SCRUMstudy
Targeting success



A Guide to the

# SCRUM BODY OF KNOWLEDGE

(SBOK® Guide)

## 8. INITIATE

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

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

# 8. INITIATE

This chapter includes the processes related to initiation of a project: *Create Project Vision*, *Identify Scrum Master and Business Stakeholder(s)*, *Form Scrum Team*, *Develop Epic(s)*, *Create Prioritized Product Backlog*, and *Conduct Release Planning*.

*Initiate*, 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.

The Initiate phase occurs at the beginning of a Scrum project. During this phase, the Scrum Core Team and business stakeholders are identified, starting with the Product Owner, who creates a Project Vision that serves as guidance throughout the entire project.

Based on Project Vision, an initial set of requirements is identified and documented in the form of Epics.

These initial requirements are prioritized and used to create an initial Prioritized Product Backlog (i.e., this is the requirements document in a Scrum project). During the last step of the Initiate phase, a Release Planning Schedule is created for the overall project. The Initiate phase does not produce a comprehensive and detailed plan for the entire project. There is no need for a comprehensive plan because change is expected and can easily be incorporated in a Scrum project due to the iterative principle incorporated into the Scrum processes. Rather, the goal of the Initiate phase of a Scrum project is to come up with a good initial plan for the project that aligns with the business needs and/or any high-priority regulations. This phase is expected to be short in order so that value creation can start as quickly as possible in the project.

*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 8-1 provides an overview of the Initiate phase processes, which are as follows:

**8.1 Create Project Vision**—In this process, the Product Owner is identified. Based on the project's business case, the Product Owner then creates a Project Vision Statement. This Project Vision Statement provides the overall guidance, inspiration, and focus for the project. The company may also choose to evaluate and select an AI-powered Scrum project tool that can significantly reduce the complexity of setting up the Scrum initiative and increase the productivity of Scrum projects within the organization.

**8.2 Identify Scrum Master and Business Stakeholder(s)**—In this process, the Scrum Master is identified using specific selection criteria that can effectively assess the soft skills and Scrum knowledge needed for this important role. Additionally, business stakeholders are also identified during this process.

**8.3 Form Scrum Team**—In this process, Scrum Team members are identified based on the skills required to complete the project deliverables, as well as considerations for the availability, costs, and soft skills important for members of a Scrum Team. Normally the Product Owner has the primary responsibility of selecting team members but often does so in collaboration with the Scrum Master.

**8.4 Develop Epic(s)**—In this process, the Project Vision Statement serves as the basis for developing Epics, which define the high-level requirements for the project. The Product Owner may use User Group meetings and other tools to collect requirements from business stakeholders.

**8.5 Create Prioritized Product Backlog**—In this process, Epics are refined, elaborated, and most importantly, prioritized according to their respective business value to create a Prioritized Product Backlog for the project. Additionally, based on the Scrum Guidance Body recommendations, the Product Owner and the Scrum Team establish the Done Criteria for the project.

**8.6 Conduct Release Planning**— In this process, the Product Owner, with inputs from business stakeholders and members of the Scrum Team, develops the initial Release Planning Schedule, which is communicated to, and shared with, all business stakeholders and Scrum Team Members. It is understood that the iterative nature of Scrum may necessitate future adjustments to the release schedule. The length of each Sprint is also determined in this process.

### 8.1 Create Project Vision

**INPUTS**
1. Project Business Case*
2. Trial Project
3. Proof of Concept
4. Company Vision
5. Company Mission
6. Market Study
7. Scrum Guidance Body Recommendations

**TOOLS**
1. Project Vision Meeting*
2. JAD Sessions
3. SWOT Analysis
4. Gap Analysis

**OUTPUTS**
1. Identified Product Owner*
2. Project Vision Statement*
3. Project Charter
4. Project Budget
5. AI-powered Scrum Project Tool

### 8.2 Identify Scrum Master and Business Stakeholder(s)

**INPUTS**
1. Product Owner*
2. Project Vision Statement*
3. People Requirements
4. People Availability and Commitment
5. Organizational Resource Matrix
6. Skills Requirement Matrix
7. Scrum Guidance Body Recommendations

