

SCRUM MANUAL



**Business
Explained**



“

**Scrum is like your
mother-in-law, it points
out ALL your faults.**

”

Ken Schwaber



**Business
Explained**

ALL RIGHTS RESERVED.

No one is permitted to reproduce or transmit any part of this book through any means or form, be it electronic or mechanical. No one also has the right to store the information herein in a retrieval system, neither do they have the right to photocopy, record copies, scan parts of this document, etc., without the proper written permission of the publisher or author.

Copyright © Business Explained (2023)
www.business-explained.com

Disclaimer

All the information in this book is to be used for informational and educational purposes only. The author will not, in any way, account for any results that stem from the use of the contents herein. While conscious and creative attempts have been made to ensure that all information provided herein is as accurate and useful as possible, the author is not legally bound to be responsible for any damage caused by the accuracy as well as the use/misuse of this information.

INTRODUCTION	5
THE SCRUM FRAMEWORK	6
SCRUM ROLES: PRODUCT OWNER, SCRUM MASTER, AND DEVELOPMENT TEAM	7
THE PRODUCT BACKLOG	9
THE DAILY SCRUM (STAND-UP) MEETING	10
SPRINT PLANNING	11
THE SPRINT REVIEW	13
THE SPRINT RETROSPECTIVE	14
WHO SHOULD RUN SPRINT RETROSPECTIVE MEETINGS?	15
WHAT HAPPENS IN A SPRINT RETROSPECTIVE MEETING?	16
THE BENEFITS OF SPRINT RETROSPECTIVE MEETINGS	18
BURNDOWN CHARTS AND VELOCITY TRACKING	20
USER STORIES AND ACCEPTANCE CRITERIA	22
ESTIMATING AND PLANNING POKER	23
WORKING WITH DEPENDENCIES	24
MANAGING AND PRIORITIZING THE PRODUCT BACKLOG	25
PRODUCT BACKLOG	26
BEST PRACTICES FOR EFFECTIVE SCRUM IMPLEMENTATION	27
COMMON SCRUM ANTI-PATTERNS AND HOW TO AVOID THEM	29
USING AGILE TOOLS AND SOFTWARE IN SCRUM	30
COLLABORATING WITH STAKEHOLDERS AND EXTERNAL TEAMS	31
MANAGING AND TRACKING PROGRESS WITH SCRUM BOARDS	33
CLOSING THE PROJECT: LESSONS LEARNED AND NEXT STEPS	35
TIPS AND TRICKS FOR RUNNING SUCCESSFUL SPRINTS	37
CONCLUSION: THE FUTURE OF SCRUM IN PROJECT MANAGEMENT	39
REFERENCES	40

INTRODUCTION

Which steps can you take to ensure that your company provides the greatest possible value to all stakeholders? If you need a means of managing projects quickly and effectively, you might want to look into Agile Scrum. But what exactly is Scrum?

Scrum is just an Agile framework for managing projects with a set of guiding principles and practices designed to assist teams in better organize and control their work. Scrum is a training method for teams that aims to promote learning by doing, self-organization while working on a problem, and to reflect on successes and failures to improve performance.

Although software development teams make up the majority of Scrum users, the principles and lessons it teaches may be applied to any type of workgroup. This is why Scrum has become so well-liked. Scrum is for organizing and managing a team's work comprising meetings, artifacts, and responsibilities.

Scrum team members often do their tasks in iterative, short bursts of time called sprints, with ample built-in review and reflection time. Scrum teams are defined by a strong commitment to continual improvement in their method and the quality of the product they deliver.

This concise guide provides a high-level overview of the roles, artifacts, and events that make up the sprint cycle, as well as a brief overview of the basics of Scrum.

Incase you want to find out more about Scrum, you have found the appropriate resource. This book will guide the many roles, objects, and events that make up a Scrum sprint.

THE SCRUM FRAMEWORK

When implementing the Scrum process, it is helpful to have a set of rules and procedures to follow, which can be found in the Scrum framework. The framework consists of three parties (the Product Owner, the Development Team, and the Scrum Master), five meetings (the Sprint Planning, the Daily Scrum, the Sprint Review, the Sprint Retrospective, and the Backlog Refinement), and three deliverables (Product Backlog, Sprint Backlog, and Increment).

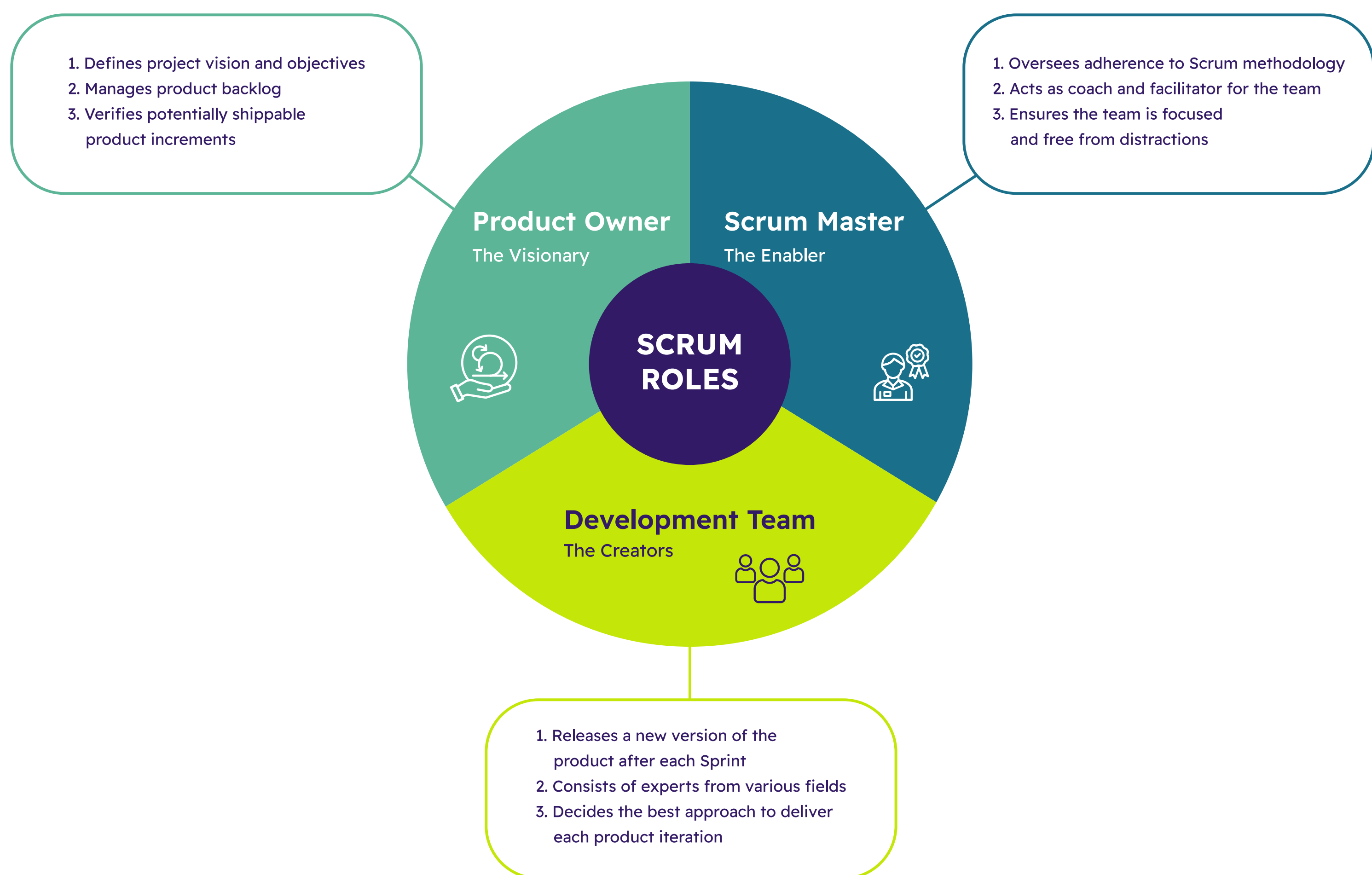
Scrum is a mature form of Agile Project Management. Scrum is an agile software development framework with strict rules for how things should be done and who should do it. It's a malleable approach that gives credit for using the Agile principles in a setting that's acceptable to everyone working on the product.

