There was problem to delivery software with producttivity,on budge with real value, certainally doing the right thing

Waterfall model had this problem with 24 months may not be expected often does not meet buget ,cost and requirements

Necesscity was to be more agile give value in smaller increments

We need gather and to deliver in smaller increments with more value(depending on the priorities for the customer)We keep gathering not delivering , or delviering it when it looses its credibility, and its value

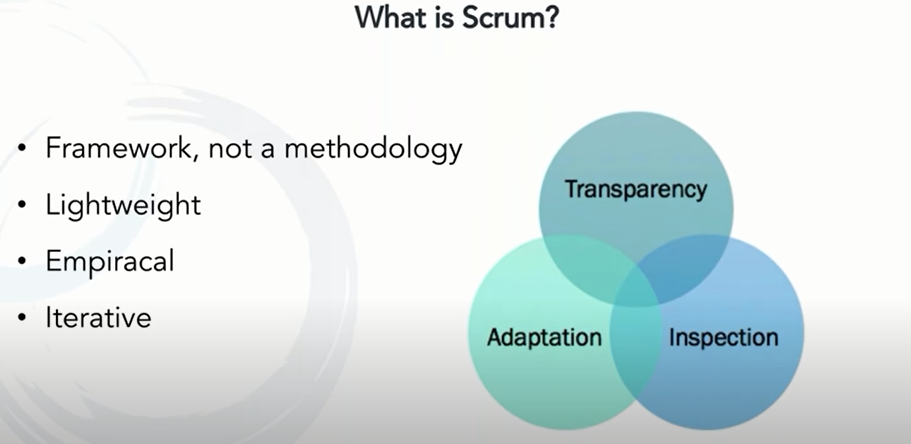
----------------------------------------------------------------------------------------------------------------------------------

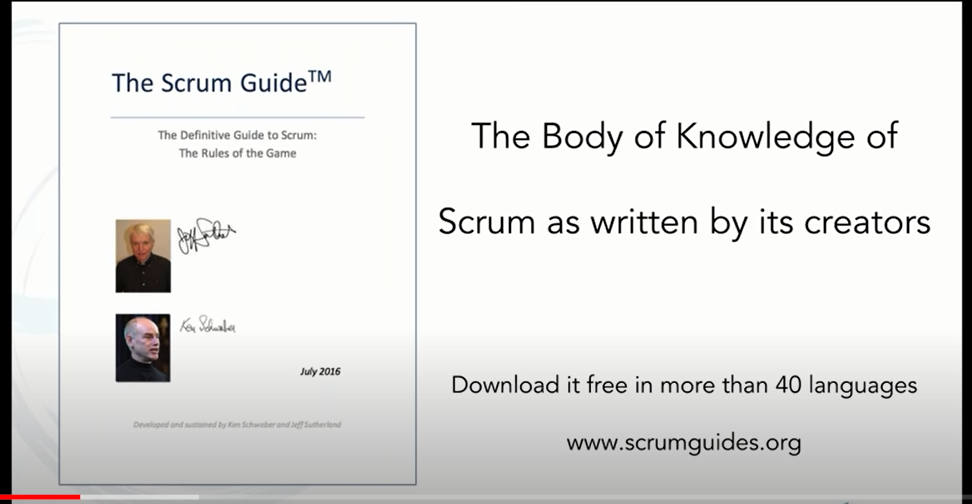
What is scrum: it is a framework : it is not methodology : it is not very descriptive and not very strict to be followed

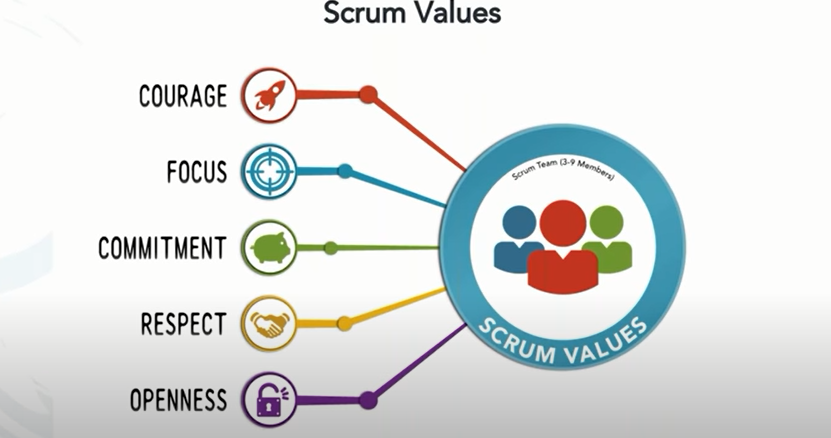
with sane methods followed by everyone in the organisation.

it is emphirical,adapt the framework work and learn. We work as a team and make the smaller increments with value