**TOOLS**
1. Selection Criteria*
2. Expert Advice from HR
3. Training
4. Resource Costing

**OUTPUTS**
1. Identified Scrum Master*
2. Identified Business Stakeholder(s)*

### 8.3 Form Scrum Team

**INPUTS**
1. Product Owner*
2. Scrum Master*
3. Project Vision Statement*
4. People Requirements
5. People Availability and Commitment
6. Organizational Resource Matrix
7. Skills Requirement Matrix
8. Resource Requirements
9. Scrum Guidance Body Recommendations

**TOOLS**
1. Scrum Team Selection Criteria*
2. Expert Advice from HR
3. People Costs
4. Training
5. Resource Costing
6. AI-powered Scrum Project Tool

**OUTPUTS**
1. Identified Scrum Team*
2. Back-ups
3. Collaboration Plan
4. Team Building Plan

### 8.4 Develop Epic(s)

**INPUTS**
1. Scrum Core Team*
2. Project Vision Statement*
3. Business Stakeholder(s) *
4. Approved Change Requests
5. Unapproved Change Requests
6. Laws and Regulations
7. Applicable Contracts
8. Previous Project Information
9. Scrum Guidance Body Recommendations

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

**OUTPUTS**
1. Epic(s)*
2. Personas*
3. Approved Changes
4. Identified Risks

### 8.5 Create Prioritized Product Backlog

**INPUTS**
1. Scrum Core Team*
2. Epic(s)*
3. Personas*
4. Business Stakeholder(s)
5. Project Vision Statement
6. Business Requirements
7. Approved Change Requests
8. Identified Risks
9. Applicable Contracts
10. Scrum Guidance Body Recommendations

**TOOLS**
1. Prioritization Methods*
2. User Story Workshops
3. Risk Assessment Techniques
4. Project Value Estimation
5. Estimation Methods
6. Dependency Determination
7. Scrum Guidance Body Expertise
8. AI-powered Scrum Project Tool

**OUTPUTS**
1. Prioritized Product Backlog*
2. Done Criteria*
3. Definition of Ready *
4. High-Level Estimates for Epics

### 8.6 Conduct Release Planning

**INPUTS**
1. Scrum Core Team*
2. Business Stakeholder(s)*
3. Project Vision Statement*
4. Prioritized Product Backlog*
5. Done Criteria*
6. Business Requirements
7. Holiday Calendar
8. Scrum Guidance Body Recommendations
9. High-Level Estimates for Epics

**TOOLS**
1. Release Planning Sessions*
2. Release Prioritization Methods*
3. AI-powered Scrum Project Tool

**OUTPUTS**
1. Release Planning Schedule*
2. Length of Sprint*
3. Target Customers for Release
4. Refined Prioritized Product Backlog

**Figure 8-1: Initiate Overview**

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

Figure 8-2 below shows the mandatory inputs, tools, and outputs for processes in the Initiate phase.

