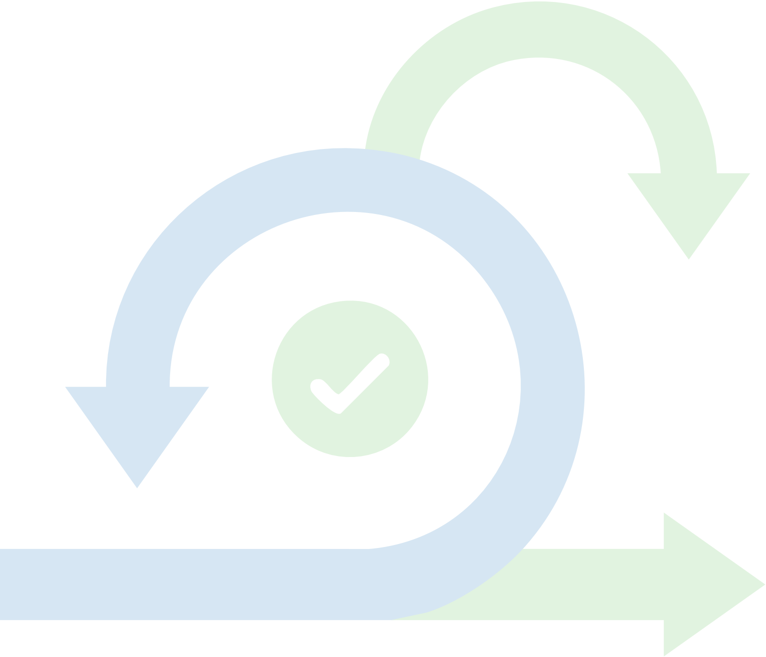
SCRUMstudy
Targeting success



A Guide to the

SCRUM BODY OF KNOWLEDGE

(SBOK® Guide)

5. QUALITY

A Comprehensive Guide to Implementing and Scaling Scrum, with Practical Examples

(Includes insights into how Artificial Intelligence can enhance Scrum processes)

5. QUALITY

5.1 Introduction

The purpose of this chapter is to define quality as it relates to projects and to present the Scrum approach to achieve the required levels of quality. *Quality*, as defined in *A Guide to the Scrum Body of Knowledge* (*SBOK® Guide*), is applicable to the following:

- Portfolios, programs, and/or projects in *any* industry
- Products, services, or any other results to be delivered to business stakeholders
- Projects of any size or complexity

The term "product" in the *SBOK® Guide* may refer to a product, service, or other deliverable. Scrum can be applied effectively to any project in any industry—from small projects or teams with as few as six team members to large, complex projects with up to several hundred members in several teams. This chapter is divided into the following sections:

5.1.1 Roles Guide—This section provides guidance on which sections are relevant for each Scrum role: Product Owner, Scrum Master, and Scrum Team.

5.2 Quality Defined—This section provides the Scrum definition of quality, with a clear distinction from scope, and describes the relationship between quality and business value.

5.3 Acceptance Criteria and the Prioritized Product Backlog—This section emphasizes the importance of Acceptance Criteria, the Prioritized Product Backlog, and their relationship. It also explains the Scrum definition of Done.

5.4 Quality Management in Scrum—This section provides details on quality planning, quality control, and quality assurance in the context of Scrum.

5.5 Summary of Responsibilities—This section describes the responsibilities relevant to quality for each person or role in a project.

5.6 Scrum vs. Traditional Project Management—This section highlights the benefits of quality management in Scrum method over traditional project management models.

5.1.1 Roles Guide

1. Product Owner—It is important for anyone assuming the role of Product Owner in Scrum projects to read this complete chapter.
2. Scrum Master—The Scrum Master should also be familiar with this entire chapter with primary focus on sections 5.3, 5.4, 5.5.3, and 5.6.
3. Scrum Team—The Scrum Team should focus on sections 5.3, 5.4, and 5.6.

5.2 Quality Defined

There are numerous ways to define quality.

In Scrum, quality is defined as the ability of the completed product or deliverables to meet the Acceptance Criteria and achieve the business value expected by the customer.

To ensure that a project meets quality requirements, Scrum adopts an approach of continuous improvement whereby the team learns from experience and stakeholder engagement to constantly keep the Prioritized Product Backlog updated with any changes in requirements. The Prioritized Product Backlog is simply never complete until the closure or termination of the project. Any changes to the requirements reflect changes in the internal and external business environment and allow the team to continually work and adapt to achieve those requirements. Since Scrum requires work to be completed in increments during Sprints, this means that errors or defects get noticed earlier through repetitive quality testing, rather than when the final product or service is near completion. Moreover, important quality-related tasks (e.g., development, testing, and documentation) are completed as part of the same Sprint by the same team—this ensures that quality is inherent in any Done deliverable created as part of a Sprint. Thus, continuous improvement with repetitive testing optimizes the probability of achieving the expected quality levels in a Scrum project. Constant discussions between the Scrum Core Team and business stakeholders (including customers and users) with actual increments of the product being delivered at the end of every Sprint, ensures that the gap between customer expectations from the project and actual deliverables produced is constantly reduced.

