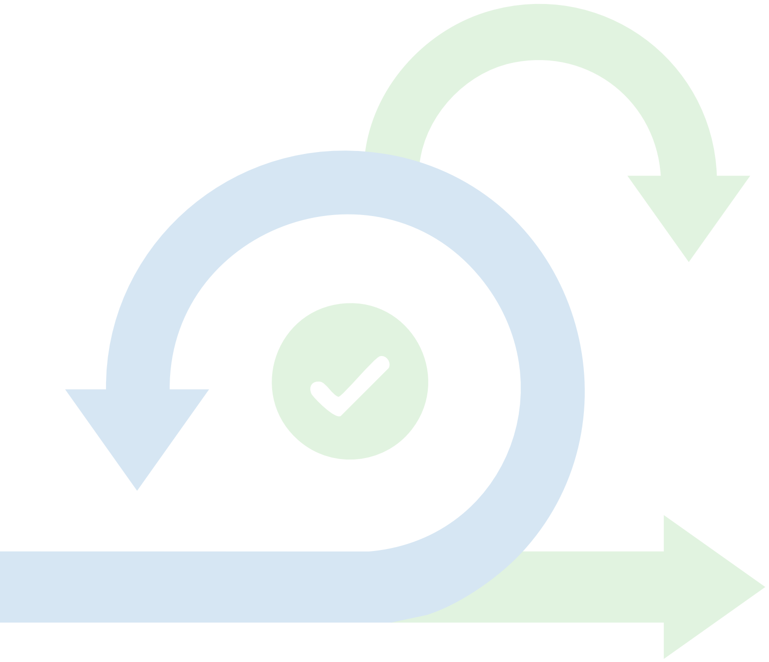
SCRUMstudy
Targeting success



A Guide to the

SCRUM BODY OF KNOWLEDGE

(SBOK® Guide)

9. PLAN AND ESTIMATE

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

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

9. PLAN AND ESTIMATE

The Plan and Estimate phase consists of processes related to planning and estimating User Stories and associated tasks, which include *Create User Stories, Estimate User Stories, Commit User Stories, Identify Tasks, Estimate Tasks,* and *Update Sprint Backlog*.

Plan and Estimate, 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.

To facilitate the best application of the Scrum framework, this chapter identifies inputs, tools, and outputs for each process as either “mandatory” or “optional.” Inputs, tools, and outputs denoted by asterisks (*) are mandatory, or considered critical for project success, whereas those with no asterisks are optional.

It is recommended that the Scrum Team and those individuals being introduced to the Scrum framework and processes focus primarily on the mandatory inputs, tools, and outputs; while Product Owners, Scrum Masters, and other more experienced Scrum practitioners strive to attain a more thorough knowledge of the information in this entire chapter.

This chapter is written from the perspective of one Scrum Team working on one Sprint to produce potentially shippable deliverables, which could be part of a larger project, program, or portfolio. Additional information pertaining to Scaling Scrum for Large Projects is available in chapter 13. Additional information pertaining to Scaling Scrum for the Enterprise can be found in chapter 14.

After the Initiate Phase is completed, the iterative Sprint cycles can commence. Plan and Estimate is the first of three phases that are done repetitively, in every Sprint cycle.

At the beginning of a Sprint, the Product Owner and Scrum Team, facilitated by the Scrum Master, plan the Sprint. The Product Owner refines the highest-priority Epics into a set of well written and estimated User Stories, which the Scrum Team commits to completing in the upcoming Sprint based on team velocity assumptions. The Scrum Master and the Scrum Team create and updates the Sprint Backlog with the list of User Stories committed to be delivered as part of the Sprint.

The Scrum Team then plans its work in more detail by identifying and optionally estimating the tasks it must complete in order to deliver the User Stories for the Sprint. As a final planning step for the Sprint, the team completes the Sprint Backlog with details of tasks and, if available, their estimates. The Sprint Backlog will be used in the Implement phase to track the team’s progress during the Sprint.

It is also important to realize that although all phases and processes are defined uniquely in the SBOK® Guide, they are not necessarily performed sequentially or separately. At times, it may be more appropriate to overlap some phases and/or processes, depending on the specific requirements of each project.

Figure 9-1 provides an overview of the Plan and Estimate phase processes, which are as follows:

9.1 Create User Stories—In this process, User Stories and their related Acceptance Criteria are created by the Product Owner (elaborated from the previously-defined Epics) and incorporated into the Prioritized Product Backlog. User Stories are designed to ensure that the customer's requirements are clearly depicted and can be fully understood by the business stakeholders. User Stories need to be tangible enough and must satisfy the Definition of Ready before they can be estimated and developed by the Scrum Team.

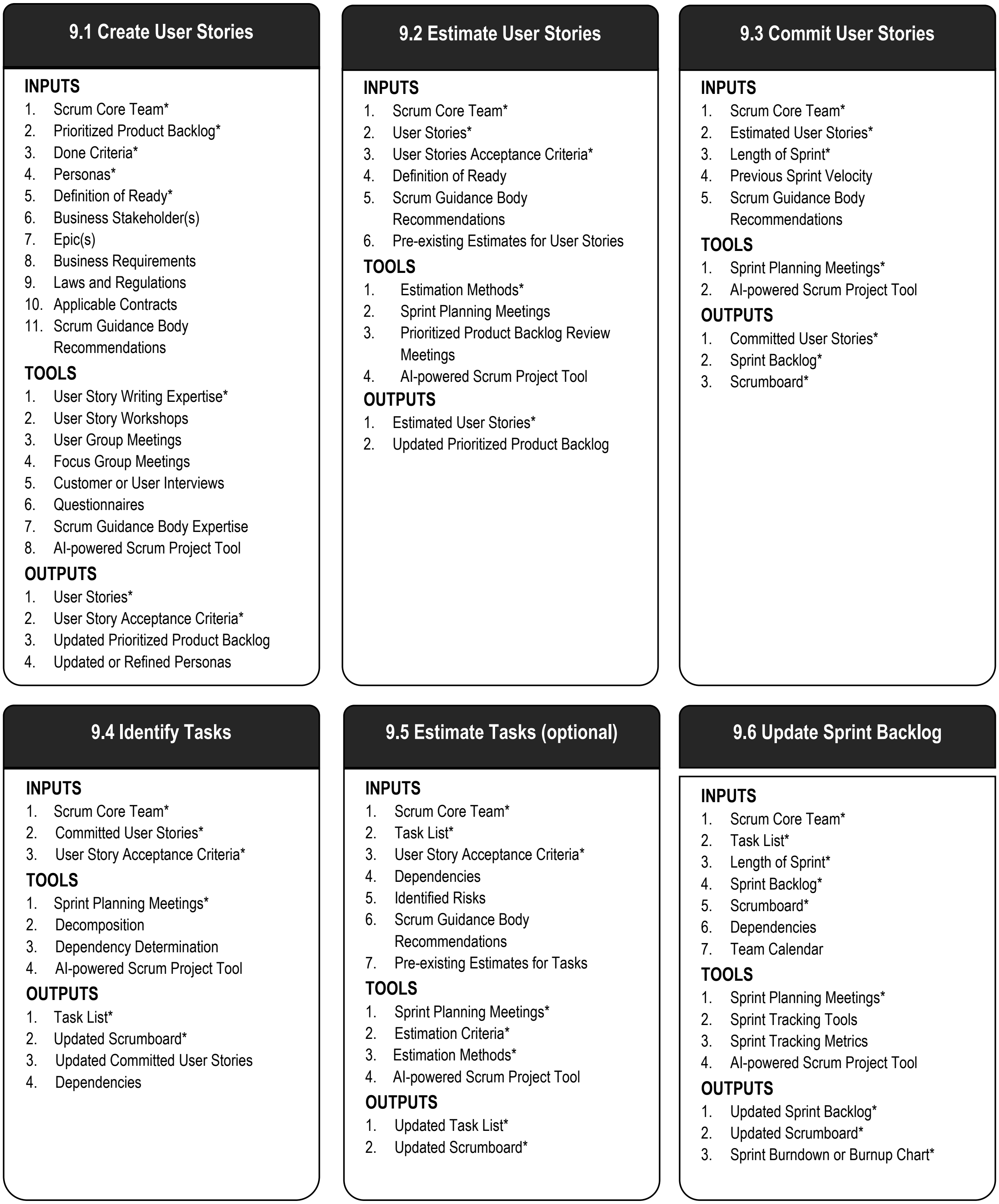
9.2 Estimate User Stories—In this process, the Scrum Team, supported by the Scrum Master, estimates the User Stories and identifies the effort required to develop the functionality described in each User Story. Only User Stories that satisfy the Definition of Ready and are properly defined by the Product Owner are estimated by the team.

9.3 Commit User Stories—In this process, the Scrum Team commits to delivering a set of User Stories for the Sprint. The decision on which User Stories will be committed to is based on the relative value-based priority of the User Stories and the estimated effort and team velocity for one Sprint. As part of this process, the Scrum Team starts the creation of the Sprint Backlog, which contains the committed User Stories that are assigned to a particular Sprint. The backlog is refined further with task-level details as Sprint Planning continues. With this commitment from the Scrum Team given at the beginning of a Sprint as part of Sprint Planning, the content of the Sprint is defined and cannot be changed once the Sprint implementation phase has begun.

9.4 Identify Tasks—In this process, the committed User Stories are decomposed into specific tasks and compiled into a task list. Identifying tasks could either be done at the beginning of the Sprint for all committed User Stories or before the team starts working on the tasks required for each User Story.

9.5 Estimate Tasks—This is an optional process which involves creating task estimates if the Scrum Team sees value in doing so. In this process, the Scrum Team estimates the effort required to accomplish each task in the Task List. Task estimates could either be determined at the beginning of the Sprint for all User Stories/tasks relevant to that Sprint, or for each task just before the team starts working on the particular User Story/task. The estimation can be done using the same methods that were used for the *Estimate User Stories* process.

9.6 Update Sprint Backlog—In this process, the Scrum Core Team updates the Sprint Backlog with task details and if available, the task estimates. The updated Sprint Backlog will be used in the Implement phase to track the team's progress during the upcoming Sprint.



9.1 Create User Stories

INPUTS

1. Scrum Core Team*
2. Prioritized Product Backlog*
3. Done Criteria*
4. Personas*
5. Definition of Ready*
6. Business Stakeholder(s)
7. Epic(s)
8. Business Requirements
9. Laws and Regulations
10. Applicable Contracts
11. Scrum Guidance Body Recommendations

TOOLS

1. User Story Writing Expertise*
2. User Story Workshops
3. User Group Meetings
4. Focus Group Meetings
5. Customer or User Interviews
6. Questionnaires
7. Scrum Guidance Body Expertise
8. AI-powered Scrum Project Tool

OUTPUTS

1. User Stories*
2. User Story Acceptance Criteria*
3. Updated Prioritized Product Backlog
4. Updated or Refined Personas

9.2 Estimate User Stories

INPUTS

1. Scrum Core Team*
2. User Stories*
3. User Stories Acceptance Criteria*
4. Definition of Ready
5. Scrum Guidance Body Recommendations
6. Pre-existing Estimates for User Stories

TOOLS

1. Estimation Methods*
2. Sprint Planning Meetings
3. Prioritized Product Backlog Review Meetings
4. AI-powered Scrum Project Tool

OUTPUTS

1. Estimated User Stories*
2. Updated Prioritized Product Backlog

9.3 Commit User Stories

INPUTS

1. Scrum Core Team*
2. Estimated User Stories*
3. Length of Sprint*
4. Previous Sprint Velocity
5. Scrum Guidance Body Recommendations

TOOLS

1. Sprint Planning Meetings*
2. AI-powered Scrum Project Tool

OUTPUTS

1. Committed User Stories*
2. Sprint Backlog*
3. Scrumboard*

9.4 Identify Tasks

INPUTS

1. Scrum Core Team*
2. Committed User Stories*
3. User Story Acceptance Criteria*

TOOLS

1. Sprint Planning Meetings*
2. Decomposition
3. Dependency Determination
4. AI-powered Scrum Project Tool

OUTPUTS

1. Task List*
2. Updated Scrumboard*
3. Updated Committed User Stories
4. Dependencies

9.5 Estimate Tasks (optional)

INPUTS

1. Scrum Core Team*
2. Task List*
3. User Story Acceptance Criteria*
4. Dependencies
5. Identified Risks
6. Scrum Guidance Body Recommendations
7. Pre-existing Estimates for Tasks

TOOLS

1. Sprint Planning Meetings*
2. Estimation Criteria*
3. Estimation Methods*
4. AI-powered Scrum Project Tool

OUTPUTS

1. Updated Task List*
2. Updated Scrumboard*

9.6 Update Sprint Backlog

INPUTS

1. Scrum Core Team*
2. Task List*
3. Length of Sprint*
4. Sprint Backlog*
5. Scrumboard*
6. Dependencies
7. Team Calendar

TOOLS

1. Sprint Planning Meetings*
2. Sprint Tracking Tools
3. Sprint Tracking Metrics
4. AI-powered Scrum Project Tool

OUTPUTS

1. Updated Sprint Backlog*
2. Updated Scrumboard*
3. Sprint Burndown or Burnup Chart*

Figure 9-1: Plan and Estimate Overview

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-2 below shows the mandatory inputs, tools, and outputs for the processes in the Plan and Estimate phase.