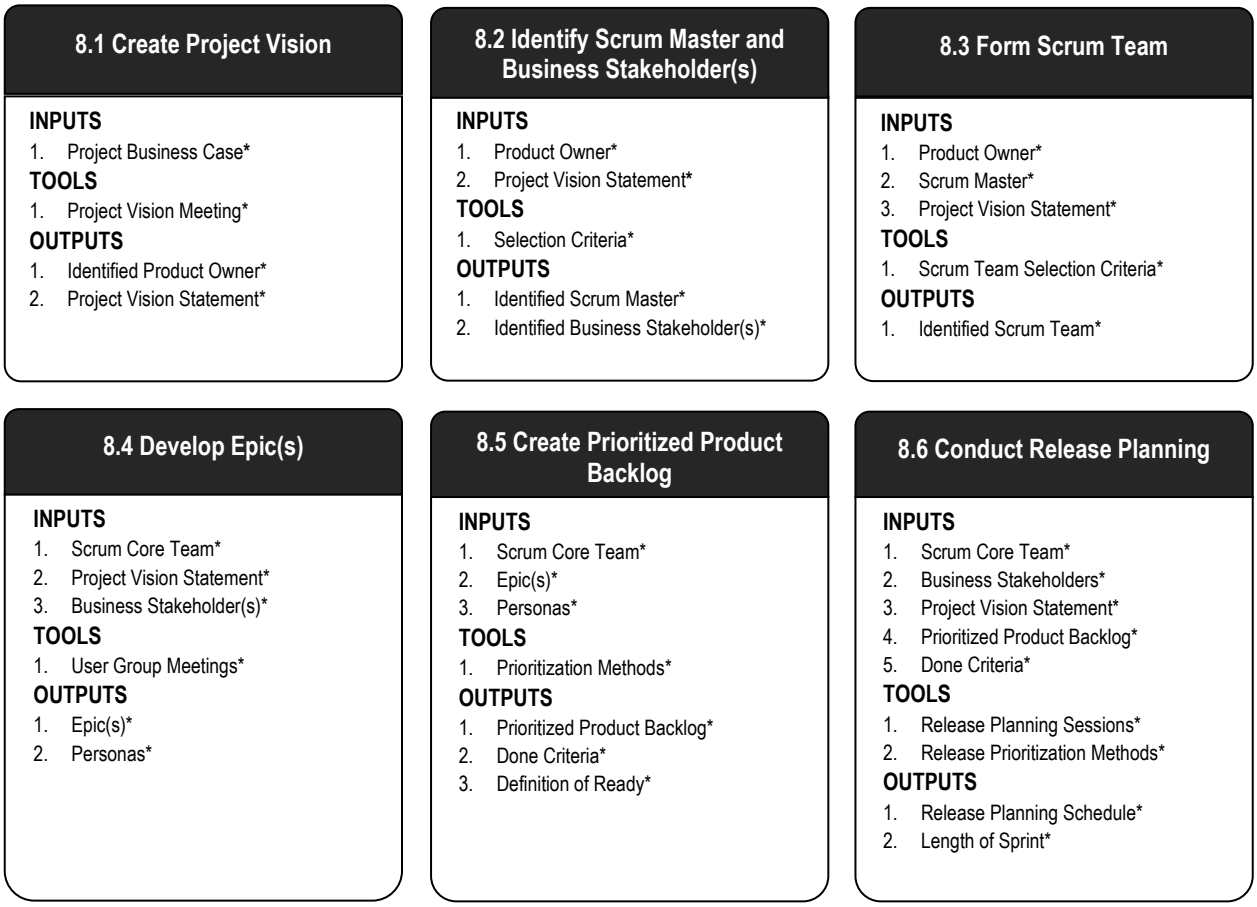


**Figure 8-2: Initiate Overview (Essentials)**

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

## 8.1 Create Project Vision

In this process, the Product Owner is identified. Based on the project's business case, the Product Owner then creates a Project Vision Statement. This Project Vision Statement provides the overall guidance, inspiration, and focus for the project. The company may also choose to evaluate and select an AI-powered Scrum project tool that can significantly reduce the complexity of setting up the Scrum initiative and increase the productivity of Scrum projects within the organization.

Figure 8-3 shows all the inputs, tools, and outputs for the *Create Project Vision* process.