5.2.1 Quality and Scope

Scope and quality requirements for a project are determined by taking into consideration numerous factors such as the following:

- The business need the project will fulfill
- The capability and willingness of the organization to meet the identified business need
- The current and future needs of the target audience

The scope of a project is the total sum of all the product increments and the work required for developing the final product. Quality is the ability of the deliverables to meet the quality requirements for the product and satisfy customer needs. In Scrum, the scope and quality of the project are captured in the Prioritized Product Backlog, and the scope for each Sprint is determined by refining the large Prioritized Product Backlog Items (PBIs) into a set of small but detailed User Stories that can be planned, developed, and verified within a Sprint.

The Prioritized Product Backlog is continuously refined by the Product Owner. The Product Owner ensures that any User Stories that the Scrum Team is expected to do in a Sprint are refined prior to the start of the Sprint. In general, the most valuable requirements in solving the customers' problems or meeting their needs are prioritized as high and the remaining are given a lower priority. Less important User Stories are developed in subsequent Sprints or can be left out altogether according to the customer's requirements. Throughout the entire project, the Product Owner, customer, and the Scrum Team discusses and changes the list of features of the product to comply with the changing needs of the customers.

5.2.2 Quality and Business Value

Quality and business value are intricately linked. Therefore, it is critical to understand the quality and scope of a project in order to correctly map the outcomes and benefits the project and its product must achieve in order to deliver business value. To determine the business value of a product, it is important to understand the business need that drives the requirements of the product. Thus, business needs determine the product required, and the product, in turn provides the expected business value.

Quality is a complex variable. An increase in scope without increasing time or resources tends to reduce quality. Similarly, a reduction in time or resources without decreasing scope also generally results in a decrease in quality. Scrum practices encourage maintaining a "sustainable pace" of work, which helps improve quality over a period of time.

The Scrum Guidance Body may define minimum quality requirements and standards required for all projects in the organization. The standards must be adhered to by all Scrum Teams in the organization.

5.3 Acceptance Criteria and the Prioritized Product Backlog

The Prioritized Product Backlog is a single requirements document that defines the project scope by providing a prioritized list of features of the product or service to be delivered by the project. The required features are described in the form of User Stories. User Stories are specific requirements outlined by relevant business stakeholders as they pertain to the proposed product or service. Each User Story will have associated User Story Acceptance Criteria (also referred to as "Acceptance Criteria"), which are the objective components by which a User Story's functionality is judged. Acceptance Criteria are developed by the Product Owner according to his or her expert understanding of the customer's requirements. The Product Owner then communicates the User Stories in the Prioritized Product Backlog to the Scrum Team members, and their agreement is sought. Acceptance Criteria should explicitly outline the conditions that User Stories must satisfy. Clearly defined Acceptance Criteria are crucial for timely and effective delivery of the functionality defined in the User Stories, which determines the success of the project.

At the end of each Sprint, the Product Owner uses these criteria to verify the completed deliverables; and can either accept or reject individual deliverables and their associated User Stories. If deliverables are accepted by the Product Owner, then the User Story is considered Done. A clear definition of Done is critical because it helps clarify requirements and allows the team to adhere to quality norms. It also helps the team think from the user's perspective when working with User Stories. User Stories corresponding to rejected deliverables are added back to the Prioritized Product Backlog to be considered for completion in future Sprints. The rejection of a few individual deliverables and their corresponding User Stories is not a rejection of the final product or product increment. The product or product increment could potentially be shippable even if a few User Stories are rejected.

Figure 5-1 is a Project Increment Flow Diagram, showing how defined acceptance criteria lead to validated product increments delivered at the end of each sprint.

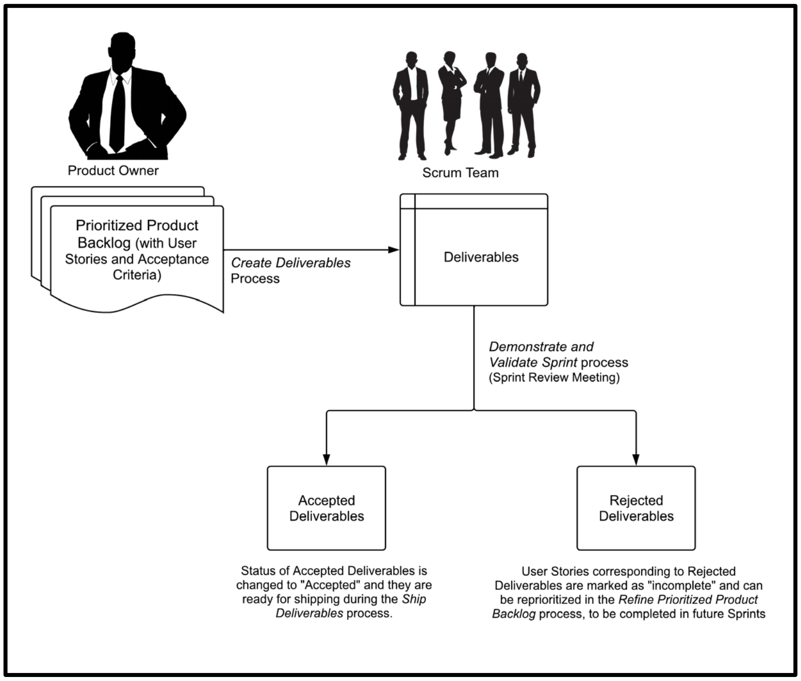

Product Owner
Scrum Team
Prioritized Product Backlog (with User Stories and Acceptance Criteria)
Create Deliverables Process
Deliverables
Demonstrate and Validate Sprint process (Sprint Review Meeting)
Accepted Deliverables
Rejected Deliverables
Status of Accepted Deliverables is changed to "Accepted" and they are ready for shipping during the *Ship Deliverables* process.
User Stories corresponding to Rejected Deliverables are marked as "incomplete" and can be reprioritized in the *Refine Prioritized Product Backlog* process, to be completed in future Sprints

