AMIT GROUP TRANFORM DIGI TOGETHER	**AMIT Group Joint Stock Company**	**Code** **Revision**	AP.EN.004 01
		Issue time **Number of pages**	09/27/2022 42

AGILE SCRUM

PROJECT MANAGEMENT AND IMPLEMENTATION PROCESS

Prepared by	Team AGILE PROCESS: - Dang Nguyen Bao Duy - Nguyen Thi Kieu Nga - Dinh Phi Long - Dang Thi Cam Tien	Date	09/22/2022
Reviewed by	Tran Nhat Anh	Date	09/22/2022
Approved by	Tran Nhat Anh	Date	09/27/2022

Table of Content

Table of Figure

Table of Form

1. PURPOSE

This document describes in detail the steps in the process of implementing and managing projects according to the AGILE SCRUM method. With towards and achieving the following purposes:

- Initializing a process of implementing and managing projects according to the AGILE SCRUM method officially applied to AMIT GROUP company.
- Creating a set of templates for the process.
- Supporting training to understand and apply the process to future projects for employees.
- Coordinating with the Marketing Department to plan related promotions.

2. SCOPE

This document is applicable to all employees who are working orinterning at AMIT GROUP Joint Stock Company. Consisting of:

- All PM, QA, BA, Design, Dev, and QC are required.
- All staff in other departments refer for appropriate preparation if needed.

All stakeholders are informed about the process and are required to understand, comply with the content in this document.

3. REFERENCES

- ***A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™GUIDE) Third Edition***, SCRUMstudy, accessed on 05/20/2022.
- ***Essential Scrum - A Practical Guide to the most popular Agile Process*** *by* ***Kenneth S.Rubin.***

And many other online articles about AGILE SCRUM.

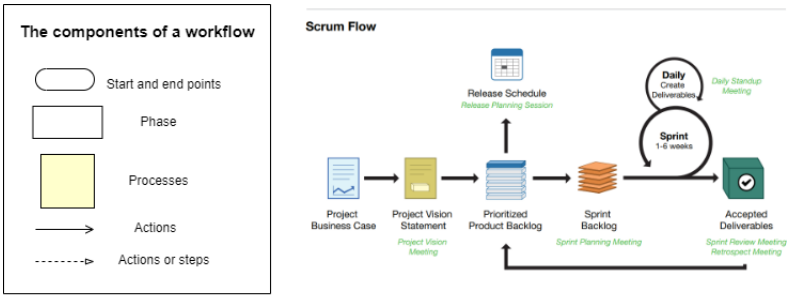
4. DEFINITION

Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster and with fewer headaches. Instead of betting everything on a "big bang" launch, an agile team delivers work in small, but consumable, increments.

Scrum is one of the most popular Agile methods. It is an adaptive, iterative, fast, flexible, and effective framework designed to deliver significant value quickly and throughout a project.

5. CONTENT

Important phases, subprocesses/steps are described here.



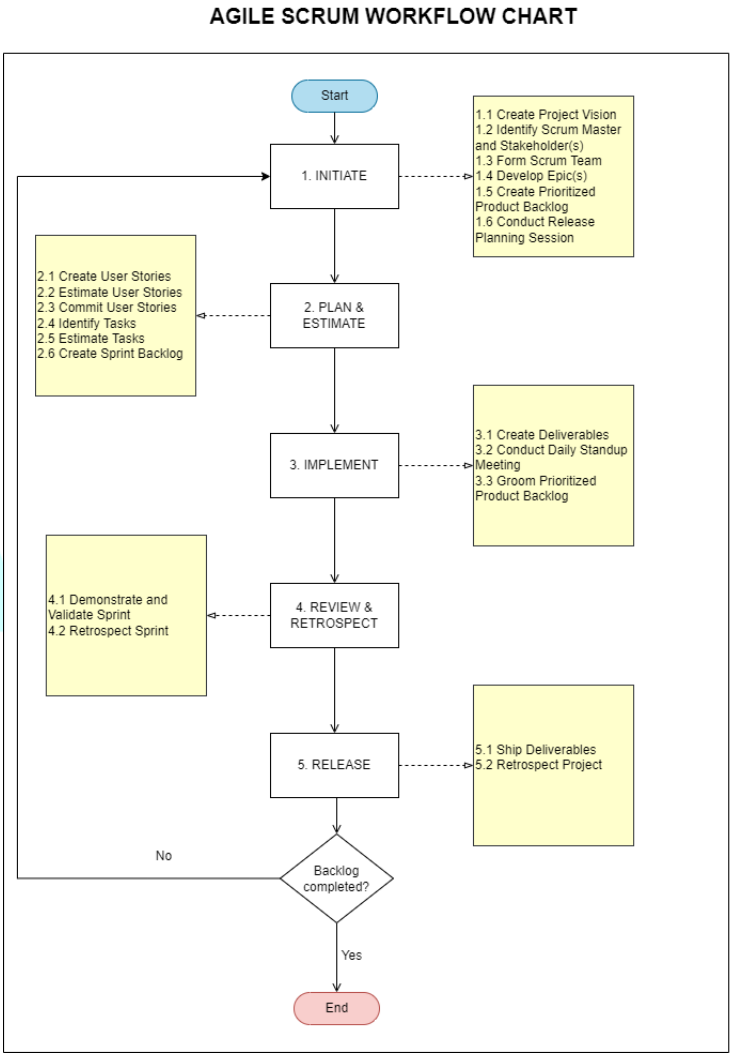


Figure 5.1 AGILE SCRUM Workflow chart

5.1 Initiate

Overview

The first phase in the Agile Scrum process is the Initiate, which includes the processes related to the initiation of a project: Create Project Vision, Identify Scrum Master and Stakeholder(s), Form Scrum Team, Develop Epic(s), Create Prioritized Product Backlog, and Conduct Release Planning.

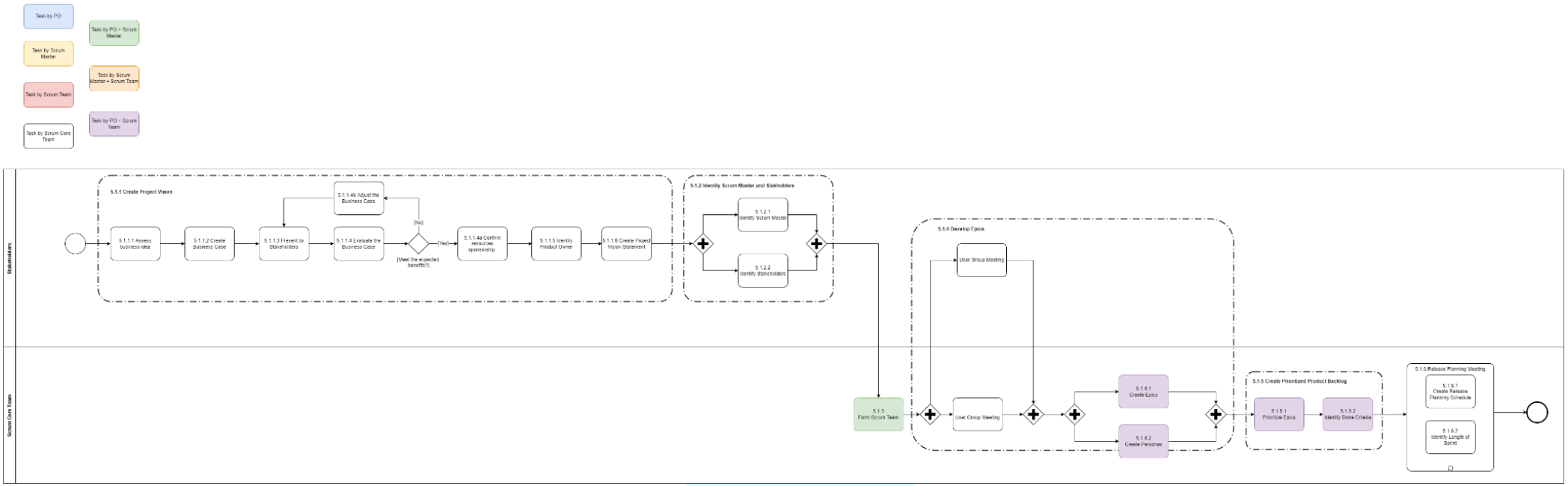


Figure 5.1.1 Initiate Phase

5.1.1 Create Project Vision

In this process, the Project Business Case is reviewed to create a Project Vision Statement that will serve as the inspiration and provide a focus for the entire project. The Product Owner is identified in this process.

5.1.1.1 Assess the business idea

With product development projects, assessing the business ideal is necessary and the first step to do when you want to develop any product. Conduct research and analysis related to your business idea, as the basis for creating a Project Business Case.

For outsourcing projects, try to understand their business as well as the problems that customers are facing, as a foundation to create a Project Business Case.

Usually, senior managers, project managers, R&D teams or maybe Business Analysts will be responsible for this. It can be as simple as searching the internet, researching specialized documents, reading relevant articles, ... conducting surveys, collecting user data, or even buying data to support analysis.

5.1.1.2 Create Project Business Case

Project Business Case often includes substantial information on the background of the project, the intended business purpose and desired outcomes, a SWOT and Gap analysis report, a list of identified risks, and estimations of time, effort, and cost (may briefly be described in the document Project Business Case). It may be a well-structured document or simply a verbal statement that expresses the rationale for initiating a project. It may be formal and comprehensive, or informal and brief.

The project sponsor prepares the business case in cooperation with team members and subject matter experts from the applicable areas, such as IT or finance. Some companies may have dedicated project management offices. In that case, the project management office prepares the business case. If an outside organization requests support, that entity prepares the business case. The project sponsor and interested parties review the business case. Based on the business case, the project may be approved, rejected, altered, or postponed. In addition, the Business Analyst (BA) can also assist in creating this document.

Related forms: AF.AS-01 [AMIT] Project Business Case Template

5.1.1.3 Present to Stakeholders

The business case is presented to the stakeholders and sponsors, usually Senior Managers or Project Manager, (possibly Business Analyst) will be responsible for this activity. The stakeholders understand the expected business benefits of the project and the sponsors confirm that they will provide the financial resources for the project.