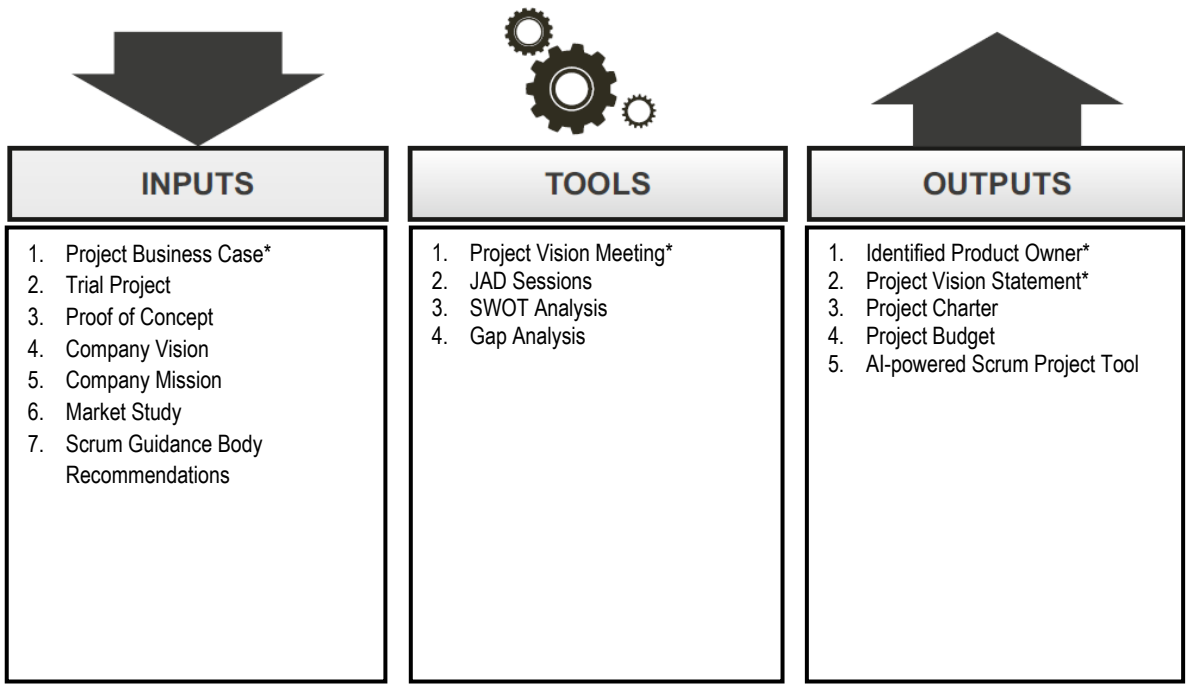


**Figure 8-3: Create Project Vision—Inputs, Tools, and Outputs**

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

Figure 8-4 illustrates the data flow during the project vision creation process. It visually connects stakeholder inputs to decisions and deliverables for initial project alignment.