Figure 5-1: Project Increment Flow Diagram

5.3.1 Writing Acceptance Criteria

Acceptance Criteria are unique to each User Story and are not a substitute for a requirements list. The following is an example of how Acceptance Criteria could be written.

Example:

Persona: Janine is a married 36-year-old working professional with a family of three children. She is a busy, successful woman who balances her professional and personal life. She is comfortable with technology and is an early adopter of innovative services and products. She is always connected to the internet through multiple devices and regularly shops on ecommerce portals. User Story: "As an online grocery shopper Janine, I should be able to save and view my draft order from any of my devices so that I can complete the order process at my convenience."

Acceptance Criteria:

- Every in-progress order is saved every five seconds to the logged-in user account as a draft order
- New draft orders show up as notifications on any devices the user logs in

It is important for a Product Owner to note that User Stories that fulfill most, but not all, Acceptance Criteria cannot be accepted as Done. Scrum projects operate in Time-boxed Sprints, with a dedicated Sprint Backlog for each Sprint. Often, the last bit of work might be the most complicated part of a User Story and might take longer than expected. If incomplete User Stories were given partial credit for being Done and carried over to the next Sprint, then the progress of the subsequent Sprint could be disrupted. Therefore, the Done status is black and white. A User Story can only be either Done or not Done.

5.3.2 Definition of Ready

The Definition of Ready is a set of rules or criteria applicable to each User Story in the Prioritized Product Backlog. A User Story must satisfy the Definition of Ready before being considered for estimation and inclusion into a Sprint. The Definition of Ready puts the onus on the Product Owner to properly define requirements for each User Story. Without properly defined requirements, it will be impossible to get reliable estimates, and the Scrum Team may not be able to effectively complete the required project work.

The Scrum Guidance Body should preferably define the Definition of Ready criteria. However, there can also be project- or organizational-specific Definition of Ready criteria that will need to be added or updated. There may also be additions or updates to the Definition of Ready from the Scrum Team.

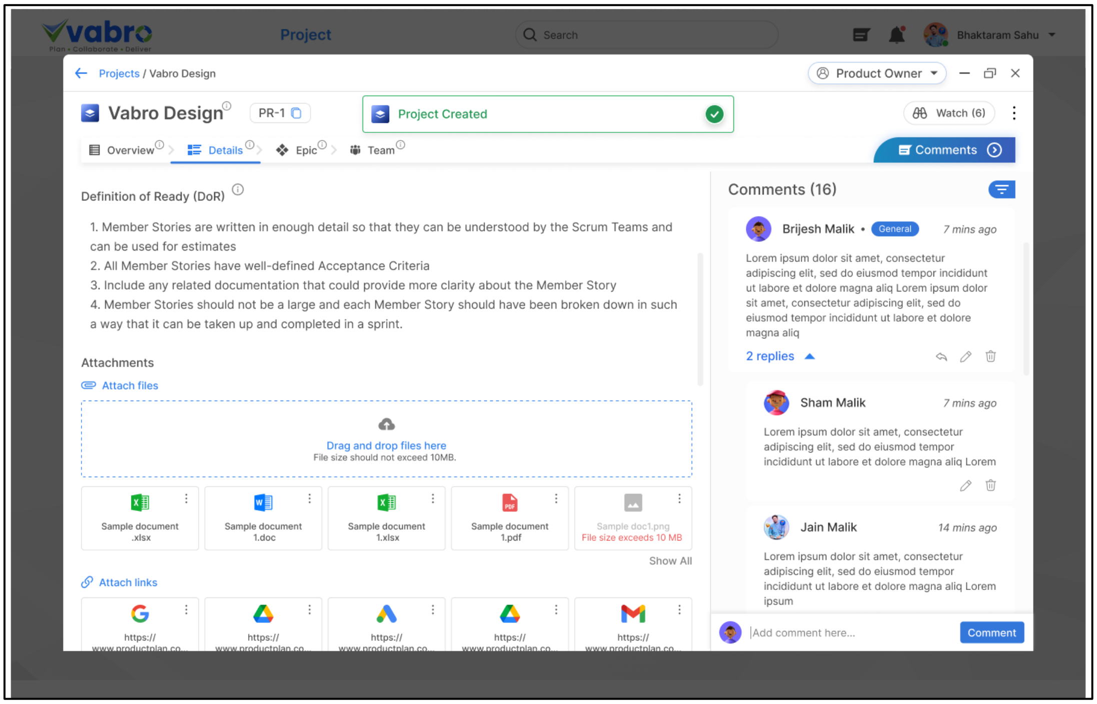
The Scrum Team commits to work on those User Stories which satisfy the Definition of Ready criteria. Review of product backlog items against the Definition of Ready criteria is a continuous activity that takes place as part of the *Refine Prioritized Product Backlog* process.

