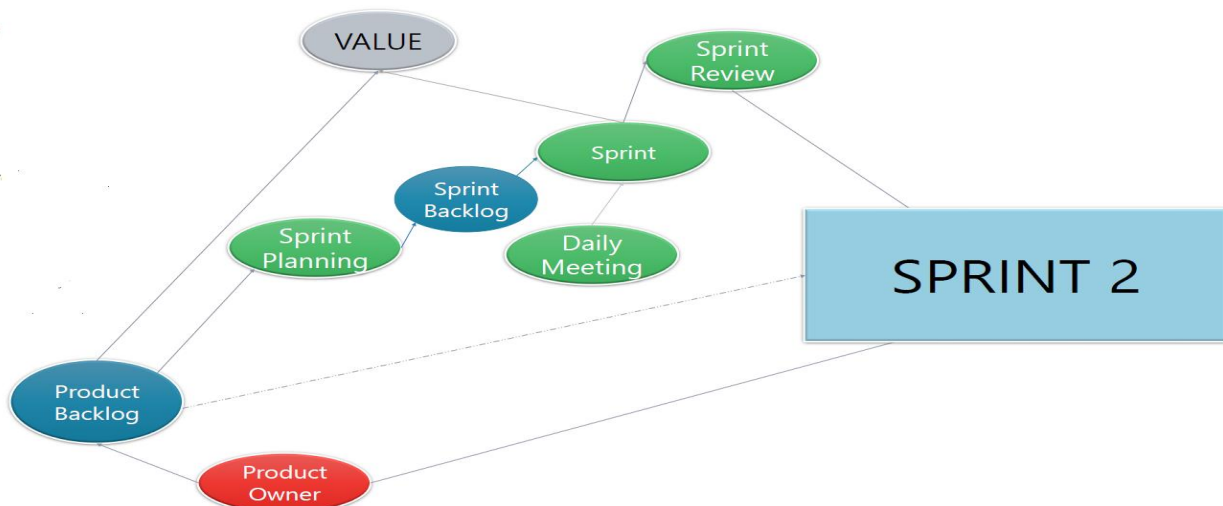
1. Scrum starts from Product Backlog which contains all work to be done or describes/represents what will fulfill the Product Goal and it is maintained by Product Owner. He/ She also make sure that top items are important one.
2. Then we have planning meeting in which developers select an item from Product Backlog (Sprint Backlog created).
3. Sprint Backlog promotes expectational alignment between business and technical team.

4. During the execution of sprint developers run Daily Scrum to synchronize their work and maximize chances of achieving product goal.
5. During the sprint the moment Product Backlog items meet the Definition of Done (DOD) an Increment is born.
6. At the end of sprint team run review meeting in which Product Owner invites the stakeholders to inspect the produced increment. During this event feedback from stakeholders might impact Product Backlog.
7. After Review meeting scrum team participate in Retrospective meeting in which they inspect and adapts how the work has been done and plan for continue improvement.
8. Scrum Master is responsible to run whole scrum process smoothly.

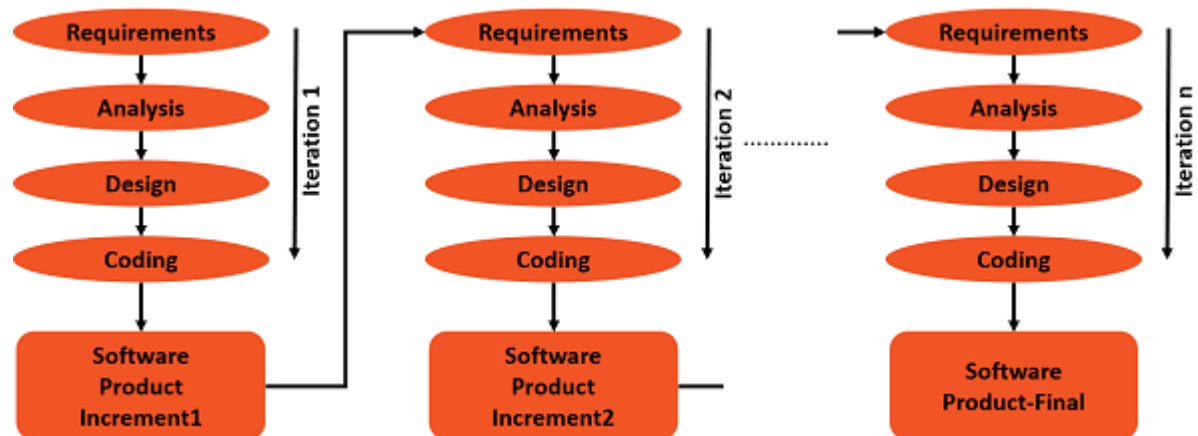
Scrum as a Value Driven Effort:



1. Ideas are register in Product Backlog which comes from Product Vision. To deliver better value product backlog must have good value too.
2. Product Owner is responsible for maintaining Product Backlog.
3. Product Backlog Refinement is ongoing process.
4. Increment is the output of the sprint also we called it vertical delivery.



- Scrum follows Agile methodology of Iterative and Incremental development. A partial implementation of a total system is constructed so that it will be in a deliverable state. Increased functionality is added. Defects, if any, from the prior delivery are fixed and the working product is delivered. The process is repeated until the entire product development is completed. The repetitions of these processes are called iterations. At the end of every iteration, a product increment is delivered.



Scrum Theory:

Scrum is founded on following:

- Empiricism:** Its mean knowledge comes from experience and making decisions based on what is being observed.
- Lean Thinking:** Its mean new way of thinking and focus on essentials (most needed) and reduces wastes. Like if daily scrum event objective completed in 5 minutes, then there is no need to stay in meeting for 15 minutes.

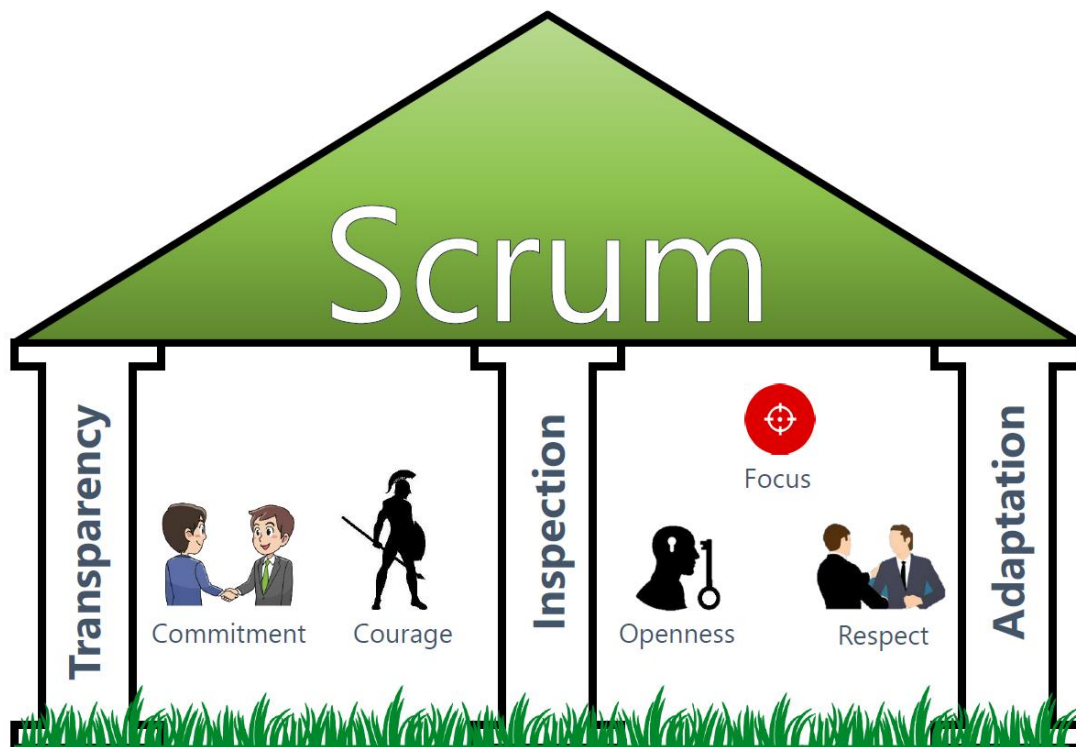
Empiricism stands on Three pillars which are following:

- Transparency:** It refers to common standard of understanding mean all observers share a common understanding on what is being seen and spoke. The significant aspect of process must be visible to those responsible for outcome. In Scrum transparency is implemented in 4 Scrum events and 3 artifacts.
- Inspection:** It is a testing hypothesis for checking what is happened also learning and adapting from it. Scrum set 4 events for inspection.
- Adaptation:** During inspection if any process seems deviating from outside limits, then adjustment must be made as early as possible.