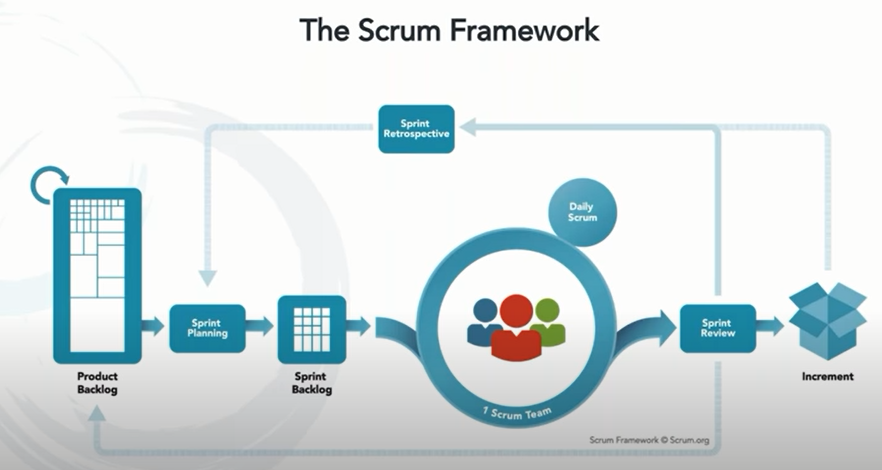
My team is very different and we do not mock others in the organisation, we need to deliver very fast with great value

working as adaptive and inspective team.





Openess is to accept the retrospective nature of the process in the scrum



The scrum framework above gives a glance of a sprint. Sprint is a duration of work. Scrum belives that it not just production, but every duration spent in developing should be made production ready to be delivered in small increments to the customer.

Sprint : generally is short duration : 1 to 4 weeks y

You can even have multiple delivery happening within a sprint if possible that’s what scrum belives in.

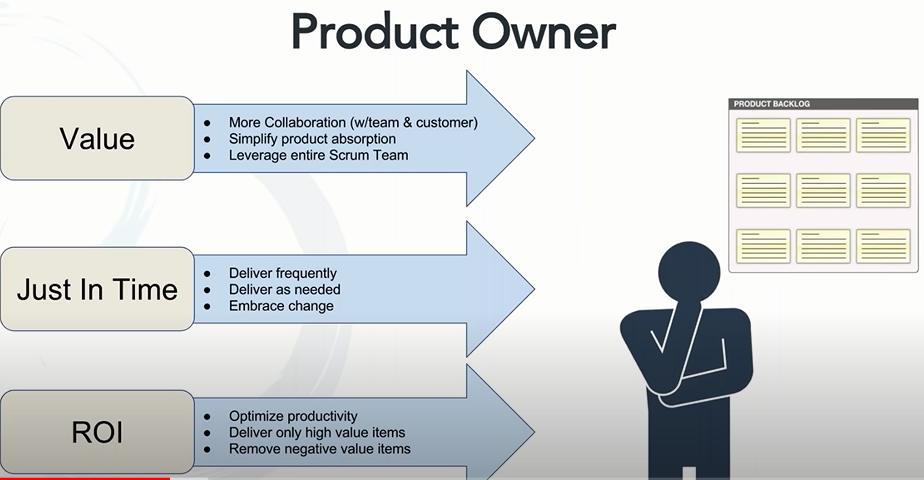


Three roles



His role and job is to support and help the team. He teaches the members how to follow the scrum rules. He is a coach for all members incl. their behaving with the team and he facilitate for the work to productive and make it delivery ready. Team to focus towards and work protecting the team as shield form other people in org making the team members do which is not part of their work;

Scrum master can lead more than one team . multiskilled and great leader and make the team work independently focosed on delivering in time.



The ROI **represents the benefits gained from the investment versus the costs that was expended**

Product owerner is the person who is focused on delivering the product to the stakeholder. Stakeholders can be the customers , or the senior managers within the organization, or any one within outside the organization.

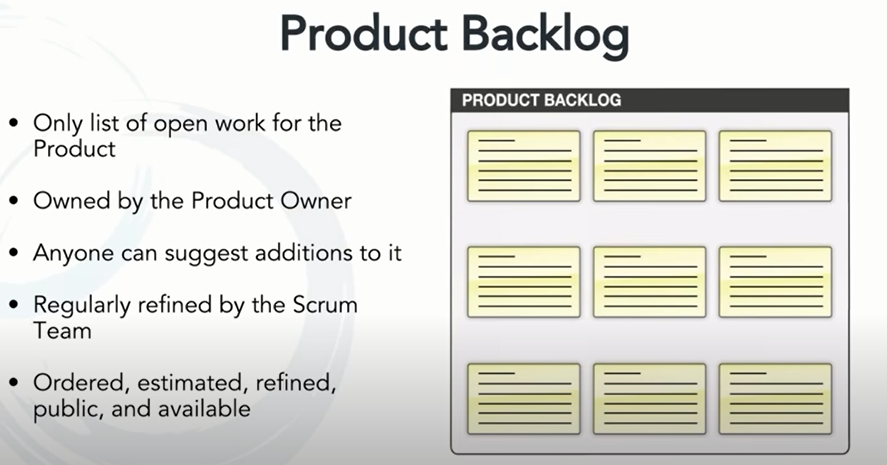
The have to bring the requirements and prepare the product backlogs by including the req that is needed and removing the non valuable req from the backlog.

They need not be too technical but they are empowered to decide on the functionality of the product whether these features have to be included or not. They have to decide on the fetures of the product because they are responsible for the product developed and communicate with stateholders reg it. They are people who get the feedbacks and they are accountable to build the right thing.They should be enabled to do that. Their job is to prioritize the work and bring out the right features to the product in time and delivered within the schedule. Great work!!!



They are responsible for delivering the product with value specified with the duration of the work ie sprint. PBI product backlog items.