Some of the Definition of Ready criteria are:

- User Stories are written in enough detail so that they are understood by the Scrum Team and can be used for estimating

- User Stories have well-defined Acceptance Criteria
- Any related documentation that provides clarity about the User Stories are included
- User Stories are broken down to be small enough to be completed in a single Sprint.

Figure 5-2 is an interface from Vabro, showing a sample Definition of Ready (DoR) with criteria ensuring backlog items are fully prepared for sprint planning.



Definition of Ready (DoR)

1. Member Stories are written in enough detail so that they can be understood by the Scrum Teams and can be used for estimates
2. All Member Stories have well-defined Acceptance Criteria
3. Include any related documentation that could provide more clarity about the Member Story
4. Member Stories should not be a large and each Member Story should have been broken down in such a way that it can be taken up and completed in a sprint.

Attachments

Attach files
Drag and drop files here
File size should not exceed 10MB.
Sample document .xlsx
Sample document 1.doc
Sample document 1.xlsx
Sample document 1.pdf
Sample doc1.png
File size exceeds 10 MB
Show All
Attach links
https:// www.productplan.co...
https:// www.productplan.co...
https:// www.productplan.co...
https:// www.productplan.co...
https:// www.productplan.co...

Comments (16)

Brijesh Malik • General 7 mins ago

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq

2 replies

Sham Malik 7 mins ago

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq Lorem

Jain Malik 14 mins ago

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq Lorem ipsum

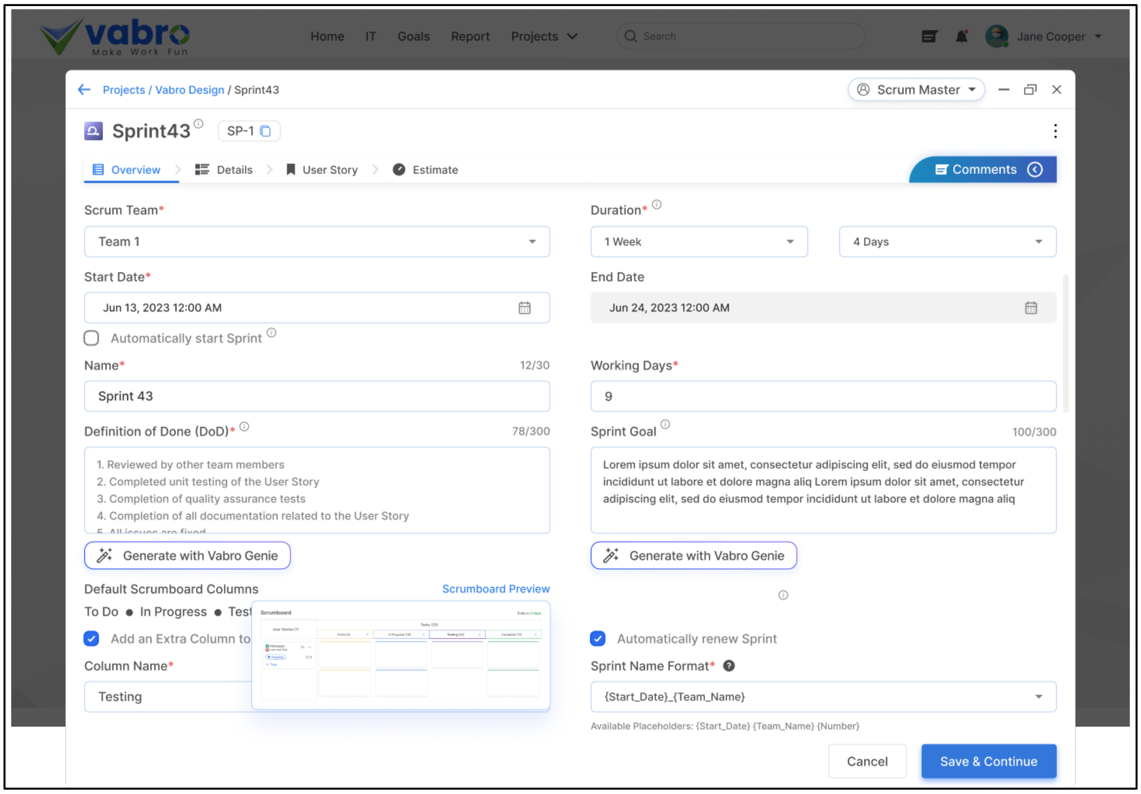
Figure 5-2: Sample DoR presented in an AI-powered Scrum Project Tool (Source: Vabro)

5.3.3 Definition of Done (or Done Criteria)

While Acceptance Criteria and the Definition of Ready are unique for individual User Stories, Done Criteria are a set of rules that are applicable to all the User Stories in a given Sprint. General Done Criteria could include any of the following:

- Reviewed by other team members
- Completed unit testing of the User Story
- Completion of quality assurance tests
- Completion of all documentation related to the User Story
- All issues are fixed
- Successful demonstration to business stakeholders and/or business representatives

Figure 5-3 is an interface from Vabro, showing a sample Definition of Done (DoD), including checklist items to ensure task completion, testing, and review.