9.1 Create User Stories

INPUTS

1. Scrum Core Team*
2. Prioritized Product Backlog*
3. Done Criteria*
4. Personas*
5. Definition of Ready*

TOOLS

1. User Story Writing Expertise*

OUTPUTS

1. User Stories*
2. User Story Acceptance Criteria*

9.2 Estimate User Stories

INPUTS

1. Scrum Core Team*
2. User Stories*
3. User Stories Acceptance Criteria*

TOOLS

1. Estimation Methods*

OUTPUTS

1. Estimated User Stories*

9.3 Commit User Stories

INPUTS

1. Scrum Core Team*
2. Estimated User Stories*
3. Length of Sprint*

TOOLS

1. Sprint Planning Meetings*

OUTPUTS

1. Committed User Stories*
2. Sprint Backlog*
3. Scrumboard*

9.4 Identify Tasks

INPUTS

1. Scrum Core Team*
2. Committed User Stories*
3. User Story Acceptance Criteria*

TOOLS

1. Sprint Planning Meetings*

OUTPUTS

1. Task List*
2. Updated Scrumboard*

9.5 Estimate Tasks (optional)

INPUTS

1. Scrum Core Team*
2. Task List*
3. User Story Acceptance Criteria*

TOOLS

1. Sprint Planning Meetings*
2. Estimation Criteria*
3. Estimation Methods*

OUTPUTS

1. Updated Task List*
2. Updated Scrumboard*

9.6 Update Sprint Backlog

INPUTS

1. Scrum Core Team*
2. Task List*
3. Length of Sprint*
4. Sprint Backlog*
5. Scrumboard*

TOOLS

1. Sprint Planning Meetings*

OUTPUTS

1. Updated Sprint Backlog*
2. Updated Scrumboard*
3. Sprint Burndown or Burnup Chart*

Figure 9-2: Plan and Estimate Overview (Essentials)

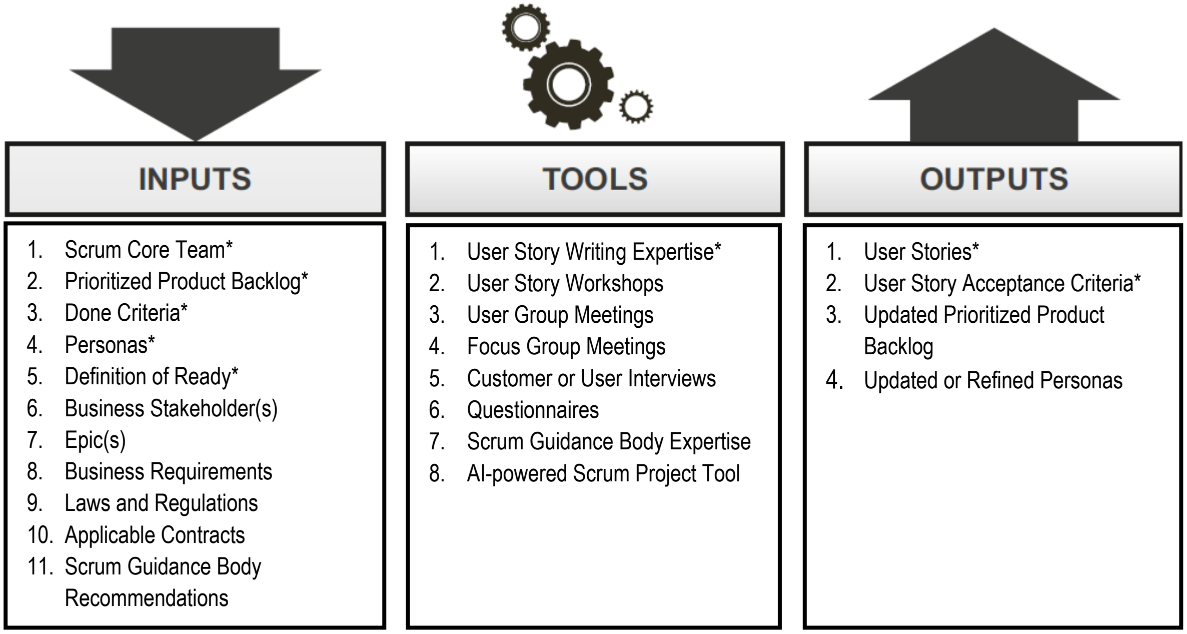
Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

9.1 Create User Stories

In this process, User Stories and their related Acceptance Criteria are created by the Product Owner (elaborated from the previously defined Epics) and incorporated into the Prioritized Product Backlog. User Stories are designed to ensure that the customer's requirements are clearly depicted and can be fully understood by all the project stakeholders. User Stories need to be tangible enough and must satisfy the Definition of Ready before they can be estimated and developed by the Scrum Team.

User Story Workshops may be used to help the Scrum Team members better understand the User Stories created by the Product Owner.

Figure 9-3 shows all the inputs, tools, and outputs for the *Create User Stories* process.



INPUTS

1. Scrum Core Team*
2. Prioritized Product Backlog*
3. Done Criteria*
4. Personas*
5. Definition of Ready*
6. Business Stakeholder(s)
7. Epic(s)
8. Business Requirements
9. Laws and Regulations
10. Applicable Contracts
11. Scrum Guidance Body Recommendations

TOOLS

1. User Story Writing Expertise*
2. User Story Workshops
3. User Group Meetings
4. Focus Group Meetings
5. Customer or User Interviews
6. Questionnaires
7. Scrum Guidance Body Expertise
8. AI-powered Scrum Project Tool

OUTPUTS

1. User Stories*
2. User Story Acceptance Criteria*
3. Updated Prioritized Product Backlog
4. Updated or Refined Personas

Figure 9-3: Create User Stories—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

Figure 9-4 illustrates the data flow for Create User Stories process. It maps how inputs such as business needs and personas are transformed into user stories and acceptance criteria.

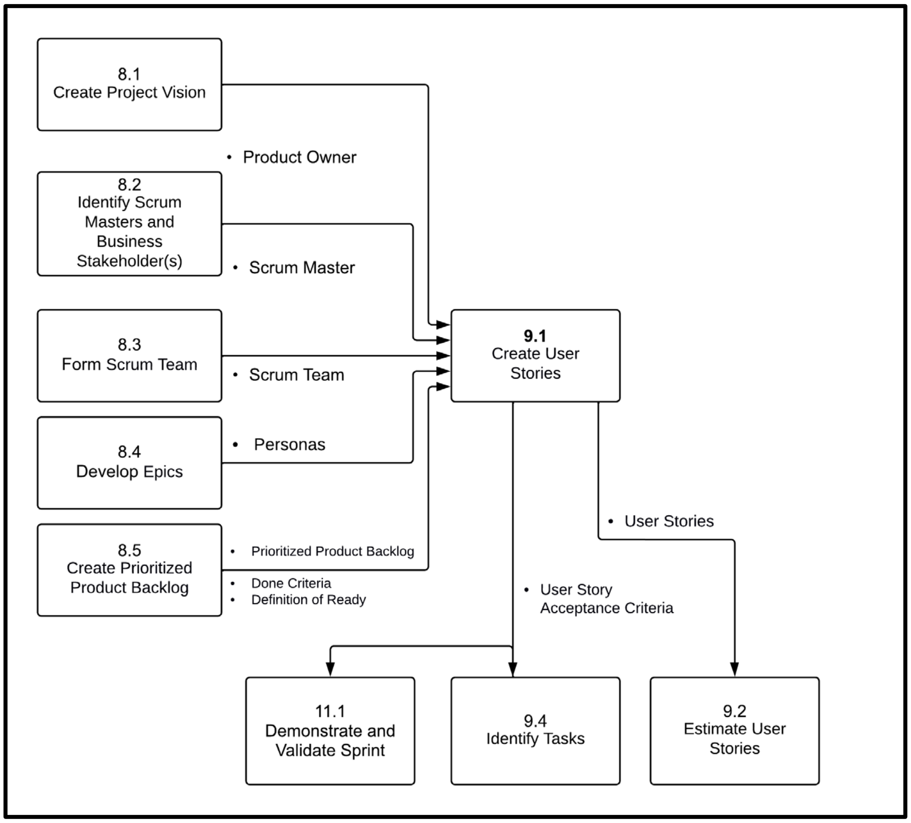


Figure 9-4: Create User Stories—Data Flow Diagram

9.1.1 Inputs

9.1.1.1 Scrum Core Team*

Described in section 3.2.1.

9.1.1.2 Prioritized Product Backlog*

Described in section 8.5.3.1.

9.1.1.3 Done Criteria*

Described in section 8.5.3.2.

9.1.1.4 Personas*

Described in section 8.4.3.2.

9.1.1.5 Definition of Ready*

The Definition of Ready defines the criteria that a User Story will have to satisfy before being considered for estimation or inclusion into a Sprint. For more information on the Definition of Ready see sections 5.3.2 and 8.5.3.3.

9.1.1.6 Business Stakeholder(s)

Described in section 8.2.3.2.

9.1.1.7 Epic(s)

Described in section 8.4.3.1.

9.1.1.8 Business Requirements

Described in section 8.5.1.6.

9.1.1.9 Laws and Regulations

Described in section 8.4.1.6.

9.1.1.10 Applicable Contracts

Described in section 8.4.1.7.

9.1.1.11 Scrum Guidance Body Recommendations

In the *Create User Stories* process, Scrum Guidance Body Recommendations may include information on rules, regulations, standards, and best practices required to create effective User Stories. For more information on Scrum Guidance Body Recommendations, see section 8.1.1.7.

9.1.2 Tools

9.1.2.1 User Story Writing Expertise*

The Product Owner, based on his or her interaction with the business stakeholders, business knowledge and expertise, and inputs from the team, develops the User Stories that will form the initial Prioritized Product Backlog for the project. The Prioritized Product Backlog represents the total sum of all the requirements that must be completed for the project. The objective of this exercise is to create elaborated and refined User Stories that can be estimated and committed to by the Scrum Team. At times, the Product Owner may bring a business analyst to assist with writing the User Stories. Although the Product Owner has the primary responsibility for writing User Stories and often carries out this exercise on his or her own, a User Story Writing Workshop can be held if desired.

9.1.2.2 User Story Workshops

Described in section 8.4.2.2.

9.1.2.3 User Group Meetings

Described in section 8.4.2.1.

9.1.2.4 Focus Group Meetings

Focus Group Meetings are a qualitative technique used to gauge and understand user needs and expectations about a proposed product. A small group of users are selected to form the focus group. This group may be selected randomly from a large pool of users or can be selected specifically to represent all the major personas being targeted. Focus Group Meetings normally adhere to a certain format in which the group is asked questions that they then discuss among themselves. Each Focus Group session can have its own rules for discussion as decided by the organizers. These meetings are usually held in the presence of a moderator.

9.1.2.5 Customer or User Interviews

Described in section 8.4.2.4.

9.1.2.6 Questionnaires

Described in section 8.4.2.5.

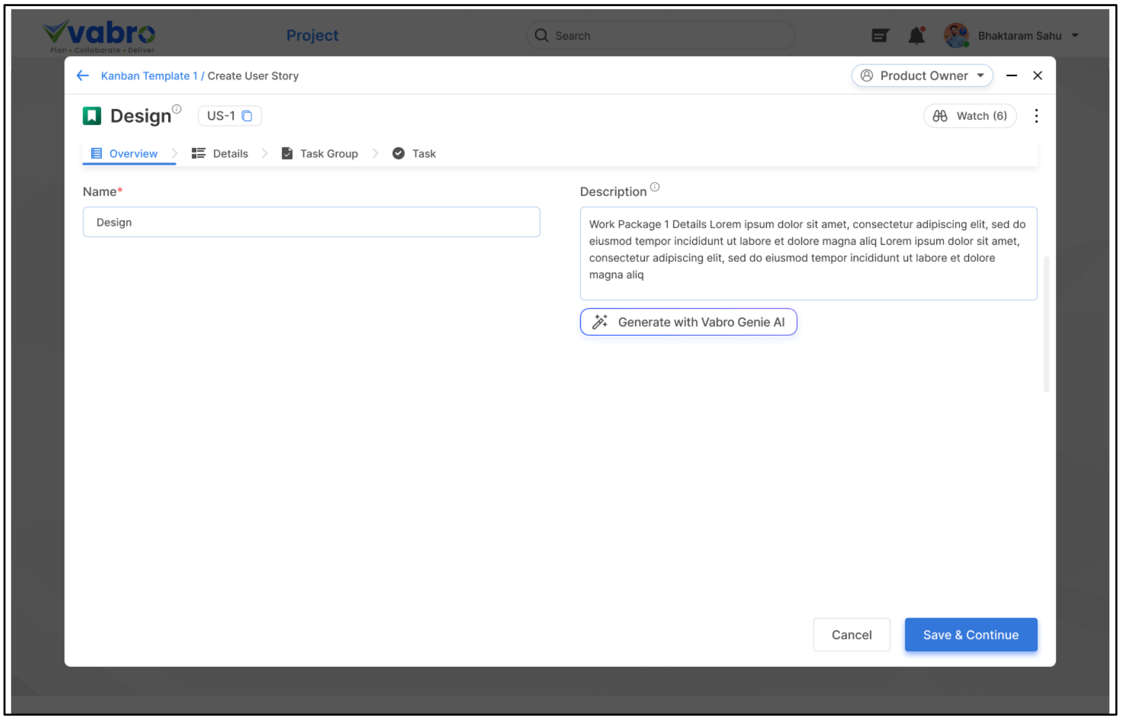
9.1.2.7 Scrum Guidance Body Expertise

While creating User Stories, the Scrum Guidance Body Expertise could refer to regulations, standards, and/or best practices for creating User Stories. There may also be a team of subject matter experts who are available to assist the Product Owner or provide guidance on how to create the User Stories. This team could include business analysts, lead architects, senior developers, Scrum experts, and other experienced people. This expert group is usually not the same team that will stay on and work on the project, as they tend to move from project to project and provide guidance to Scrum Teams when required. For more information on the Scrum Guidance Body see section 8.4.2.7.

9.1.2.8 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 9-5 displays the Vabro interface for adding user stories to the prioritized product backlog. It supports team collaboration and helps organize stories based on business value.