The backlog of the items to be worked oout the open work to be done. it can be longer and growing. Sometimes become shorter as the per the deliverables moved to the customers.

But it is not fully defined in the beginning . we have some big items and some smaller items and also these items refined over time. As we work with it we add product its priority and value

Product owner is responsible for the product backlog. They work with project managers and business analysts. PBI define the product value delivered by the system.

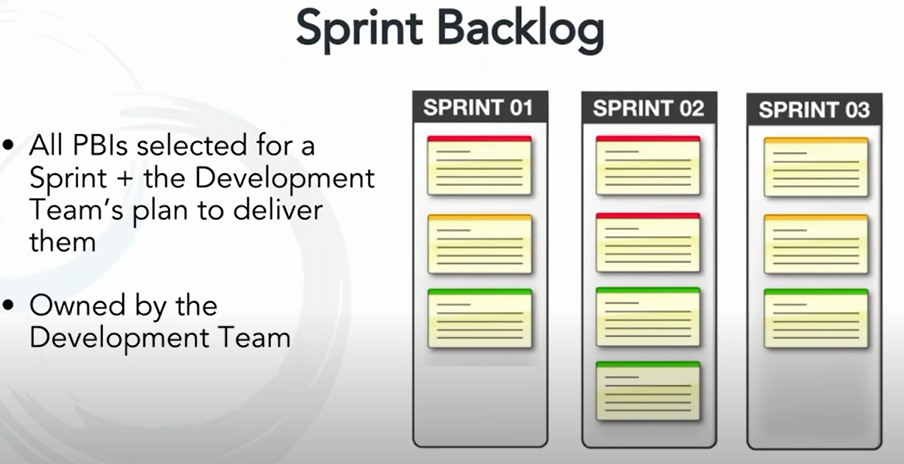
Proudct backlog items:

Identify them Like we need a mongodb , sqlserver db, ect. Queirees I have get done, the uI features to done, the classes to be included. Ect. Ect. With the estimated time . and also sometimes the remaining estimated time, with start date and end date

1. User Story Item description moduleworked at estimated time remaining time start end date

How can they be made public and easily accessable by all in the team.

Tools as axissoft helps to build the scrum as a frameworkl



Sprint backlog is Sprint work that is decided to be completed with a small duration of time. These items are got from the PB.

The developer teams now owns the sprint and is specified with work that they are going to do to add value to the product to be delivered.

You can drag and drop items form PB during the sprint plan so as to fit I th duration . As the scrum master can closely inspect it by getting the view of the module worked of by each member in developing team. That’s called the swinlanes what are their items , and

this gives way to conversations that need to be communicated in the next scrum standup meeting

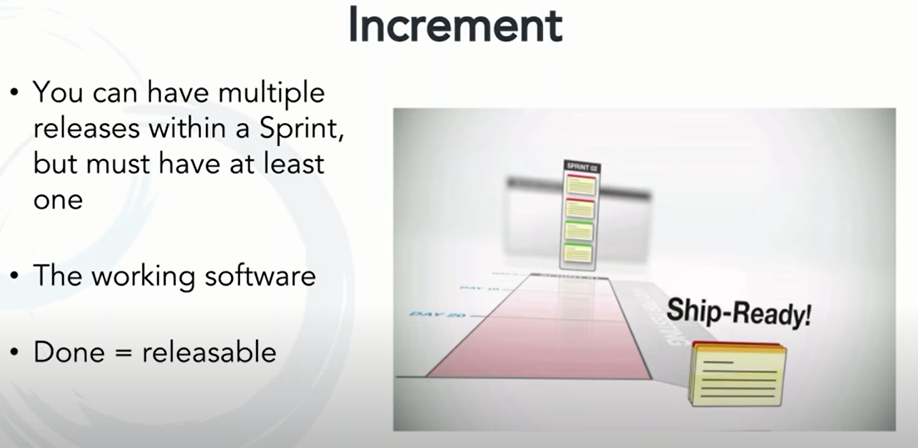
this gives way to see if someone is not given too much of work and someone with no work

this give way to know the work in progress In each swinlane of the member in team ie same person has three things in progress

How are you working out with three things, focussed on which one,

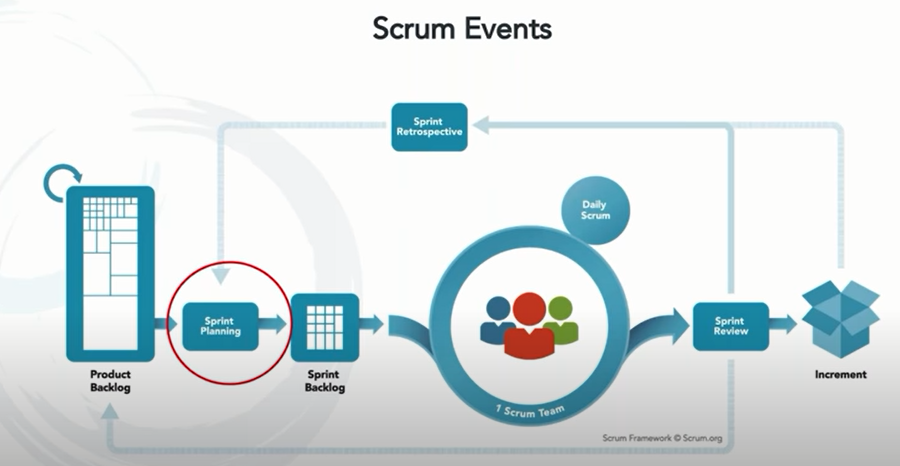
He might have two already working with him to be brought into the build by the completion of the one he is working and others sequenced one after other after this item is complted .