Scrum Team*: Team 1

Duration*: 1 Week | 4 Days

Start Date*: Jun 13, 2023 12:00 AM

End Date: Jun 24, 2023 12:00 AM

□ Automatically start Sprint

Name* (12/30): Sprint 43

Working Days*: 9

Definition of Done (DoD)* (78/300):

1. Reviewed by other team members
2. Completed unit testing of the User Story
3. Completion of quality assurance tests
4. Completion of all documentation related to the User Story
5. All issues are fixed

Sprint Goal (100/300): Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliq

Default Scrumboard Columns

To Do ● In Progress ● Test

☑ Add an Extra Column to

☑ Automatically renew Sprint

Column Name*: Testing

Sprint Name Format*: {Start_Date}_{Team_Name}

Available Placeholders: {Start_Date} {Team_Name} {Number}

Figure 5-3: : Sample DoD presented in an AI-powered Scrum Project Tool (Source: Vabro)

As with the Acceptance Criteria, all conditions of the Done Criteria must be satisfied for each User Story to be considered Done. The Scrum Team should use a checklist of the general Done Criteria to ensure a task is finished and the result meets the Definition of Done (DoD). A clear Definition of Done is critical because it helps remove ambiguity and allows the team to adhere to required quality norms.

The Definition of Done (or the Done Criteria) is typically determined and documented by the Scrum Guidance Body. However, there can be project- or organization- specific Done Criteria that may need to be added or updated. There may also be additions or updates to the Done Criteria by the Scrum Team.

The required records and data to comply with the project's documentation requirements can be generated as the team proceeds through Sprints and Releases. The inclusion of activities such as holding review meetings and writing design documents can help ensure compliance with internal and external quality standards. The basic principles of Scrum such as short iterations, incremental building, customer involvement, adaptation to changing requirements, and constantly adjusting scope, time, and cost within the project will still apply.

5.3.4 Minimum Done Criteria

A higher-level business unit may announce mandatory minimum Done Criteria, which then become part of the Done Criteria for any User Story for that business unit. Any functionality defined by the business unit must satisfy these minimum Done Criteria if it is to be accepted by the respective Product Owner. The introduction of these Done Criteria may lead to a cascading set of Done Criteria for the portfolio, program, and project (see Table 5-1). Thus, the Done Criteria for a User Story in a project will implicitly include all the minimum Done Criteria from the higher levels, as applicable.

Portfolio Product Owner	• Sets the minimum Done Criteria for the entire portfolio • Reviews portfolio deliverables
Program Product Owner	• Sets the minimum Done Criteria for the entire program, which includes the Done Criteria from the portfolio • Reviews program deliverables
Chief Product Owner/ Product Owner	• Sets the minimum Done Criteria for the project, which includes the Done Criteria from the program • Reviews project deliverables

Table 5-1: Cascading Done Criteria

Once the minimum Done Criteria are defined, such criteria may then be documented in the Scrum Guidance Body documents and referred to by Scrum Teams as required.

5.3.5 Acceptance or Rejection of Prioritized Product Backlog Items

Toward the end of any iteration, the respective business unit and business stakeholders participate in a Sprint Review Meeting in which the product increment is demonstrated to the Product Owner, sponsor, customer, and users. While feedback from all the business stakeholders is gathered, only the Product Owner has the power to accept or reject a particular User Story as Done, according to the agreed upon Acceptance Criteria. Thus, the role of Acceptance Criteria in maintaining quality is critical and needs to be clearly understood by the team. It is the responsibility of the Scrum Master to ensure that the Acceptance Criteria for a User Story are not changed by the Product Owner in the middle of a Sprint. Partially completed User Stories are rejected as not Done and moved back into the Prioritized Product Backlog.

Figure 5-4 is an interface from Vabro, showing the user story approval process, detailing the steps for accepting or rejecting stories based on stakeholder validation.