It could be a short workshop to present a business case project or simply send them a document they will review and respond to later.

5.1.1.4 Evaluate the Business Case

Stakeholders review and evaluate Project Business Case:

- If it meets their expected benefits, they can confirm resource sponsorship.
- If it doesn't meet their expected benefits, they can adjust the business case and present it again for them to evaluate, until their expectations are met.

5.1.1.5 Identify Product Owner

Identifying Product Owner of the project. The Product Owner represents the interests of the stakeholder community to the Scrum Team. The Product Owner is responsible for ensuring clear communication of product or service functionality requirements to the Scrum Team, defining Acceptance Criteria, and ensuring those criteria are met. In other words, the Product Owner is responsible for ensuring that the Scrum Team delivers value.

Senior managers or project managers will assign the Product Owner for the project.

The Product Owner must always maintain a dual view. He or she must understand and support the needs and interests of all stakeholders, while also understanding the needs and workings of the Scrum Team. Because the Product Owner must understand the needs and priorities of the stakeholders, including customers and users, this role is commonly referred to as the Voice of the Customer.

5.1.1.6 Create Project Vision

Project vision explains the intended business needs for the project.

The Product Owner will define the Project Vision.

A Project Vision Meeting will be held with stakeholders and the Product Owner. It helps identify the business context, business requirements, and stakeholder expectations in order to develop an effective Project Vision Statement.

The Project Vision Statement should not be too specific and should have room for flexibility. It's possible that the current understanding of the project may be based on assumptions that will change as the project progresses, so it is important that the project vision is flexible enough to accommodate these changes. The project vision should focus on the problem rather than the solution. As well as explain the business needs the project is intended to meet rather than how it will meet the need.

Related forms: AF.AS-02 [AMIT] Project Vision Statement Template

5.1.2 Identify Scrum Master and Stakeholder(s)

The Scrum Master and Stakeholders are identified using specific Selection Criteria.

5.1.2.1 Identify Scrum Master

A Scrum Master is a facilitator and 'servant leader' who ensures that the Scrum Team is provided with an environment conducive to completing the project successfully. The Scrum Master guides, facilitates, and teaches Scrum practices to everyone involved in the project; clears impediments for the team; and, ensures that Scrum processes are being followed.

When there is flexibility in choosing the Scrum Master(s), the following are important Selection Criteria:

- Problem-solving skills: This is one of the primary criteria to be considered while selecting Scrum Master(s). The Scrum Master(s) should have the necessary skills and experience to help remove any impediments for the Scrum Team.
- Availability: The Scrum Master should be available to schedule, oversee, and facilitate various meetings, including the Release Planning Meeting, Daily Standup Meeting, and other Sprint-related meetings.
- Commitment: The Scrum Master should be highly committed to ensuring that the Scrum Team is provided with a conducive work environment to ensure the successful delivery of Scrum projects.
- Servant Leadership Style: Servant leaders employ listening, empathy, commitment, and insight while sharing power and authority with team members. Servant leaders are stewards who achieve results by focusing on the needs of the team.

The Product Owner will be responsible for defining the Scrum Master for the project, refer to Project Vision Statement for the appropriate selection.

Besides Selection Criteria, the Product Owner also should consider the **Resource Costs** factor when defining the Scrum Master. One of the primary considerations in selecting people has to do with the trade-offs related to experience versus salary. There are other people elated factors impacting cost that may also need to be considered.

Ideally, the Scrum Master(s), team members, and Stakeholder(s) should be colocated, so that they can communicate frequently and easily. If colocation is not possible and there are distributed teams, additional resources will have to be devoted to facilitate communications, understand cultural differences, synchronize work, and foster knowledge sharing.

5.1.2.2 Identify Stakeholders

Stakeholder(s), which is a collective term that includes customers, users, and sponsors, frequently interface with the Scrum Core Team and influence the project throughout the product development process. It is for the stakeholders that the project produces collaborative benefits.

When identifying the Stakeholder(s), it is important to remember that stakeholders are all the customers, users, and sponsors, who frequently interface with the Product Owner, Scrum Master, and Scrum Team to provide inputs and facilitate the creation of the project's products. The stakeholders influence the project throughout its lifecycle.

Customer

The customer is the individual or the organization that acquires the project's product, service, or other result. For any organization, depending on the project, there can be both internal customers (i.e., within the same organization) or external customers (i.e., outside of the organization).

Users

Users are the individual or the organization that directly uses the project's product, service, or other result. Like customers, for any organization, there can be both internal and external users. Also, in some industries customers and users may be the same.

Sponsor

The sponsor is the individual or the organization that provides resources and support for the project. The sponsor is also the stakeholder to whom everyone is accountable in the end.

The Product Owner will be responsible for Identifying Stakeholder(s).

Additionally, **Expert Advice from Human Resource** managers can be valuable in identifying the Scrum Master and the Stakeholder(s). The HR department possesses specialized knowledge about the organization and various techniques that might help in identifying the Scrum Master and Stakeholder(s).

5.1.3 Form Scrum Team

In this process, Scrum Team members are identified. Normally the Product Owner has the primary responsibility of selecting team members, but often does so in collaboration with the Scrum Master. They refer to the Project Vision Statement for proper selection.

Scrum Team members are generalists/specialists in that they have knowledge of various fields and are experts in at least one. Beyond their subject-matter expertise, it is the soft skills of team members that determine the success of self-organizing teams. Ideal members of the Scrum Team are independent, self-motivated, customer-focused, responsible, and collaborative. The team should be able to foster an environment of independent thinking and group decision making in order to extract the most benefits from the structure.

Personnel Selection

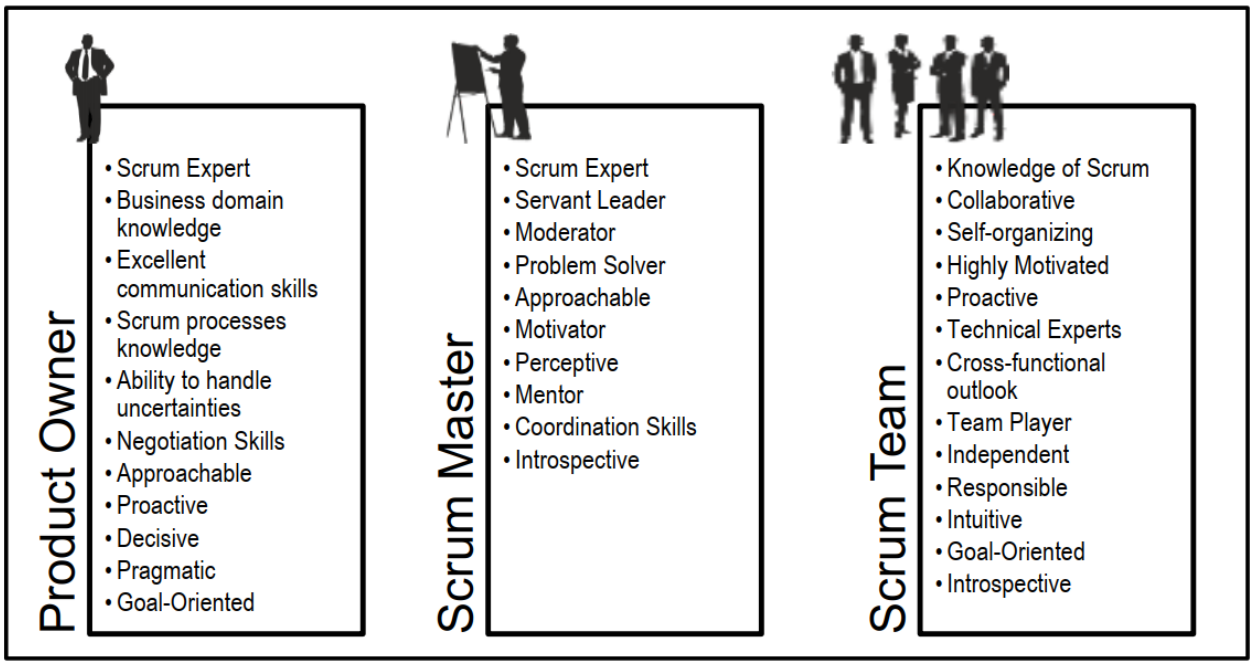


Figure 5.1.2 Personnel selection

Besides, **Expert Advice from Human Resource (HR)** managers can be valuable while forming a Scrum Team. The HR department possesses specialized knowledge about the employees of an organization and of

the numerous techniques that might help Product Owners, Scrum Masters, and sponsors identify the right team members.

Moreover, also should consider the **Resource Costs** factor when defining the Scrum Team. The costs associated with all non-people requirements must be assessed, analyzed, approved, and budgeted for. A resource in the project environment is anything used to perform a task or activity including but not limited to equipment, material, outside services, and physical space.

Team size

The optimum size for a Scrum Team is six to ten members—large enough to ensure adequate skill sets, but small enough to collaborate easily. A key benefit of a six-to-ten-member team is that communication and management are typically simple and require minimal effort. However, there may also be drawbacks. One major drawback is that smaller teams are more significantly impacted by the loss of a team member than larger teams, even for a short period of time. To address this problem, it may be possible for team members to have expert knowledge and skills outside their own specific roles. However, this may be difficult and depends on the type of project, industry, and size of the organization. It is also recommended to have backup persons to replace any person who may have to leave the Scrum Team.

5.1.4 Develop Epic(s)

The Project Vision Statement serves as the basis for developing Epics. User Group Meetings may be held to discuss appropriate Epics.

5.1.4.1 Create Epics

Epics are a big chunk of work that has one common objective, high-level functionalities, customer requests, or business requirements are broadly defined. It is the final output of user needs and may not contain all the details to work on.

The Product Owner is the person responsible for creating the Epics.