This can communicated t the clearly to the scrum master.



Incremennt is to deliver at one release of the product by increasing its capability . It is the real working model realeased to be expericed by the stakeholders, not a document of jpg image file which are not actual real.

We get to know the real feedback with this one increment of the real working model of the product.Exmaple like a homepage. They scroll, they can read text , something repeated or overlapped ect. They tend to work on it and quickly get accepting it and also giving the real feedback about the incremented product.

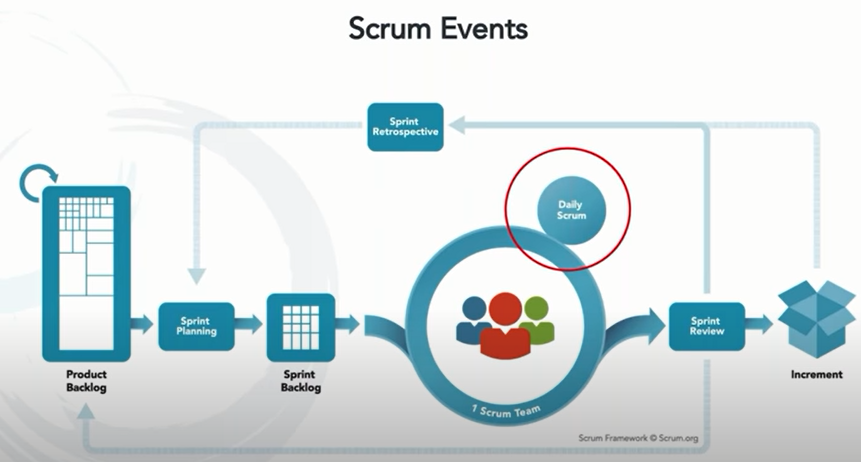


Sprint planning is done in the beginging of every sprint. The whole process above is a sprint as discussed early, once one sprint is delivrerd the next sprint takes off.

Sprint planning is a meeting is priorities the items in the PB given and add it to the sprint , Sometimes the PB item is to be split into smaller items because it s too a big and also sometimes we bring to fit in the sprint schedule time. That means it’s the work to be going on by the delveopers and scrums masters monitoring it.

It not just speed, ofcourse necessary but it basically focused to deliver value. Not garbage.

What are items to be fit in that sprint is the major goal.



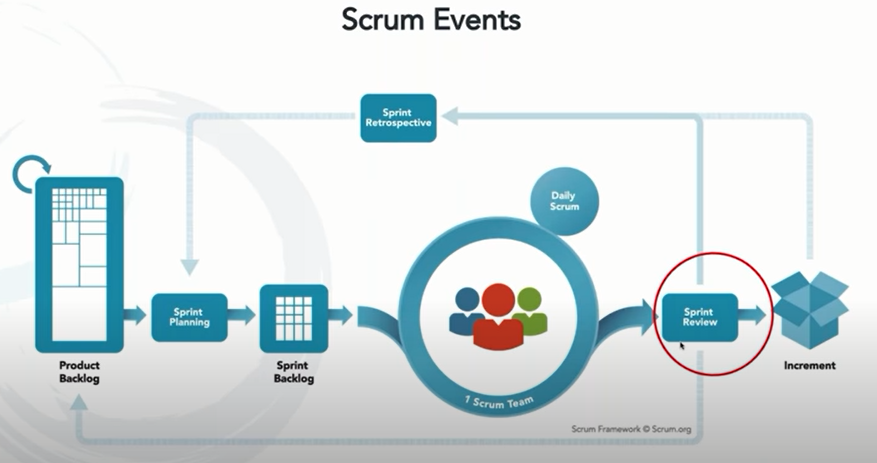
Conducted by the scrum master as the same time same place every day . Inception and transperncy is the main goals , Not a meeting, but scrum meeting not too long not more than 15mits.

To find the flow of the work , if any blockages to be resolved. At least list them so they can be even taclked individually. This problem inception and adaption.

Its not just what you did yesterday, what you are doing today , what is on the way. These are optional questions. What is need is to understand the changes to be brought to the product and find if any blockages present.

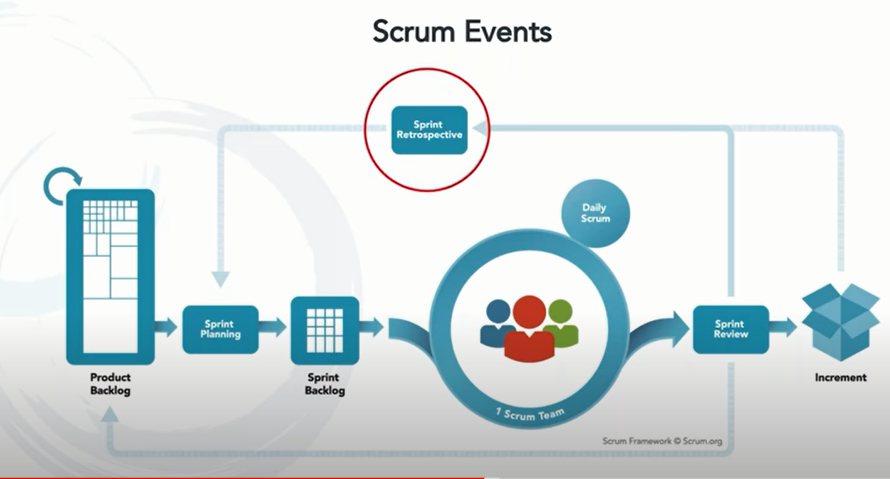
There are matrix maintained to know the decrement the spring. backlog : ie the duration with completion of each task in the sprint .