Scrum Values:

If all accountabilities implemented, artifacts maintained but values missed its mean Scrum is not implemented from its core. Scrum refers five values:

1. **Commitment:** People personally commits for achieving the goal of scrum team. In scrum business and technical people must work together. Entire team is held accountable for the results. It does not matter your areas of specialization, if one succussed all succussed and if one failed all failed.
2. **Courage:** Scrum Team members must have courage to do right things and work on tough problems. This is necessary while working on though problems members must go through hard path when they have to go through tough problem.
3. **Focus:** Every team member focus on work of the sprint and the goal of the scrum team.
4. **Openness:** This is strongly related to transparency pilar. Scrum team and stakeholders agreed to open about all the work and challenges while performing the work. For this most commonly (Information radiator such as Scrum Board etc. was used).
5. **Respect:** Team members must respect each other to be capable independent people. It is essential for complex growing high performing team.

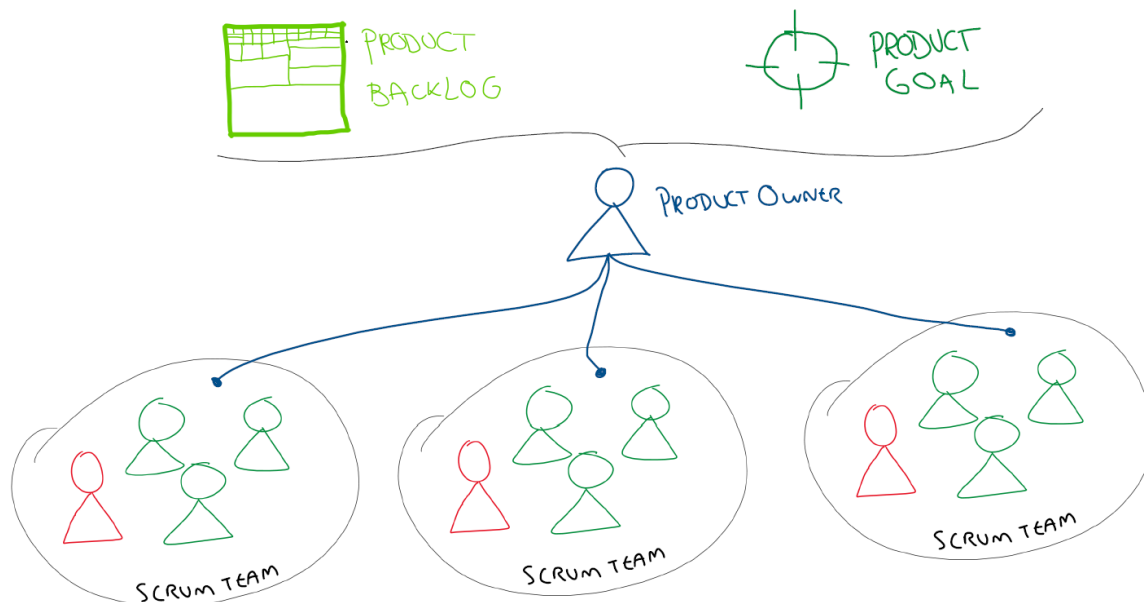


The Scrum Team:

Scrum team is small team of people. Also, there is no sub-teams or hierarchies concept exist. Team consists of one Product owner, one Scrum master and 10 or less developers.

Scrum team have following three accountabilities/ roles:

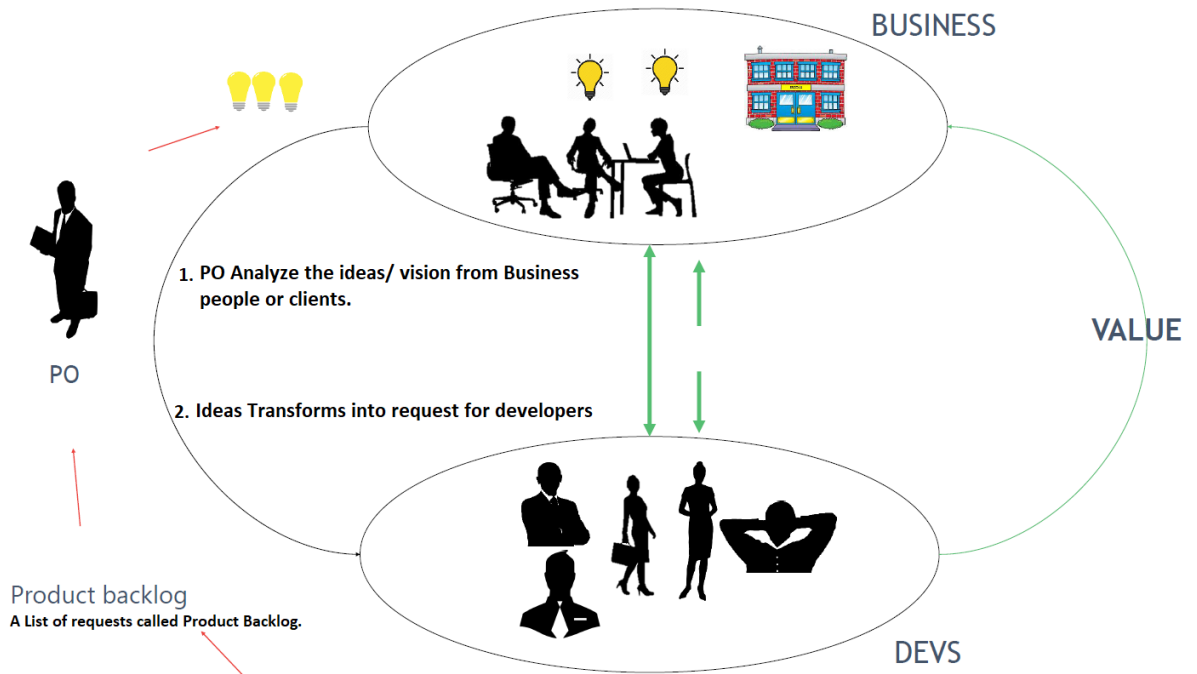
1. **Product Owner:** It is a business person who take care of product backlog.
 2. **Developers:** These are responsible for transforming the ideas reflecting in product backlog into usable increment.
 3. **Scrum Master:** Responsible to run scrum smoothly.
- If Scrum team become too large than it should be decomposed into multiple cohesive (same nature of thinking) teams and each should focus on same product goal and have same product backlog.



- Collectively whole scrum team is responsible for all type of activities from stakeholders' collaboration to end delivery. Also, whole team is accountable for delivering usable increment.
- It is cohesive unit of professionals which focused on one objective at one time that is product goal.
- Team should be (Self-Managed → They know what, when and how to do?) and (Cross Functional → All necessary skills to deliver valuable increment).

Product Owner:

- Responsible for the value delivery chain or one for maximizing the value of product resulting from scum team. Following image represents how product owner maximize the value of product.



- He/ She manages the product backlog. Which included communicating and developing Product goal and ordering of items in product backlog.
- PO is not a necessary bridge between a client/business team and developers. Both developers and client can take to each other if mandatory but they can't command and interrupt developers. Scrum Master need to make sure all this.
- Weak Product Owner can't work effectively. He/ She should have authority to prioritize the work, inspect the increment and adapt in timely manner.