User Group Meetings may be held to discuss appropriate Epics, User Group Meetings involve relevant stakeholders (primarily users or customers of the product). They provide the Scrum Core Team with firsthand information about user expectations. This helps in formulating the Acceptance Criteria for the product and provides valuable insights for developing Epics. It's vital in the prevention of expensive rework that may result from a lack of clarity regarding expectations and requirements. These meetings

also promote buy-in for the project and create a common understanding among the Scrum Core Team and relevant Stakeholder(s).

More meetings can be taken if the team is not clear enough to develop Epics:

- User Story Workshops: are held as part of the Develop Epic(s) process. The Scrum Master facilitates these sessions, where the entire Scrum Core Team is involved, and at times, it is desirable to include other Stakeholders. They ensure that the Epics and User Stories describe the functionality from the users' point of view, are easy to understand, and can be reliably estimated. User Story Workshops are useful in understanding user expectations for the deliverables and are excellent for team building. A User Story Workshop is a good platform to discuss and clarify every element of a product and often delve into the smallest details to ensure clarity.
- Focus Group Meetings: Focus groups assemble individuals in a guided session to provide their opinions, perceptions, or ratings of a product, service, or desired result. Focus group members have the freedom to ask questions to each other and to get clarification on particular subjects or concepts. Through questioning, constructive criticism, and feedback, focus groups lead to a better quality product and thereby contribute to meeting the expectations of the users. In these meetings, the focus group members sometimes reach consensus in certain areas, while in other areas their opinions may differ. Where group members have differing opinions or perspectives, every effort is made to resolve the differences in order to reach a consensus.

In addition, there are some other tools:

- User or Customer Interviews: Engaging stakeholders, including the sponsor, users, and customers of the product, is important to gain the necessary context and insight required to develop Epics. Quality time spent interviewing users and customers will result in ensuring that the requirements in Epics align with the overall Project Vision, thereby delivering greater value.
- Questionnaires: are a research instrument that contains questions to be asked to a respondent in order to collect information about a specific issue or topic. Questionnaires can be self-administered or administered by an interviewer. Great care must be exercised in the design of Questionnaires, selecting the right target audience, and determining an appropriate method of survey deployment to avoid errors and bias.

Approved Changes

Unapproved Change Requests may be approved by the Product Owner during the Develop Epic(s) process, at times with suggestions provided by relevant stakeholders. Such changes are categorized as Approved Changes and can be prioritized and implemented in future Sprints.

Change Requests will be treated as issues, and logged in the Impediment Log.

Related forms: AF.AS-06 [AMIT] Impediment Log Template

Identified Risks

When creating Epics, new risks may be identified and these risks contribute to the development of the Prioritized Product Backlog (also referred to as the Risk Adjusted Product Backlog). The Product Owner can make notes to manage these risks or update them in the Risk section in the **Project Business Case** document.

5.1.4.2 Create Personas

Personas are created to identify the needs of the target user base. Creating specific Personas can help the team better understand users and their requirements and goals. Based on a Persona, the Product Owner can more effectively prioritize features to create the Prioritized Product Backlog.

The Product Owner is responsible for creating it. This involves assigning a fictional name and preferably a picture, like a stock image, to the character.

Related forms: Figma tool

5.1.5 Create Prioritized Product Backlog

In this process, Epic(s) are refined, elaborated, and then prioritized to create a Prioritized Product Backlog for the project. The Done Criteria is also established at this point.

Related forms: AF.AS-03 [AMIT] Product Backlog Template

5.1.5.1 Prioritize Product Backlog

The Product Owner refines, elaborates, and then creates prioritized Epics in the Product Backlog. The Scrum Master helps Product Owner in the creation of the Prioritized Product Backlog. And assure Scrum Team understands the Epics/User Stories in the Prioritized Product Backlog. Referring to Personas can help the team better understand users and their requirements and goals.

Some techniques used to prioritize the Epics/User Stories or requirements in the Prioritized Product Backlog such as MoSCow, Paired Comparison, T-shirt, 100 Point.

Which is recommended to be used in this procedure is MoSCow. The MoSCoW prioritization scheme derives its name from the first letters of the phrases “Must have,” “Should have,” “Could have,” and “Won’t have” (Would). This prioritization method is generally more effective than simple schemes. The labels are in decreasing order of priority with “Must have” User Stories being those without which the product will have no value and “Won’t have” User Stories being those that, although they would be nice to have, are not necessary to be included.

Related forms: AF.AS-0301 [AMIT] Epic List

5.1.5.2 Create Done Criteria

Done Criteria are a set of rules that are applicable to all User Stories, it removes ambiguity from requirements and helps the team adhere to mandatory quality norms. A User Story is considered Done when it is demonstrated to and approved by the Product Owner who judges it on the basis of the Done Criteria and the User Story Acceptance Criteria.

The Product Owner is responsible for defining Done Criteria with the support of the Scrum Master. It is usually listed as a list and does not usually change much throughout the project.

Related forms: AF.AS-0302 [AMIT] Done Criteria

5.1.6 Conduct Release Planning

In this process, the Scrum Core Team reviews the Epics/User story in the Prioritized Product Backlog to develop a Release Planning Schedule, which is essentially a phased deployment schedule that can be shared with the project stakeholders. The Length of Sprints is also determined in this process.

5.1.6.1 Create Release Planning Schedule

A Release Planning Schedule states which deliverables are to be to the customers, along with planned intervals, and dates for releases. The goal of Release Planning is to when various sets of usable functionality or products will be delivered to the customer which to enable the Scrum Team to have an overview of the releases and delivery schedule for the product being developed.

The Product Owner is responsible for creating the Release Planning Schedule. The Scrum Master also collaborates to create it. By reviewing relevant documents such as Project Vision Statement, Prioritized Product Backlog, Done Criteria, and stakeholder expectations (primarily the sponsor). A Release Planning Meeting is held to enable the Scrum Team to have an overview of the releases and delivery schedule for

the product they are developing so that they can align with the expectations of the Product Owner and relevant stakeholders.

The plan defines when various sets of usable functionality or products will be delivered to the customer. There may not be a release scheduled at the end of every Sprint iteration. At times, a release may be planned after a group of Sprint iterations are completed. Depending on the organization's strategy, Release Planning sessions in projects may be driven by functionality, in which the objective is to deliver once a predetermined set of functionality has been developed, or the planning may be driven by date, in which the release happens on a predefined date. The deliverable should be released when it offers sufficient business value to the customer.

Release Planning Sessions need not produce a detailed Release Plan for the entire project. The Release Plan can be updated continually as relevant information is available.

Related forms: AF.AS-0303 [AMIT] Release Planning Schedule

5.1.6.2 Decide Length of Sprint

Length of Sprint is the duration of a sprint in a project. Based on the various inputs including business requirements and Release Planning Schedule, the Product Owner and the Scrum Team decide on the Length of Sprint for the project. Once determined, the Length of Sprint often remains the same throughout the project.

Sprints need to be short; no more than four weeks long in order to enable rapid feedback loops and timely production of increments. They must be just long enough for the Team to achieve their Sprint Goal and deliver something of value which is one reason different durations work better for different teams and organizations.

However, the Length of Sprint may be changed if and as the Product Owner and the Scrum Team deem appropriate. Early in the project they may still be experimenting to find the best Sprint length. Later in the project a change in the Length of Sprint normally means it can be reduced due to improvements in the project environment.

A Sprint could be Time-boxed from 1 to 6 weeks. However, to get maximum benefits from a Scrum project, it is always recommended to keep the Sprint Time-boxed to 4 weeks, unless there are projects with very stable requirements.

5.2 Plan & Estimate

Overview

The Plan and Estimate phase consists of processes related to planning and estimating tasks.

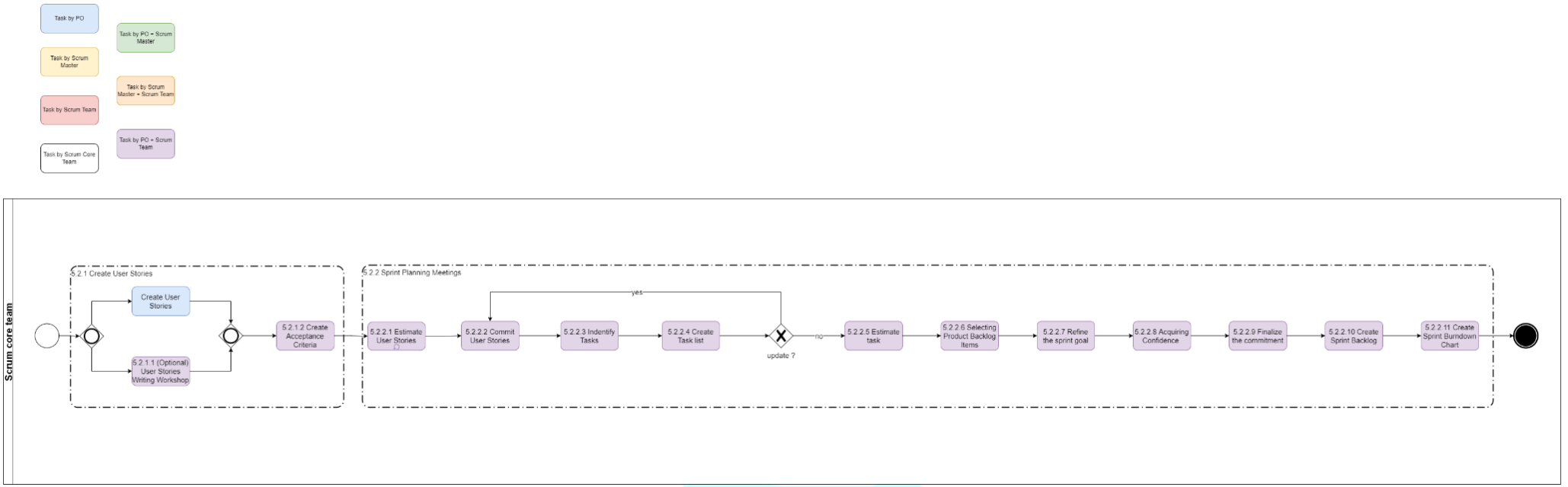


Figure 5.2.1 Plan & Estimate Phase

5.2.1 Create User Stories

In this process, User Stories and their related User Story Acceptance Criteria are created.