Sprints, the time frames across which Scrum is implemented and during which feedback and reflection can be provided, typically last between two and four weeks. Each Sprint is self-contained; it yields an entire deliverable, a variant of the ultimate product that must be readily available for client delivery upon demand with as little hassle as feasible. Project planning begins with a list of goals and prerequisites. The project's client sets the priorities for these goals by weighing the value and cost of each option.

Scrum is a paradigm for managing product development that emphasizes exposing and celebrating wins early and often. Scrum teams use the scrum framework to foster openness, inspection, and adaptability as they take a product from concept to reality, working from a moving list of the most useful things to do. Scrum's ultimate goal is to improve team collaboration so that clients are more than satisfied.

SCRUM ROLES: PRODUCT OWNER, SCRUM MASTER, AND DEVELOPMENT TEAM

The Scrum framework includes three key roles: Product Owner, Scrum Master, and Development Team.



Product Owner

The Product Owner's job is to share the project's vision and objectives with the team and manage the product backlog. The Product Owner is the group's official spokesperson and the person ultimately responsible for the product's success. They determine which issues in the product backlog are most pressing and ensure the Development Team works on those. Each Sprint will actually result in a potentially shippable product increment, which the Product Owner must verify

Scrum Master

The Scrum Master's job is to oversee the team's adherence to the Scrum methodology and ensure that it runs well.

The Scrum Master's role is that of coach and facilitator, ensuring that the team follows the Scrum process and works together efficiently. The Scrum Master's responsibility is to ensure that the team is free from distractions and able to focus on getting their work done.

Development Team

Following each Sprint, the Development Team must release a new version of the product. The Development Team consists of experts from many fields who have come together to create the final product. The team is autonomous and must decide how to best deliver each product iteration. The Development Team must also complete the tasks listed in the Sprint backlog and report their progress during the Daily Scrum.

All of the positions depend on one another and must work together for the project to succeed. Each Scrum position relies on the others for its success, as they were all created to function together.

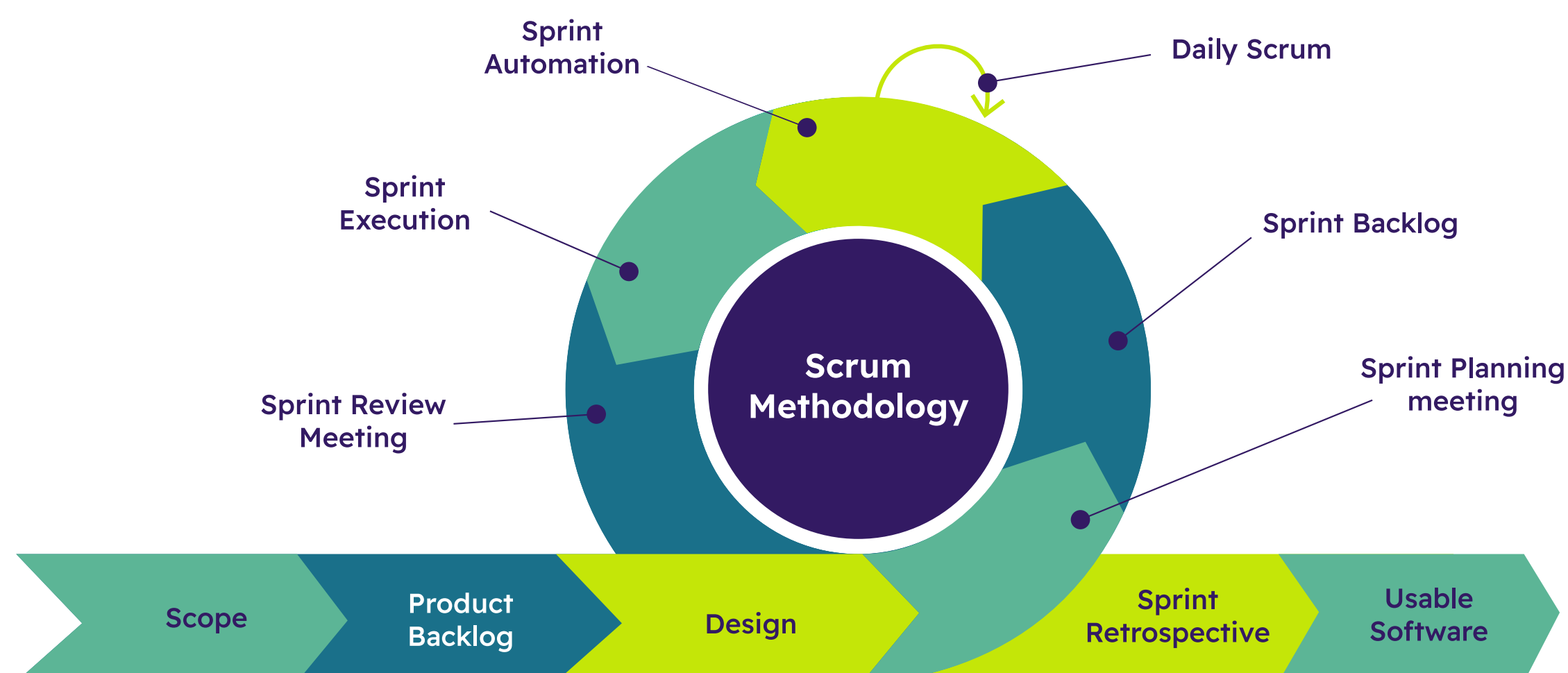
THE PRODUCT BACKLOG

Among the three artifacts that make up the Scrum framework is the Product Backlog, which is a prioritized collection of the product's features and needs. The Product Owner is responsible for keeping track of this list, which is an alphabetized compilation of all features and functionality that are currently anticipated to be included in the final product. The Product Backlog will be revised and updated as necessary throughout a project.