Product Owner – Key take aways:

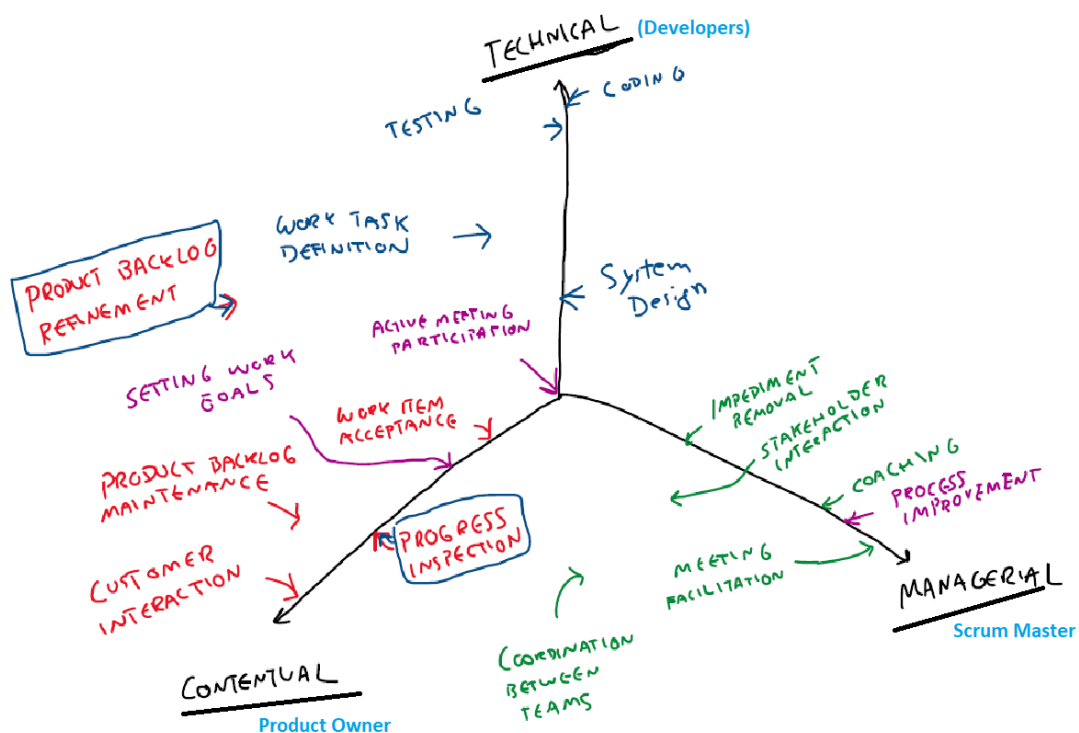
- Both technical and business people must understand scrum to benefit from it. If it is not a case Scrum Master need to arrange a coaching sessions or work with traditional Project Management approach by closing the scope of work.
- He/ she must have business knowledge and authority.
- He/ She must be available to team whenever needed.
- Other might do work related to product backlog management but Product Owner remains accountable of it.

The Developers:

- Responsible to transform Product Backlog items into a usable increment as per Definition of Done.
- They are Cross Functional and Self-Managed.
- They do commitment for achieving sprint goal.
- Accountable for creating sprint plan which called sprint backlog.
- Accountable to run Daily Scrum and adapting their plan daily towards sprint goal.
- Hold each other accountable as professionals.

The Scrum Master:

- He serves organization and scrum team.
- He helps outsiders to positively interact with scrum team.
- He/ She accountable for scrum team effectiveness.
- Coaches the team and organization on scrum understanding.
- Causes the removal of impediments. These can be raised by anyone at any time.
- Leads and facilitate scrum adaptation like facilitating meetings.
- Scrum Master is also a manager who manages scrum framework. Although he hasn't authority on team or manage people.
- Scrum master must not "hunt" impediments/ Blockers. Also, does not necessarily removes all impediments. (Team should be self-managed and they should know how to remove impediments and scrum master should coach them on self-management).



Dealing with Undone work and Impediments:

1. Should the undone work be part of the increment?

Answer: No, it should not be a part of increment because of following reasons:

- a. Undone part of increment will hinder the autonomy of increment.
- b. Presenting undone work in review meeting may cause communication issues with stakeholders because they might think work was done and hindering transparency.
- c. Further developers think it is OK to deliver undone work. They need to know how to manage undone work.

2. Should Undone work be presented to stakeholders during sprint review?

Answer: Such items should not be a part of review meeting but these should move back to product backlog. These items might come in discussion during review meeting.

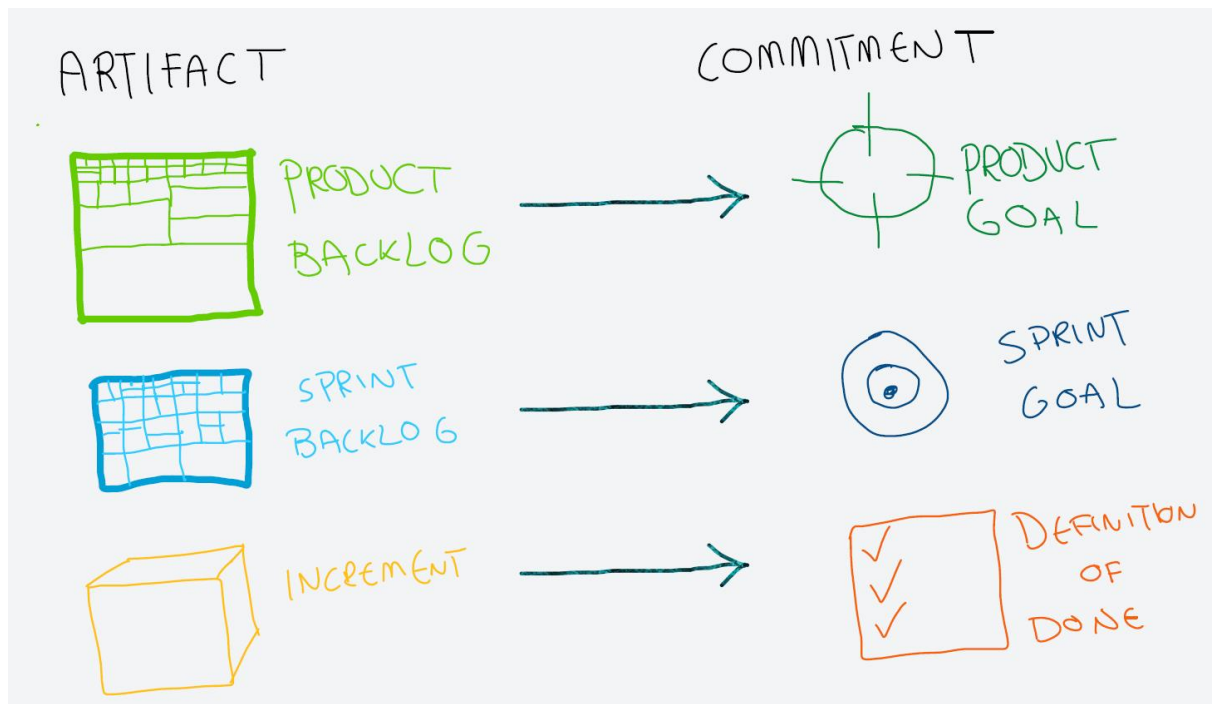
3. Dealing with hostile developer. Such developer has conflicts with others in team and creating a hostile environment. If necessary, who is responsible to remove developer? PO? SM? Developers or HR?

Answer: Scrum Master is responsible because the concern developer is considered as impediment? **NO**, Developers should remove it scrum master help them in case they are unable to remove such impediments.

- Scrum Master should foster self-management.
- If team raised any impediment than scrum master should not solve it by own but train them how to resolve it because team is self-managed.