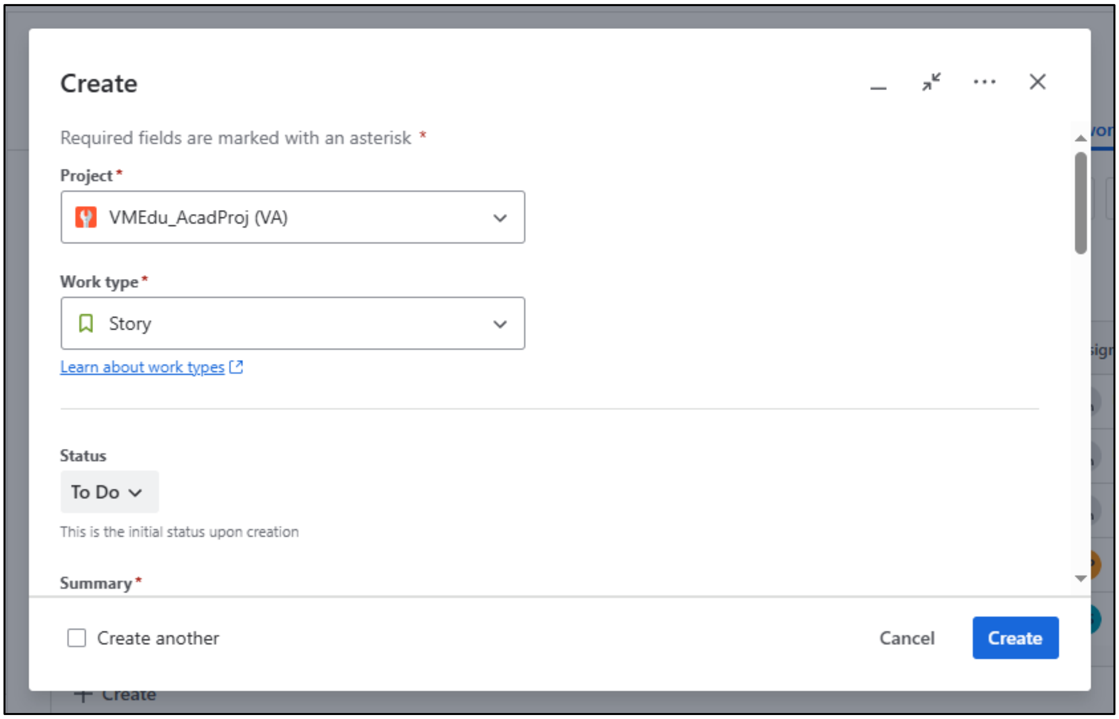


Name*: Design

Description: Work Package 1 Details 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

Figure 9-5: Adding User Stories in Prioritized Product Backlog (Source: Vabro)

Figure 9-6 shows Jira's interface for managing product backlog. It allows teams to add, prioritize, and track user stories within an agile framework.



Create

Required fields are marked with an asterisk *

Project *

VMEdu_AcadProj (VA)

Work type *

Story

Learn about work types

Status

To Do

This is the initial status upon creation

Summary *

☐ Create another

Cancel

Create

Figure 9-6: Adding User Stories in Product Backlog (Source: Jira)

9.1.3 Outputs

9.1.3.1 User Stories*

User Stories adhere to a specific, predefined structure and are a simplistic way of documenting the requirements and desired end-user functionality of a product. A User Story describes three things about a requirement—Who, What, and Why. The requirements expressed in User Stories are short, simple, and easy-to-understand statements. The predefined standard format results in enhanced communication among the business stakeholders and better estimations by the team. Some Epics/User Stories may initially be too large to handle within a single Sprint. Once Epics move 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. These updates to the Product Backlog are typically the result of changing business requirements. For more information on the Prioritized Product Backlog see section 8.5.3.1.

User Story Format:

As a <role/persona>, I should be able to <requirement> so that <benefit>.

User Story Example:

As a Database Administrator, I should be able to revert a selected number of database updates so that the desired version of the database is restored.

9.1.3.2 User Story Acceptance Criteria*

Every User Story has an associated Acceptance Criteria that is defined by the Product Owner and communicated to the Scrum Team. User Stories are subjective, so the Acceptance Criteria provide the objectivity required for the User Story to be considered as Done or not Done (i.e., accepted or rejected) during the *Demonstrate and Validate Sprint* process. Acceptance Criteria provide clarity to the team on what is expected of a User Story, remove ambiguity from requirements, and help in aligning expectations. During the Sprint Review Meeting, 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.

9.1.3.3 Updated Prioritized Product Backlog

The Prioritized Product Backlog created in the *Create Prioritized Product Backlog* process is updated with information on User Stories, Epics, estimates for User Stories, and the User Story Acceptance Criteria. For more information on the Prioritized Product Backlog, see section 8.5.3.1.

9.1.3.4 Updated or Refined Personas

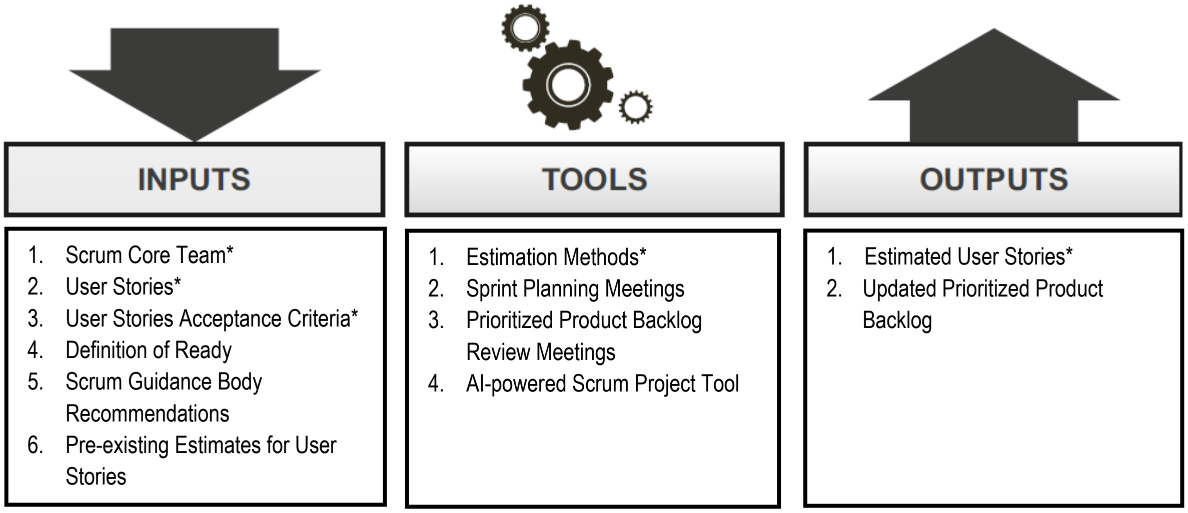
Personas are initially created in the *Develop Epic(s)* process. While writing User Stories, the Scrum Team may come to a collective decision that some of those initial personas created are inadequate and need refinement. If refining personas is required, it is normally done near the end of the *Create User Stories* process. For more information on personas see section 8.4.3.2.

9.2 Estimate User Stories

In this process, the Scrum Team, supported by the Scrum Master, estimates the User Stories and identifies the effort required to develop the functionality described in each User Story. Only User Stories that satisfy the Definition of Ready and are properly defined by the Product Owner are estimated by the team.

The Product Owner does not play an active role in estimating User Stories, but may be consulted to clarify any questions the Scrum Team might have related to the User Stories being estimated.

Figure 9-7 shows all the inputs, tools, and outputs for the *Estimate User Stories* process.



INPUTS

1. Scrum Core Team*
2. User Stories*
3. User Stories Acceptance Criteria*
4. Definition of Ready
5. Scrum Guidance Body Recommendations
6. Pre-existing Estimates for User Stories

TOOLS

1. Estimation Methods*
2. Sprint Planning Meetings
3. Prioritized Product Backlog Review Meetings
4. AI-powered Scrum Project Tool

OUTPUTS

1. Estimated User Stories*
2. Updated Prioritized Product Backlog

Figure 9-7: Estimate User Stories—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-8 explains the data flow for estimating user stories. It connects backlog items with planning tools to produce updated, estimated stories ready for commitment.

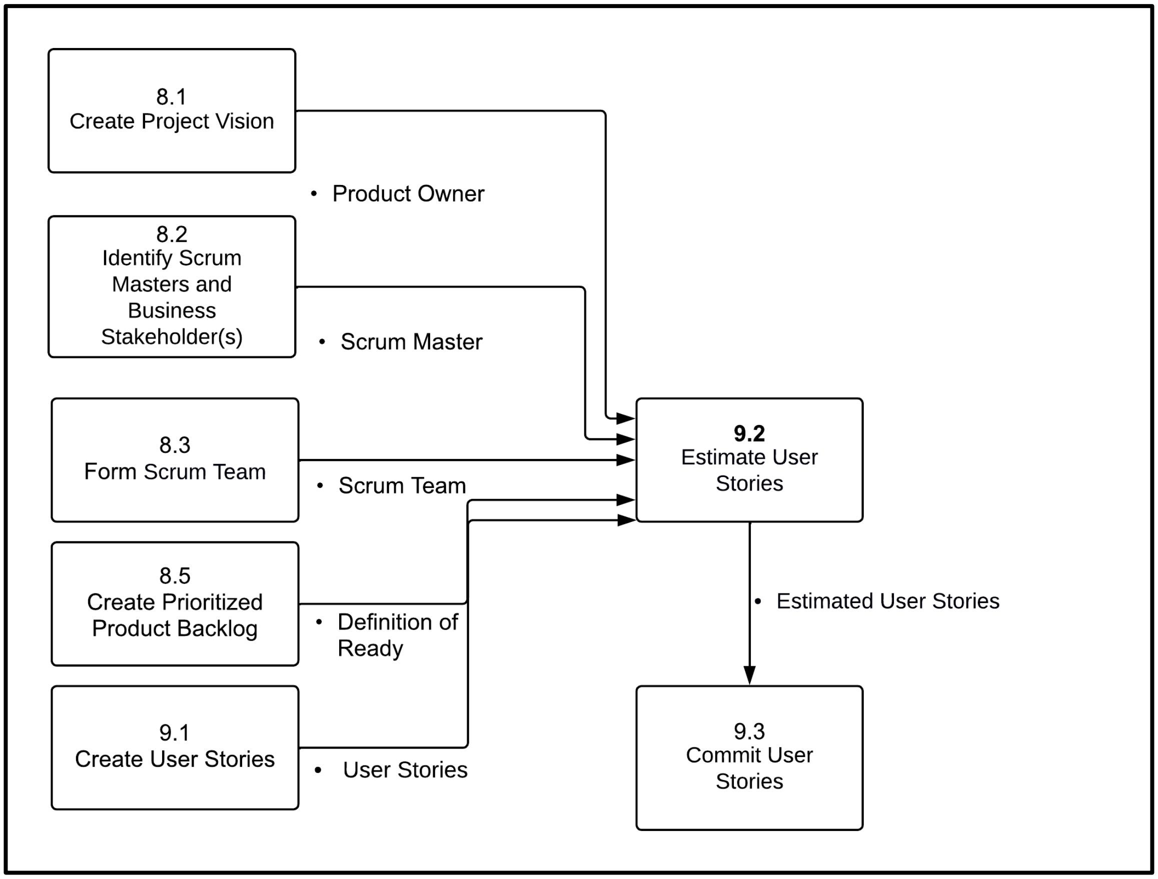


Figure 9-8: Estimate User Stories—Data Flow Diagram

9.2.1 Inputs

9.2.1.1 Scrum Core Team*

Described in section 3.2.1.

9.2.1.2 User Stories*

Described in section 9.1.3.1.

9.2.1.3 User Stories Acceptance Criteria*

Described in section 9.1.3.2.

9.2.1.4 Definition of Ready

The Definition of Ready defines the criteria that a User Story will have to satisfy before being considered for estimation or inclusion into a Sprint. For more information on the Definition of Ready see sections 5.3.2 and 8.5.3.3.

9.2.1.5 Scrum Guidance Body Recommendations

In the *Estimate User Stories* process, Scrum Guidance Body Recommendations may include information on the rules, regulations, standards, and best practices required to effectively estimate User Stories. For more information on the Scrum Guidance Body Recommendations, see section 8.1.1.7.

9.2.1.6 Pre-Existing Estimates for User Stories

Some pre-existing estimates of the effort needed to complete User Stories may already exist, particularly if similar User Stories were already completed in the same project or related past projects. The effort and time taken to complete similar User Stories can be used to derive estimates for the time needed to complete existing User Stories. Experts who have implemented similar project requirements in the past may also be able to provide some effort estimates for the User Stories. These pre-existing estimates will help the Scrum Team to thoroughly estimate and commit User Stories for the Sprint. It is important to ensure that the Scrum Team creates their own estimates of User Stories before the User Stories are committed as a part of the Sprint, instead of solely relying on any pre-existing estimates available for User Stories.

9.2.2 Tools

9.2.2.1 Estimation Methods*

As new or updated User Stories are refined in the Product Backlog, the Scrum Team will assign or update any pre-existing estimates for each User Story. Relative sizing or story points can be used for estimating the overall size of a User Story or feature. This approach assigns a story point value based on an overall assessment of the size of a User Story with consideration given to its associated risk, the amount of effort required, and the level of complexity. 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 the story point calibration for each team is different so the number of User Story points completed cannot be used for comparison across teams. Also, the estimation method selected depends on the level of estimation detail required by the team.

Some techniques that can be used to estimate User Stories are as follows:

1. **Wideband Delphi**

   Wideband Delphi is a group-based estimation technique for determining how much work is involved and how long it will take to complete the work. Individuals within the team anonymously provide estimations for each item and the initial estimates are plotted on a chart. The team then discusses the factors that influenced their own estimates and proceed to a second round of estimation. This process is repeated until the individual estimates are close to each other and a consensus for the final estimate can be reached.

2. **Planning Poker**

   Planning Poker, also called Estimation Poker, is a derivative of the Wideband Delphi technique that uses consensus to estimate the relative sizes of User Stories, or the effort required to create them.

   In Planning Poker, each team member is assigned a deck of cards. Each card is numbered in sequence with each number representing the complexity of the User Story (or task) in terms of time or effort. The Scrum Team members assess the User Story (or task) to better understand it. Each member then picks a card from his/her deck that represents his/her estimate for the time or effort required to complete the User Story (or task). If the majority or all team members select the same card, then the estimate indicated by that card value will be the estimate recorded for that item. If there is no consensus, then the team members discuss reasons for selecting different cards or estimates. After this discussion, each member picks a card again. This sequence continues until all the assumptions are understood, misunderstandings are resolved, and a majority or consensus is reached. Planning Poker advocates greater interaction and enhanced communication among Scrum Team members. It also facilitates independent thinking by participants, thus avoiding the phenomenon of group think.

3. **Fist of Five**

   Fist of Five is a simple and fast mechanism that can be used as an estimation tool, as well as a general group consensus-building technique. After an initial discussion about a particular User Story (or task) being estimated, Scrum Team members are each asked to vote on a scale of one to five using their fingers, with the number of fingers indicating the relative estimate value. Team members with outlier estimates (i.e., the highest and lowest values) explain their rational for their estimates to the group and these are discussed. After this discussion, another Fist of Five round is conducted, or a collective decision is made.