PO creates User Stories: User Stories are usually written by the Product Owner and are designed to ensure that the customer's requirements are clearly depicted and can be fully understood by all stakeholders.

The Product Owner, based on his or her interaction with the stakeholders, business knowledge and expertise, and inputs from the team, develops User Stories that will form the initial Prioritized Product Backlog for the project. At times, the Product Owner may bring a Business Analyst to assist with writing User Stories.

Related forms: AF.AS-03 [AMIT] Product Backlog Template

5.2.1.1 (optional) User Story Writing Workshops

User Story Writing Workshops may be held which involves Scrum Team members creating the User Stories. User Stories are incorporated into the Prioritized Product Backlog.

User Stories

User Stories adhere to a specific, predefined structure and are a simplistic way of documenting the requirements and desired end-user functionality. A User Story tells you three things about the requirement: Who, What, and Why. Some User Stories may be too large to handle within a single Sprint. These large User Stories are often called Epics. Once Epics come up in the Prioritized Product Backlog to be completed in an upcoming Sprint, they are further decomposed into smaller User Stories.

The Prioritized Product Backlog is a dynamic list that is continuously updated because of reprioritization and new, updated, refined, and sometimes, deleted User Stories.

User Story Format:

AS A [type of user], I NEED TO [do some task], SO THAT I CAN [get some result]

User Story Example:

As a credit card holder, I want to view my statement (or account) balance, so that I can pay the balance due.

Related forms: AF.AS-0304 User Story - Acceptance Criteria

5.2.1.2 Create User Story Acceptance Criteria

Every User Story has an associated Acceptance Criteria. The Acceptance Criteria provide the objectivity required for the User Story to be considered as Done or not Done during the Sprint Review.

Acceptance Criteria: provide clarity to the team on what is expected of a User Story, remove ambiguity from requirements, and help in aligning expectations.

The Product Owner defines and communicates the Acceptance Criteria to the Scrum Team. In the Sprint Review Meetings, the Acceptance Criteria provide the context for the Product Owner to decide if a User Story has been completed satisfactorily.

It is important and the responsibility of the Scrum Master to ensure that the Product Owner does not change the Acceptance Criteria of a committed User Story in the middle of a Sprint.

The format for writing User Story Acceptance Criteria is according to the checklist.

Example:

- Display statement balance upon authentication. Say for example $1560

- Display total balance. For example $3560. Here the balance due from the current period is $2560 and past balance due is $2000.

- Show Minimum payment due. For example $140

- Show Payment due date. For example May 16th of the current month

- Show error message if service does not respond or timeout. For example 'Sorry, something went wrong with the service. Please try again.'

The Prioritized Product Backlog is updated based on User Stories, Epic(s), estimates for User Stories, and User Story Acceptance Criteria.

Related forms: AF.AS-0304 User Story - Acceptance Criteria

5.2.2 Sprint Planning Meeting

This meeting is conducted prior to the Sprint as part of the Commit User Stories, Identify Tasks, Estimate Tasks, and Create Sprint Backlog processes. It is timeboxed to eight hours for a one-month Sprint, four hours for a two-week Sprint.

During Sprint Planning Meetings, the User Stories are taken up for discussion by the Scrum Core Team. If they have not already done during the Creation or the Grooming of the Product Backlog, each User Story is evaluated and assigned a high-level estimate based on relative story points.

From the created user stories, PO and Scrum Master with the Scrum core team clarify them one by one, after clarifying, they will continue to estimate each user story.

5.2.2.1 Estimate User Stories

The Product Owner clarifies User Stories in order for the Scrum Master and Scrum Team to estimate the effort required to develop the functionality described in each User Story.

After the User Stories are estimated by the Scrum Team using the various estimation techniques discussed in this section, they are considered to be Estimated User Stories. Relative sizing, or story points can be used for estimating the overall size of a User Story or feature.

This assessment will be conducted by the Scrum Team and a story point value will be assigned. Once an evaluation is done on one User Story in the Prioritized Product Backlog, the Scrum Team can then evaluate other User Stories relative to that first story. It should be noted that since the story point calibration for each team would be different, the number of story points completed could not be used for comparison across teams.

5.2.2.1.1 Determine Capacity

An important first activity during sprint planning is determining the available capacity of the team to perform work during the sprint. The development team begins by determining its capacity to complete work. Based on available capacity, the sprint goal may need to be refined.

Capacity in Story Points

Express capacity in story points, determining capacity is the same as predicting our team's target velocity for the upcoming sprint.

Capacity in Effort-Hours

An alternative way to express capacity is in effort-hours. The team members then determine how many hours per day they could dedicate to work in this sprint. Each person gives a range that takes into account any overhead work not associated with items in the sprint backlog.

Estimation methods

Numerous estimation methods can be used as tools to estimate User Stories.

Some important tools are: **Wideband Delphi, Planning Poker,** **Fist of Five, T shirts…**

Planning Poker

In Planning Poker, each team member is assigned a deck of cards. Each card is numbered in a sequence and the numbers represent complexity of the problem, in terms of time or effort, as estimated by the team member. The Scrum Team members assess the item (User Story or Task) to understand it better before providing their estimate for developing it. Then, each member picks a card from the deck that represents their estimate for the item. If the majority or all team members select the same card then the estimate indicated by that card will be the estimate for that item.

If the majority or all team members select the same card then the estimate indicated by that card will be the estimate for that item. If there is no consensus, then the team members discuss reasons for selecting different cards or estimates.

Related forms: AF.AS-0304 User Story - Acceptance Criteria

5.2.2.2 Commit User Stories

In this process, the Scrum Team commits to deliver User Stories approved by the Product Owner for a Sprint. The result of this process would be Committed User Stories.

The scrum team will be based on the Length of Sprint, Sprint Velocity. Then discuss with each other, and confirm with the PO the User Stories will be completed in the sprint(Scrum promotes and effective communication outstanding through colocation of the Scrum Team. Scrum also favors informal, face-to-face interactions over formal written communications).

The Scrum Team commits to a subset of Estimated User Stories that they believe they can complete in the next Sprint based upon velocity. The Committed User Stories should always be selected per the priorities defined by the Product Owner.

5.2.2.3 Identify Tasks

In this process, the Committed User Stories are broken down into specific tasks and compiled into a Task List. The team reviews each Committed User Story for the Sprint and identifies actionable activities, or tasks required to implement the deliverables necessary to fulfill the User Story and meet acceptance criteria. The Product Owner is present during this meeting in case clarification is required related to the Committed User Stories to help the team make design decisions. consider risks and dependencies for each User Story.

The scrum core team are the people who will define the tasks to be done in a User Stories.

5.2.2.4 Create Tasks List

This is a comprehensive list that contains all the tasks to which the Scrum Team has committed for the current Sprint. It contains descriptions of each task along with estimates derived during the Identify Tasks process.

5.2.2.4.1 Updated Committed User Stories

Update committed User Stories if necessary.

5.2.2.5 Estimate Tasks

The Scrum Core team estimates the effort required to accomplish each task in the task list. The result of this process is an Effort Estimated Tasklist.

The same estimation methods used to estimate User Stories can be applied to tasks as well. See also ***5.2.2.1.1.***

The Effort Estimated Task List is a list of tasks associated with the Committed User Stories included in a Sprint. Typically the accuracy of estimates varies with team skills. Estimated effort is expressed in terms of the Estimation Criteria agreed on by the team. The Effort Estimated Task List is used by the Scrum Team during Sprint Planning Meetings to create the Sprint Backlog and Sprint Burndown Chart.

Note: Update Tasks List.

As part of the Sprint Planning Meetings, the Scrum Team estimates the effort required to complete a task or set of tasks and to estimate the people's effort and other resources required to carry out the tasks within a given Sprint.

The Scrum Team members use the Task List to estimate the duration and effort for the User Stories to be completed in the Sprint.

5.2.2.6 Selecting Product Backlog Items

Selection can be done in several ways. If we have a sprint goal, we would select product backlog items that align with that goal. If there is no formal sprint goal, our default is to select items from the top of the product backlog. We would start with the top-most item and then move to the next item and so forth. If the team were not able to commit to the next-highest-priority item (perhaps there is a skills capacity issue), it would select the next appropriate higher priority backlog item that looks as if it can be completed within the constraints.

5.2.2.7 Refine the Sprint goal

The sprint goal summarizes the business purpose and value of the sprint. The product owner should come to sprint planning with an initial sprint goal. That initial goal, however, can be refined during the course of sprint planning as the sprint-planning participants work together to determine what can realistically be delivered.

5.2.2.8 Acquiring Confidence

One way to acquire confidence is to use predicted velocity to see if the commitment is realistic. The risk of using velocity as the sole means of establishing confidence is that even though the numbers look right, the commitment might still be unachievable.

Most Scrum teams gain the necessary level of confidence by breaking the product backlog items down into the tasks that are required to complete them to the Scrum team's agreed-upon definition of done.

5.2.2.9 Finalize the commitment

During Sprint Planning Meetings, User Stories are committed for a Sprint, and Tasks are identified and estimated by the Scrum Team. Each Scrum Team member also uses the Effort Estimated Task List to select the tasks they plan to work on in the Sprint, based on their skills and experience. The Scrum Team also creates the Sprint Backlog and Sprint Burndown Chart using the User Stories and the Effort Estimated Task List during the Sprint Planning Meetings.

At the completion of sprint planning the development team finalizes its commitment to the business value it will deliver by the end of the sprint. The sprint goal and the selected product backlog items embody that commitment.

The nuanced differences between these terms might affect only the scope of what the development team determines it can deliver and how the Scrum team deals with new information that arrives during sprint execution

5.2.2.10 Create Sprint Backlog

The list of the tasks to be executed by the Scrum Team in the upcoming Sprint is called the Sprint Backlog.