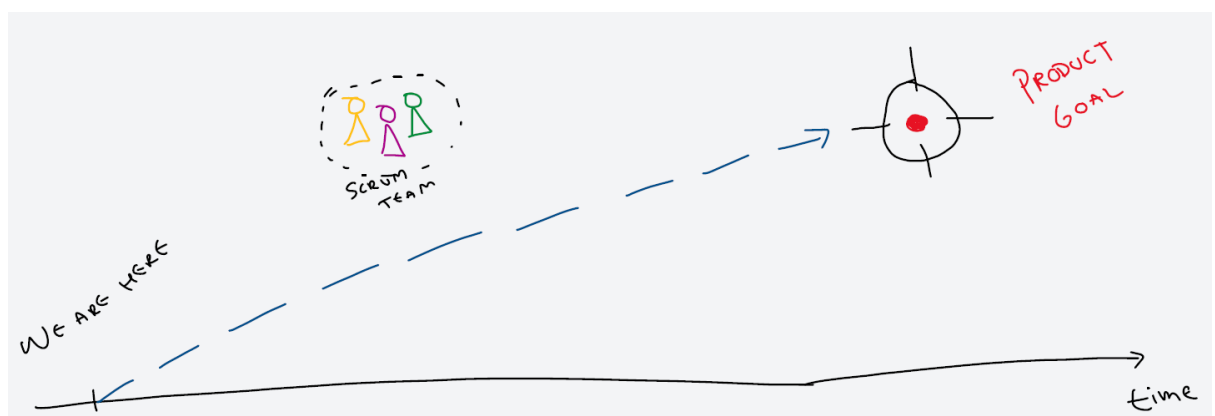
Scrum Artifacts:

- Scrum have three artifacts (Product Backlog, Sprint Backlog and Increment). These artifacts support all three pillars of Transparency, Inspection and Adaptation.
- Product Backlog represents the work “TODO”, Sprint Backlog represents work “Plan” for sprint and Increment represents “Value” and “Transparency” for inspection and adaptation.
- Artifacts are designed to maximize transparency and all stakeholders should align with it.
- Each artifacts focus can be measured by commitment.

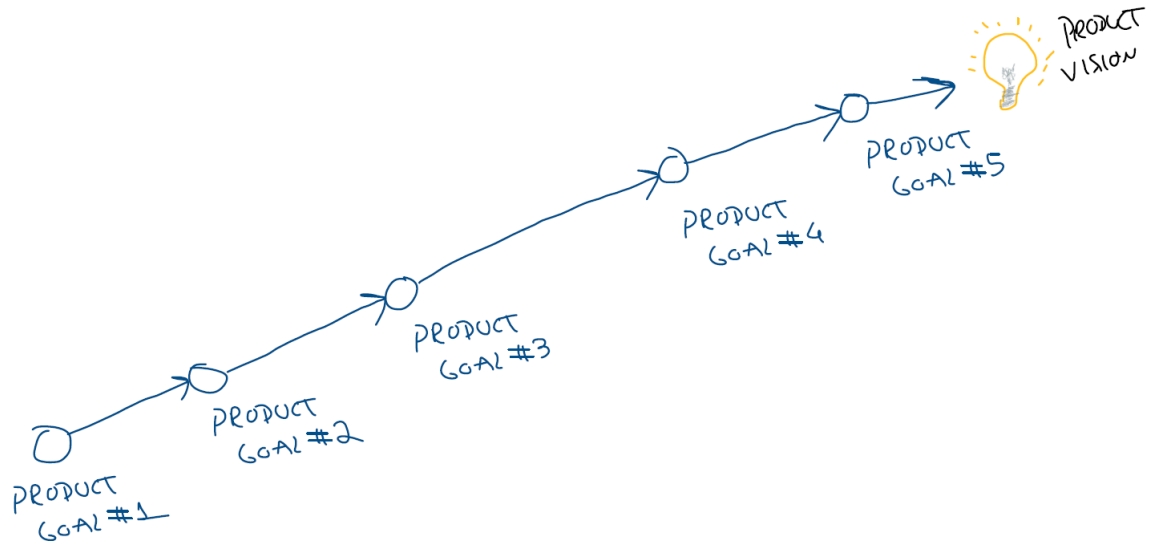


Product Goal:

- It describes the future state of the product which can serve as target for scrum team to plan against.
- It exists within the product backlog and usually based on assumptions.
- It describes the future state of the product.



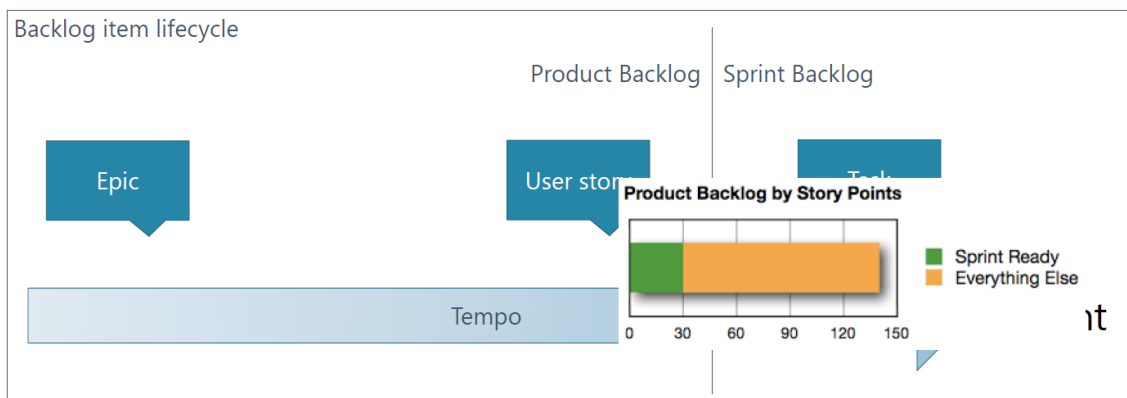
- Product itself can not be a goal. It just means to achieve high level objects for example earn money or automate the manual process to save time.
- There can be multiple product goals which might be connected and organized to a product roadmap. (Product roadmap is not a part of official Scrum Guide).



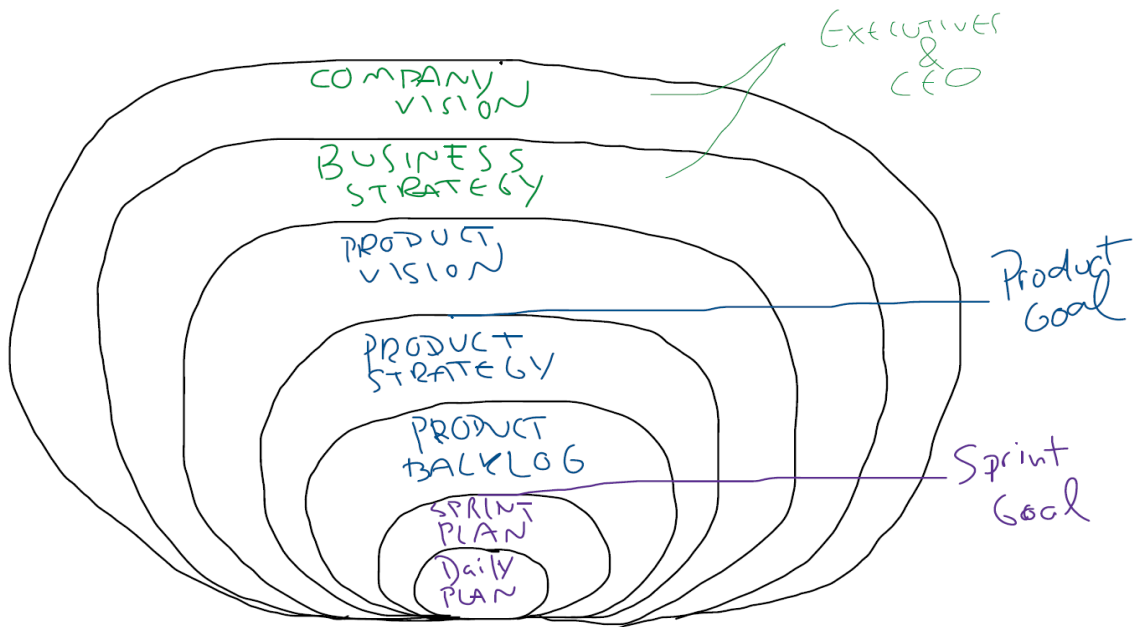
- If there are multiple product goals than team should focus on one product goal at one time.
- Product Owner is responsible for developing and explaining product goal.
- Common Product goals in organizations areas following:
 - ✓ Acquiring or retaining Customers.
 - ✓ Increasing Engagement
 - ✓ Generating Revenue and Reducing cost.

Product Backlog – Overview:

- It is an emergent ordered list of what is needed to improve the product. It is a single source of work to do by scrum team. List may contain requirements, features or enhancements.