   The value in using this technique is not only consensus building but also driving discussion because team members are asked to explain the reasons for their estimates. They are also given the opportunity to express any issues or concerns. Used as a general consensus building technique, the proposal or pending decision under consideration is initially discussed, then the team members vote based on their level of agreement and desire for discussion as follows:

   One finger: I disagree with the group's conclusion and have major concerns.
   Two fingers: I disagree with the group's conclusion and would like to discuss some minor issues.
   Three fingers: I am not sure and would like to go with the group's consensus conclusion.
   Four fingers: I agree with the group's conclusion and would like to discuss some minor issues.
   Five fingers: I wholeheartedly agree with the group's conclusion.

4. **Affinity Estimation**

   Affinity Estimation, such as T-shirt sizing, is a technique used to quickly estimate a large number of User Stories. Using sticky notes or index cards and tape, the team places User Stories on a wall or other surface, in order from small to large. Each team member begins with a subset of User Stories from the Prioritized Product Backlog to place in order by their relative size. This initial placement is done in silence. Once everyone has placed their User Stories on the wall, the team reviews all of the placements and can move User Stories around as appropriate. This second part of the exercise involves a discussion about the placements. Finally, the Product Owner will indicate some sizing categories on the wall. These categories can be small, medium, or large, or they may be numbered using story point values to indicate relative size. The team will then move the User Stories into these categories as the last step in the process. Some key benefits of this approach are that the process is very transparent, visible to everyone, and is easy to conduct.

9.2.2.2 Sprint Planning Meetings

During Sprint Planning Meetings, the User Stories are discussed by the Scrum Core Team. If it is not already done during the creation or the refining of the Product Backlog, each User Story is evaluated and assigned a high-level estimate based on relative story points. See also sections 9.3.2.1, 9.4.2.1, 9.5.2.1, and 9.6.2.1.

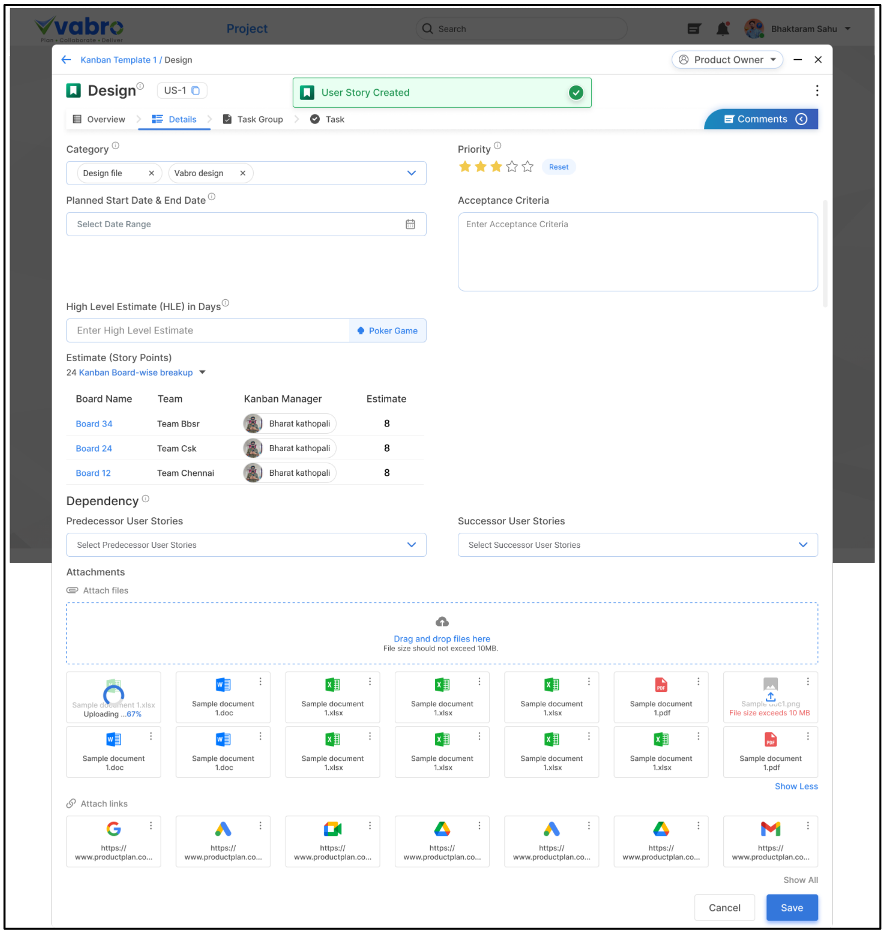
9.2.2.3 Prioritized Product Backlog Review Meetings

Prioritized Product Backlog Review Meetings are held as part of the *Refine Prioritized Product Backlog* process. Information from these meetings provides additional clarity about the User Stories and helps in determining their estimates.

9.2.2.4 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 9-9 shows Vabro's interface used for estimating user stories. It provides tools for assigning effort values and organizing sprint-ready tasks.



vabro Plan • Collaborate • Deliver | Project | Search | Bhaktaram Sahu

Kanban Template 1 / Design | Product Owner

Design | US-1 | User Story Created

Overview > Details > Task Group > Task | Comments

Category: Design file × | Vabro design ×

Priority: ★★★☆☆ Reset

Planned Start Date & End Date: Select Date Range

Acceptance Criteria: Enter Acceptance Criteria

High Level Estimate (HLE) in Days: Enter High Level Estimate | Poker Game

Estimate (Story Points)

24 Kanban Board-wise breakup

Board Name	Team	Kanban Manager	Estimate
Board 34	Team Bbsr	Bharat kathopali	8
Board 24	Team Csk	Bharat kathopali	8
Board 12	Team Chennai	Bharat kathopali	8

Dependency

Predecessor User Stories: Select Predecessor User Stories

Successor User Stories: Select Successor User Stories

Attachments

Attach files

Drag and drop files here
File size should not exceed 10MB.

Sample document 1.xlsx Uploading ...67% | Sample document 1.doc | Sample document 1.xlsx | Sample document 1.xlsx | Sample document 1.xlsx | Sample document 1.pdf | Sample doc1.png File size exceeds 10 MB

Sample document 1.doc | Sample document 1.doc | Sample document 1.xlsx | Sample document 1.xlsx | Sample document 1.xlsx | Sample document 1.xlsx | Sample document 1.pdf

Show Less

Attach links

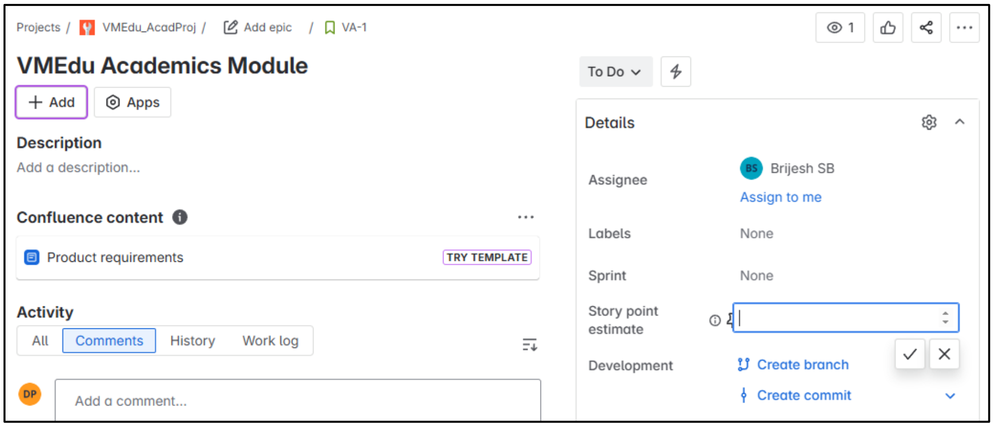
https://www.productplan.co... | https://www.productplan.co... | https://www.productplan.co... | https://www.productplan.co... | https://www.productplan.co... | https://www.productplan.co... | https://www.productplan.co...

Show All

Cancel | Save

Figure 9-9: User Stories Estimation (Source: Vabro)

Figure 9-10 depicts Jira's view of user story estimation. It enables teams to assign story points and align user stories with sprint planning workflows.



Projects / VMEdu_AcadProj / Add epic / VA-1 | 1

VMEdu Academics Module

+ Add | Apps

Description
Add a description...

Confluence content

Product requirements | TRY TEMPLATE

Activity
All | Comments | History | Work log

DP Add a comment...

To Do

Details

Assignee: BS Brijesh SB
Assign to me

Labels: None

Sprint: None

Story point estimate:

Development: Create branch
Create commit

Figure 9-10: Estimating User Stories in Scrum (Source: Jira)

9.2.3 Outputs

9.2.3.1 Estimated User Stories*

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.

9.2.3.2 Updated Prioritized Product Backlog

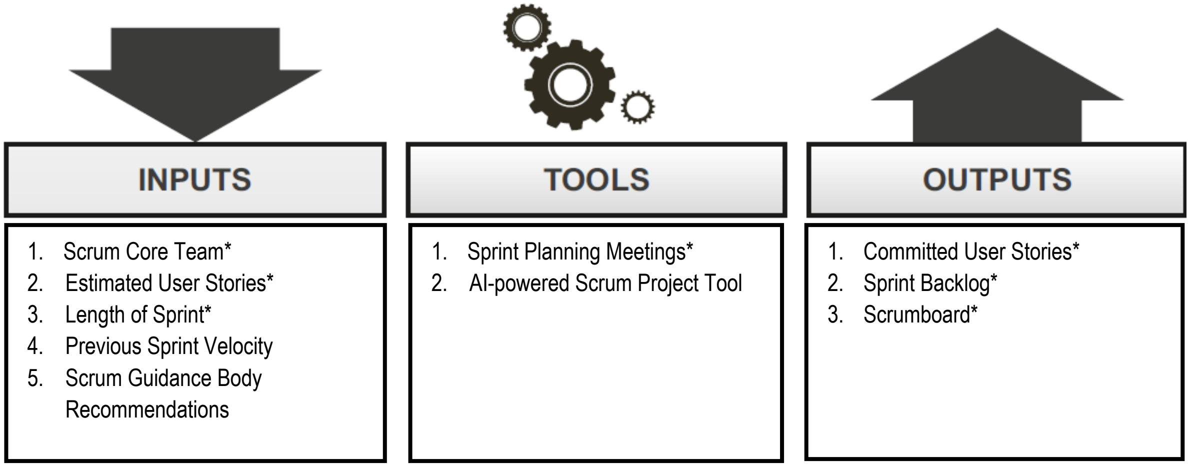
Described in section 9.1.3.3.

9.3 Commit User Stories

In this process, the Scrum Team commits to delivering a set of User Stories for the Sprint. The decision on which User Stories will be committed to is based on the relative value-based priority of the User Stories and the estimated effort and team velocity for one Sprint. As part of this process, the Scrum Team starts the creation of the Sprint Backlog, which contains the committed User Stories that are assigned to a particular Sprint. The backlog is refined further with task-level details as Sprint Planning continues.

With this commitment from the Scrum Team given at the beginning of a Sprint as part of Sprint Planning, the content of the Sprint is defined and cannot be changed once the Sprint implementation phase has begun.

Figure 9-11 shows all the inputs, tools, and outputs for the *Commit User Stories* process.



INPUTS

1. Scrum Core Team*
2. Estimated User Stories*
3. Length of Sprint*
4. Previous Sprint Velocity
5. Scrum Guidance Body Recommendations

TOOLS

1. Sprint Planning Meetings*
2. AI-powered Scrum Project Tool

OUTPUTS

1. Committed User Stories*
2. Sprint Backlog*
3. Scrumboard*

Figure 9-11: Commit User Stories—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-12 illustrates the data flow of the commit user stories process. It maps how selected and estimated stories become sprint tasks with clear acceptance criteria.

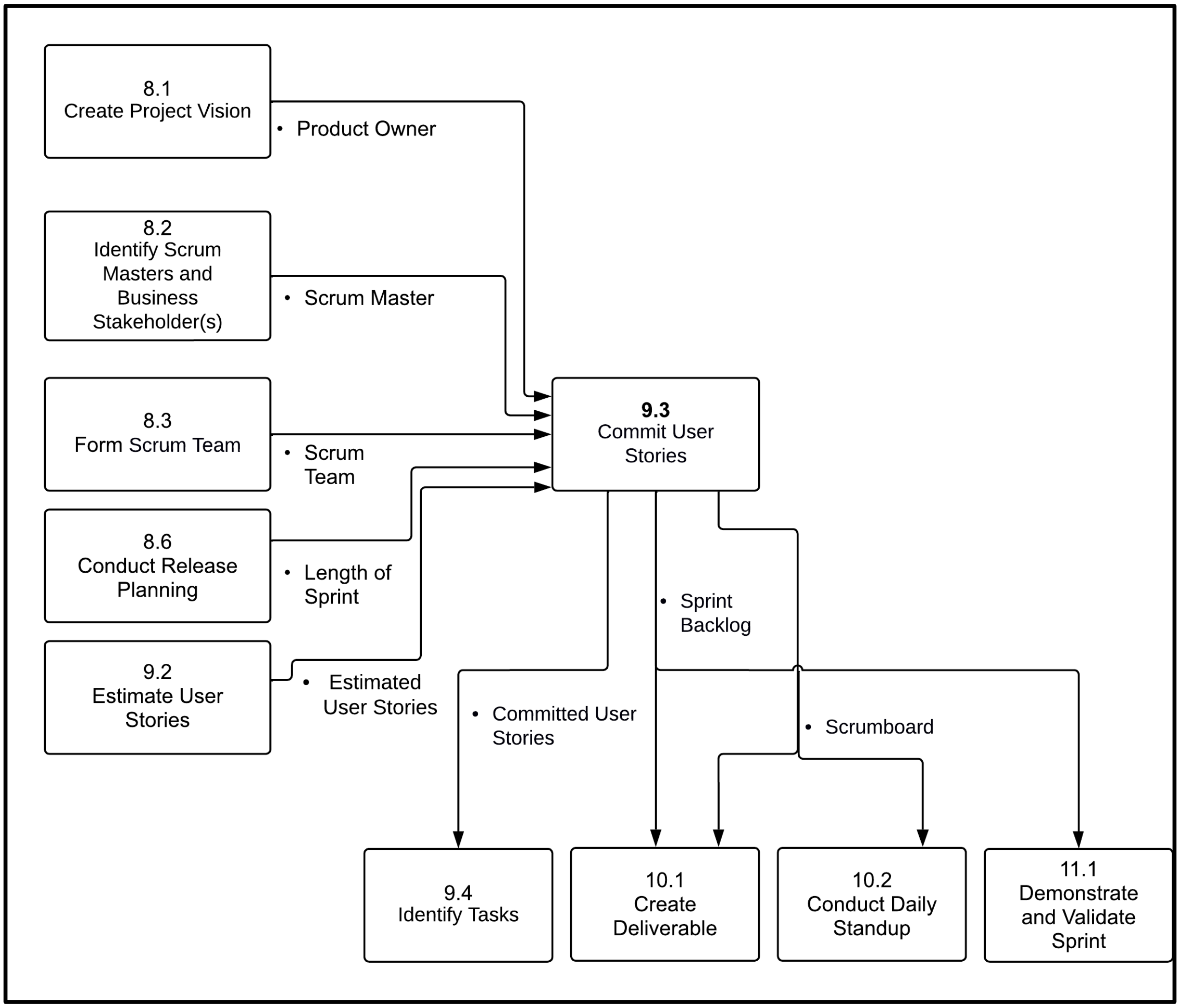


Figure 9-12: Commit User Stories—Data Flow Diagram

9.3.1 Inputs

9.3.1.1 Scrum Core Team*

Described in section 3.2.1.