We can continue the discussion offline not in the meeting if any blockages found so that it can be fixed quickly and the sprint moves in the planned schedule.



Sprint review is the operation which happens for the incremente feature of the product . Increments of the product definetly happens whether review is conducted or not.

IT is the review inclusive of the developers and stakeholders feedback , so the we are sure we delivered what is the required for customers within the sprint .



Retrospective is an opportunity to make changes how we worked as team, what aspects could have been better in the sprint , what could have been handled better. Looking back

1. What is PBI
2. Who will place the items from PBI into the sprint backlog during sprint backlog by developer team.
3. Does is scrum meeting to be done on the everyday morning
4. 15mit time box? Extend the meeting?

Any comes issues only the member is to be involved not the whole team so we can go to the rabbit hole and quickly incorporate the changes. In the offline which is post scrum discussion.

1. How to bring changes who were following the waterfall model: not to be rigid and be transparent with the deliverables.
2. How to change from half completed task item to a prioritesed item? Based on the project owners decision. He is accountable for prioritising the value of the product
3. Sprint backlog is designed by the scrum master an developing team
4. Any time a sprint can be cancelled by the product owner depending on the priority

What Is a Burn Up Chart?



A burn up chart is a visual diagram commonly used on Agile projects to help measure progress. Agile burn up charts allow [project](https://www.wrike.com/project-management-guide/faq/what-is-a-project-in-project-management/) managers and teams to quickly see how their workload is progressing and whether project completion is on schedule.

But what's the meaning of this Agile buzzword? In this article, we’ll discuss the purpose and benefits of a burn up chart, how to read and create one, and the difference between a burn up and [burn down chart](https://www.wrike.com/blog/what-is-burndown-chart/).

## **Introducing the Agile burn up chart**

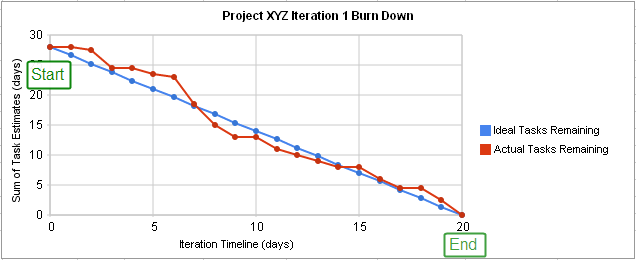
A burn up chart is [a graph](https://www.wrike.com/blog/guide-to-setting-up-your-agile-workflow-in-wrike/) that shows project progress over time. There are two main lines shown on the chart: one for the total project work planned, and the other for tracking the work completed to date.

By comparing the work your team has accomplished so far with the total amount of work planned, you can understand how efficiently they’re working and better estimate how long it will take to complete the work remaining.

### **Burn up vs. burn down chart — What is the difference?**

A burn up chart and [a burn down chart](https://www.wrike.com/blog/what-is-burndown-chart/) are both popular [project management](https://www.wrike.com/teams/marketing-project-management/) tools for visually tracking work completed over time. But, there are key differences between the two charts.

First, a burn down chart starts with the total amount of work and then graphs the amount remaining over time. As time progresses, the amount of work to be done lessens, and the line decreases toward the right.

[](https://web-static.wrike.com/blog/content/uploads/2020/10/What-Is-a-Burn-Up-Chart-2.png?av=37eed518815539f012e34968f847d10e)(Source: [*Wikipedia*](https://en.wikipedia.org/wiki/Burn_down_chart))

A burn up chart tracks the amount of work to be completed as one straight line across the top of the graph (unless [a scope change](https://www.wrike.com/project-management-guide/faq/what-is-scope-creep-in-project-management/) occurs). A second line is then used to track work completed, starting at zero and increasing to the right as more tasks are finished.

Because of this difference, a burn down chart is great at emphasizing what’s left to be completed, but a burn up chart is better at illustrating the work that’s been accomplished.