- Common attributes of product backlog items are Description, Order and Size.
- Product backlog is manifesto of product strategy mean “How vision should be realized”.
- Following onion diagram shows how Company/ Client vision transform into request for developers to deliver increment.

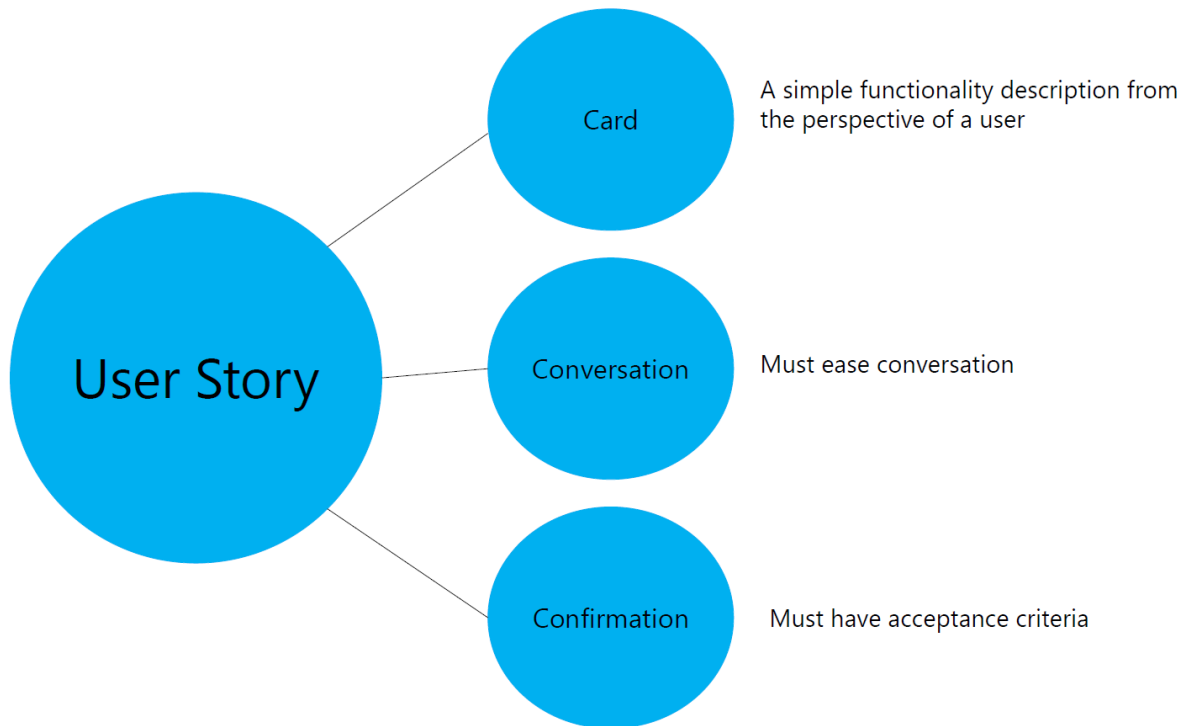


- Product vision represents the vision of company or client that reflects the purpose to develop a product.

THE PRODUCT VISION BOARD

<p>VISION What is your purpose for creating the product? Which positive change should it bring about?</p> <p><i>Enable customers worldwide to buy in Japan with ease and low cost</i></p>			
<p>TARGET GROUP</p> <p>Which market or market segment does the product address? Who are the target customers and users?</p> <p><i>Customers</i></p> <p><i>Collectors</i></p> <p><i>Store owners</i></p> <p><i>Internal</i></p> <p><i>Customer support</i></p> <p><i>Operator</i></p> <p><i>Marketing</i></p>	<p>NEEDS</p> <p>What problem does the product solve? Which benefit does it provide?</p> <p><i>Current services to buy in Japan are complex, expensive or low quality</i></p> <p><i>You lost a lot of time "mining" for items</i></p>	<p>PRODUCT</p> <p>What product is it? What makes it stand out? Is it feasible to develop the product?</p> <p><i>Web app</i></p> <p><i>Mobile app</i></p> <p><i>Easy and safe</i></p> <p><i>Real-time bidding</i></p> <p><i>Manage store orders</i></p> <p><i>Schedule bid</i></p> <p><i>Recommend auctions</i></p>	<p>BUSINESS GOALS</p> <p>How is the product going to benefit the company? What are the business goals?</p> <p><i>Reduce cost to sell Japanese products</i></p> <p><i>Expand from selling only to Brazil to sell worldwide</i></p>

- Most commonly items exist in the form of “**User Stories**” in product backlog. But User story is not an official part of scrum guide.



Ordering the Product Backlog:

- Scrum guide did not prescribe any detail or rule about ordering of items in product backlog.
- Product owner orders considering what he/ she judges as appropriate. Like as per size, complexity or business value.

		BV	SIZE	ROI		
2	US1	10	5	2	2	4
1	US2	12	10	1,2	4	3
3	US3	6	3	2	3	2
4	US4	5	2	2,5	1	1
5	US5	4	5	8	5	5

- Attributes like Size, Order and Description often vary with the domain of work.

Sizing the Product Backlog Items:

- Developers are responsible for Sizing/ Estimating the product backlog items.
 1. **Note:** The word estimation is not used in Scrum guide.
- Scrum guide did not prescribe any rule for sizing/ estimation.
- Most popular unit is “Story Pointing”.
- Most Popular estimation or sizing technique in agile is “Planning Poker”.
- In agile teams follow “Relative” approach for estimation instead of absolute.

- Story pointing is done as per “Size” and “complexity” of problem. (**Note: Story Pointing is not an official part of scrum guide**).

Planning Poker – Estimation Technique (Not an official Scrum Guide Part):

- It is collaborative and consensus based agile estimation technique.
- In this each developer contains a card with number and each number represent a story point.
- During this activity product owner describes/ explain the product backlog items after that each developer privately select card to represent his/her estimation or sizing.
- If team haven’t consensus on provided estimation, then resource with high and resource with low estimations need to explain reason for estimating. In case team does not agree again this activity will be performed.

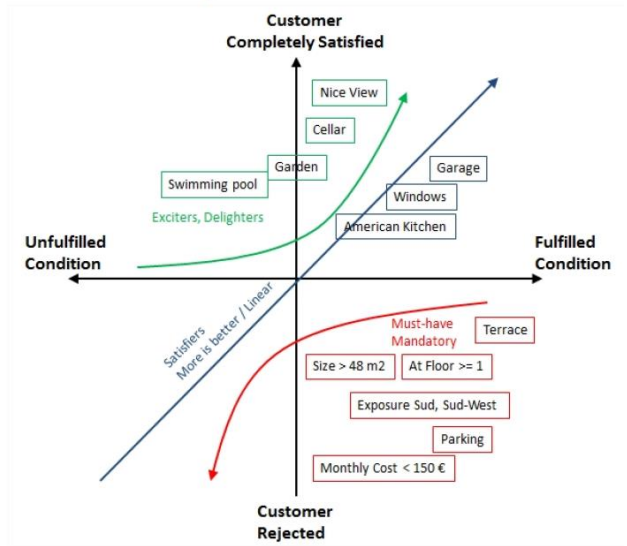
Product Backlog Item	Estimate
Register and authentication	13
Real-time bidding Yahoo Auctions	40
PayPal Deposit	8
Search	13
Shipment	20
CMS	20
Buy on Yahoo Japan	20
Notifications	8
Localization	13
Buy on other Japanese stores	13
Bitcoin deposit	13
Login Gmail	2
Login Facebook	3
Recommend Auctions	20

Product Backlog Item Value:

- Scrum guide did not prescribe any technique for defining value of product backlog items. Its totally depends upon Product Owner what he/ she thinks appropriate for this.
- One of the popular technique is “Kano Model”. It classifies product backlog items in one of three following categories. (**Not a scrum guide Part**).

Desired-based prioritization (Kano Model)

- Features as one of three categories
 - Mandatory
 - Basic stuff. Its presence won't add much value, but its absence has a tremendous negative impact
 - Linear
 - The more, the better. The more linear features, the more customers pay!
 - Exciter
 - Unknown desires



Product Backlog Refinement:

- It is an act of breaking down and further defining product backlog items into smaller and more precise items.
- Refinement done collaboratively by the Scrum team usually by the developers and product owner.
- Scrum team defines how and when refinement is done.