9.3.1.2 Estimated User Stories*

Described in section 9.2.3.1.

9.3.1.3 Length of Sprint*

Described in section 8.6.3.2.

9.3.1.4 Previous Sprint Velocity

Sprint Velocity is the rate at which the team can complete the work in a Sprint. It is usually expressed in the same units as those used for estimation, normally story points or ideal time. A record of the team's velocity for each Sprint is maintained and used as a reference point for future Sprints. The previous Sprint Velocity becomes the most important factor in determining the amount of work the team can commit to in a subsequent Sprint. Any changes in the situation or conditions since the last Sprint are accounted for to ensure a better estimation of the Sprint Velocity for the upcoming Sprint.

9.3.1.5 Scrum Guidance Body Recommendations

In the *Commit User Stories* process, recommendations from the Scrum Guidance Body may include information on the rules, regulations, standards, and best practices required for the team to effectively commit User Stories and incorporate them into the Sprint Backlog. For more information on Scrum Guidance Body recommendations, see 8.1.1.7.

9.3.2 Tools

9.3.2.1 Sprint Planning Meetings*

In the Sprint Planning Meeting, the Scrum Team comes together to plan the work to be done in the upcoming Sprint. The Product Owner is present during this meeting in case any clarifications of the User Stories or priorities are needed. The team reviews the estimates for those User Stories that are at the top of the Prioritized Product Backlog. To help ensure that the group stays on topic, this meeting should be Time-boxed, with the standard length limited to two hours per week of Sprint duration (e.g., four hours for two-week Sprints). This assists in preventing the tendency to stray into discussions that should occur at other meetings (such as the Release Planning Meeting or the Sprint Review Meeting). As part of this meeting, the entire Scrum Team will commit to delivering a subset of User Stories from the Prioritized Product Backlog in the upcoming Sprint. These committed User Stories are then incorporated into the Sprint Backlog. For more information on Sprint Planning Meetings, see sections 9.2.2.2, 9.4.2.1, 9.5.2.1, and 9.6.2.1.

9.3.2.2 AI-powered Scrum Project Tool

Described in section 2.4.4

9.3.3 Outputs

9.3.3.1 Committed User Stories*

The Scrum Team commits to a subset of the estimated User Stories that they believe they can complete in the upcoming Sprint based upon team velocity. The committed User Stories should always be selected based on the priorities defined by the Product Owner (as incorporated into the Prioritized Product Backlog).

9.3.3.2 Sprint Backlog*

The list of the User Stories to be executed by the Scrum Team in the Sprint is called the Sprint Backlog. This is a subset of the Prioritized Product Backlog that contains the committed User Stories that are assigned to a particular Sprint. The Sprint Backlog will be further refined with task-level details as Sprint Planning continues.

It is common practice for the Sprint Backlog User Stories (and associated tasks) to be represented on a Scrumboard or similar task board, which provides a constantly visible depiction of the current status of the User Stories in the Product Backlog.

Once the Sprint Backlog is finalized and committed to by the Scrum Team, new User Stories should not be added. If new requirements arise during a Sprint, they should be added to the Prioritized Product Backlog to be considered for a future Sprint.

9.3.3.3 Scrumboard*

It is important to track the progress of a Sprint and to know where the Scrum Team stands in terms of completing the User Stories (and tasks) in the Sprint Backlog**.** A variety of tools can be used to track the work in a Sprint, but one of the most common is a Scrumboard, also known as a task board or a progress chart. Scrum's transparency comes from openly viewable information tools like the Scrumboard, which shows the ongoing progress of the team. The team uses a Scrumboard to plan and track progress during each Sprint.

The most basic version of a Scrumboard has the board divided into three sections— "Work Not Started" (also referred to as "To Do"), "Work In Progress" (also referred to as "In Progress"), and "Completed Work" (also referred to as "Complete." Sticky notes representing each User Story and their associated tasks are placed in the appropriate category to reflect the status of the work to be complete in the current Sprint. The task notes are moved forward to the next category as the work progresses.

A typical Scrumboard is shown in Figure 9-13. The Scrumboard shows all the User Stories in the left column and has three columns labelled "To Do," "In Progress," and "Complete." As tasks associated with the User Stories are identified and worked on in later processes, those tasks would be depicted under their respective columns.

Figure 9-13 displays a typical Scrumboard in Vabro. It helps visualize sprint progress by showing committed stories, their statuses, and related tasks.

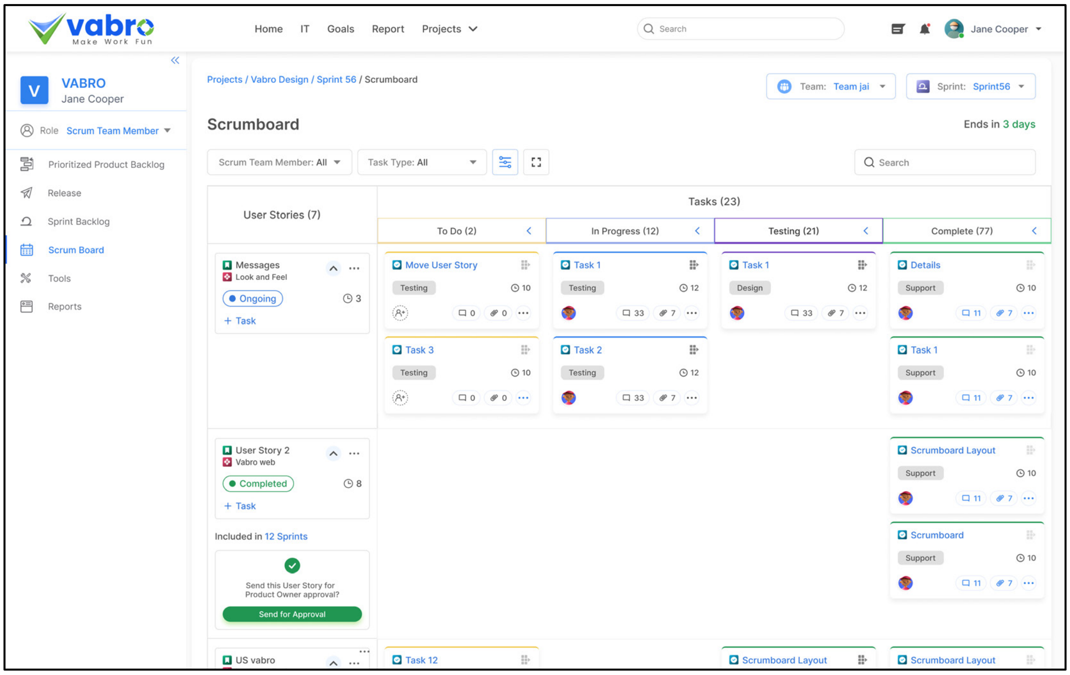


Figure 9-13: Typical Scrumboard (Source: Vabro)

Variations of the typical Scrumboard may be used to depict the status of the team's work more accurately. For example, one variation contains an additional "Testing" column to indicate that the task is complete, but the work result is currently being tested. The "Complete" column in this case is used to represent those tasks that are fully complete and also successfully tested. Instead of a Testing column, the team may include any other column on the Scrumboard that the team thinks will be helpful to track progress.

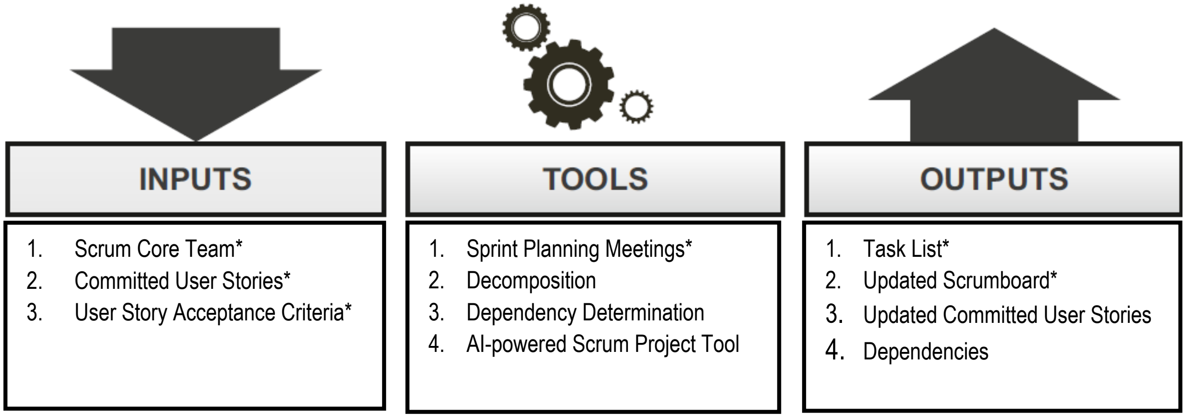
The Scrumboard can be maintained manually on paper or on a large whiteboard, or it can be maintained electronically in a spreadsheet or using an AI-powered Scrum Project Tool. The Scrum Team should change or add to the Scrumboard as required so that the Scrumboard continues to provide accurate visual information and control about the status of the work being done (as agreed to and committed by the team).

9.4 Identify Tasks

In this process, the committed User Stories are decomposed into specific tasks and compiled into a task list. Identifying tasks could either be done at the beginning of the Sprint for all committed User Stories or before the team starts working on the tasks required for each User Story.

The Product Owner does not play an active role in identifying tasks but needs to be available to answer any questions from the Scrum Team that may arise while decomposing the User Stories into tasks.

Figure 9-14 shows all the inputs, tools, and outputs for the *Identify Tasks* process.



INPUTS

1. Scrum Core Team*
2. Committed User Stories*
3. User Story Acceptance Criteria*

TOOLS

1. Sprint Planning Meetings*
2. Decomposition
3. Dependency Determination
4. AI-powered Scrum Project Tool

OUTPUTS

1. Task List*
2. Updated Scrumboard*
3. Updated Committed User Stories
4. Dependencies

Figure 9-14: Identify Tasks—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-15 depicts the data flow for identifying tasks. It illustrates how stories are broken into tasks and arranged in the Scrumboard to support sprint planning.

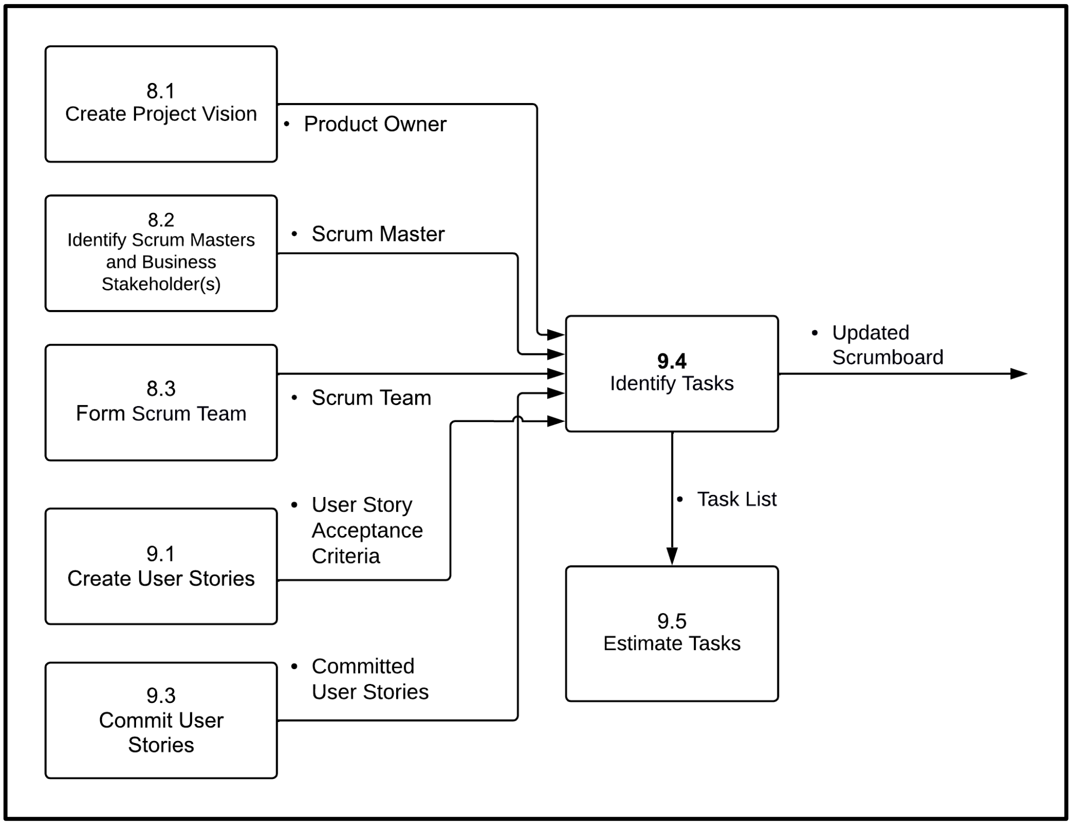


Figure 9-15: Identify Tasks—Data Flow Diagram

9.4.1 Inputs

9.4.1.1 Scrum Core Team*

Described in section 3.2.1.

9.4.1.2 Committed User Stories*

Described in section 9.3.3.1.

9.4.1.3 User Story Acceptance Criteria*

Described in section 9.1.3.2.