Once the Sprint Backlog is finalized and committed to by the Scrum Team, new User Stories should not be added; however, tasks that might have been missed or overlooked from the committed User

Stories may need to be added. If new requirements arise during a Sprint, they will be added to the overall Prioritized Product Backlog and included in a future Sprint.

Related forms: AF.AS-04 [AMIT] Sprint Backlog Template

5.2.2.11 Create Sprint Burndown Chart

The Sprint Burndown Chart is a graph that depicts the amount of work remaining in the ongoing Sprint. The initial Sprint Burndown Chart is accompanied by a planned burndown. The Sprint Burndown Chart should be updated at the end of each day as work is completed. This chart shows the progress that has been made by the Scrum Team and also allows for the detection of estimates that may have been incorrect.

If the Sprint Burndown Chart shows that the Scrum Team is not on track to finish the tasks in the Sprint on time, the Scrum Master should identify any obstacles or impediments to successful completion and try to remove them.

Related forms: AF.AS-05 Burndown Chart

5.3 Implement

Overview

The Implement phase is related to the execution of the tasks and activities to create a project's product.

These activities include creating various deliverables, conducting Daily Standup Meetings, and grooming (i.e., reviewing, fine-tuning, and regularly updating) the Product Backlog at regular intervals.

After conducting the Sprint planning meeting at the beginning of a Sprint, we will start the implementation phase. There is a small note about the implementation that we should focus. The first daily meeting in the implementation phase will be occurred in the next day after sprint planning meeting instead of being occurred the same day as sprint planning meeting.

Related forms: AF.AS-06 [AMIT] Impediment Log Template

AF.AS-07 [AMIT] Daily Check List Sprint [Number] Template

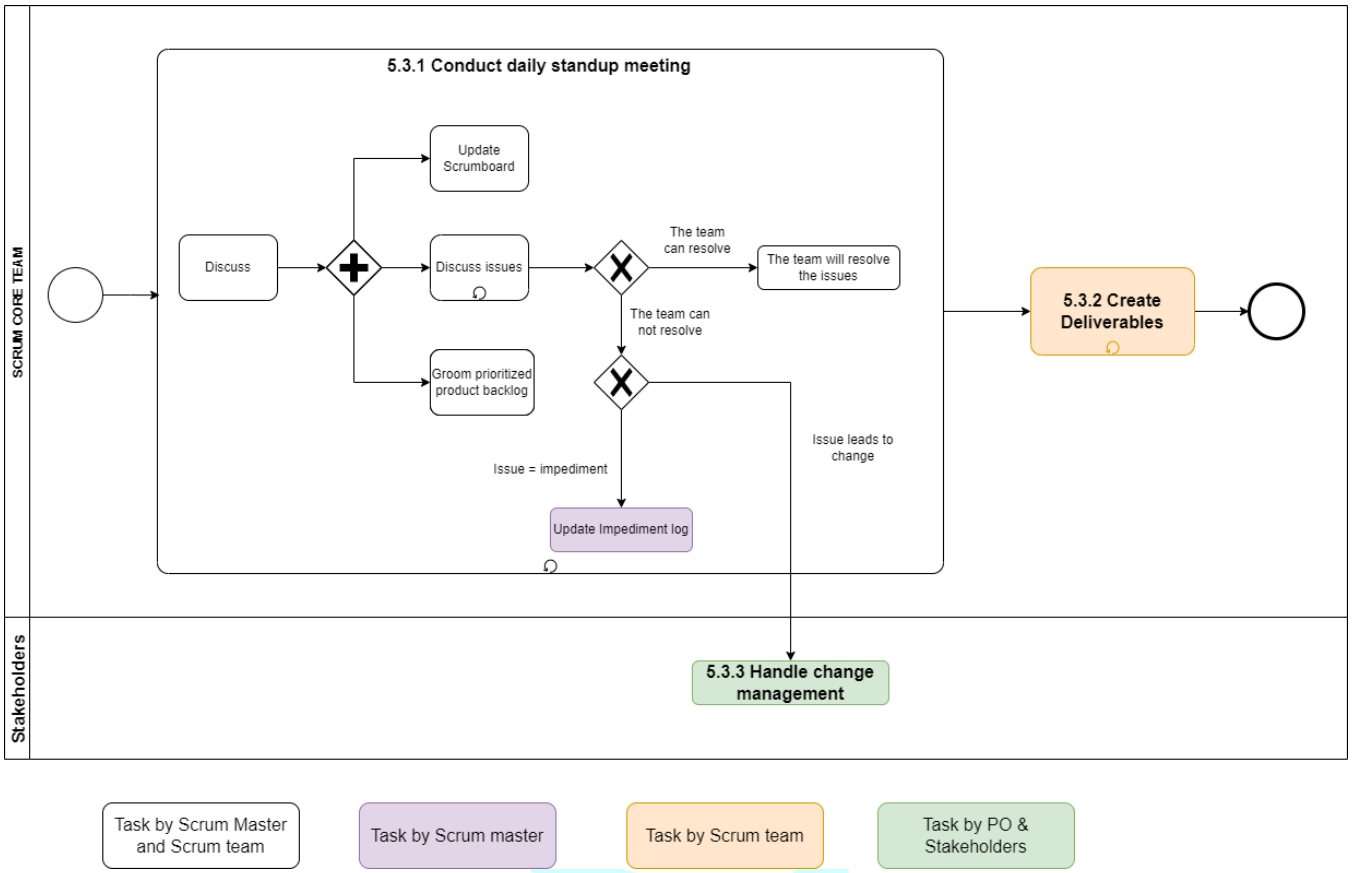


Figure 5.3.1 Implement Phase

5.3.1 Conduct daily standup meeting

Overview

Everyday a highly focused, Time-boxed meeting is conducted referred to as the Daily Standup Meeting. This is the forum for the Scrum Team to update each other on their progress and any impediments they may be facing. Time for Daily Standup Meeting should not be more than 15 mins.

1. In daily standup meetings, the Scrum team will discuss the progress of work they did.
2. In the discussion, they will update the Scrum Board - which is used to track the work and activities being carried out, discuss about some issues if have and update the prioritized product backlog to prepare next sprint better
3. When discussing the issue, if the scrum team is able to handle it, they will be in charge of the issue. However, if they can not handle the issue, they must clarify the issue into impediment or change.

3.1 If the issue becomes an impediment, The scrum master will update this issue in Impediment log

3.2 If the issue leads to some changes, the team will put it in the change management process.

4. After daily standup meeting, they will continue working on the tasks in the sprint backlog to create sprint deliverables

5.3.2 Create Deliverables

All member in scrum team focus on working the tasks in the Sprint backlog to create Sprint Deliverables

5.3.3 Handle Change Management

If there is a Change Request that may have a significant impact on a Sprint in progress.

The Product Owner, after consultation with relevant stakeholders, decides whether the change can wait until the next Sprint or represents an urgent situation which may require ending the current Sprint and starting a new one.

There is only one exception to this rule about not changing the scope of a Sprint once a Sprint begins. If the Scrum Team determines it has heavily overestimated the effort during the Sprint and has spare capacity to implement additional User Stories, the team can ask the Product Owner which additional User Stories should be included in the current Sprint.

By locking down the scope of every Sprint, the team is able to efficiently optimize and manage their work and effort. An additional benefit is that the team does not have to worry about managing changes once they start working on a Sprint. This is a big advantage of the Scrum framework as compared with traditional project management.

Moreover, since the Product Owner and Stakeholders are aware that changes are not allowed once a Sprint begins and a Sprint lasts between 1 and 6 weeks, they define and prioritize requirements during the appropriate processes of Create Epic(s), Create Prioritized Product Backlog, and Groom Prioritized Product Backlog.