Definition of Ready (DOR): (Not a Scrum guide official part)

- It is a set of criteria that is defined when product backlog items are ready for execution.
- It enhances transparency and shared understanding of product backlog.

FRONT

As a buyer

*I want to bid in real-time on Yahoo-Auctions Japan
so that I can apply auctions winning strategies*

- DOR
- Acceptance criteria
 - Team understands
 - Small enough

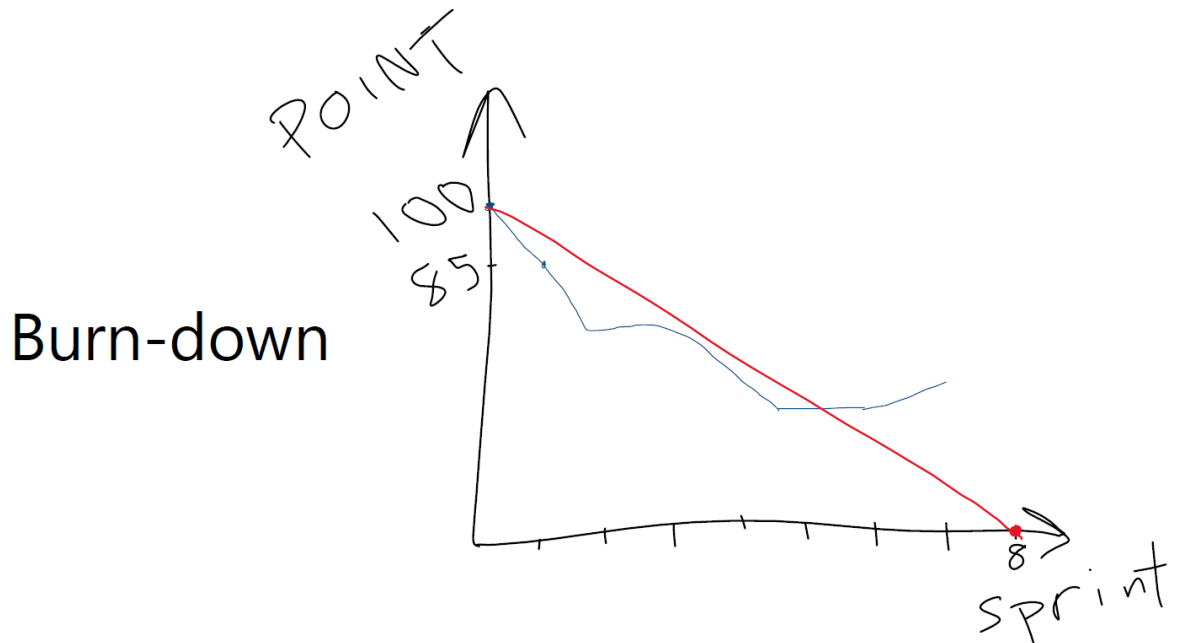
BACK

- User selects the value of the bid
- Bid is executed in real-time
- Bidding is only possible if the user has enough balance
- The value is discounted from the user's balance and registered in the transactions history
- User can increase bidding anytime
- User cannot cancel a bid
- If the user wins an action, the system must refund him for the extra costs if the auction's final value is lower than his bid value
- Update RUNNING AUCTIONS list on user's home page
- If user wins, add it to WON AUCTIONS list on user's home page
- If user wins, charge for service fee

Monitoring Progress Towards Goal:

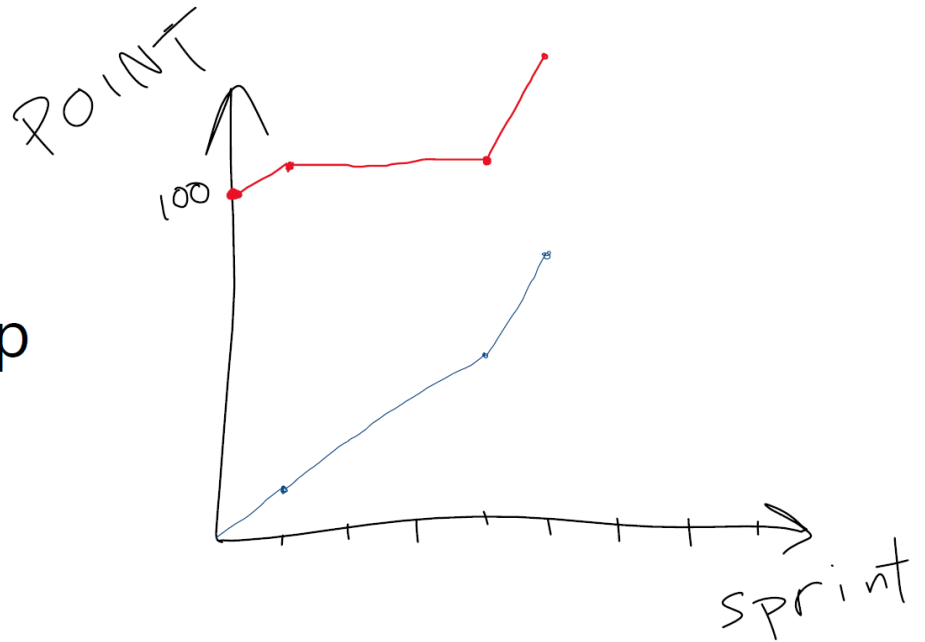
- At any pointing time, the total work remaining to reach a goal.
- Scrum guide writes three possible practices which are as following:

1. **Burn Down Chart:** It shows work left to do versus time.



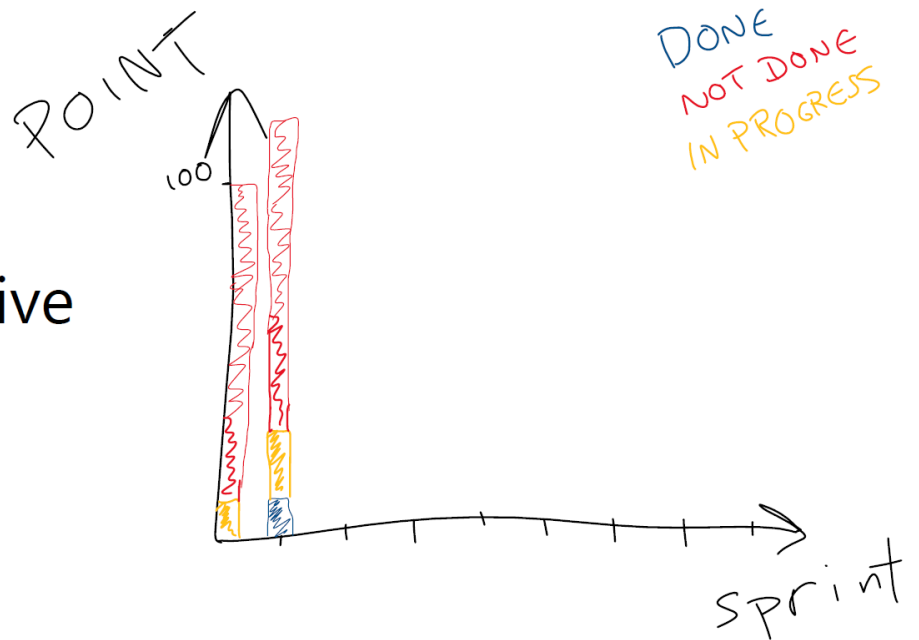
2. **Burn Up Chart:** It shows how much work has been completed (Red line) and shows scope variation (Red line).

Burn-up



3. **Cumulative Flow:** It shows the status of issues over time. This help you to identify potential bottlenecks that needs to be investigated.