The Product Owner must ensure that the defined Acceptance Criteria are appropriate for the User Stories and must also provide clarity regarding the requirements for the Scrum Team. Understanding Acceptance Criteria by the Scrum Team helps them determine which tasks are needed to satisfy the requirements of the User Story.

Acceptance testing refers to the assessment of the ability of the completed deliverable to meet its Acceptance Criteria. This provides information to the Product Owner to help decide about approving or rejecting the deliverables. The Acceptance Criteria should be crisp, unambiguous, and specific. They should be defined to ensure that the team is able to verify that the outcomes are in alignment with the sponsor organization's goals and objectives.

9.4.2 Tools

9.4.2.1 Sprint Planning Meetings*

In the Sprint Planning Meetings, the Scrum Team convenes to plan the work to be done in the upcoming Sprint. 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 the Acceptance Criteria. The Product Owner is present at this meeting in case clarification is required related to the committed User Stories to help the team make design decisions. For more information on Sprint Planning Meetings, see sections 9.2.2.2, 9.3.2.1, 9.5.2.1, and 9.6.2.1.

9.4.2.2 Decomposition

The Scrum Team uses decomposition to break down the User Stories in the upcoming Sprint into detailed tasks. The User Stories should be sufficiently decomposed to a level that provides the Scrum Team with adequate information needed to create the deliverables using the tasks captured in the Task List.

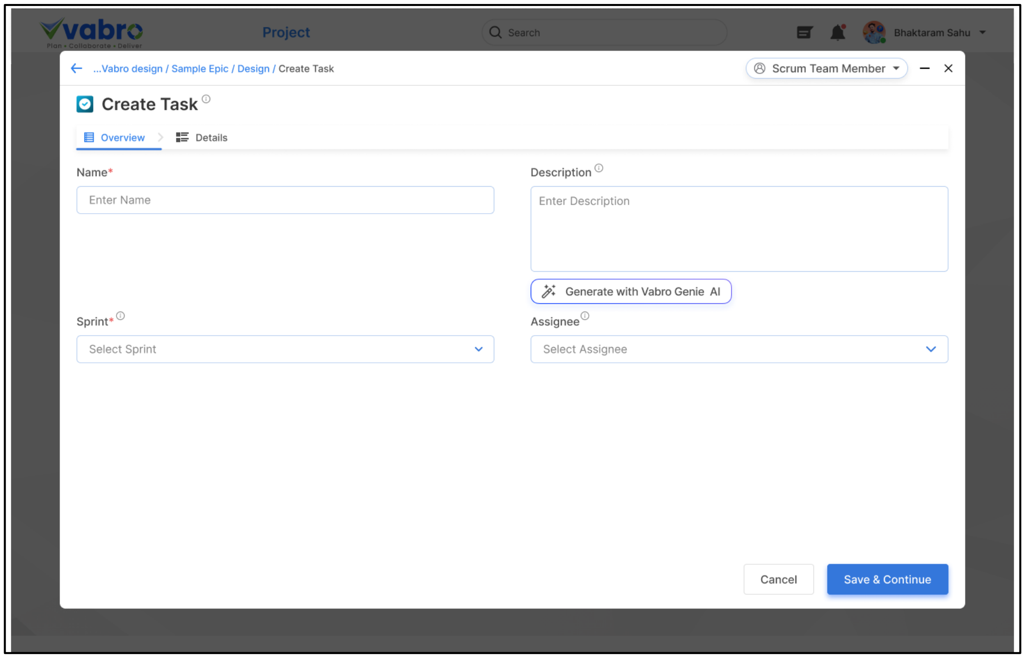
9.4.2.3 Dependency Determination

Once the Scrum Team has selected the User Stories for the upcoming Sprint, the team should then consider any dependencies, including those related to the availability of people, as well as any technical dependencies. Properly documenting dependencies helps the Scrum Team determine the relative order in which tasks should be executed to create the Sprint deliverables. Dependencies also highlight the relationship and interaction between tasks both within the Scrum Team working on a given Sprint and across other Scrum Teams in the project. For more information on determining dependencies, see section 8.5.2.6.

9.4.2.4 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 9-16 shows the Vabro interface used to create tasks from user stories. It supports task assignments, estimates, and dependency tracking.


vabro
Plan • Collaborate • Deliver
Project
Search
Bhaktaram Sahu
...Vabro design / Sample Epic / Design / Create Task
Scrum Team Member
Overview
Details
Generate with Vabro Genie AI
Cancel
Save & Continue

Create Task

Name*
Enter Name

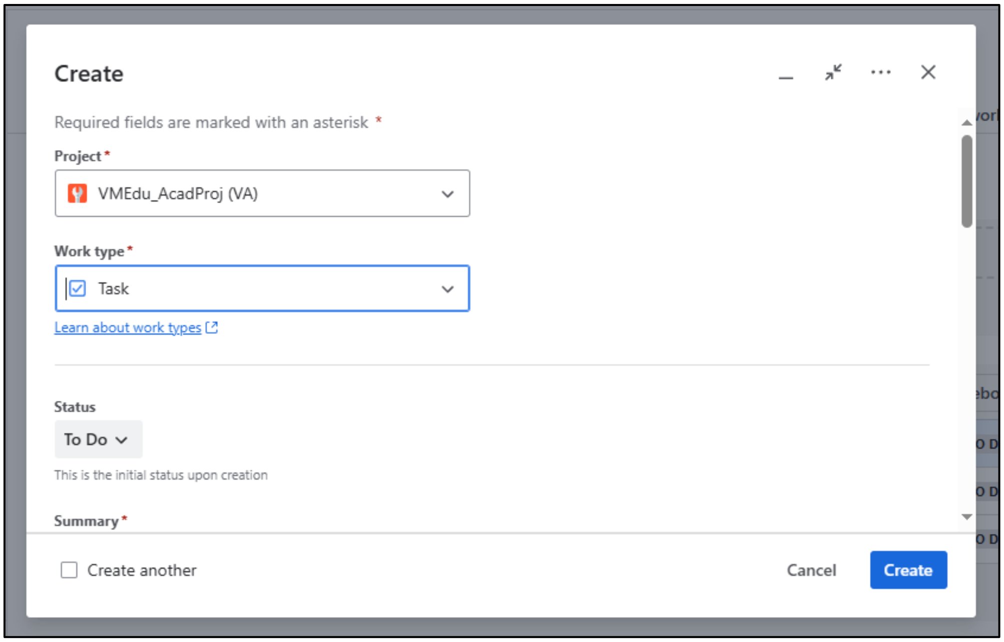
Description
Enter Description

Sprint*
Select Sprint

Assignee
Select Assignee

Figure 9-16: Task Creation in Scrum Project Tool (Source: Vabro)

Figure 9-17 displays Jira's task creation interface, enabling teams to generate and manage sprint tasks linked to specific user stories.



Create

Required fields are marked with an asterisk *

Project*
VMEdu_AcadProj (VA)

Work type*
Task

Status
To Do
This is the initial status upon creation

Summary*

□ Create another

Figure 9-17: Task Creation in Scrum Project Tool (Source: Jira)

9.4.3 Outputs

9.4.3.1 Task List*

The Task List is a comprehensive list that contains all the tasks which the Scrum Team has committed to for the current Sprint and their corresponding descriptions. The level of granularity to which the tasks are decomposed is decided by the Scrum Team. The Task List must include any testing and integration efforts so that the product increment from the Sprint can be successfully integrated into the deliverables from previous Sprints. The Task List is used by the Scrum Team during Sprint Planning Meetings to update the Sprint Backlog and to create the Sprint Burndown Chart. It is also used to determine if the team needs to reduce its commitment, or if they can take on additional User Stories during Sprint Planning for the next Sprint.

9.4.3.2 Updated Scrumboard*

As tasks are identified, the Scrumboard is updated to show the tasks associated with each User Story. Tasks are typically shown on sticky notes placed on a physical Scrumboard or as entries under the applicable User Stories when using an electronic AI-powered Scrum Project Tool. During implementation, as the team adds, assigns, and updates tasks being worked on, the Scrumboard keeps getting updated with the additional tasks and the status of each task. If the team has estimated the tasks, the task estimates are also depicted on the Scrumboard. In the example in Figure 9-18, the Scrumboard shows four User Stories. User Story 1, 2, and 3 have been decomposed into tasks, but User Story 4 has not yet been decomposed into tasks. At the beginning of a Sprint, all tasks for that Sprint are placed in the ‘To Do’ column and are subsequently moved forward according to their progress. For example, User Story 1 has 7 tasks, all of which are categorized as “To Do,” which indicates that the Scrum Team has not started working on any of those tasks.

For more information on the Scrumboard, see sections 9.3.3.3 and 10.1.1.3.

Figure 9-18 depicts a scrumboard with identified tasks for multiple user stories. Shows decomposition status, with tasks in 'to do' column, and differentiates user stories that have or have not been broken into tasks yet. Useful for tracking sprint readiness.

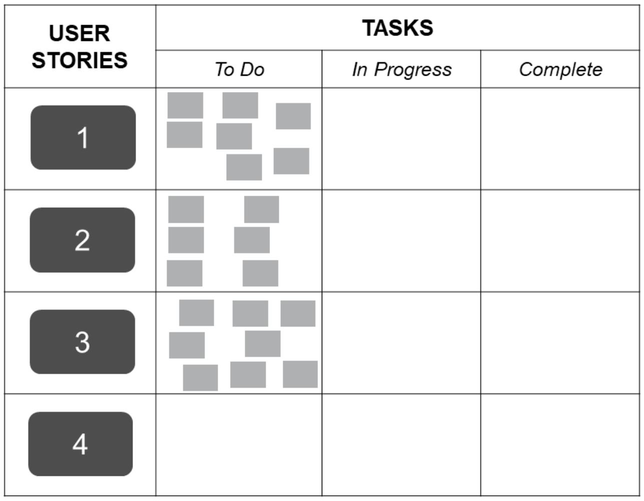


Figure 9-18: Scrumboard with Identified Tasks

9.4.3.3 Updated Committed User Stories

The User Stories are updated during this process. Updates can include revisions to the original User Story estimates based on task creation and complexity factors discussed during the Sprint Planning Meeting. Committed User Stories are described in section 9.3.3.1.

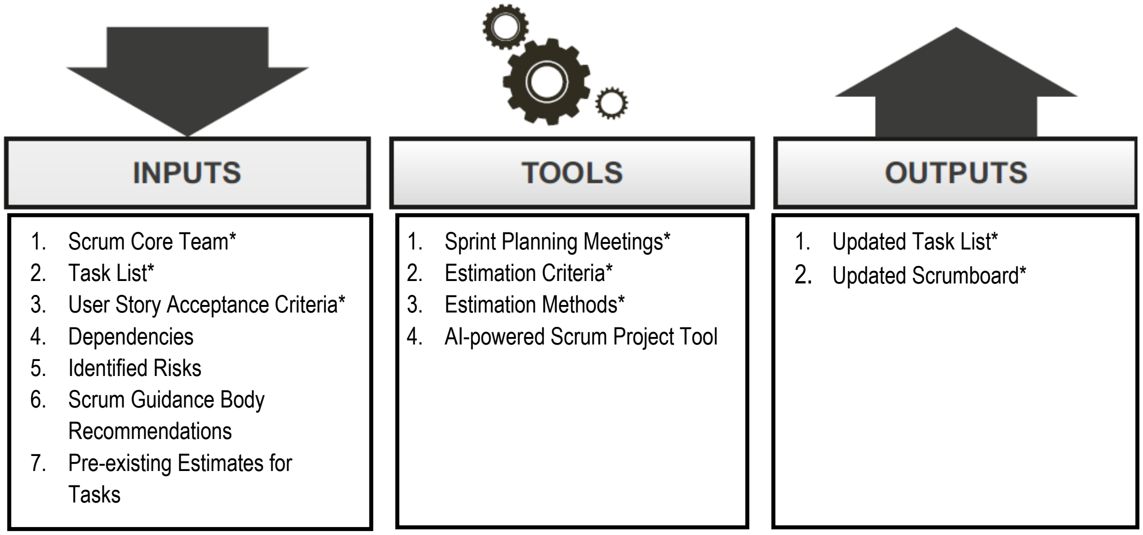
9.4.3.4 Dependencies

Dependencies describe the relationship and interaction between different tasks in a project and can be classified as mandatory, discretionary, internal, or external, as discussed in section 8.5.2.6.

9.5 Estimate Tasks

This is an optional process which involves creating task estimates if the Scrum Team sees value in doing so. In this process, the Scrum Team estimates the effort required to accomplish each task in the Task List. Task estimates could either be determined at the beginning of the Sprint for all User Stories/tasks relevant to that Sprint, or for each task just before the team starts working on the particular User Story/task. The estimation can be done using the same methods that were used for the *Estimate User Stories* process.

Figure 9-19 shows all the inputs, tools, and outputs for the *Estimate Tasks* process.



INPUTS

1. Scrum Core Team*
2. Task List*
3. User Story Acceptance Criteria*
4. Dependencies
5. Identified Risks
6. Scrum Guidance Body Recommendations
7. Pre-existing Estimates for Tasks

TOOLS

1. Sprint Planning Meetings*
2. Estimation Criteria*
3. Estimation Methods*
4. AI-powered Scrum Project Tool

OUTPUTS

1. Updated Task List*
2. Updated Scrumboard*

Figure 9-19: Estimate Tasks—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-20 is a data flow diagram representing the "estimate tasks" process. Shows the flow from task inputs to estimation activities and the resulting outputs such as updated scrumboard and task lists.