Each item in the Product Backlog can be turned into a user story, which is a concise explanation of a feature or a set of requirements from the customer's point of view. The standard formula for such user stories is "As a [user], I desire [feature], so that [benefit]." Items on the Product Backlog can be viewed as either technical needs or bugs that need to be fixed.

The Product Backlog is a prioritized list, with the most critical items at the top and the less important items at the bottom. The Product Owner determines the priorities; however, the Development Team is welcome to offer feedback. Priorities are determined by weighing each feature's benefits against the time and resources needed to execute them.

During each Sprint, the Development Team will use the Product Backlog as a roadmap for their work. The Development Team prioritizes the items on the Product Backlog to determine which ones to include in the Sprint backlog, the list of tasks that the team agrees to finish in the timeframe of the Sprint.



inapps.net/everything-you-need-to-know-about-scrum-methodology-in-2022/

THE DAILY SCRUM (STAND-UP) MEETING

As one of Scrum's five events, the Daily Scrum (or Daily Stand-up) is a meeting amongst team members to discuss the day's progress. The daily stand-up is a brief, time-limited meeting that takes place at the same time and location each day to check in on the progress of the team's work and make sure that nothing is standing in the way of their getting it done.

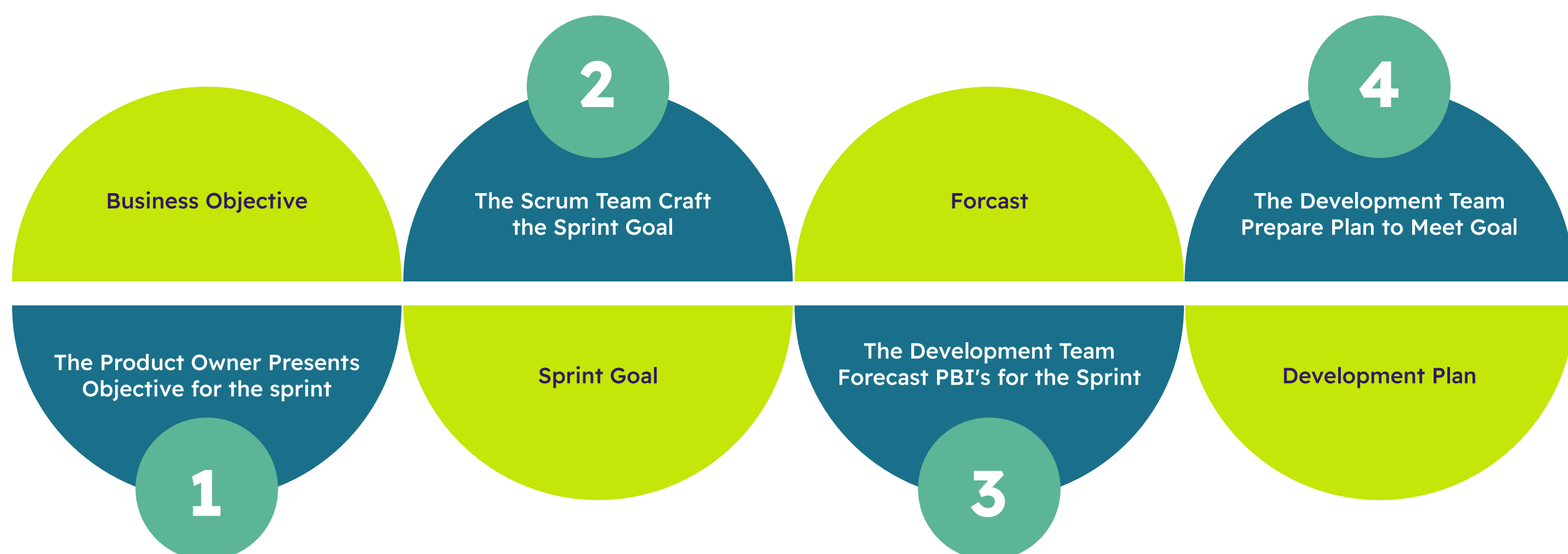
Each member of the team responds to the following three questions during the Daily Scrum:

- What did I do the day before to contribute to the Development Team's success in reaching the Sprint Goal?
- What can I do today to aid the Development Team in achieving the Sprint Goal?
- Does the Development Team have any roadblocks to achieving the Sprint Goal?

Members of the team are only allotted 15 minutes total to talk at the Daily Scrum, so they should get right to the point. This meeting is just to check in and see how things are doing, not to solve any problems. While conducting the Daily Scrum, the Scrum Master will look for any roadblocks to the team's progress and take steps to eliminate them.

SPRINT PLANNING

One of Scrum's five events is called "Sprint Planning." Sprint Planning is a time-limited meeting held at the start of each Sprint to map out the tasks that the Development Team will tackle during that Sprint. Members of the Development Team, Product Owner, and the Scrum Master are all expected to be present at the Sprint Planning event.



scrum.org

At the Sprint Planning meeting, the group decides what they want to accomplish in the upcoming Sprint by discussing and answering the following two questions:

- To what extent may the Sprint Goal and the anticipated Increment be accomplished through actions taken during this Sprint?
- To what extent will the Increment require additional effort, and how will it be accomplished?

The Product Owner then provides the items from the Product Backlog that are, in their opinion, the most critical for the team to focus on during the upcoming Sprint in response to the first question. The Development Team then meets to discuss these tasks and determine which ones they can reasonably expect to complete during the next Sprint.

The Development Team provides an answer to the second question by decomposing the prioritized features into manageable jobs and determining how much time will be needed to finish each one. The risks and dependencies that may affect the Development Team's ability to finish the task are also identified.

The Development Team knows exactly what needs to be done and how to execute it when the Sprint Planning event has concluded. Both parties are in agreement over the Sprint Objective and the anticipated Increment.

THE SPRINT REVIEW

One of the Scrum framework's five events is the Sprint Review. The Sprint retrospective is a meeting/gathering held at the end of each Sprint to assess the progress made by the Development Team during that Sprint. Members of the Development Team, Product Owner, the Scrum Master, and other stakeholders (including customers, users, and managers) are expected to attend the Sprint Review event.

The Development Team presents the work they've accomplished during the Sprint for review during the Sprint Review. This may take the form of demonstrations of newly implemented features, data displays, or code tours. The group also talks about the difficulties they faced during the Sprint and the lessons they were able to learn as a result.