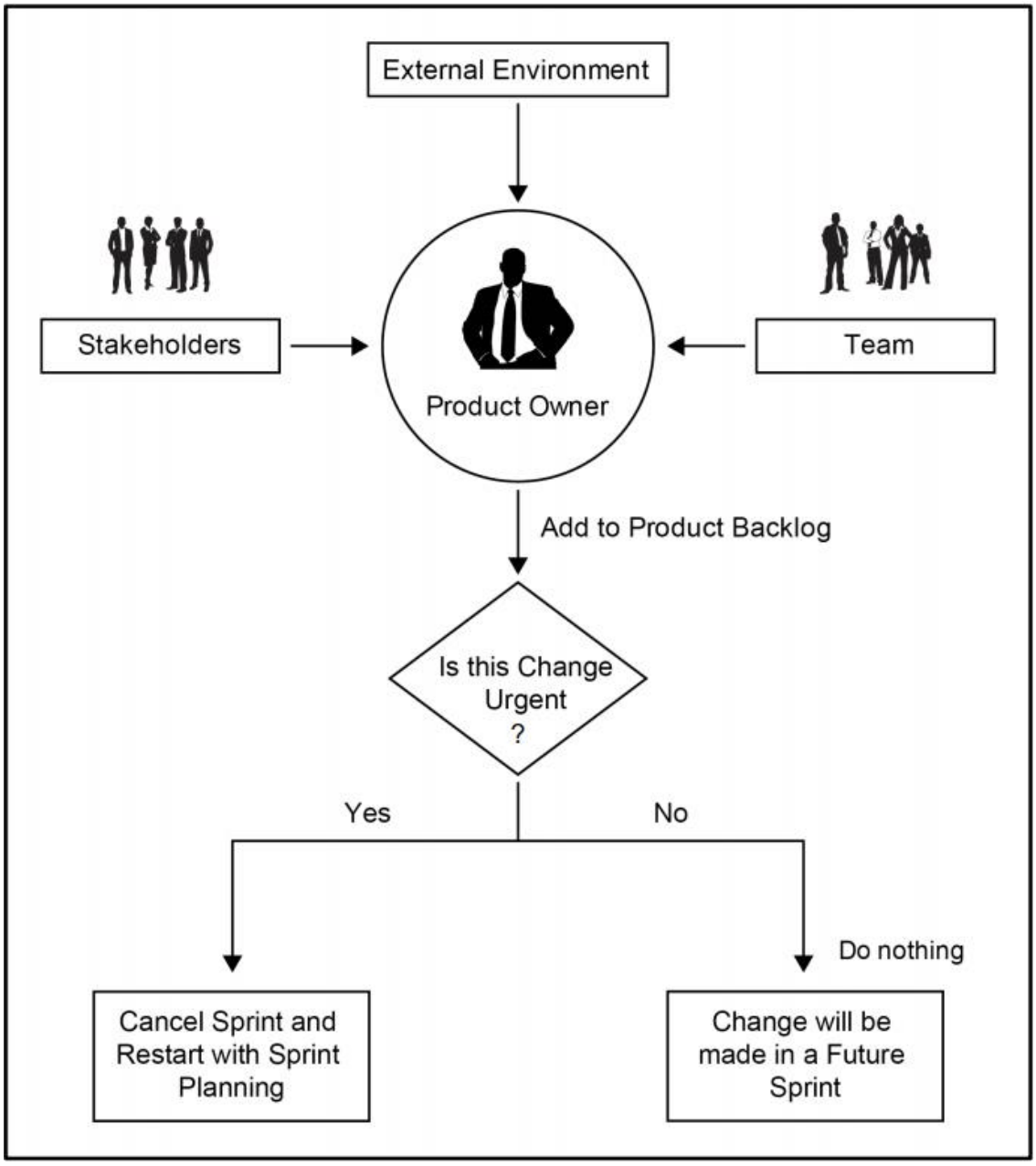


Figure 5.3.2 Change request

5.4 Review & Retrospect

Overview

Scrum provides two inspect-and-adapt opportunities at the end of each sprint: the sprint review and the sprint retrospective.

5.4.1. Demonstrate and Validate Sprint

The Scrum Team demonstrates the Sprint Deliverables to the PO and relevant stakeholders in a Sprint Review Meeting.

The purpose of this meeting is to secure approval and acceptance of the product or service by the PO.

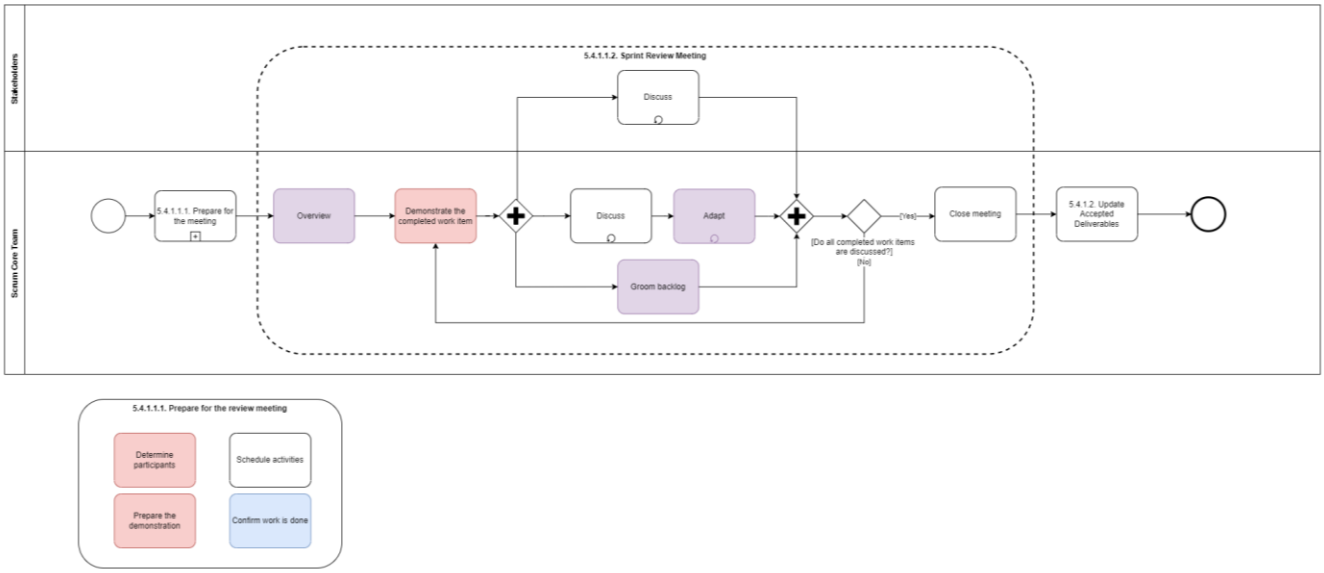


Figure 5.4.1 Demonstrate and Validate sprint Process

5.4.1.1. Conduct Sprint Review Meetings

At the end of every Sprint, the Sprint Review Meeting is convened so that the Scrum Team can demonstrate the achievements from the Sprint, including the new functionalities or products created. The Scrum Core Team members and relevant Stakeholder(s) participate in Sprint Review Meetings to accept the deliverables which meet the User Story Acceptance Criteria and reject unacceptable deliverables.

This provides an opportunity for the Product Owner and Stakeholder(s) to inspect what has been completed so far and to determine if any changes should be made in the project or processes in subsequent Sprint.

5.4.1.1.1. Prepare for the Sprint Review Meetings

Determine whom to invite

The participants of the Sprint Review Meetings should be all interested parties, who can come from a number of different sources, as summarized below.

Source	Description
Scrum team	The product owner, ScrumMaster, and development team should all be present so that they can all hear the same feedback and be able to answer questions regarding the sprint and the product increment.
Internal stakeholders	Business owners, executives, and managers should see the progress firsthand so that they can suggest course corrections. For internal product development, internal users, subject matter experts, and the operations manager of the business function to which the product relates should attend.
Other internal teams	Sales, marketing, support, legal, compliance, and other Scrum and non-Scrum development teams might want to attend sprint reviews to provide area-specific feedback or to sync their own groups' work with the Scrum team.
External stakeholders	External customers, users, and partners can provide valuable feedback to the Scrum team and other attendees.

Figure 5.4.2 Sprint Review Attendee (Source: Essential Scrum by Kenneth S.Rubin)

To get the right set of people into the room to extract the highest possible value, identify a core group that should be invited to every review and then issue a separate invitation to certain groups or clients on a sprint-by-sprint basis.

Schedule the activity

The sprint review needs to be scheduled (when, where, and how long). If we use consistent-duration sprints, we can schedule all, or at least most, of the sprint review meetings using a regular cadence (ex: every second Friday at 2 pm).

Sprint reviews vary in duration depending on several factors, including sprint length, team size, and whether multiple teams are participating in the same review. Typically, however, the sprint review does not exceed a four-hour timebox. Many teams have found the one-hour-per-sprint-week rule helpful. In other words, for a two-week sprint the review should take no more than two hours; for a four-week sprint it should take no more than four hours.

Confirm that the sprint work is done

Ultimately it is the Product Owners' responsibility to determine if the work is done or not. The Product Owner should be performing just-in-time reviews of product backlog items as they become

available during sprint execution. This way, by the time the sprint review happens, the team knows which items are complete.

Prepare for the demonstration

Because all of the work the team presents at the sprint review is done (potentially shippable), it shouldn't take much preparatory work to demonstrate it. The sprint review is supposed to be an informal meeting with low ceremony and high value.

However, in some cases, for example there are representatives with high positions, the team understandably invested a bit more time in prep and polish.

Determine who does what

Prior to the sprint review, the team needs to decide who on the Scrum team is going to facilitate the review and who will demonstrate the completed work. Typically the ScrumMaster facilitates, but the product owner might kick things off by welcoming members of the stakeholder community and providing a synopsis of the sprint results. As for demoing the completed work,every member of the development team should have an opportunity at some sprint review to go hands-on and demonstrate, rather than the same person always dominating the demo every sprint review.