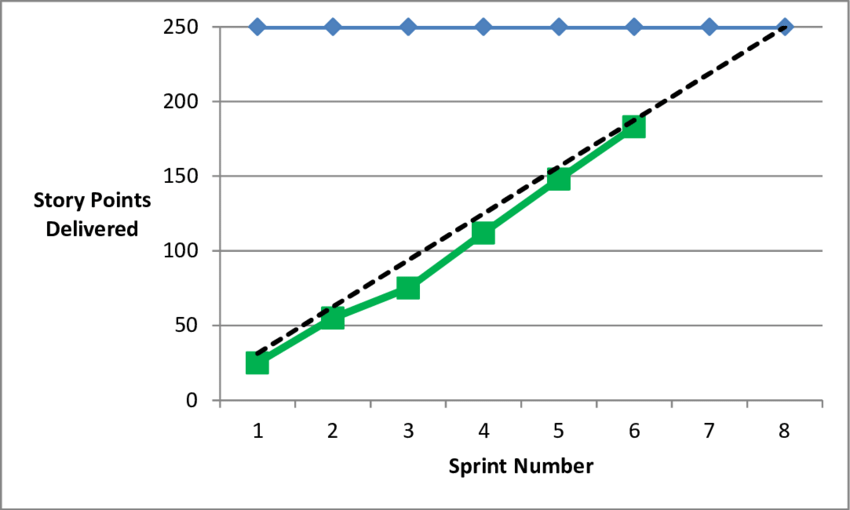
For example, imagine [your Agile team](https://www.wrike.com/project-management-guide/faq/what-is-an-agile-team/) had 50 [story points](https://www.scrum.org/resources/blog/why-do-we-use-story-points-estimating) (SPs) left to complete. (A story point is a unit of measurement that [Scrum](https://www.wrike.com/project-management-guide/faq/what-is-scrum-in-project-management/) teams assign to tasks or stories to indicate how much effort they think will be required.) Then, a change occurred, which added 10 more SPs. Let’s also assume that, during your current sprint, your team managed to get through 15 SPs.

On a burn down chart, before your sprint, it would show 50 SPs remaining, and after your sprint, it would show 45 SPs remaining (50 + 10 - 15). The amount remaining is correct, but if someone wasn’t aware of the [scope](https://www.wrike.com/project-management-guide/faq/what-is-scope-in-project-management/) change, they could look at this graph and assume you had a really unproductive sprint and your team only completed 5 SPs instead of the 15 they actually did.

With a burn up chart, you’d easily see the 10 SP increase in the line illustrating overall work to be completed, and see the 15 SP increase in the line tracking work accomplished.

### **How do you read a burn up chart?**

Here’s an example of a burn up chart:

[](https://web-static.wrike.com/blog/content/uploads/2020/10/What-Is-a-Burn-Up-Chart-3.png?av=30da980824a49989309746089cb4dd88)(Source: [*researchgate.net*](https://www.researchgate.net/figure/Sample-Release-Burn-Up-Chart_fig7_266049888))

Along the y-axis is the number of story points delivered or the amount of work to be completed. Across the x-axis is the sprint number or time period.

The blue line across the top shows the total planned number of story points that will be delivered by the end of the project.

The dotted black line illustrates the planned completion of story points. In this example, that’s roughly 31 SPs per sprint (250/8).

The green line is reporting the actual number of story points completed. You can see that for sprints 1 and 2, actual and planned were essentially the same. Then, during sprint 3, less work was delivered than expected, but the team caught back up by sprint 5.

The green line only goes to sprint 6, indicating that the final two sprints haven’t been completed yet.

Some key things to watch out for when reading a burn up chart are:

* If the blue line changes, you know work has been added or removed from the project.
* If the black line changes, a change has been made to the baseline plan for how much work will be completed in each sprint.
* If the green line is beneath the black line, you’re behind schedule.
* If the green line is above the black line, you’re ahead of schedule.

## **What are the benefits of a burn up chart?**

A sprint burn up chart helps your [Agile](https://www.wrike.com/project-management-guide/agile-methodology-basics/) team monitor how efficient they are from one sprint to the next. It also helps everyone keep track of how much work is left and whether you can expect to complete the project on time based on your progress rate so far.

If your team was completing 20 SPs per sprint and now suddenly they only completed 10 SPs, your burn up chart will highlight that change. This makes it easier for you to spot differences and investigate them to see if a bottleneck or issue is slowing your team.

During a project, you can extend the actual work completed line to help predict when the project will end. This can help you estimate project length. It also allows you to be proactive about monitoring progress and discussing what you can do if it looks like the project isn’t going to finish on schedule.

Plus, a burn up chart monitors your overall scope and how that impacts your project length. [What does scope mean in project management](https://www.wrike.com/professional-services-guide/project-definition/)? It's essentially all of the requirements and objectives that a project needs to be a success. Your burn up chart is an easy way to show your client and other stakeholders how much longer the project should take if they want additional work added.

### **How do you create a burn up chart?**

First, draw your two axes with your time periods or sprint numbers across the bottom (x-axis) and your number of story points or planned work up the left side (y-axis). Next, find the number of total work planned and draw a line straight across the top.

Now it’s time to [plot your trend line](https://www.chilimath.com/lessons/introductory-algebra/plotting-points-graph-xy-plane/) for planned work over time. The easiest way to do this is to plot your start point (0,0) and your end point (where the last planned sprint intersects with the total volume of planned work). Then connect the two points with a straight line.

As work progresses, you can plot the actual work level completed at the end of each sprint or reporting period. Connecting these dots will create your line showing actual work.

While you can create an Agile burn up chart using a pencil, ruler, and piece of paper, it makes it challenging to quickly edit it and share it with your team.

With [Wrike Analyze](https://www.wrike.com/add-on-wrike-analyze/), you can create a sprint board that highlights all of the essential project information you need to see — including burn up and burn down charts, percentage of work completed, the total number of hours spent, and more.