The Sprint Review is where the Product Owner checks in on the product's development about Sprint's intended outcomes. In this way, everyone involved can see how far the product has come, what new features have been added, and how well the job has been done thus far. It also lets people with vested interests in the project's outcome weigh in on its course.

The Sprint Review's main purpose is to examine the Increment and modify the Product Backlog as needed. After each Sprint, the Development Team and the people who were affected by the work discuss it together. The Product Owner can use this information to add or remove things from the Product Backlog ahead of the next Sprint.

THE SPRINT RETROSPECTIVE

As one of Scrum's five events, the Sprint Retrospective is meant to help teams reflect on their most recent Sprint progress. The Sprint Review is a time-limited meeting held at the end of every Sprint during which the team discusses its performance during the Sprint and brainstorms methods to improve. The Development Team, the Scrum Master, and occasionally the Product Owner all participate in the Sprint Retrospective meeting.

The Sprint Retrospective is the time when the team discusses how well they did during the Sprint by answering three questions.

- What worked successfully throughout the Sprint?
- What didn't go well during the Sprint?
- How can we make the most of the upcoming Sprint?

The group then examines the results of these questions and decides what they can do differently in the next Sprint to achieve better results. The Scrum Master leads the conversation and directs the team's attention to problem areas.

Sprint Retrospectives are held primarily to foster a culture of constant enhancement. The team discusses the successes and failures of the previous iteration and then settles on specific next steps for advancement. The Scrum Master guides the group in determining the sources of issues and developing practical, measurable, and achievable solutions.

The Sprint Retrospective can also be used as a trust-building and team-bonding exercise. Communication and cooperation can be boosted when team members open up to one another about their personal experiences and perspectives.

WHO SHOULD RUN SPRINT RETROSPECTIVE MEETINGS?

The Scrum Master is responsible for coordinating the Sprint Retrospective meeting. The Scrum Master's job is to monitor the team's progress and ensure they stay true to Scrum's guiding principles and practices. It's the Scrum Master's job to lead the team through the Sprint Retrospective meeting, where they can help them figure out where they can become better and what they can do to make big strides forward in the next iteration.

The Scrum Master plays a critical role in the Sprint Retrospective by keeping the team on track and ensuring that they achieve the meeting's stated objectives. They also aid the group in determining the sources of issues and coming up with practical, quantifiable, and attainable remedies. The Scrum Master also aids in making sure that the meeting stays within its allotted time and that everyone gets a chance to speak.

The Scrum Master also fosters a safe and trusting setting in which team members feel comfortable expressing their opinions and concerns. The Scrum Master's role at the Sprint Retrospective meeting is to facilitate an environment where the team can honestly discuss and seek solutions to any problems that may be hindering their performance or their ability to produce high-quality work.

WHAT HAPPENS IN A SPRINT RETROSPECTIVE MEETING?

The Development Team reviews their work from the previous Sprint and makes suggestions on how they may do better during the upcoming Sprint during the Sprint Retrospective meeting. The meeting is often steered by the Scrum Master, who ensures that everyone gets a chance to speak and that the conversation stays on topic.

It is common practice for the Sprint Retrospective meeting to follow the stages below.

The Development Team reviews their work from the previous Sprint and makes suggestions on how they may do better during the upcoming Sprint during the Sprint Retrospective meeting. The meeting is often steered by the Scrum Master, who ensures that everyone gets a chance to speak and that the conversation stays on topic.

It is common practice for the Sprint Retrospective meeting to follow the stages below.

- The team reviews the previous Sprint: The team reviews the Sprint prior and talks about what worked and what didn't. They consider whether or not the Sprint Objective was accomplished.
- The team identifies areas for improvement: The team decides on specific areas for improvement in the next Sprint based on the discussion from step 1.
- The team develops action items: In order to boost their productivity, the team makes a list of measures they can do during the upcoming Sprint. These steps need to be clear, quantifiable, and achievable.

- The team assigns action items: Members of the team are given individual tasks to do.
- The team sets goals for the next Sprint: The team decides what they want to accomplish in the upcoming Sprint based on the items they've prioritized.
- The team reviews progress: The team discusses how far along they are in completing the tasks from the prior Sprint and any problems they have run into.
- The team closes the meeting: The meeting is then closed by the Scrum Master after any other relevant subjects have been discussed.

It's also worth noting that the team's performance and the Scrum process as a whole can benefit from regular refinement at the Sprint Retrospective meeting. The team should be prompted to discuss any problems they had with the Scrum methodology during the Sprint and suggest ways to fix them. The Scrum Master should also strive to enhance the Sprint Retrospective meeting by soliciting the team's input and making incremental changes to the meeting's format and structure.

THE BENEFITS OF SPRINT RETROSPECTIVE MEETINGS

The Development Team and the company as a whole can benefit greatly from holding Sprint Retrospective sessions.



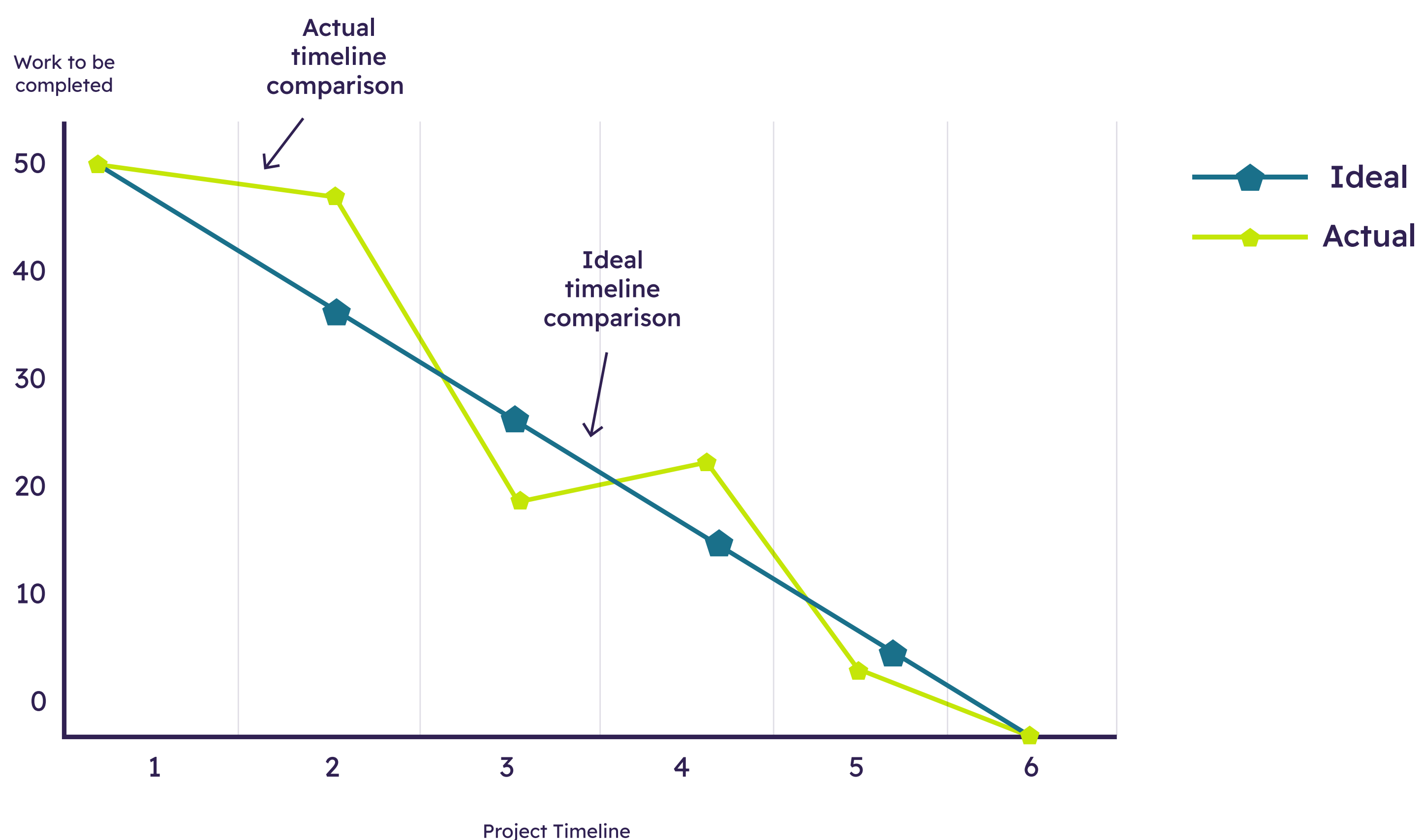
Benefits include, but are not limited to, the following