5.4.1.1.2. Sprint Review activities

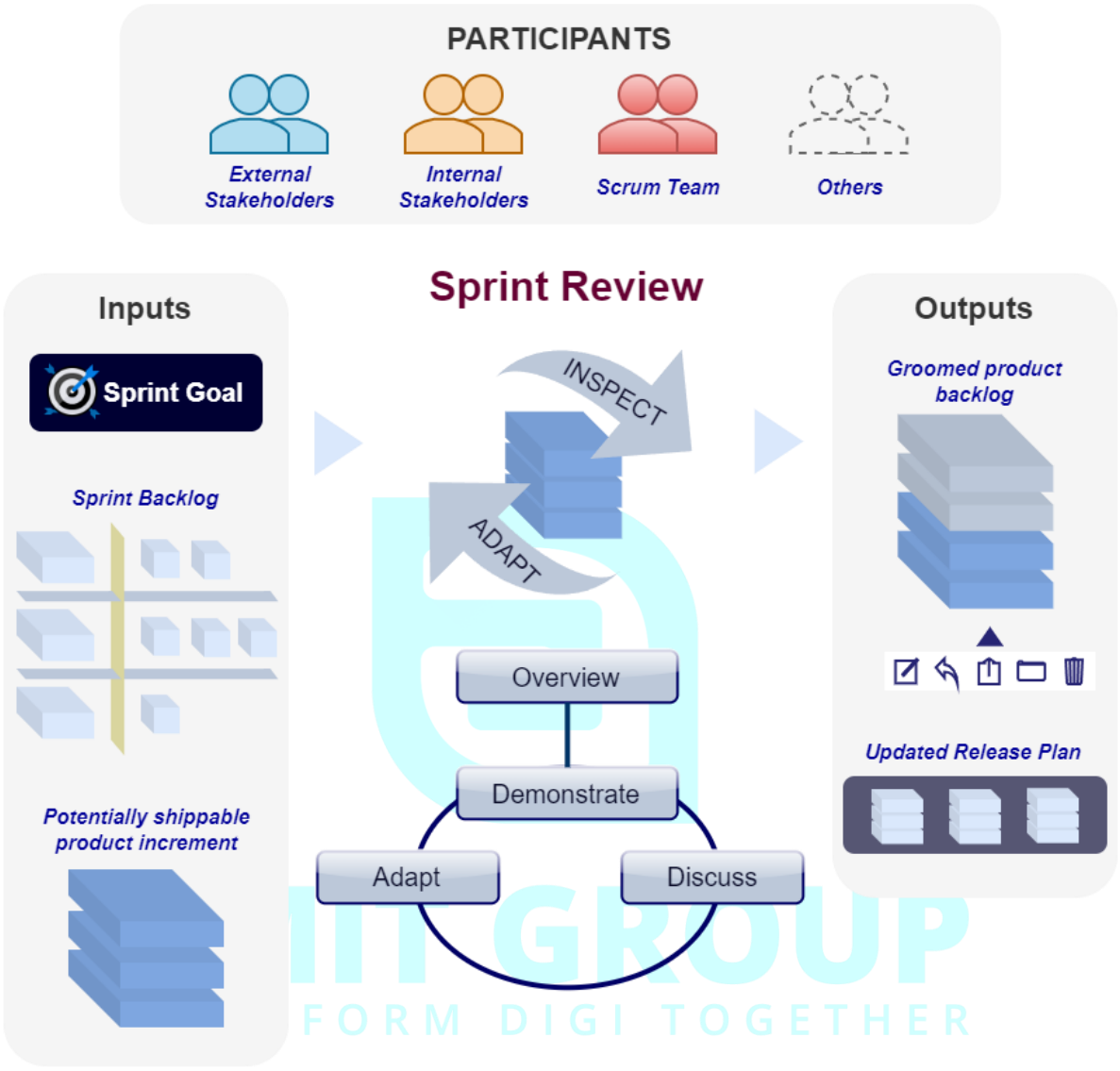


Figure 5.4.3 Sprint Review Activity (Reference: Essential Scrum by Kenneth S.Rubin)

Overview

The sprint review kicks off with a Scrum team member (frequently the Product Owner) presenting the sprint goal, the product backlog items associated with the sprint goal, and an overview of the product increment that was actually achieved during the sprint. This information provides a summary or synopsis of how the sprint results compare with the sprint goal.

If the results don't match, the Scrum team provides an explanation. It is important that the sprint review be a blame-free environment. If the goal wasn't met, everyone participating should refrain

from trying to assess blame. The purpose of the review is to describe what was accomplished and then to use the information to determine the best course of action for moving forward.

Demonstrate

The most important aspect of the sprint review is in-depth conversation and collaboration among the participants to enable productive adaptations to surface and be exploited. The demonstration of what actually got built is simply a very efficient way to energize that conversation around something concrete. Nothing provides focus to the conversation like being able to actually see how something works.

As determined in the prework, one or more Scrum team members will demonstrate all relevant aspects of the product increment that was built during the sprint. In certain organizations, such as game studios, it can be even more effective to let the stakeholders actually give themselves the demo, perhaps by playing the increment of the game that was developed during the sprint.

If the team didn't get anything done and there is truly nothing to show, the sprint review will likely focus on why nothing got done and how the future work will be affected by the lack of progress during this sprint.

If, on the other hand, what was built can't easily be demoed, the Product Owner must understand the value of doing the work and also must know how to determine if the work is done. At a minimum, the team must have some set of tests to demonstrate that the work is done to the satisfaction of the product owner. Those tests must have passed because the team can show **only completed work** at the sprint review. So, at the very least, the team can use those tests to demonstrate at the sprint review.

Discuss

Demonstrating the product increment becomes the focal point for having an in depth conversation. Observation, comments, and reasonable discussion regarding the product and direction are strongly encouraged among the participants. The sprint review, however, is not the place for deep problem solving; that type of work should be deferred to another meeting.

Vigorous discussion allows participants who aren't on the Scrum team to ask questions, understand the current state of the product, and help guide its direction. At the same time, the Scrum team members gain a deeper appreciation for the business and marketing side of their product by getting feedback on the convergence of the product toward delighted customers or users.

Adapt

Through demonstration and discussion, the team is able to ask and answer questions, including the following:

- Do the stakeholders like what they see?
- Do they want to see changes?
- Is what we're building still a good idea in the marketplace or to our internal customers?
- Are we missing an important feature?
- Are we overdeveloping/investing in a feature where we don't have to?

Asking and answering these questions provides input on how to adapt the product backlog and release plans.

Grooming

Most teams naturally do some grooming as part of the sprint review. As everyone involved gains a better understanding of the current development effort and where it is going, new PBIs are often created or existing PBIs are reprioritized or deleted if they are no longer needed. This grooming might affect what the team will work on in the next sprint.

5.4.1.2. Maintain and Update Accepted Deliverable

Accepted Deliverables are a list of deliverables which meet the User Story Acceptance Criteria accepted by the Product Owner. The objective of a Sprint is to create potentially shippable deliverables, or product increments, which meet the Acceptance Criteria defined by the customer and Product Owner. These are considered Accepted Deliverables that may be released to the customer if they so desire. A list of Accepted Deliverables is maintained and updated after each Sprint Review Meeting.

5.4.1.3. Handle Rejected Deliverables

If a deliverable does not meet the defined Acceptance Criteria, it is not considered accepted and will usually be carried forward into a subsequent Sprint to rectify any issues. This is highly undesirable because the objective of every Sprint is for the deliverables to meet the criteria for acceptance.

User Stories associated with such Rejected Deliverables get added to the Prioritized Product Backlog so that such deliverables may be considered as part of a subsequent Sprint.

5.4.2. Retrospect Sprint

The Scrum Master and Scrum Team meet to discuss the lessons learned throughout the Sprint. This information is documented as lessons learned which can be applied to future Sprint.

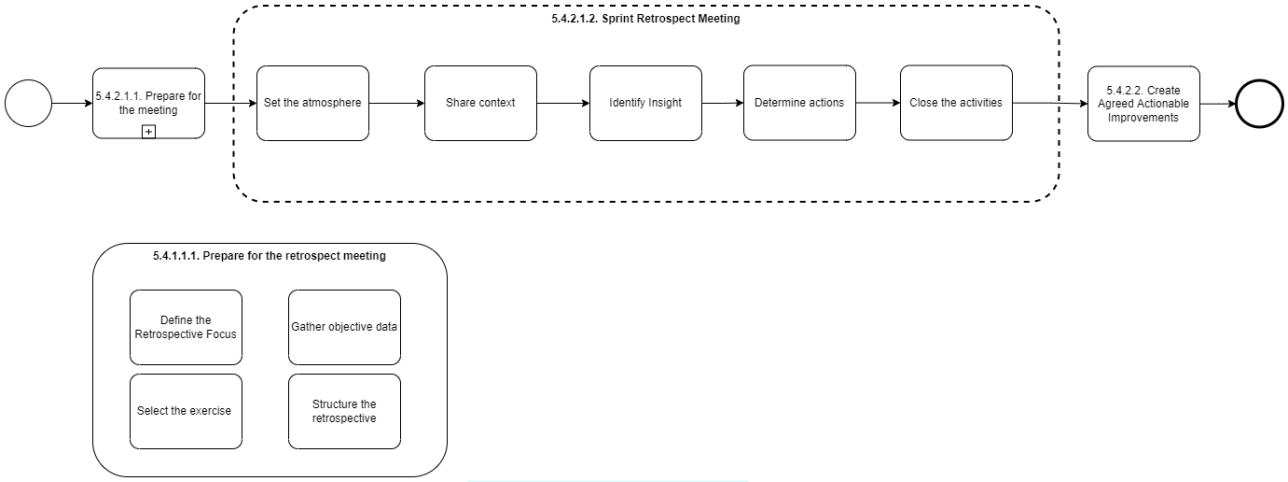


Figure 5.4.4 Sprint Retrospect Process

5.4.2.1. Conduct Retrospective Sprint Meeting

The Retrospect Sprint Meeting is an important element of the 'inspect-adapt' Scrum framework and it is the final step in a Sprint. All Scrum Team members attend the meeting, which is facilitated or moderated by the Scrum Master. It is recommended, but not required for the Product Owner to attend. One team member acts as the scribe and documents discussions and items for future action. It is essential to hold this meeting in an open and relaxed environment to encourage full participation by all team members. Discussions in the Retrospect Sprint Meeting encompass both what went wrong and what went right. Primary objectives of the meeting are to identify three specific items:

1) Things the team needs to keep doing: best practices

2) Things the team needs to begin doing: process improvements

3) Things the team needs to stop doing: process problems and bottlenecks

These areas are discussed and a list of Agreed Actionable Improvements is created.

5.4.2.1.1. Prepare for the meeting

Define the Retrospective Focus

Each sprint retrospective should have a well-defined focus. The default focus is to review all relevant aspects of the process the Scrum team used during the current sprint. However, there are

times when a team might select a different retrospective focus based on what is currently important to the team and where it is energetic about seeing improvement. For example:

- Focus on how to improve our skills with test-driven development (TDD).
- Focus on why we build what we think the customers want, but when they see it they frequently believe we misunderstood their desires or missed an important facet of the requirement.

Establishing and communicating the focus before the start of the retrospective allows the Scrum team to determine if any non-Scrum team members should be invited. In addition, knowing the focus before the start of the retrospective allows the team to select appropriate retrospective exercises and gives people time to gather and prepare any data needed to ensure a smooth performance of the retrospective.

Select the exercises

Once we have established the focus and final participants for the upcoming retrospective, we can determine which exercises might help participants to engage, think, explore, and decide together. A typical retrospective includes the following exercises:

- Create and mine a sprint event timeline.
- Brainstorm insights.
- Group and vote on insights.

Gather objective data

Because a sprint retrospective is performed in a focused, short period of time (many teams establish a timebox), any legwork to collect needed data should be done before the retrospective begins.

We know both the focus and the exercise options for the upcoming retrospective, so we should have a good idea of what, if any, objective data should be gathered. Objective data is hard data (not opinions), such as what events happened and when, or counts of the number of PBIs that were started but not finished, or the feature burnup chart for the sprint illustrating the flow of completed work. At this point we are not organizing or analyzing any data; we are just collecting it so that it is available during the retrospective.