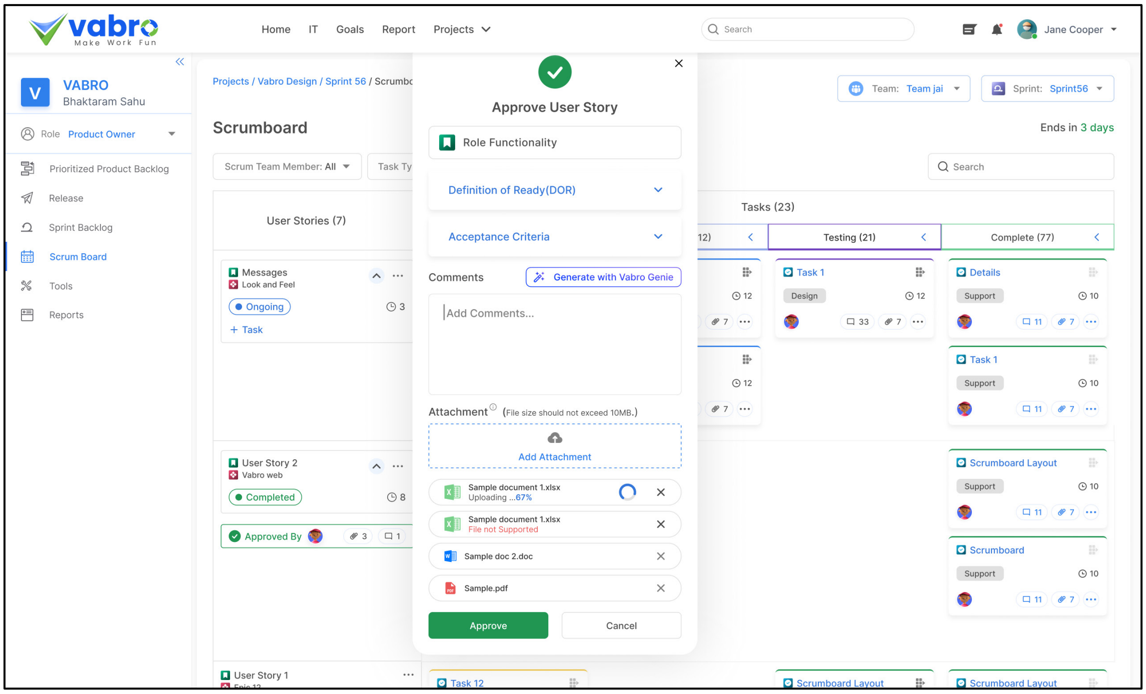


Figure 5-4: User Story Approval Process in an AI-powered Scrum Project Tool (Source: Vabro)

5.4 Quality Management in Scrum

The customer is the most important stakeholder in any project. Therefore, it is important to understand the customer's needs and requirements. The Voice of the Customer (VOC) can be referred to as the explicit and implicit requirements of the customer, which must be understood prior to the designing of a product or service. In a Scrum environment, the Product Owner's focus is on business requirements and objectives, which together represent the Voice of the Customer. The Product Owner can benefit from the guidance available from the Scrum Guidance Body (either through quality documents or standards, or from quality experts). These specialists should work with the Product Owner and the customer to ensure the appropriate level of detail and information in the User Stories, since User Stories are the basis for the success of any Scrum project.

It should be noted that external business stakeholders are not directly involved at the Scrum Team level and, instead, interact primarily with the Product Owner. For any Scrum project, the customer may be either of the following:

- Internal (that is, within the same organization)
- External (that is, outside the organization)

Quality management in Scrum enables customers to become aware of any problems in the project early and helps them recognize if a project is going to work for them or not. In Scrum, quality is about customer satisfaction and a working product, not necessarily meeting arbitrary metrics. This distinction becomes especially important from the customer's point of view because they are the ones investing time and money in the project.

Quality management in Scrum is facilitated through three interrelated activities:

1. Quality Planning
2. Quality Control
3. Quality Assurance

5.4.1 Quality Planning

One of the guiding principles of Scrum is to develop the functionality of the highest priority to the customer first. Less key features are developed in subsequent Sprints or can be left out altogether according to the customer's requirements. This approach gives the Scrum Team the required time to focus on the quality of essential functionality. A key benefit of quality planning is the reduction of technical debt. Technical debt—also referred to as design debt or code debt—refers to the work that teams prioritize lower, omit, or do not complete as they work toward creating the primary deliverables associated with the project's product. Technical debt accrues and must be paid in the future.

Some causes of technical debt can include the following:

- Quick-fix and building deliverables that do not comply with standards for quality, security, long-term architecture goals, etc.
- Inadequate or incomplete testing
- Improper or incomplete documentation

- Lack of coordination among different team members, or if different Scrum Teams start working in isolation, with less focus on final integration of components required to make a project or program successful
- Poor sharing of business knowledge and process knowledge among the business stakeholders and project teams
- Too much focus on short-term project goals instead of the long-term objectives of the organization. This oversight can result in poor-quality Working Deliverables that incur significant maintenance and upgrading costs.

In Scrum projects, any technical debt is not carried over beyond a Sprint, because there should be clearly defined Acceptance and Done Criteria. The functionality must satisfy these criteria to be considered Done. As the Prioritized Product Backlog is refined and User Stories are prioritized, the team creates Working Deliverables regularly, preventing the accumulation of significant technical debt. The Scrum Guidance Body may also include documentation and definition of processes which help in decreasing technical debt.

To maintain a minimal amount of technical debt, it is important to define the product required from a Sprint and the project along with the Acceptance Criteria, any development methods to be followed, and the key responsibilities of Scrum Team members in regard to quality. Defining Acceptance Criteria is an important part of quality planning, and it allows for effective quality control to be carried out during the project.

Technical debt is a very big challenge with some traditional project management techniques where development, testing, documentation, and so on are done sequentially and often-times by different people, with no one person being responsible for any particular Working Deliverable. As a result, technical debt accrues, leading to significantly higher maintenance, integration, and product release costs in the final stages of a project's release. Also, the cost of changes is extremely high in such circumstances as problems surface in later stages of the project. Scrum framework prevents the issues related to technical debt by ensuring that Done deliverables with Acceptance Criteria are defined as part of the Sprint Backlog and key tasks including development, testing, and documentation are done as part of the same Sprint and by the same Scrum Team.