# Product Backlog Template - How to Build and Prioritize Agile Product Backlog?

## **What is an Agile Product Backlog?**

**Product backlog templates are standardized repositories for tracking PBIs through prioritization and inclusion in sprints, usually retained in standard Excel format. In Agile methodology, the product owner gathers required tasks or requirements as Product Backlog Items (PBIs), also referred to as “stories.” By retaining an ever-changing but all-inclusive list in a central repository, the prioritization, and visibility of PBIs facilitate planning. It allows for pulling product backlog items into Sprints for development and implementation.**

|  |  |
| --- | --- |
| |  | | --- | | [Download This Template Now!](https://shop.techno-pm.com/pages/agile-templates) | |

|  |
| --- |
| [agile product backlog template, product backlog template, product backlog example](https://3.bp.blogspot.com/-6uAzD-FhMzE/XBISSvTikrI/AAAAAAAAByQ/SmUZToXdsn8RkS9NB8A52b3WzyHRORJ7QCLcBGAs/s1600/product-backlog-with-user-stories-agile-product-backlog-template.png) |
| Agile Product Backlog |

## **Product Backlog with User Stories –**

## **What is a user story?**

**User stories** are short, simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system. They typically follow a simple template:

As a < type of user >, I want < some goal > so that < some reason >.

User stories are often written on index cards or sticky notes, stored in a shoe box, and arranged on walls or tables to facilitate planning and discussion.

## **How is detail added to user stories?**

Detail can be added to user stories in two ways:

* By splitting a user story into multiple, smaller user stories.
* By adding “conditions of satisfaction.”

As a vice president of marketing, I want to select a holiday season to be used when reviewing the performance of past advertising campaigns so that I can identify profitable ones.

Detail could be added to that user story example by adding the following conditions of satisfaction:

* Make sure it works with major retail holidays: Christmas, Easter, President’s Day, Mother’s Day, Father’s Day, Labor Day, New Year’s Day.
* Support holidays that span two calendar years (none span three).
* Holiday seasons can be set from one holiday to the next (such as Thanksgiving to Christmas).
* Holiday seasons can be set to be a number of days prior to the holiday.

## **Who writes user stories?**

Anyone can write user stories. It's the product owner's responsibility to make sure a product backlog of agile user stories exists, but that doesn’t mean that the product owner is the one who writes them. Over the course of a good agile project, you should expect to have user story examples written by each team member.

Also, note that who writes a user story is far less important than who is involved in the discussions of it.

## **When are user stories written?**

User stories are written throughout the agile project. Usually a story-writing workshop is held near the start of the agile project. Everyone on the team participates with the goal of creating a product backlog that fully describes the functionality to be added over the course of the project or a three- to six-month release cycle within it.

Some of these agile user stories will undoubtedly be epics. Epics will later be decomposed into smaller stories that fit more readily into a single iteration. Additionally, new stories can be written and added to the product backlog at any time and by anyone.

## **Do user stories replace a requirements document?**

Agile projects, especially Scrum ones, use a product backlog, which is a prioritized list of the functionality to be developed in a product or service. Although product backlog items can be whatever the team desires, user stories have emerged as the best and most popular form of product backlog items.

While a product backlog can be thought of as a replacement for the requirements document of a traditional project, it is important to remember that the written part of an agile user story (“As a user, I want …”) is incomplete until the discussions about that story occur.

It’s often best to think of the written part as a pointer to the real requirement. User stories could point to a diagram depicting a workflow, a spreadsheet showing how to perform a calculation, or any other artifact the product owner or team desires.

Product backlog contined ….

Product backlog information is continuously being updated with new items and status information of previous items. This varies considerably from a sprint backlog, which is static in nature once PBIs have been designated as included in the sprint. Product owners and scrum masters can benefit greatly from standardization and effective use of the product backlog template.

In an Agile approach - including Scrum - the Product Backlog can be considered the backbone of project delivery. Simply put, it represents an ordered list of all the things (features, requirements, bugs) that need to be done in the project.

* **Entries in the Product Backlog add value**: they do not exist as mere reminders but instead are perceived as valuable for the user journey.
* **Entries in the Product Backlog are sorted by their priority**: they are not randomly organized but are instead prioritized by the Product Owner.
* **Entries in the Product Backlog are estimated**: they are not just a call to action but are actually the primary source for additional knowledge about a certain task, such as its estimated duration for completion.

The Product Backlog specifies what the project's product is about, that is, its features and requirements. Yet, it is not your traditional requirements specifications document either, since it is not built as a narrative but as a list of atomized items. The Backlog doesn’t follow any specific format and can be produced in an Excel spreadsheet, document, or simply sticky notes on a wall. In a nutshell, the Product Backlog is the ultimate list to go to when wanting to know more about a certain requirement, such as their priority or status.

## **How to Build and Prioritize a Product Backlog?**

The best way to build a Product Backlog is by starting with the 2-R approach: (1) roadmap and (2) requirements, which form the foundation for the Product Backlog. The Roadmap, also called vision, represents what the product aims to achieve for its customer. The requirements are an expression of the needs and wants of the customer. Both are combined into the Product Backlog, with displays the roadmap by breaking it down into what is called epics which, in turn, are composed of requirements and user stories, that is, a description of functionality from the perspective of the user, such as “As a frequent traveler, I would like to get my flight details added to my Outlook calendar, so I could easily plan ahead.”

### **The following conditions should be met when building a Product Backlog:**

* **100% rule**: only work required for the delivery of the product should be part of the Product Backlog; hence: If a task bears no value to the delivery of the product, it should be removed from the Product Backlog. If a task is later found to be necessary for the product's delivery, it should be added to the Product Backlog.
* **The Product Backlog is a living document**: a Product Backlog is never completed, but it is refined (Product Backlog Grooming) by the Product Owner over time.