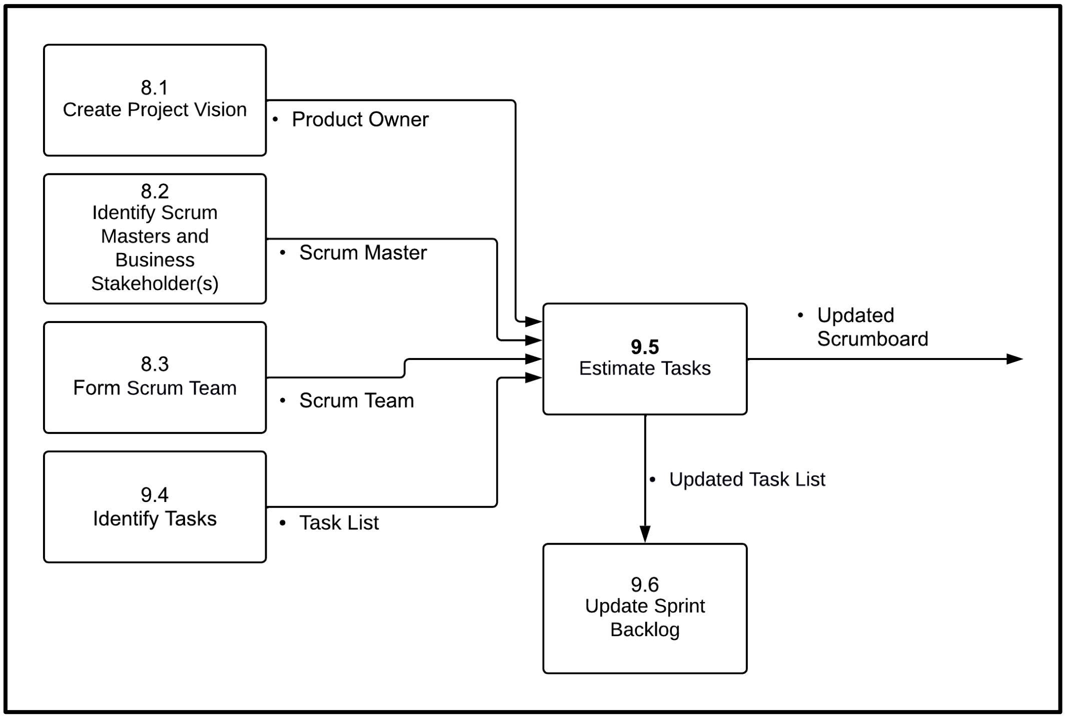


Figure 9-20: Estimate Tasks—Data Flow Diagram

9.5.1 Inputs

9.5.1.1 Scrum Core Team*

Described in section 3.2.1.

9.5.1.2 Task List*

Described in section 9.4.3.1.

9.5.1.3 User Story Acceptance Criteria*

Every User Story has associated Acceptance Criteria. User Stories are subjective, so the Acceptance Criteria provide the objectivity required for the User Story to be considered as Done or not Done during the Sprint Review (that occurs during the *Demonstrate and Validate Sprint* process). For more information on the User Story Acceptance Criteria, see section 9.1.3.2.

9.5.1.4 Dependencies

Described in section 9.4.3.4.

9.5.1.5 Identified Risks

Described in section 8.4.3.4.

9.5.1.6 Scrum Guidance Body Recommendations

In the *Estimate Tasks* process, Scrum Guidance Body Recommendations may include information on rules, regulations, standards, and best practices required to effectively estimate tasks in the Task List. For more information on Scrum Guidance Body Recommendations, see section 8.1.1.7.

9.5.1.7 Pre-existing Estimates for Tasks

Some pre-existing task estimates may be considered by the Scrum Team members when estimating tasks. Similar tasks may have been previously completed in the same project, or in other past projects, and the effort and time taken to complete those similar tasks can help the Scrum Team to create better task estimates during this process. Experts who have worked on similar tasks in the past may also be able to provide some effort estimates for tasks. However, it is important to ensure that the Scrum Team creates their own task estimates, instead of solely relying on any pre-existing estimates. Pre-existing task estimates may also help the Scrum Team reevaluate the commitment made to the Product Owner at the User Story level.

9.5.2 Tools

9.5.2.1 Sprint Planning Meetings*

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 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 effort for the User Stories to be completed in the Sprint. One of the key benefits of this technique is that it enables the team to have a shared perspective of the User Stories and requirements so that they can reliably estimate the effort required. For more information on Sprint Planning Meetings, see sections 9.2.2.2, 9.3.2.1, 9.4.2.1, and 9.6.2.1.

9.5.2.2 Estimation Criteria*

Estimation criteria can be expressed in numerous ways, with two common examples being story points and ideal time. Story point values are used to represent relative or comparative effort to complete tasks. Whereas ideal time normally describes the number of hours a Scrum Team member works exclusively on developing the project's deliverables, without including any time spent on other activities or work that is outside the project. Estimation criteria make it easier for the Scrum Team to estimate effort and enable them to evaluate and address inefficiencies when necessary.

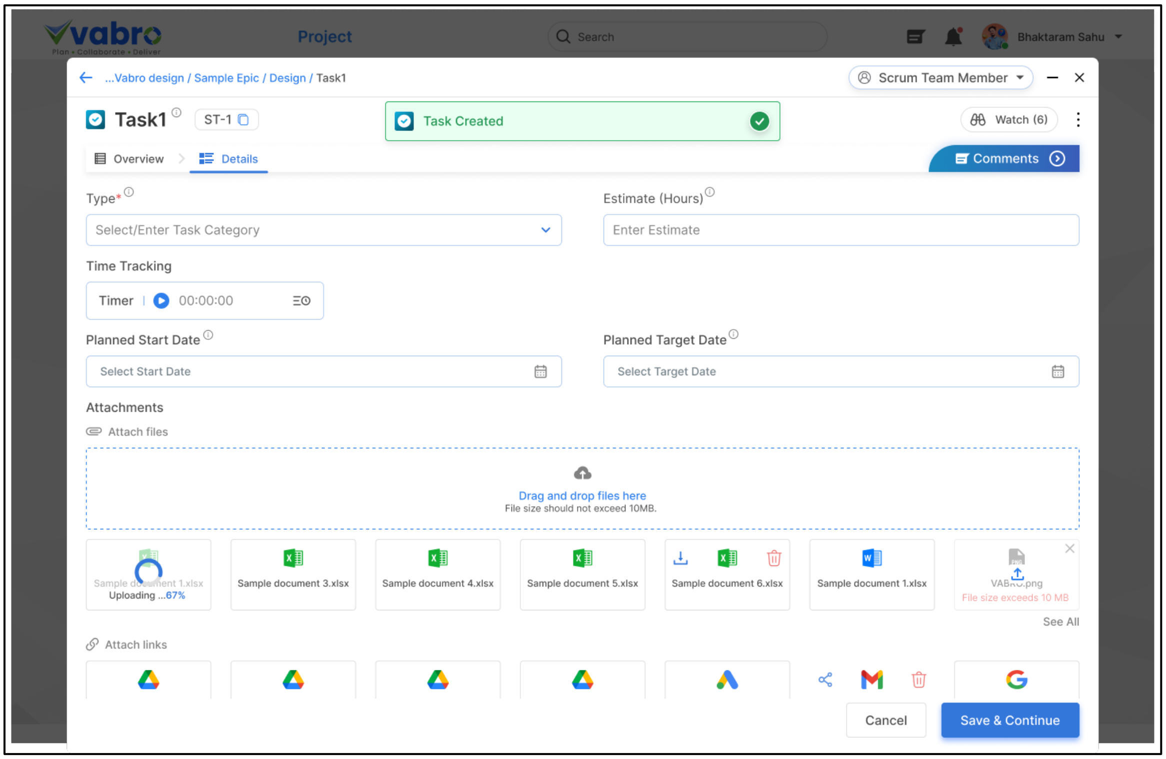
9.5.2.3 Estimation Methods*

The same estimation methods used to estimate User Stories can be applied to tasks as well. For more information on estimation methods, see section 9.2.2.1.

9.5.2.4 AI-powered Scrum Project Tool

Described in section 2.4.4.

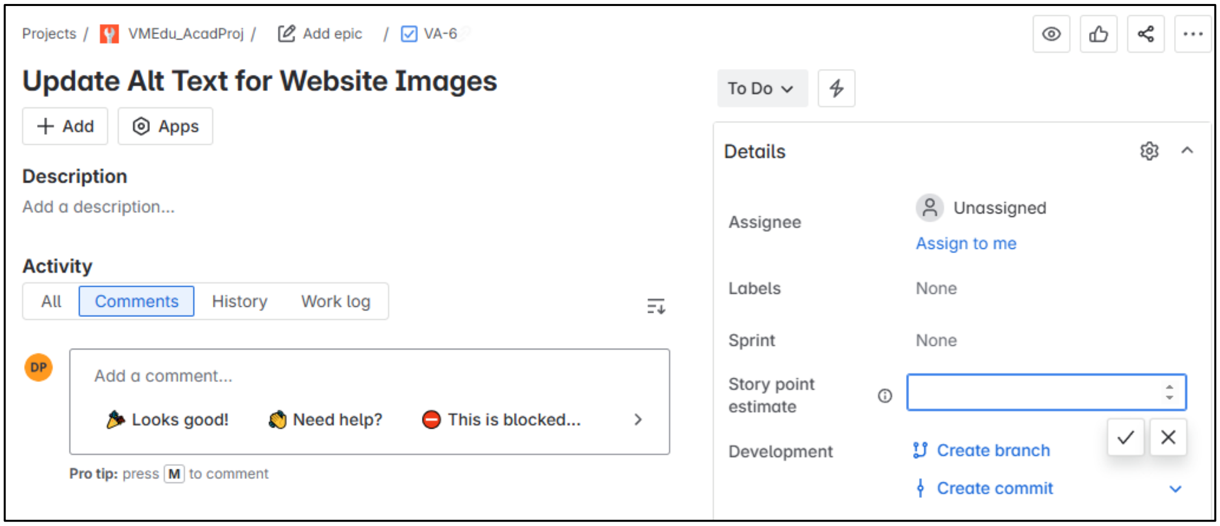
Figure 9-21 shows how Vabro facilitates task estimation in scrum using ai-powered features. Includes automated suggestion tools and effort estimation inputs for scrum teams to plan more effectively.



Task Created

Figure 9-21: Task Estimation in Scrum (Source: Vabro)

Figure 9-22 demonstrates Jira's task estimation interface. Includes visuals for inputting task data, viewing estimates, and tracking sprint effort. It helps streamline the planning process using real-time collaboration.



Projects / VMEdu_AcadProj / Add epic / VA-6

Update Alt Text for Website Images

Description
Add a description...

Activity

Add a comment...

Pro tip: press M to comment

Details

Assignee: Unassigned

Labels: None

Sprint: None

Story point estimate:

Development

Figure 9-22: Use of Scrum Tool for Task Estimation (Source: Jira)

9.5.3 Outputs

9.5.3.1 Updated Task List*

The Task List is updated to include the estimated efforts that were determined using the detailed estimation activities undertaken in the *Estimate Tasks* process. There may also be re-estimations resulting from changes in the Scrum Team's collective understanding of User Stories and requirements. Estimated effort is expressed in terms of the estimation criteria agreed on by the team. Typically, the accuracy of the estimates varies with team skills. The updated Task List is used by the Scrum Team during Sprint Planning Meetings to update the Sprint Backlog and to create the Sprint Burndown Chart. It is also used to determine if the team needs to reduce its commitment, or if they can take on additional User Stories during Sprint Planning for the next Sprint.

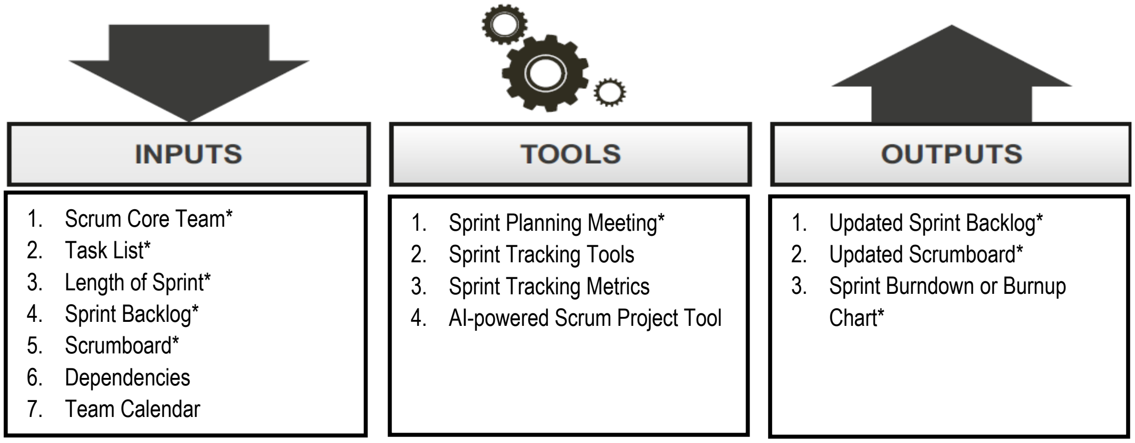
9.5.3.2 Updated Scrumboard*

As tasks are estimated, the effort estimates are then updated in the Scrumboard. For more information on the Scrumboard, see sections 9.3.3.3 and 9.4.3.2.

9.6 Update Sprint Backlog

In this process, the Scrum Core Team updates the Sprint Backlog with task details and if available, the task estimates. The updated Sprint Backlog will be used in the Implement phase to track the team’s progress during the upcoming Sprint.

Figure 9-23 shows all the inputs, tools, and outputs for the *Update Sprint Backlog* process.



INPUTS

1. Scrum Core Team*
2. Task List*
3. Length of Sprint*
4. Sprint Backlog*
5. Scrumboard*
6. Dependencies
7. Team Calendar

TOOLS

1. Sprint Planning Meeting*
2. Sprint Tracking Tools
3. Sprint Tracking Metrics
4. AI-powered Scrum Project Tool

OUTPUTS

1. Updated Sprint Backlog*
2. Updated Scrumboard*
3. Sprint Burndown or Burnup Chart*

Figure 9-23: Update Sprint Backlog—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a “mandatory” input, tool, or output for the corresponding process.

Figure 9-24 shows data flow diagram for the “update sprint backlog” process. Visualizes how various tools and inputs lead to an updated sprint backlog and scrumboard.