- **Continuous improvement:** The team can assess their progress and make adjustments during the Sprint Retrospective meeting. As a result, the team's performance improves with time.
- **Increased transparency:** During the Sprint Retrospective meeting, everyone on the team can be completely candid about their experiences and ideas. This is a great step forward to boost communication and openness among the team.
- **Increased collaboration:** Members of the team are prompted to collaborate to pinpoint problem areas during the Sprint Retrospective meeting. As a result, team members are better able to work together, and collaboration is boosted.

- Increased accountability: Team members are encouraged to take responsibility for their performance and to make concrete, measurable commitments to improve in the upcoming Sprint during the Sprint Retrospective meeting. As a result, team members feel more responsible for their actions, and the team as a whole is better able to achieve its goals.
- Alignment with organizational goals: At the Sprint Retrospective meeting, the team is prompted to reflect on the bigger picture and consider how their efforts relate to the larger aims and objectives of the business. As a result, the team's efforts become more focused and productive, contributing to the organization's overall success.
- Increased team morale: Participants in the Sprint Retrospective are urged to be open and honest with one another as they discuss Sprint's successes and failures and brainstorm ways to enhance the process moving forward. This boosts morale and makes everyone on the team happier with their jobs.
- Improving the Scrum Process: Throughout the Sprint Retrospective, the team can discuss and come up with solutions to any problems that arose during the Sprint that was relevant to the Scrum methodology. This contributes to the enhancement of the Scrum process and its usefulness.

BURNDOWN CHARTS AND VELOCITY TRACKING

Scrum teams frequently utilize burndown charts and velocity monitoring to monitor their projects' development.



<https://www.villanovau.com/>

The amount of work remaining over time can be seen using a burndown chart. Its purpose is to monitor a project's development and anticipate any problems that may arise. For most projects, the chart will look like a line graph that compares the amount of work left to do versus the remaining amount of time (usually measured in days). A successful burndown chart will demonstrate a declining backlog over time, demonstrating the team's progress toward project completion. Having the line on the graph plateau or rise could mean that the team is hitting roadblocks.

The velocity tracking approach is used to determine how much work a team can accomplish in a particular time frame (usually a sprint). The team's velocity is the sum of all the user stories or tasks that were finished in a sprint divided by the total amount of time spent on those stories or activities. The size or complexity of a user story is typically quantified in story points. Future sprints may be planned more effectively, and the team's overall performance can be monitored with the help of velocity. If the team's velocity is steadily going up, that could mean they're getting better at what they do. If the team's velocity keeps dropping, it could be because they're having trouble completing their tasks.

Burndown charts and velocity tracking are both useful for monitoring a project's development and discovering any problems that may arise. By combining them, you may gain a fuller picture of the team's development. The burndown chart can aid in the early detection of problems by providing a visual picture of the work remaining over time. The team's progress can be monitored over time, and future sprints may be more effectively prepared with the help of velocity tracking.

USER STORIES AND ACCEPTANCE CRITERIA

A user story is a short, conversational description of a feature or needs written from the customer's point of view used in the Scrum framework. A common format is "As a [user], I want [feature], so that [benefit]." A user story is a short, self-contained narrative representing a single feature the Development Team is responsible for implementing.

The acceptance criteria define the conditions under which a user narrative can be considered complete. The Development Team can use these standards to gauge the user story's progress and determine when it is finished. Writing acceptance criteria in a straightforward, testable language helps verify that the user narrative actually delivers on its promise of satisfying the user.

User stories and acceptance criteria work together to clarify what's expected of a given feature, so the Development Team can be sure it's making progress toward a solution that will satisfy customers. With the use of acceptance criteria, user stories can be used to break down huge, complex requirements into smaller, more manageable portions. It's also important to note that tools like wireframes, mockups, and prototypes may be utilized with user stories and acceptance criteria to help get a better grasp on what's needed for a certain feature.

ESTIMATING AND PLANNING POKER

Estimation in Scrum is the act of calculating how much time will be needed to finish a user story or a given task. The Development Team does this during sprint planning to assist them in organizing and prioritizing their tasks for the upcoming Sprint.

Planning Poker is often used as an estimation method in Scrum. Planning Poker is a method for estimating the time and resources needed to accomplish a user narrative through group consensus. The method employs a deck of cards, each of which is numbered according to the Fibonacci sequence (1, 2, 3, 5, 8, 13, 21, 34, 55, 89, etc.).

Members of the Development Team are often given their deck of cards to use during a Planning Poker session. Next, the group discusses the user story, and then each member chooses a card from his or her deck to represent his or her estimate of the time commitment involved in completing the tale. The group then concurrently shares their predictions. If there's a discrepancy between the estimates, the group will talk about what could be causing the discrepancy and re-estimate the tale.

Planning Poker is a technique used to determine how much work will be necessary to finish a user story. This method allows team members to share their thoughts and opinions to arrive at a more precise estimate.

T-shirt sizing, Affinity estimate, and other ways exist in addition to Planning Poker. However, Scrum teams frequently use Planning Poker because of the success it has brought them.