5.4.1.1 Continuous Integration and Sustainable Pace

Maintaining a sustainable pace is an important tenet of Scrum. Sustainable pace translates to increased employee satisfaction, stability, and increased estimation accuracy, all of which leads to increased customer satisfaction. To develop a truly high-quality product and maintain a healthy work environment, it is important to carry out integration-type activities regularly, rather than delaying the integration work until the end of the project. To provide value at frequent intervals, the team should continuously develop, test, and integrate the functionalities of each Prioritized Product Backlog Item (PBI) in every Sprint with the use of techniques, such as continuous integration and automated product testing. It is also important, from the team's point of view, to ensure that the effort expended in the current Sprint is similar to the effort spent in the preceding Sprint in order to sustain an even pace throughout the project Sprints. This helps the team avoid phases of intense periods of work, ensuring they are always able to put forth the level of effort required to accomplish the work that needs to be done. Maintaining a sustainable pace is one of the most important tenets of Scrum and other Agile practices such as DevOps.

5.4.2 Quality Assurance and Quality Control

Quality is required not only in products, but also in processes. Quality assurance refers to the evaluation of processes and standards that govern quality management in a project to ensure that they continue to be relevant. Quality assurance activities are carried out as part of the work. In fact, quality assurance is a significant factor of the Definition of Done. The deliverable is not complete if appropriate quality assurance has not been conducted. Often, quality assurance is demonstrated during the Sprint Review Meeting.

Product Owners for respective projects, programs, and portfolios can monitor and evaluate quality assurance activities to ensure each team continues to agree and comply with the quality standards that have been set. End-to-end quality assurance may be addressed during final testing of the product, a Release, or a Sprint. A comparison of the number of issues encountered versus the number of User Stories completed can be done. The product components that have defects can be incorporated as Prioritized Product Backlog Items (PBIs), which can be worked upon by either the team or by one person at certain times during the Sprint, depending on the number of defects.

Quality control refers to the execution of the planned quality activities by the Scrum Team in the process of creating deliverables that are potentially shippable. It also includes learning from each set of completed activities in order to achieve continuous improvement. Within the cross-functional team, it is important to have the skills necessary to perform quality control activities. During the Sprint Retrospect Meeting, team members discuss lessons learned. These lessons act as inputs into continuous improvement and contribute to the improvement of ongoing quality control.

At times, the Scrum Guidance Body can define the processes and documents that can be referred to by Scrum Teams when doing their projects to ensure that uniform quality norms are followed by all projects within the organization.

5.4.3 Plan-Do-Check-Act (PDCA) Cycle

The Plan-Do-Check-Act Cycle—also known as the Deming or Shewhart Cycle—was developed by Dr. W. Edwards Deming, considered the father of modern quality control and Dr. Walter A. Shewhart. The following are some important points of Deming's philosophy:

- Management guidelines define quality. When management is able to provide a conducive environment and is able to motivate its employees to improve quality on a continuous basis, each employee will be able to contribute toward a superior quality product. Deming's "Theory of Profound Knowledge" advocates what management should do in order to create an environment in which each employee can make significant contributions to quality improvement.

Deming modified the Plan-Do-Check-Act to Plan-Do-Study-Act (PDSA) because he felt the term 'Study' emphasized analysis rather than simply inspection, as implied by the term 'Check.'

Both Scrum and the Deming/Shewhart/PDCA Cycle are iterative methods that focus on continuous improvement.

Figure 5-5 presents the PDCA Cycle in Scrum, linking Plan-Do-Check-Act stages with Scrum ceremonies like planning, review, and retrospective for continuous improvement.