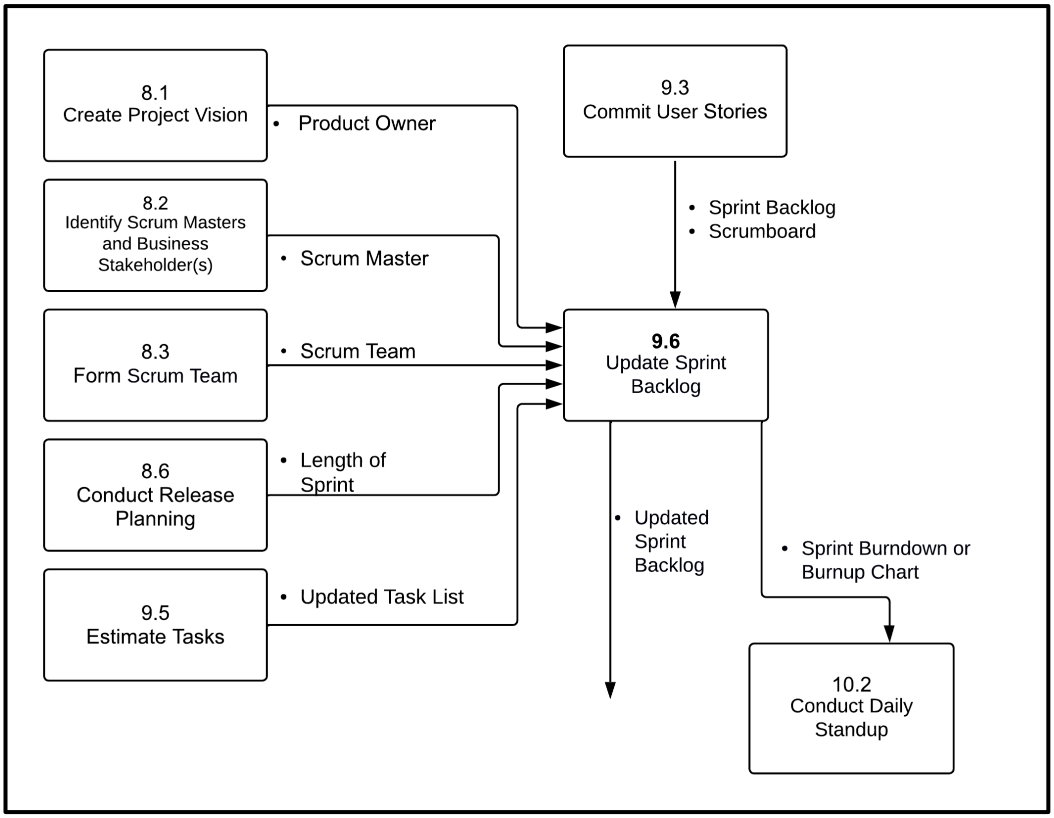


Figure 9-24: Update Sprint Backlog—Data Flow Diagram

9.6.1 Inputs

9.6.1.1 Scrum Core Team*

Described in section 3.2.1.

9.6.1.2 Task List*

Described in section 9.5.3.1.

9.6.1.3 Length of Sprint*

Described in section 8.6.3.2.

9.6.1.4 Sprint Backlog*

Described in section 9.3.3.2.

9.6.1.5 Scrumboard*

Described in section 9.3.3.3.

9.6.1.6 Dependencies

Described in section 9.4.3.4.

9.6.1.7 Team Calendar

A team calendar contains information regarding the availability of team members, including information related to employee vacation, leaves, major events, and holidays. One of the major objectives of using a Team Calendar is to track what each team member is working on throughout the project. It helps the team not only in planning and executing the Sprints efficiently but also in aligning the Sprints with release dates.

9.6.2 Tools

9.6.2.1 Sprint Planning Meetings*

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 Task List to select the tasks they plan to work on in the Sprint, based on their skills and experience. The Scrum Team also updates the Sprint Backlog and creates Sprint Burndown Chart using the User Stories and the Task List during the Sprint Planning Meetings. For more information on Sprint Planning Meetings, see sections 9.2.2.2, 9.3.2.1, 9.4.2.1, and 9.5.2.1.

9.6.2.2 Sprint Tracking Tools

It is important to track the progress of a Sprint and to know where the Scrum Team stands in terms of completing the tasks in the Sprint Backlog**.** A variety of tools can be used to track the work in a Sprint. One of the most common tools is a Scrumboard, also known as a task board or a progress chart. For more information on the Scrumboard, see sections 9.3.3.3 and 9.4.3.2.

9.6.2.3 Sprint Tracking Metrics

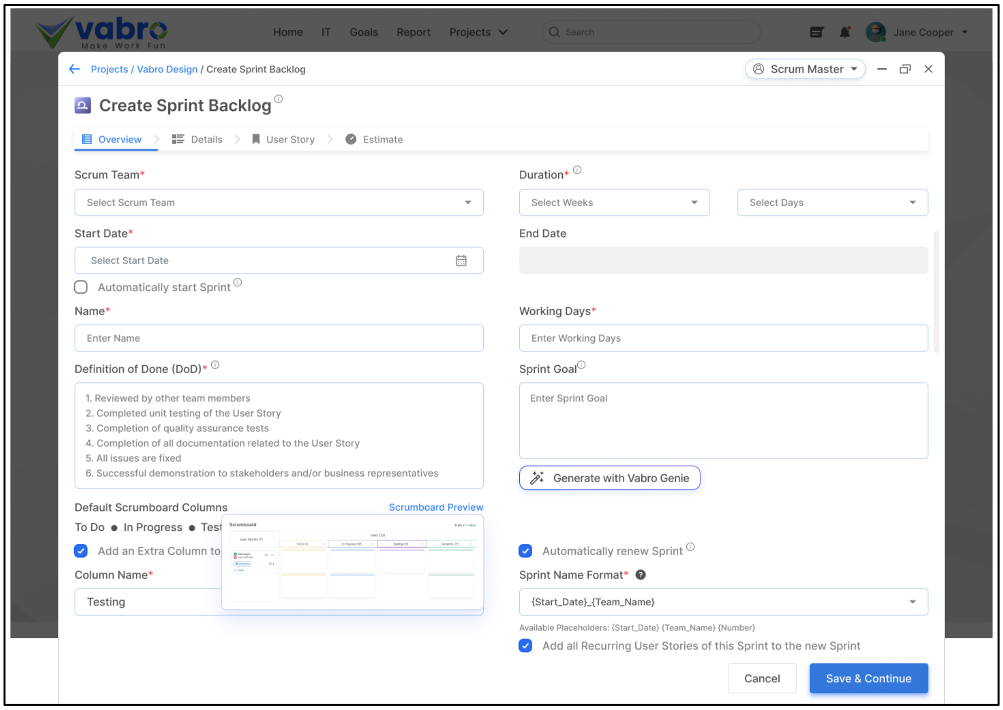
Tracking metrics used in Scrum projects include velocity, business value delivered, and number of stories.

- **Velocity**—represents the number of User Stories or functionalities delivered in a single Sprint.
- **Business value delivered**—measures the value of the User Stories delivered from the business perspective.
- **Number of stories**—refers to how many User Stories are delivered as part of a single Sprint. It can be expressed in terms of simple count or weighted count.

9.6.2.4 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 9-25 displays how to set up the sprint backlog in Vabro. Shows visual categorization of tasks, priority levels, and assignment options to prepare for sprint execution.



Create Sprint Backlog

Scrum Team* Select Scrum Team

Duration* Select Weeks Select Days

Start Date* Select Start Date

End Date

□ Automatically start Sprint

Name* Enter Name

Working Days* Enter Working Days

Definition of Done (DoD)*

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
6. Successful demonstration to stakeholders and/or business representatives

Sprint Goal Enter Sprint Goal

Default Scrumboard Columns

To Do ● In Progress ● Test

☑ Add an Extra Column to

Column Name* Testing

☑ Automatically renew Sprint

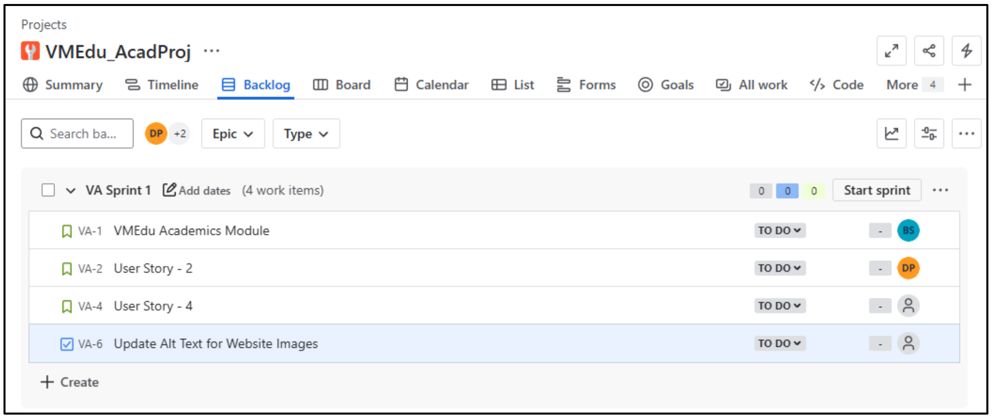
Sprint Name Format* {Start_Date}_{Team_Name}

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

☑ Add all Recurring User Stories of this Sprint to the new Sprint

Figure 9-25: Setting up Sprint Backlog in Scrum (Source: Vabro)

Figure 9-26 illustrates Jira's interface for using the sprint backlog. Includes filtering, prioritizing, and linking tasks to user stories for smooth backlog management.



VA Sprint 1 Add dates (4 work items) 0 0 0

VA-1	VMEdu Academics Module	TO DO	-	BS
VA-2	User Story - 2	TO DO	-	DP
VA-4	User Story - 4	TO DO	-	
VA-6	Update Alt Text for Website Images	TO DO	-	

Figure 9-26: Use of Sprint Backlog in Scrum Project Tool (Source: Jira)

9.6.3 Outputs

9.6.3.1 Updated Sprint Backlog*

The Scrum Core Team updates the Sprint Backlog with details of the tasks associated with the committed User Stories in the Sprint Backlog. If available, task estimates are also updated in the Sprint Backlog. The Sprint Backlog is used in the Implement phase to track the team's progress during the Sprint. 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 Prioritized Product Backlog and included for consideration in a future Sprint.

9.6.3.2 Updated Scrumboard*

The Scrumboard is updated to reflect the information in the updated Sprint Backlog including any updates to the tasks, task statuses, and task estimates, if available. For more information on the Scrumboard, see sections 9.3.3.3 and 9.4.3.2.

9.6.3.3 Sprint Burndown or Burnup Chart*

Burn Charts (Burndown or Burnup) are used to track progress in a Scrum project. A Burndown Chart is a graph that depicts the amount of work remaining in relation to the remaining time. Unlike the Burndown Chart, a Burnup Chart depicts what has been completed in relation to the remaining time.

For more details, please refer to Section 4.6.1.

9.7 Plan and Estimate Phase Data Flow Diagram

Figure 9-27 is a comprehensive data flow diagram for the entire 'plan and estimate' phase. Maps interdependencies between tasks, sprints, and backlog elements to align planning activities.

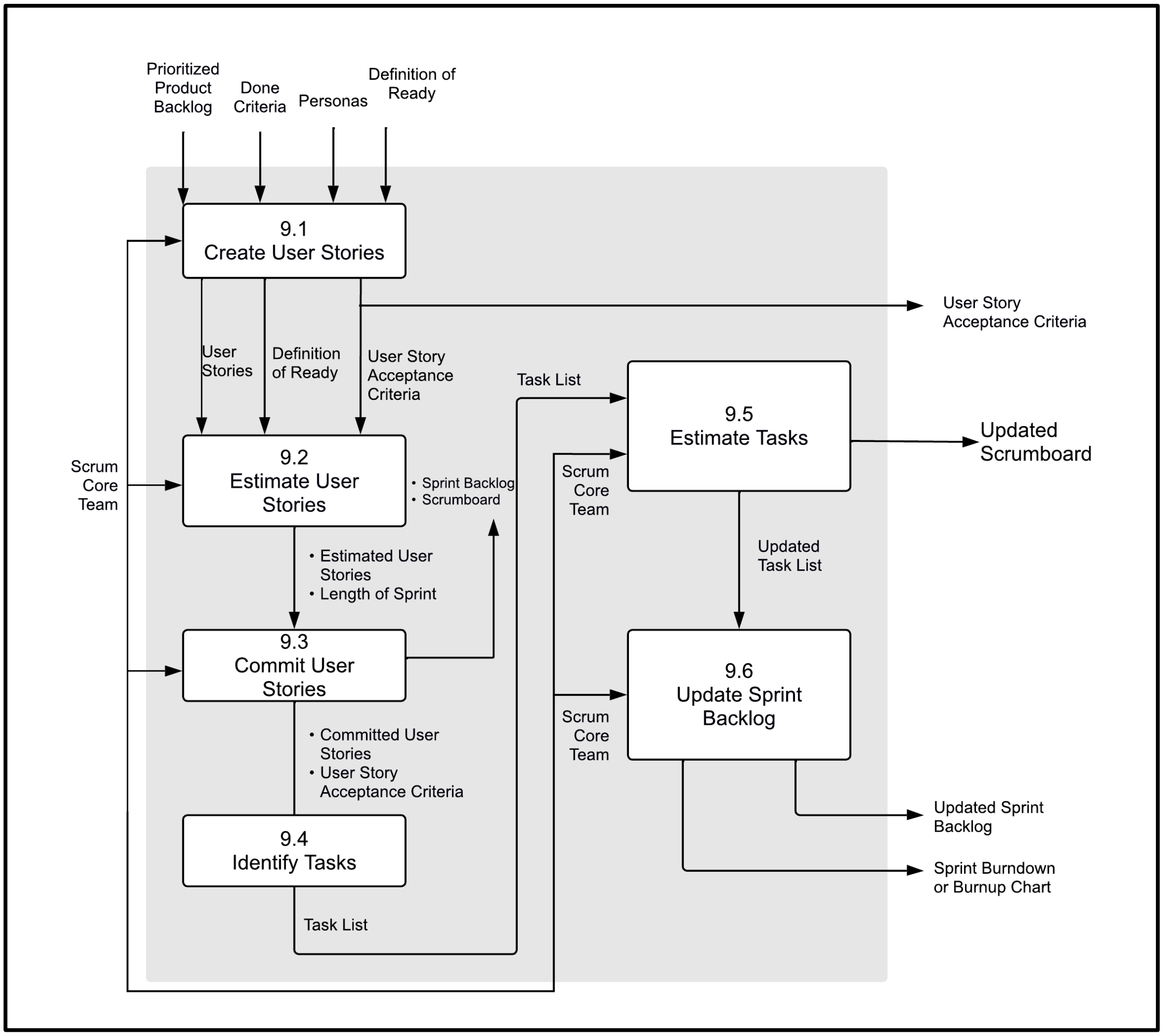


Figure 9-27: Plan and Estimate Phase—Data Flow Diagram

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.