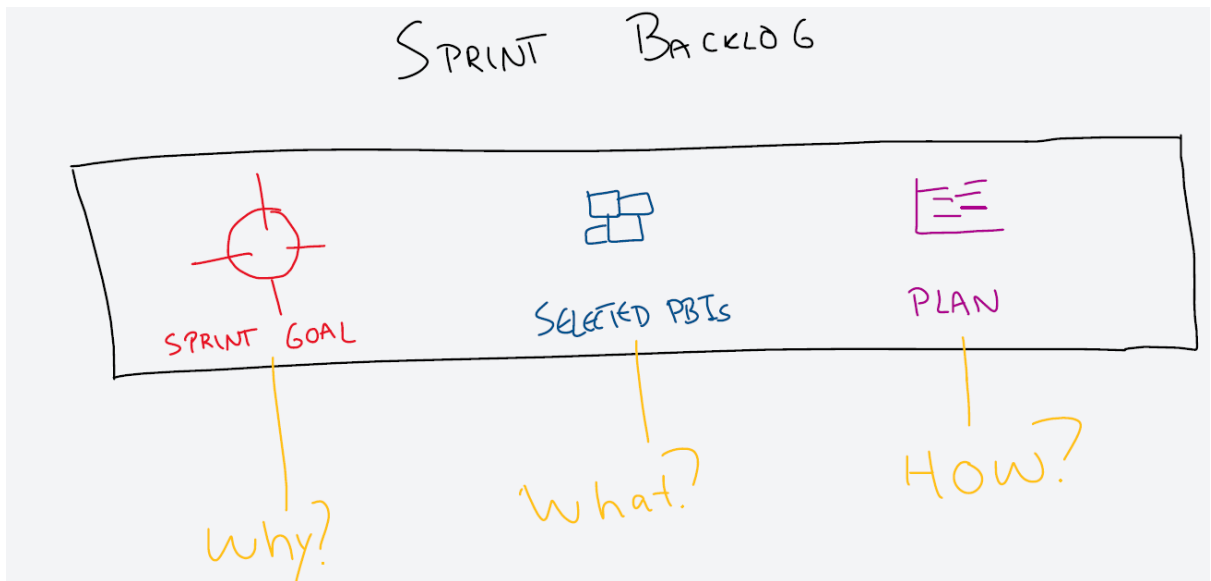
Cumulative Flow



- **It is not necessary for the product goal to be fulfilled all the product backlog items must be done.**

Sprint Backlog:

- Sprint backlog created after sprint planning meeting.



- Sprint goal is fixed but sprint backlog is flexible.
- It is real time picture of work that the developer committed to accomplish during sprint. It is highly visible.
- If developers want to negotiate the scope of sprint than they need to talk with Product Owner reminding one thing that goal of sprint should not effect.
- Sprint backlog is plan by and for the developers.
- Sprint goal is commitment by the developers not sprint backlog.
- Sprint goal is created during sprint planning meeting and then added to sprint backlog.
- Sprint backlog also can be emerged during the sprint.

Items	Sprint Goal <i>Users access with ease the system using a valid email</i>		Done
	To Do	WIP	
<i>User can register on the system with email</i>	Register page Validate email Compare informed passwords Persiste data access on database Do URL routing Encrypt password Create register route Send email confirmation Authenticate user		
<i>Registered user can log into the system with email</i>	Create login and password fields Declare provider Store temporary data (remember me) Cryptograph password on database Authenticate user Create OAuth service Do URL routing Associate bidirectional data		
<i>As a user I want to recover my password so that I can get access to my account if I lose it</i>	Modify login page Generate default password Validate informed email Authenticate user Create notification service Send email notification Create screen to define new password		

Increment:

- It is sum of all product backlog items completed in sprint and value delivered in previous sprints.
- Multiple increments would be created in sprint but sum of all increments should be presented in review meeting.
- Increment supports empiricism at the end of sprint.
- It should always be in usable condition.
- There is no need to release increment at the end of each sprint. It releases when Product owner thinks it is valuable.
- Sprint is also not a gate for release. Release might be performed during a sprint.
- Increment born when it meets the Definition of Done (DOD).

Definition of Done:

- It is criteria or defined as what is needed for work to be considered as done. It promotes transparency and visibility.
- Everyone should have same understanding on what done is.
- As per scrum guide it is a formal description of the state of the increment when it meets the quality measures required for the product.
- When Backlog items meets the definition of done as increment was born and if does not meet than it moved back to product backlog.
- DOD ≠ Acceptance Criteria.
- DOD is same for all items while Acceptance criteria is different for each item.
- Each increment must have a single Definition of Done.
- Define by the organization as standard it not than Scrum Team will define it mutually.
- To define DOD is a continue effort and might evolve during the execution mean it is not fixed during sprint. There might some addition or subtractions from criteria list.

Technical Debt: (Not a part of Scrum guide but present in Scrum Glossary)

- It is implied cost of additional rework caused by choosing a poor solution now instead of using a better approach that would take longer.
- Consequences of poor software development practices will majorly face by non-technical stakeholders.
- Code with high technical debt is really hard to work with and cost of change is high.
- Scrum framework manage this through Definition of Done and product quality is topic during retrospective meeting and things could be managed in coming sprints.
- It is not necessarily bad always. (Might be used for learning purpose).

Scrum Events:

- There are five Scrum Events, namely the Sprint, Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective. In this article we will take a deep dive into each of the Scrum Events that happen within a Sprint. In Scrum, the Sprint is a fixed length event of maximum one month, which contains all other events.
- The Scrum Events are often also called “scrum ceremonies”.
- All the events in scrum are time boxed. Timeboxing means allotting a fixed unit of time for an activity. The unit of time is called a time box. The maximum length of a time box should be 15 minutes.

1. The Sprint:

- Sprint is a repeatable cycle during which work is completed and made ready for review. The duration of the Scrum Sprint depends on the size of the project and the team working on it.
- Its also called heart of Scrum Mean there is no lag between sprints, a new sprint starts immediately after previous sprint closed.
- A sprint is only cancelled when timeboxed expired or sprint goal become obsolete and only Product Owner have authority to cancel it.
- Sprint consist of four formal events and during a sprint a done and usable increment must be created.
- It must have fixed duration. Not maximum than 1 month.
- It also helps scrum team to control risks.
- During Sprint no changes should be made that that would affect sprint goal. Also, quality would not be compromised at any.
- Product Backlog would be refined when needed.
- Team should clarify and re-negotiate scope with product owner as more as learned.
- Following are invalid sprints as per scrum guide and these are not allowed as per guide:
 - Hardening (or Integration or Stabilization Sprint.
 - Sprint 0 and Design Sprint
 - Release sprint
- To monitor sprint progress scrum guide did not prescribed any projective practice for this. Team can use Burn Down or Up charts or WBS etc.
- In scrum every sprint must be delivered as per Definition of Done.

2. Sprint Planning:

- The Sprint Planning event takes place on the first day of the Sprint. Its purpose is to plan the work to be done during the Sprint and the whole Scrum Team is involved in this event.
- Scrum team participate in this event and they may also invite other stakeholders for advice.
- Timebox for 8 hrs. for one month sprint.

- In this event scrum team defines the goal (What to be build) of sprint.
- A Plan is built during this event to achieve goal that's called sprint backlog.
- Only developers can change sprint backlog and sprint backlog is only source of work for developers.
- Sprint goal is commitment from developers not sprint backlog.
- **How to run a Sprint Planning event:**
 - **Introduction**
The Scrum Master, who acts as facilitator, introduces the event by welcoming the participants, explaining the purpose of the meeting to the people invited by the Scrum Team and showing the agenda for the meeting. (5-10 min)
 - **Topic One: Why?**
The Product Owner takes the lead, provides an overview of the Product, and reminds everyone about the long-term aim for the Product as well as the current Product Goal. They come prepared and propose a draft Sprint Goal. For example, assuming that we are in a payment domain, a Sprint Goal could be: By the end of the Sprint, we will support card payments for all EU customers. (10-15 min)
 - **Topic Two: What?**
The Product Owner and the Developers quickly go through the Product Backlog. The Developers should have contributed to refining the Product Backlog, spending "just enough" time to do some final clarification and possible changes after the last Product Backlog Refinement session. (30 min) The Developers make a forecast of how much capacity they think they can spend on building the Increment during the Sprint and then select a list of PBIs they believe they can complete by the end of the Sprint. If those PBIs seem to be sufficient to achieve the Sprint Goal proposed by the PO, this part of Sprint Planning is complete. Otherwise, the PO and Developers collaborate to craft a new Sprint Goal which better aligns with what the Developers can deliver. In the example above they might come up with something like: By the end of the Sprint, we will support only debit card payments for all EU customers. (15-30 min)
 - **Topic Two: How?**
The Developers collectively create an actionable plan to deliver the Increment. They self-manage on how to best achieve that. It is not necessary to create a detailed plan for the Sprint. Instead, the Scrum Team should anticipate emergent opportunities and be prepared to adapt their plan as they learn. It is probably enough to have an idea of what each Developer will do in the coming 2 to 3 days. In this phase of the Sprint Planning the PO is not really necessary. However, it is good if they are available to answer questions or clarify customer needs. (30-45 min)
 - **Closing**
The Scrum Master thanks the participants for their contribution and officially closes the event. (5 min)