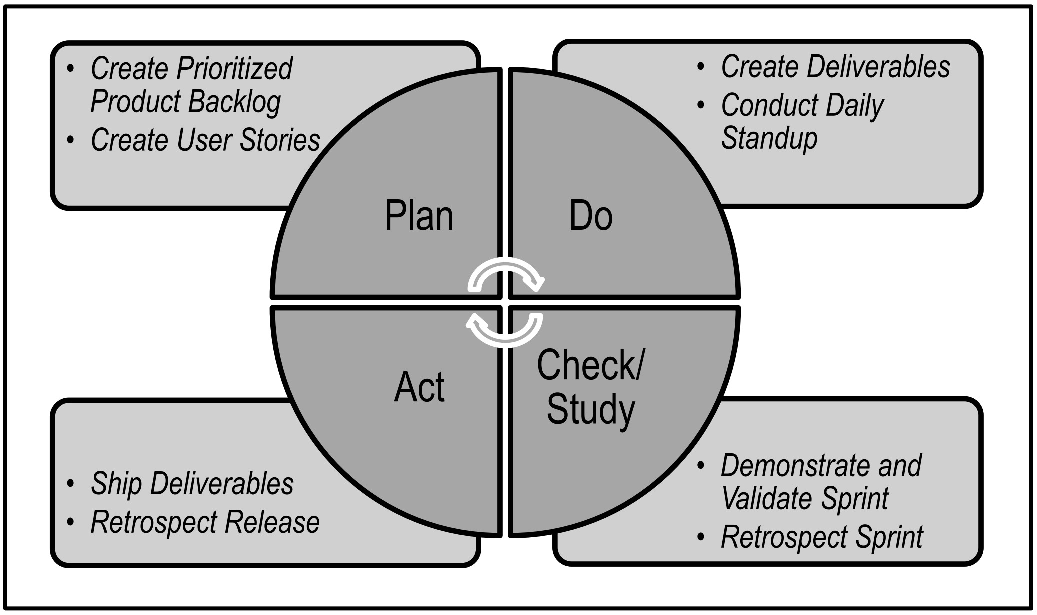


Figure 5-5: PDCA Cycle in Scrum

5.5 Summary of Responsibilities

Role	Responsibilities
Scrum Team	• Develops and maintains all deliverables during Sprints until they are handed over to end users • Practices and encourages effective communication so that requirements are clarified and fully understood • Shares knowledge to ensure that team members are familiar with the entire feature set and benefit from the experience of others • Makes appropriate changes to deliverables swiftly • Complies with the Definition of Done criteria for each deliverable
Product Owner/ Chief Product Owner	• States the business requirements for the product and defines requirements clearly in the Prioritized Product Backlog • Assesses viability and ensures that deliverables meet the quality requirements • Sets the minimum Done Criteria for the entire project, including the Acceptance Criteria for the respective projects • Facilitates creation of Acceptance Criteria for User Stories • Reviews and validates the deliverables during *Demonstrate and Validate Sprint*
Scrum Master/ Chief Scrum Master	• Facilitates a “team first” mentality when it comes to quality • Eliminates environmental obstructions that may affect the quality of deliverables and processes • Ensures that a sustainable pace is maintained in which the focus is on quality of features rather than strictly on velocity • Ensures that Scrum processes are correctly followed by all team members, including the Product Owner
Program Product Owner	• Sets the minimum Done Criteria for the entire program • Reviews program deliverables
Program Scrum Master	• Ensures that a sustainable pace is maintained in which the focus is on quality of features rather than strictly on velocity
Portfolio Product Owner	• Sets minimum Done Criteria for the entire portfolio • Reviews portfolio deliverables
Portfolio Scrum Master	• Ensures that a sustainable pace is maintained in which the focus is on quality of features rather than strictly on velocity
Business Stakeholder(s)	• Review and provide feedback about product deliverables • Work collaboratively with the Product Owner and the Scrum Team
Scrum Guidance Body	• Provides the Definition of Ready and the Definition of Done • Provides the framework and guidance for developing the Acceptance Criteria • Defines the range of tools that can be used by the Scrum Team to develop and verify the product components

Table 5-2: Summary of Responsibilities Relevant to Quality

5.6 Scrum vs. Traditional Project Management

Although there are similarities in Scrum and traditional project management methods with regard to definition of ‘quality’ (i.e., the ability of the product to meet the agreed Acceptance Criteria and achieve the business value expected by the customer), differences exist in terms of how the approaches address the implementation and achievement of the required quality levels.

In traditional project management methods, the users clarify their expectations; the project manager defines those expectations in measurable terms and gains agreement from the users. After detailed planning, the project team develops the product over an agreed period of time. If any of the agreed criteria are to be changed, changes can happen only through a formal change management system where impact of changes is estimated and the Project Manager gets approval from all relevant business stakeholders.

In Scrum, however, the Product Owner collaborates with the Scrum Team and defines the Acceptance Criteria for the User Stories related to the product to be delivered. The Scrum Team then develops the product in a series of short iterations called Sprints. The Product Owner can make changes to the requirements to keep pace with the user needs and these changes can be addressed by the Scrum Team either by terminating the current Sprint or including the adjusted requirements in the next Sprint as each Sprint is of short duration (i.e., one to four weeks).

One of the major advantages of Scrum is the emphasis on creating potentially shippable deliverables at the end of each Sprint cycle, instead of at the end of the entire project. So, the Product Owner and customers constantly inspect, approve, and accept deliverables after each Sprint. Also, even if a Scrum project is terminated early, there is some value created prior to termination through the deliverables created in individual Sprints.

The Essential Guide to Successfully Deliver Projects using Scrum

A Guide to the Scrum Body of Knowledge (SBOK® Guide) provides comprehensive guidelines for the successful implementation of Scrum—the most popular Agile product development and project delivery approach. Defined in the SBOK® Guide as a flexible framework, Scrum can be applied to portfolios, programs, or projects of any size or complexity across industries to deliver products, services, or other results.

This Fifth Edition is based on the collective knowledge gained from thousands of projects across diverse organizations and industries. It reflects contributions from a large number of experts in Scrum and project delivery. Feedback from the global Scrum community played a vital role in shaping improvements and additions, making the SBOK® Guide a truly collaborative effort.

Unlike other Scrum references, the SBOK® Guide is available for free on Scrumstudy.com, along with free certifications, webinars, videos, and study guides. It is ideal for professionals seeking a foundational understanding of Business Analysis or exploring careers in related fields. The Guide addresses real-life challenges faced by Scrum practitioners and explains how to solve them using modern tools and Artificial Intelligence (AI).

Designed to be accessible and engaging, the SBOK® Guide follows the 80-20 rule—80% of key concepts can be learned by reading just 20% of the content, with the remainder available for deeper reference. It is more readable than most Scrum books, which are often either too simplistic or overly detailed.

The SBOK® Guide serves as a reference for both experienced practitioners and those with no prior knowledge of Scrum or project delivery methods. Organized for easy navigation, the SBOK® Guide aims to inform, support, and inspire all readers through its rich, collaborative content.

SCRUMstudy
Targeting success

SCRUMstudy™ is a brand of VMEdu, Inc.