The Product Backlog is owned, prioritized, and maintained by the Product Owner, who starts by listing all the requirements he can think of at the beginning of the project. However, as mentioned, the Product Backlog is a living document, and additional requirements can be added as the team progresses into the next iteration of work.  The composition of the Product Backlog varies, but, as a minimum, it should contain a unique identifier for each item, the item name, their priority, and status. Once drafted, the Product Owner should prioritize each of the Product Backlog items based on their value to the user experience.

Different prioritization techniques exist; however, one of the simplest and most popular is MoSCoW, where items are classed as:

M – Must do

S – Should do

C – Could do

W – Won’t have this time

|  |
| --- |
| Once the items are prioritized, the Product Owner takes the Product Backlog to the [Sprint Planning Meeting](https://www.techno-pm.com/2016/08/sprint-planning-and-tracking-excel-template.html), where the backlog is presented. The team determines which items will be completed in the upcoming sprint. |

|  |
| --- |
|  |
|  |

## **Backlog Grooming**

At the end of each sprint – and in preparation for the next [Sprint Planning Meeting](https://www.techno-pm.com/2016/08/sprint-planning-and-tracking-excel-template.html) – it’s also usual to do the Product Backlog Grooming. The Backlog Grooming can be a continuous activity, or you can set up a meeting just for it (who doesn’t love meetings?), and its purpose is to refine and make improvements to the Product Backlog. Grooming, one may say!

The grooming of the Backlog is a responsibility of the Product Owner where, in conjunction with some or all of the team, the Product Owner reviews the items in the Backlog to ensure that all items are still relevant, their prioritization is still right, and that the items first in the backlog (hence, with higher priority) are ready for delivery.

A Grooming meeting should also allow for any questions by the team to be answered by the Product Owner, ensuring that everyone has a clear understanding of what is involved and expected for each Backlog item. To save time and unnecessary explanations at the Sprint Planning Meeting, the Grooming session should address whatever is necessary for the upcoming sprint, which can include:

* Clarifying requirements
* Removing and/or adding user stories
* Re-evaluate priorities
* Re-evaluate estimates

At the end of the day, the Product Backlog's Grooming should allow for a well-groomed backlog: neat, clean, and clear to all.

## **Advantages of using a Product Backlog**

Product backlog information forms the basis of pulling PBIs into sprints for development and delivery through:

* Providing a central repository of items/tasks identified by the product owner and stakeholders
* At-a-glance view of prioritized items for consideration in sprint planning
* A quick reference of relative task difficulty, sprint preparedness, and status
* Story points contribute to establishing a reasonable number of tasks that may be accomplished within any sprint execution scope.
* Product owners can utilize this information to make accurate decisions on PBIs to be included in sprints based on priority and level of resource requirements.
* PBIs may be submitted by any stakeholder and added to the product backlog, but the product owner makes the decisions on product backlog priorities.

## **What does a Product Backlog contain?**

Product Backlog Template content includes high-level information that can readily be utilized by the product owner and Sprint leaders to accurately evaluate, monitor backlog items and sprint planning:

* **Task name/description** – This is a brief identifying description of the task to be performed by the backlog item.
* **Story**– This is a Y/N indicator to identify a clear story behind the item, stating the item’s purpose and benefits.
* **Dev Ready** – Is the item ready for pulling into a sprint action? There are varying views of what “ready” means, but the values of Y/N tell the product owner that there is sufficient information for the PBI to be pulled forward into a sprint.
* **Priority** – the product owner establishes the priority of each PBI. This is based on a combination of business value, the cost to implement (or cost savings), and story points. Values for priority are Low, Medium, and High.
* **Status**– Initially, a story will be in a “Not Started” status. Once it has begun work as part of an active sprint, it will move to “In Progress” until such time that the task is delivered, making the status “Complete.”
* **Story Points** – Story points are unique to Agile estimating techniques, taking traditional estimating in increments of hours or weeks. Story points are assigned as factors of difficulty for development, either due to complexity or unknown factors required to complete the task. Scrum teams reach a consensus on the story points to be assigned to a PBI without a true relationship to estimates of hours. This value presents a realistic gauge of complexity as related to other tasks on the product backlog. Values for story points are generally set as numerical values such as 1,2,4,8,16, indicating microscopic comparisons through extensive efforts required.
* **Sprint**– Once an item has been combined into a sprint exercise, this entry tracks the sprint associated with the task. This column may include values of Y/N, or if sprint numbers are utilized, it may contain the actual sprint project designation.
* **Release number and/or date** can be utilized to record when the sprint results were actually implemented.
* **Acceptance criteria** – Where the product owner or [designated stakeholders](https://www.techno-pm.com/2017/03/stakeholder-management-plan-template-ppt.html) have specific requirements that must be met before acceptance, this column can be utilized to provide such details.

PBIs form a key part of all the sprint activities, including but not limited to [Sprint Review Meeting,](https://www.techno-pm.com/2017/01/sprint-retrospective-meeting-template.html) [Sprint Capacity Planning](https://www.techno-pm.com/2016/11/sprint-capacity-planning-excel-template.html), and [Agile Reporting](https://www.techno-pm.com/2016/11/agile-project-management-dashboard.html).