WORKING WITH DEPENDENCIES

A dependency in Scrum is an association between two or more tasks where one task's completion is reliant on the other task's completion. User stories, tasks, and even sprints themselves can all have interdependencies.

In managing a Scrum project, dealing with dependencies is crucial because they can majorly affect the team's output. If one user narrative depends on another story that has not been finished, the dependent story cannot be finished until the dependence is resolved.

Finding dependencies early and making sure everyone on the team understands them is one approach to managing them. This can be achieved by conducting frequent reviews of the product backlog, during which any dependencies between stories can be uncovered, and the team's understanding of those dependencies can be improved.

Alternatively, you can use a dependency matrix or a Gantt chart to see the interdependencies between your stories and manage them that way. The team's ability to see the interconnected nature of the stories will improve as a result of this insight.

It is also crucial to recognize and control inter-group interdependencies. One way to handle this is to establish an open line of communication between teams and designate a single point of contact for each dependence.

MANAGING AND PRIORITIZING THE PRODUCT BACKLOG

The product backlog in Scrum is a prioritized list of features and requirements for a product or project. When the team isn't working on the most pressing issues, the product owner manages and prioritizes the product backlog.

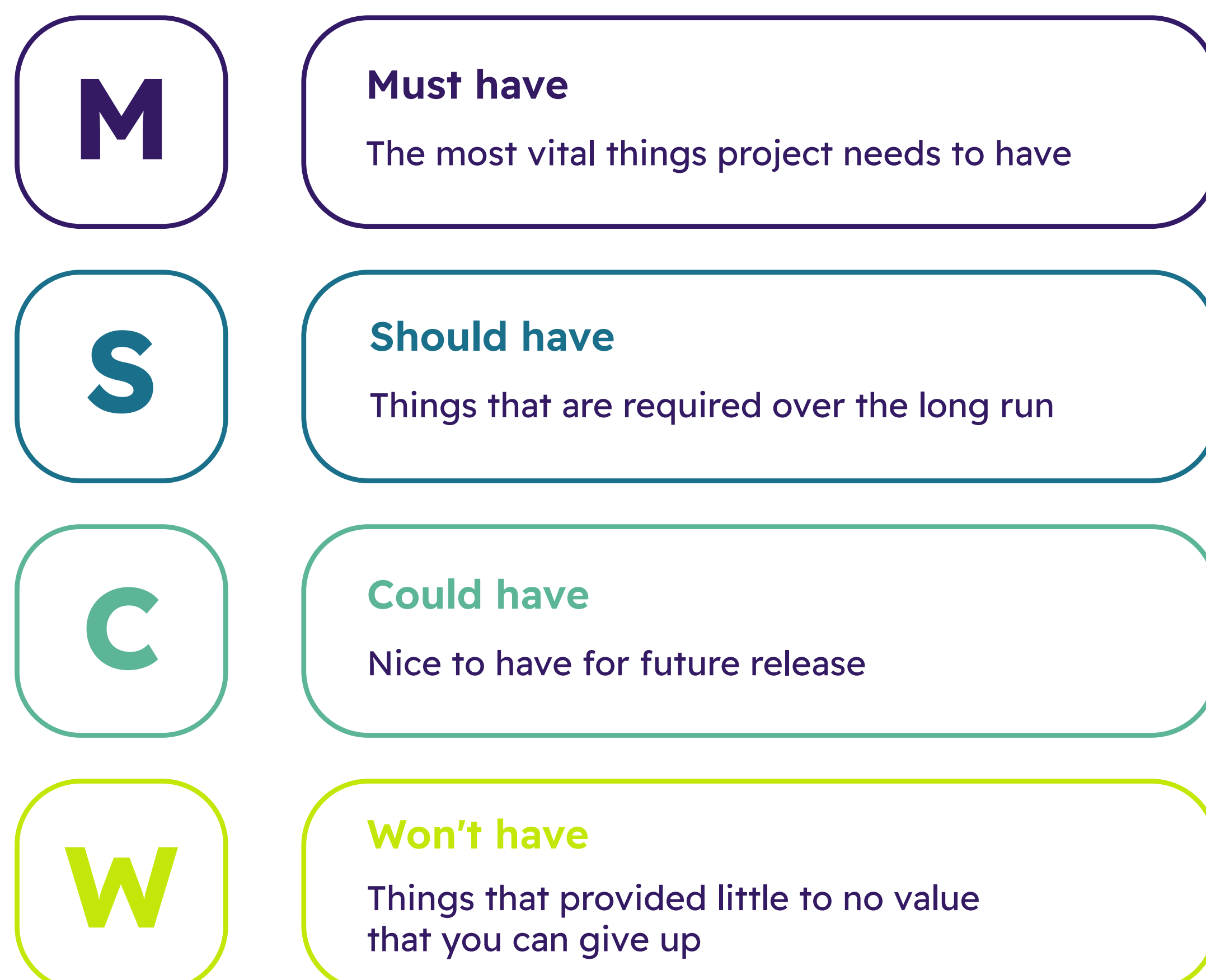
There are many components to managing the product backlog.

- Finding and writing down new requirements or features for the backlog.
- Putting items in the backlog in order of priority to the business or the end user.
- Clarifying and elaborating on the items in the backlog to give the Development Team precise instructions.
- Spreading knowledge of the backlog's contents among the Development Staff and other interested parties.
- Coordinating the team's ability to deliver the items in the backlog by managing the dependencies between them.

PRODUCT BACKLOG

The product backlog needs to be prioritized so that the team may focus on the most important tasks at any given time. Evaluating considerations such as item value, risk, and dependencies, the product owner should set priorities based on their significance to the business or end user.

We have the MoSCoW (Must have, Should have, Could have, Won't have) method for prioritizing the product backlog. The product owner can prioritize the items in the backlog by assigning a higher priority to those that are closer to the top of the list using this method.



Using a Kano model or similar tool is another option for prioritizing the product backlog. To ensure the team concentrates on the things that will have the most influence on the end user, this tool helps prioritize items based on the level of happiness they will bring to the end user.

BEST PRACTICES FOR EFFECTIVE SCRUM IMPLEMENTATION

The following are some of the most helpful techniques for ensuring a successful Scrum implementation in any organization:

- Start with a small pilot project: Starting with a modest pilot project can assist a team in grasping the fundamentals of Scrum before moving on to larger, more complex projects after the framework has been introduced.
- Emphasize teamwork: Scrum is a framework that is built on teams, so communication and cooperation must be fostered among employees. Facilitating honest dialogue, instilling a feeling of mutual accountability, and motivating team members to cooperate effectively can all help.
- Appoint a skilled Scrum Master: Appointing a competent and experienced Scrum Master who can keep the team on task is essential for the success of a Scrum project.
- Encourage continuous improvement: Since Scrum is an iterative methodology, the entire company should strive to improve. This can be achieved by conducting periodic reviews of the team's operations, finding areas for improvement, and subsequently putting into place modifications to enhance the team's performance.
- Emphasize transparency: In Scrum, open communication is essential. All team members and interested parties must be kept up to date on the project's development.