Structure the retrospective

The exact length of the retrospective is influenced by factors such as how many people are on the team, how new the team is, whether any team members are located remotely, and so on. In my experience, teams new to Scrum have a tendency to budget too little time for their retrospectives. It's difficult to hold a meaningful sprint retrospective in less than 60 minutes. As a rule, budget abou**t** *1.5 hours for the sprint retrospective when using two-week sprints, and proportionally more when using longer sprints.*

5.4.2.1.2. Sprint Retrospect Meeting Activities

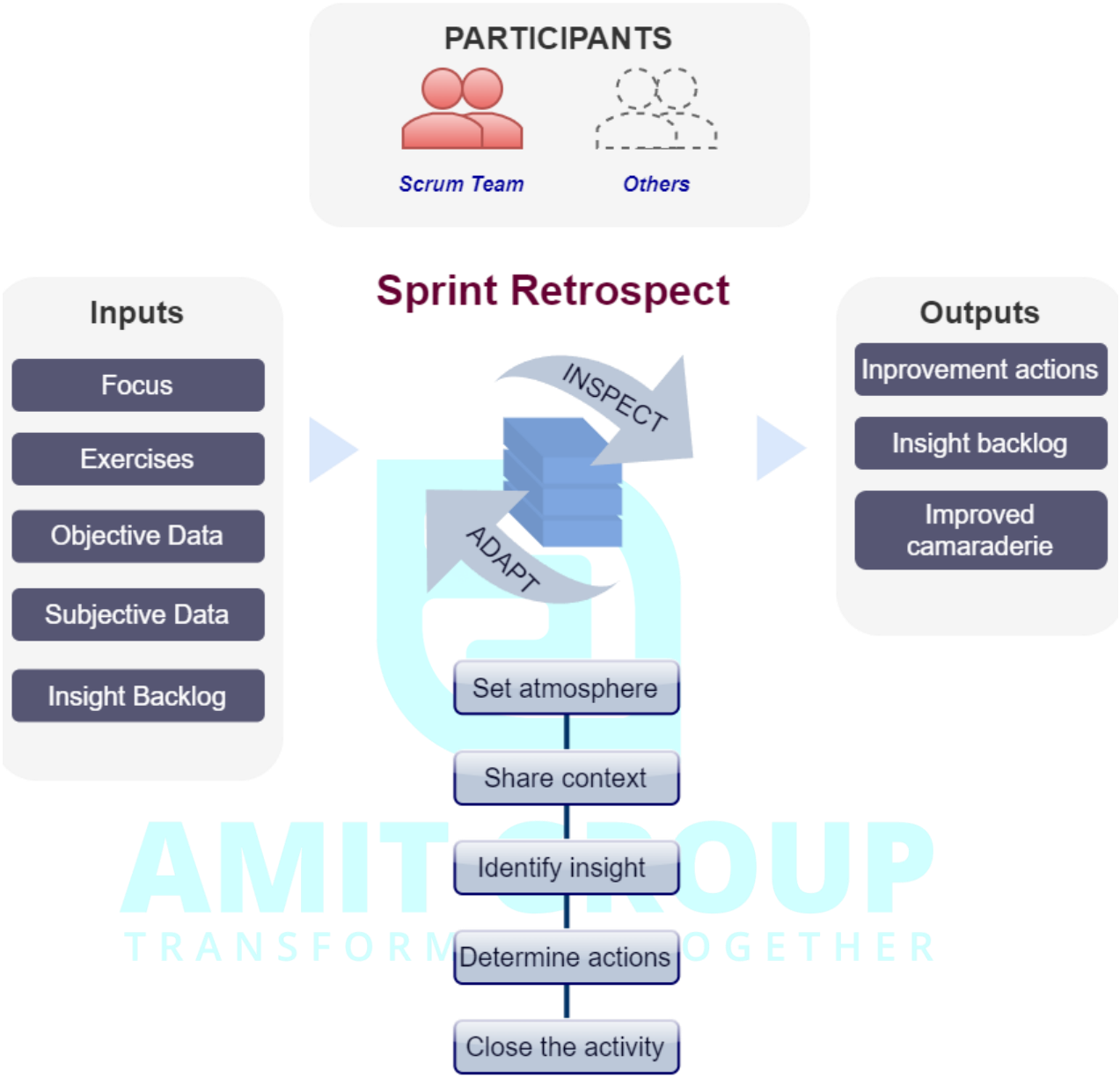


Figure 5.4.5 Sprint retrospective activity (Reference: Essential Scrum by Kenneth S.Rubin)

Set the Atmosphere

People must feel it is safe to express their opinions without fear of retribution. Teams should have established ground rules, or a working agreement, which make it clear that expressing opinions and airing dirty laundry are safe things to do. It is helpful for the ground rules to make clear that the focus is on the organizational system and process, not the individuals, thus making it safe to explore what

went wrong. There will be times when problems are people problems; the retrospective is not the place to solve them. The retrospective is about improving the Scrum team's process, not about assigning blame or reprimanding individual behavior. When setting the atmosphere, ensure that the ground rules reinforce the concept of a blame-free environment.

Share Context

To establish a shared context the participants must align their diverse individual Perspectives into a shared team perspective.

Just because we are grounded in objective data doesn't mean subjective data is irrelevant, however. Each person brings to the retrospective subjective data reflecting her interpretation of the sprint. If that subjective data is not exposed and discussed, participants might just assume that everyone else experienced the sprint in a similar way. This misalignment will make it difficult for people to understand one another's comments and suggestions.

There are a number of exercises that the participants can use to develop a shared context of both objective and subjective data. Two of the most common exercises are an event timeline and an emotions seismograph.

Identify insight

The participants should start by mining the shared context data. For example, they could look at their event timeline and emotions seismograph and ask the following questions to help uncover insights:

- What worked well?
- What didn't work well?
- Where are some opportunities to do things differently?

Determine Actions

Insights are our ideas or perceptions of things that can be improved. To extract long term value from these insights we need to move from discussing them to taking demonstrable actions to leverage them. For example, if the insight is "We're wasting too much time because the code management system keeps failing," the improvement action moving forward might be "Have Talya apply the vendor patches to the code management system to make it more stable." Talya, a member of the development team, can take this action in the next sprint.

Close the activities

Once the final improvement actions have been determined, the participants close out the retrospective. Many close by recapping what actions the team has decided to take based on what the participants learned. This might be as simple as describing each committed action item and who is going to work on it.

5.4.2.2. Create Agreed Actionable Improvements

Agreed Actionable Improvements are the primary output of the Retrospect Sprint process. They are the list of actionable items that the team has come up with to address problems and improve processes in order to enhance their performance in future Sprints.

Related form: AF.AS-08 [AMIT] Agreed Actionable Improvement Sprint Template

5.5. Release

Overview

The Release phase focuses on delivery of the Accepted Deliverables to the Customer and on identifying, documenting, and internalizing the lessons learned during the project. Release is the relevant option for the following:

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

Scrum can be applied effectively to any project in any industry—from small projects or Teams with as few as sixTeam members to large, complex projects with up to several hundred Team members.

5.5.1. Ship Deliverable

Accepted Deliverables are delivered or transitioned to the relevant stakeholders. A formal Working Deliverables Agreement documents the successful completion of the Sprint.

5.5.1.1. Deliver Accepted Deliverables

5.5.1.1.1. Approve Accepted Deliverables

Deliverables that meet the Acceptance Criteria receive formal business sign-off and approval by the customer or sponsor. To get formal customer acceptance is critical for revenue recognition and

the responsibility for obtaining it will be defined by the company policies and is not necessarily the responsibility of the Product Owner.

5.5.1.1.2. Release Product

The Product Releases should include the following:

• Release Content—This consists of essential information about the deliverables that can assist the Customer Support Team.

• Release Notes—Release Notes should include external or market facing shipping criteria for the product to be delivered.

Related form: AF.AS-09 [AMIT] Project Management Deliverables Template

5.5.2. Retrospect Project

In this process, which completes the project, organizational stakeholders and Scrum Core Team members assemble to retrospect the project and identify, document, and internalize the lessons learned. Often, these lessons lead to the documentation of Agreed Actionable Improvements, to be implemented in future projects.

5.5.2.1. Conduct Retrospect Project Meeting

The Retrospect Project Meeting is a meeting to determine ways in which team collaboration and effectiveness can be improved in future projects. Positives, negatives, and potential opportunities for improvements are also discussed. This meeting is not Time-boxed and may be conducted in person or in a virtual format. Attendees include the Project Team, Chief Scrum Master, Chief Product Owner, and Stakeholder(s). During the meeting, lessons learned are documented and participants look for opportunities to improve processes and address inefficiencies.

5.5.2.2. Create Agreed Actionable Improvements

Similar to the Retrospect Sprint process, Agreed Actionable Improvements are the primary output of the Retrospect Project process. They are the list of actionable items that the team has come up with to address problems and improve processes in order to enhance their performance in future projects within the organization.

Related form: AF.AS-10 [AMIT] Agreed Actionable Improvement Project Template

5.5.2.3. Create Assigned Action Items and Due Dates

Once the Agreed Actionable Improvements have been elaborated and refined, action items to implement the improvements may be considered by the Scrum Team. Each action item will have a defined due date for completion.

6. FORM ISSUED (Template)

The full process template is listed here: Template of AGILE SCRUM process.

No	Document ID	Document Template
1	AF.AS-01	[AMIT] Project Business Case Template
2	AF.AS-02	[AMIT] Project Vision Statement Template
3	NA	[AMIT] Personas Template (Figma Tool, it's not official document so no required code)
4	AF.AS-03	[AMIT] Product Backlog Template
5	AF.AS-04	[AMIT] Sprint Backlog Template
6	AF.AS-05	[AMIT] Impediment Log Template
7	AF.AS-06	[AMIT] Daily Check List Sprint [Number] Template
8	AF.AS-07	[AMIT] Agreed Actionable Improvement Sprint Template
9	AF.AS-08	[AMIT] Project Management Deliverables Template
10	AF.AS-09	[AMIT] Agreed Actionable Improvement Project Template