3. Daily Scrum:

- The Daily Scrum is the moment where the Developers step back for 15 minutes, analyze where they are in respect to the Sprint Goal, and collectively decide what is the most important thing each Developer has to do in the next 24 hours to get closer to the Sprint Goal.
- Scrum Master only ensure that event/ meeting took place and coach developers to keep it in 15 minutes timebox.
- Other stakeholder would not be allowed to attend meeting.
- Product Owner and Scrum Master also not invited in meeting unless they both actively working as developer and participate, they're as a developer.
- Developers can select whatever structure and techniques they want to run this event, as main focus is on inspection towards sprint goal.
- **When:** Every working day at the same time
Duration: A maximum of 15 minutes
Participants: The Daily Scrum is the only event that is self-managed by the Developers. The role of the SM is to teach and coach the Developers to run it autonomously.
Output: An adapted Sprint Backlog, if necessary
Structure: The Developers decide how they want to run it.

4. Sprint Review:

- The Sprint Review is an opportunity for the whole Scrum Team to stand in front of their users, stakeholders and customers and inspect and adapt the Product. It takes place at the end of the Sprint. The meeting should include an overview of the state of the product in terms of progress, budget and next steps. Usually, there is also a hands-on demonstration of the actual product. The users and stakeholders provide feedback, and then the developers incorporate relevant feedback into the Product Backlog.
- Sprint review is about “Transparency” and “Collaboration”.
- **When:** Last day of the Sprint
Duration: A maximum of four hours for a one-month Sprint. For shorter Sprints it is usually shorter: roughly two hours for a two-week Sprint.
Participants: The whole Scrum Team: (Product Owner, Developers, Scrum Master) as well as the customers, stakeholders and users. The event is usually facilitated by the Scrum Master.
Output: A revised Product Backlog and changes to the product or release strategy, if necessary

Structure: An inspection phase followed by an adaptation phase (working session).

- **How to Run a Sprint Review:**

- **Introduction:**

The SM, who acts as facilitator, introduces the event by welcoming the participants. They explain the purpose of the meeting to the participants which can include users, stakeholders and customers. Then they show the agenda for the meeting. (5-10 min)

- **Inspection Phase:**

The PO takes the lead and provides an overview of the state of the product in terms of progress, budget and next steps. They remind everyone of the Product Goal and describe the Sprint Goal the Developers were trying to achieve. (10 min)

The PO leaves the stage to the Developers to demonstrate the Increment. This is not a PowerPoint presentation of what the team has done, but a hands-on demonstration of the actual product. Usually, Developers will go through the scenarios described in the acceptance criteria of each PBI. Some teams even let users or customers try the product themselves, while the Developers and the Product Owner observe how they interact with the Increment. (30 min)
The Scrum Team collects feedback on the Increment from all invited participants. (15-30 min)

- **Adaptation phase:**

Users, stakeholders and customers might leave the meeting at this point, while the Scrum Team continues with a working session to incorporate relevant feedback into the Product Backlog. Should the feedback be related to something which does not impact the upcoming Sprint, they can simply take notes to address the feedback in one of the upcoming Product Backlog Refinement sessions. However, if the feedback potentially affects the next Sprint Goal, the Scrum Team will perform a quick Product Backlog Refinement session during the Sprint Review to get ready for the upcoming Sprint Planning. (30 min)

- **Closing:**

The Scrum Master thanks the participants for their contribution and officially closes the event. (5 min)

5. Sprint Retrospective:

- The Sprint Retrospective is the only event in Scrum that is exclusive to the Scrum Team. The intention is to create a safe space where everyone in the Scrum Team feels comfortable to openly share their observations and express their views and ideas. The purpose of the event is to inspect how the last Sprint went and plan ways to increase quality and effectiveness.
- It is held for making things better and continue improvement.
- **When:** Last day of the Sprint, right after the Sprint Review
Duration: A maximum of three hours for a one-month Sprint. For shorter Sprints it is usually shorter.
Participants: Only the Scrum Team: Product Owner (PO), Developers, Scrum Master (SM). The Scrum Master usually facilitates the event
Output: An improvement plan
Structure: A good structure is suggested in the book “Agile Retrospective – Making Good Teams Great” from Esther Derby and Diana Larsen, which comprises the following 5 steps:
 - Set the stage
 - Gather data
 - Generate insights
 - Decide what to do
 - Close the retrospective
- **How to run a Sprint Retrospective:**
 - Set the stage
 - The SM, who acts as facilitator, introduces the event by welcoming the participants and showing the agenda for the meeting. They work to create the right environment for focused and open communication. Maybe it was a tough Sprint, with conflicts and failures, and it is essential for the team to step back from the stress and consider different perspectives to find opportunities for improvement. (5 min)
 - Here is a simple yet effective activity you can do. Ask every individual to write on one sticky note the worst thing going on in their head at the moment and on another sticky note what would make them happy after the event. Then ask them to throw away the first sticky note and keep the second one.
 - Gather data
 - Now it is time to gather both objective (e.g. number of found bugs) and subjective data (e.g. how the different people felt throughout the Sprint) from the Sprint. (10-15 min)
 - Sometimes it is hard for people to recollect data after a few weeks. Consider asking everyone to report events when they happen (for example by adding sticky notes on a Miro board) with a short description and a time stamp. During the retrospective you can look at

the sticky notes and order them on a timeline. Then each individual visualizes their mood related to those events. Then you have both objective and subjective data in a single visual artifact.

- Generate insights
- Now it is time to look at data and try to find patterns. These will allow the team to identify one or two improvement areas. (15-30 min)
- Decide what to do
- Once the most important improvement area is identified, the team collaborates to define a few actionable items to incorporate in the next Sprint. Consider those items experiments. Once the team validates, they actually lead to a positive impact, they can incorporate them into the team's routines and practices. Using the SMART technique enables the definition of action items which are Specific, Measurable, Attainable, Relevant and Time-bound. (30-45 min)
- Close the retrospective
- The Scrum Master thanks the participants for their contribution, closes the event and encourages everyone to celebrate the Sprint. In an in-person setup, this might be a good moment for the PO to share a cake to thank everyone for their effort. (5 min)

What went well?	What did not go well?	How can we improve?
<i>User stories were better written</i>	<i>Everyone not making to Daily Scrum on time</i>	<i>Change meeting to 11h30</i>
<i>Product Owner very responsive during the sprint</i>	<i>Team members could be sitting closer to each other to improve communication</i>	<i>Rearrange workspace</i>
<i>Improved understanding of agile processes</i>	<i>All user stories closed on the last day of the sprint</i>	<i>Define WIP limits</i>
<i>Key stakeholders participated in the Sprint Review giving awesome feedback</i>	<i>It has been hard to change the code</i>	<i>Configure static code analysis for Continuous Integration</i>
	<i>Technology used to develop app not compatible with non-functional requirements</i>	

Scaled Scrum:

- Scaled Scrum' is any implementation of Scrum in which multiple Scrum Teams create one software system or product. The purpose of Scrum, independent of the scale at which it is used, is to create high-quality, releasable versions of product by the end of every Sprint.
- It is based on Scrum as defined in scrum guide but modifying Events, Roles and Artifacts that better suites scaled.
- All Scrum teams in scaled have one PO, one PB and one Product Goal.
- 3 – 9 teams should be in nexus scrum.
- At last, there should be one integrated increment.
- To better manage dependencies nexus, define integration team that compose of PO, one Scrum Master and few members of developers from other scrum teams. This team is responsible for defining Definition of Done and other collaboration related technical tasks.

←=====The End =====→