- Use agile metrics: Burndown charts, velocity, and lead time are just a few examples of Agile metrics that may be used to gauge progress, pinpoint problem areas, and monitor the project's health performance over time.
- Make sure the team is properly trained: Scrum is a complex structure; thus, it's crucial that every member of the team receives adequate training and knows the fundamentals. To achieve this goal, teams can receive training and mentoring and be encouraged to participate in and attend Scrum-related events and conferences.

COMMON SCRUM ANTI-PATTERNS AND HOW TO AVOID THEM

Scrum anti-patterns are the problems that can arise when a company tries to use the Scrum framework. Scrum anti-patterns are prevalent in many projects, but luckily there are techniques to fix them. Scrum Anti-pattern includes:

- Not having a dedicated product owner
- Not having a dedicated Scrum Master
- Not having a cross-functional team
- Not having a clear definition of “done.”
- Not having a clear product backlog
- Not having regular sprint retrospectives

To avoid these anti-patterns, organizations should take the following steps:

- Appoint a dedicated product owner
- Appoint a dedicated Scrum Master
- Create a cross-functional team
- Define a clear definition of “done.”
- Prioritize the product backlog
- Hold regular sprint retrospectives
- Monitor and track the progress
- Provide regular training

Organizations have a better chance of implementing Scrum and seeing the results they want if they adhere to these best practices and avoid prevalent Scrum anti-patterns.

USING AGILE TOOLS AND SOFTWARE IN SCRUM

Organizations can improve their Scrum implementation with the help of several accessible, agile tools and technologies. Some common tools are:

- Project management software: Work management tools like Jira, Trello, and Asana can help teams keep track of tasks, set priorities for their backlog, and organize their sprints.
- Collaboration tools: Slack, Microsoft Teams, and Zoom are just a few of the tools that can help teams communicate and cooperate more efficiently.
- Planning tools: Teams may estimate and plan their work with the use of tools like Planning Poker, Estimator, and Agile Estimation.
- Tracking tools: Burndown charts, velocity, and lead time are just a few of the tools that may be used to monitor the efficiency and effectiveness of a team's work.
- Retrospective tools: Sprint Retrospectives, Retrium, and Agile Retrospectives are just a few of the retrospective tools that may help teams have productive sprint retrospectives.

Keep in mind that while agile technologies can help control and track the project's progress, they shouldn't be used in isolation. It is crucial that the team get thorough Scrum training and that all team members have a firm grasp of how to operate inside the Scrum framework.

Tools should be chosen with the team's needs, the project's goals, and the team's familiarity with agile and Scrum in mind. It's helpful to evaluate a variety of tools and settle on the one that proves most useful to the team and the undertaking at hand.

COLLABORATING WITH STAKEHOLDERS AND EXTERNAL TEAMS

Working with stakeholders and other groups outside of the Scrum team can be difficult, but it's essential to delivering a product that satisfies the needs of the stakeholders. Scrum's recommended practices for working with external teams and stakeholders are as follows:

- **Involve stakeholders in the planning process:** Participate in sprint planning, backlog grooming, and retrospectives by including stakeholders in the process. They'll be in a better position to contribute to the team's efforts and understand the team's priorities if they do.
- **Communicate regularly:** To keep stakeholders abreast of the team's work and any difficulties that may develop, it is important to set up frequent communication channels, such as weekly or biweekly meetings.
- **Prioritize their needs:** Ensure that the product backlog takes into account and prioritizes stakeholder input. By doing so, the team can better guarantee that the final result fully satisfies all concerned parties.
- **Encourage feedback:** Initiate regular opportunities for stakeholders, such as sprint reviews, to provide input on the product. The team will be able to spot problems and make corrections with this information in hand.
- **Facilitate collaboration:** Inspire and ease the interaction between your group and any interested parties or other groups. This will help guarantee that the team is all pulling in the same direction and producing a result that pleases the stakeholders.

- Define clear roles and responsibilities: Make sure that all parties participating in the project, including the Scrum team, the stakeholders, and any external teams, all know their specific roles and responsibilities. All parties involved will thus be better able to understand their roles and cooperate efficiently.

By adhering to these standards, businesses increase their chances of cooperating with stakeholders and external teams in a Scrum setting and producing a product that is satisfactory to those teams.

MANAGING AND TRACKING PROGRESS WITH SCRUM BOARDS

Teams in a Scrum setting utilize visual tools called “Scrum boards” to manage and track their progress. You can find them in both digital and analogue forms, but the common denominator is a board with columns labelled “To Do,” “In Progress,” and “Done” to reflect the many steps in a Scrum process. A note or digital card represents each job or item in the product backlog, and the team pushes them down the columns as they are done.

- **Prioritizing the backlog:** The team can utilize the board to determine what items in the product backlog should be tackled first.
- **Tracking progress:** The team can utilize the board to keep tabs on the status of each item and task and to pinpoint any obstacles standing in the way of completion.
- **Identifying dependencies:** The team can use the board to track the status of interdependent tasks and make sure they’re being worked on promptly.
- **Managing work in progress:** The board can be used to monitor the team’s workload and prevent overscheduling.
- **Improving communication:** The board is a great tool for enhancing the team’s ability to communicate and collaborate by allowing everyone to keep tabs on the progress of their colleagues’ tasks.
- **Reviewing performance:** The team can utilize the board to assess its performance by monitoring its evolution over time and pinpointing problem spots.

Scrum boards are an effective method for managing and monitoring work in progress in an agile Scrum setting, but they should be used in tandem with other agile metrics like burndown charts, velocity, and lead time for a complete picture of the project's performance.

CLOSING THE PROJECT: LESSONS LEARNED AND NEXT STEPS

When a project comes to a close, it's a good idea to think about what was accomplished and what comes next. The best practices for finishing a project in a Scrum setting are as follows:

- Hold a final retrospective: Gather everyone for one last meeting to discuss the impact of the project. This can be useful in figuring out where the team shined and where they could use some work.
- Document lessons learned: You should write down the team's triumphs, failures, and suggestions for improvement. It can assist the group in avoiding making the same mistakes twice.
- Review the product backlog: Verify that everything on the product backlog has been addressed or removed if it is no longer needed. Doing so can aid in getting the product backlog prepared for the next endeavor.
- Close out the project: Finish up in Jira or Trello or whatever else was utilized to keep tabs on the project's development and progress.
- Communicate with stakeholders: Notify anyone interested in the project's outcome and hand up any remaining deliverables.
- Celebrate the team's achievements: Don't forget to pause and honor the team's efforts. The positive effects on morale and drive can be significant.

- Next steps: Create a strategy based on what you've discovered. Things like training team members in new skills and preparing for future projects fall under this category.

By adhering to these guidelines, businesses can make sure the project is wrapped down properly, the team has a chance to reflect on what they've learned, and the business is ready to move forward with future endeavors.

TIPS AND TRICKS FOR RUNNING SUCCESSFUL SPRINTS

In a Scrum setting, satisfying stakeholders depend on smoothly executed sprints. You can learn how to sprint faster and more efficiently by reading the following.

- Start with a clear goal: Verify that your team understands the Sprint's intended outcome. In this way, you can be sure that everyone in the team is pulling in the same direction.
- Prioritize the backlog: Set a priority order for the items in the product backlog so that the team can focus on the most critical initiatives first.
- Estimate tasks accurately: If you want to be sure your job estimates are spot on, you might try using planning poker or another estimation method. The team's ability to deliver on its promises will be bolstered in this way.
- Keep the sprint backlog visible: Maintain a constantly up to-date public sprint backlog. This will make it more likely that everyone in the team is on the same page and contributing to the same result.
- Hold daily stand-up meetings: Keep the team apprised of developments with daily stand-up meetings, where problems can be discussed and resolved.
- Track progress: Tracking progress and spotting potential bottlenecks can be done with the use of burndown charts, velocity, and other indicators.
- Encourage collaboration: Foster teamwork and open lines of communication, both internally and with external stakeholders. This will make it more likely that all hands are on deck for the same mission.

- Reflect and adjust: At the end of each Sprint, the team should hold a retrospective meeting to discuss the experience and make any necessary revisions to the next Sprint's plan.

Using these guidelines, businesses have a better chance of implementing Scrum successfully and delivering value to stakeholders during sprints.

CONCLUSION: THE FUTURE OF SCRUM IN PROJECT MANAGEMENT

Scrum is a popular Agile project management framework, and for a good reason. Scrum continues to be a popular and successful paradigm for managing projects in an Agile setting, and its future in project management appears bright as organizations tackle the difficulties of an evolving business world. Teams can swiftly adapt to new requirements and bring value to stakeholders with the help of the Scrum framework, which provides a highly adaptable and flexible approach. This will become increasingly crucial as businesses embrace more complicated and ever-changing projects. Adopting Scrum and its methods can help businesses deal with the uncertainties of the modern marketplace and satisfy their stakeholders.

REFERENCES

An Overview of the Scrum Framework | Scrum Alliance. (n.d.). <https://resources.scrumalliance.org/Article/overview-scrum-framework>

Atlassian. (n.d.-a). Scrum Metrics 101. <https://www.atlassian.com/agile/scrum/scrum-metrics>

Atlassian. (n.d.-b). What is Scrum? <https://www.atlassian.com/agile/scrum>

Berteig, M. (2020, November 19). Scrum Rules: The Daily Scrum Is Time-Boxed to 15 Minutes. BERTEIG Consulting and Training. <https://berteig.com/how-to-apply-agile/scrum-rules-the-daily-scrum-is-time-boxed-to-15-minutes/>

Can You Rotate the Scrum Master Role? (n.d.). Scrum.org. <https://www.scrum.org/resources/blog/can-you-rotate-scrum-master-role>

Chappell, E. (2023, January 30). 50 Agile Scrum Terms Everyone Must Know (Detailed Glossary). ClickUp. <https://clickup.com/blog/agile-scrum-terms/>

Cupps, J. (2023, February 1). Leading an Agile Team: A Breakdown of Key Roles. <https://www.linkedin.com/pulse/leading-agile-team-breakdown-key-roles-james-cupps>

Donato, H. (2023, February 3). How To Be A Scrum Master. Workamajig. <https://www.workamajig.com/blog/scrum-methodology-guide/scrum-master>

Nedre, N. (2020, September 29). Scrum Boards vs Kanban: 11 Major Differences | Miro. MiroBlog. <https://miro.com/blog/scrum-kanban-boards-differences/>

Owner, & P. (2023, January 25). Scrum in 5 Minutes: A Simplistic and Humorous Guide. What Is Scrum. <https://www.whatisscrum.org/scrum-in-5-minutes-a-simplistic-and-humorous-guide/>

Perfect, A. (2021, July 23). Scrum Events: Myths and Facts. Approach Perfect. <https://approachperfect.com/agile-training/scrum-events-myths-and-facts/>

Scrum: A Breathtakingly Brief And Agile Introduction | Agile Learning Labs. (2023a, February 6). <https://agilelearninglabs.com/resources/scrum-introduction/>

Scrum: A Breathtakingly Brief And Agile Introduction | Agile Learning Labs. (2023b, February 6). <https://agilelearninglabs.com/resources/scrum-introduction/>

Scrum Guide 2017 | Scrum Guides. (n.d.-a). <https://scrumguides.org/scrum-guide-2017.html>

Scrum Guide 2017 | Scrum Guides. (n.d.-b). <https://scrumguides.org/scrum-guide-2017.html>

ShahJan 05, K. 2. (n.d.). Scrum Master and Product Owner: Their Roles and Responsibilities | Thirdock Techkno. Third Rock Techkno. <https://www.thirdrocktechkno.com/blog/scrum-master-and-product-owner-their-roles-and-responsibilities/>

Should the Product Owner Accept Stories aka PBI's? (n.d.). Scrum.org. <https://www.scrum.org/forum/scrum-forum/37569/should-product-owner-accept-stories-aka-pbis>

Singh, M. (2022, June 3). What Is Scrum Methodology? & Scrum Project Management. Digite. <https://www.digite.com/agile/scrum-methodology/>
Skillsu/Advisedskills, A. (2023, February 3). How to do effective sprint planning as a product owner? Reddit. https://www.reddit.com/user/advisedskills/comments/10sk041/how_to_do_effective_sprint_planning_as_a_product/
Sutherland, J. J. (2022, June 21). Product Backlog. Scrum Inc. <https://www.scruminc.com/product-backlog/>

Thakkar, B. (2022, November 22). What are the five types of scrum meetings? LogRocket Blog. <https://blog.logrocket.com/product-management/what-are-five-types-of-scrum-meetings/>

Understanding and Managing Product Backlogs. (n.d.). UniversalClass.com. <https://www.universalclass.com/articles/business/understanding-and-managing-product-backlogs.htm>

Van Osnabrugge, B. R. (2013, October 17). TFS as perfect tool for Scrum (Part 4) – Sprint. The Road to ALM. <https://roadtoalm.com/2013/07/11/tfs-as-perfect-tool-for-scrum-part-4-sprint/>

Verwijls, C. (2020, October 15). How To Kickstart A Great Scrum Team (10 practical things to do). Medium. <https://medium.com/the-liberators/how-to-kickstart-a-great-scrum-team-10-practical-things-to-do-2143bdde1a8d>