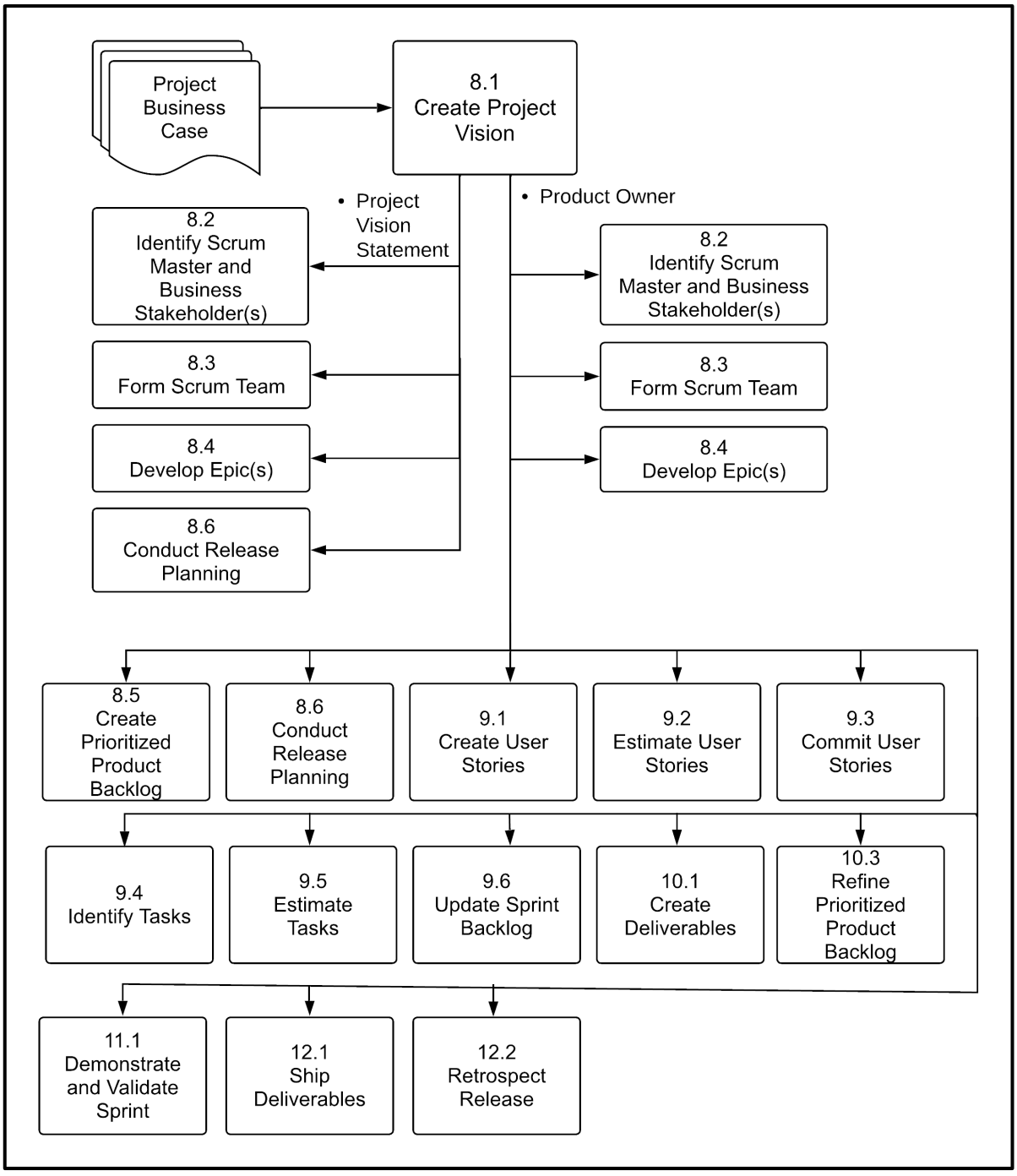


**Figure 8-4: Create Project Vision—Data Flow Diagram**

## 8.1.1 Inputs

### 8.1.1.1 Project Business Case*

The project commences with the presentation of the business case to the business stakeholders and sponsor(s). Business stakeholders should understand the expected business benefits of the project and the sponsor(s) should confirm that they will provide the financial resources for the project. A business case 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 business case for a project often includes substantial information pertaining to the project's background, the intended business purpose and desired outcomes, SWOT and/or Gap analysis reports, a list of already identified risks, and any high-level estimations of time, effort, and cost. The project business case should also take into consideration any relevant external factors, such as legal regulations, government requirements, data privacy concerns, and so on.



### 8.1.1.2 Trial Project

If feasible, a small-scale demo or trial project could be run as an experiment to predict and evaluate the viability, time, costs, risks, and effects of the actual project. This helps evaluate the practical environment and guides the actual project design prior to the initiation of the project on a full scale.

### 8.1.1.3 Proof of Concept

Proof of concept demonstrates and verifies that the idea behind the current project is potentially viable in the real-world environment. Often in the form of a prototype, it is designed to determine financial and technical viability, help understand requirements and assist in the assessment of design decisions. However, Proof of Concept does not need to necessarily represent actual project deliverables.

### 8.1.1.4 Company Vision

Understanding the company vision helps the project keep its focus on the organization's long-term objectives and the future direction of the company. The Product Owner takes guidance and direction from the company vision to create the Project Vision Statement.

### 8.1.1.5 Company Mission

The company mission provides a framework for formulating the strategies of the company and guides overall decision making within the company. The Project Vision Statement must be framed such that its fulfillment helps the organization fulfill its mission.

#### 8.1.1.6 Market Study

Market Study refers to the organized research, gathering, collation, and analysis of data related to customers' preferences for products. It often includes extensive data on market trends, market segmentation, and marketing processes. Market study could also include an analytical study of competitors which provides a better understanding of competitors' strengths and weaknesses and can help decision makers formulate better positioned products.

#### 8.1.1.7 Scrum Guidance Body Recommendations

The Scrum Guidance Body (SGB) is an optional role. It consists of a group of documents and/or a group of experts who are typically involved with defining objectives related to quality, government regulations, security, and other key organizational parameters. These objectives guide the work carried out by the Product Owner, Scrum Master, and Scrum Team. The Scrum Guidance Body also helps to capture the best practices that should be used across all Scrum projects in the organization. The Scrum Guidance Body does not make decisions related to the project. Instead, it acts as a consulting or guidance structure for all the hierarchy levels in the project organization—portfolios, programs, and projects. Scrum Teams have the option of asking the Scrum Guidance Body members for advice when needed. It is important to ensure that the project vision aligns with recommendations provided by the Scrum Guidance Body and that processes comply with any standards and guidelines that are established.

### 8.1.2 Tools

#### 8.1.2.1 Project Vision Meeting*

A Project Vision Meeting is a meeting with the relevant business stakeholders who understand the company's vision and the justification for the project. This may include the Chief Product Owner, Chief Scrum Master, Program Product Owner, Program Scrum Master, Portfolio Product Owner, and Portfolio Scrum Master if the project is a part of a larger project, program, or portfolio. This meeting helps identify the business context, business requirements, and business stakeholder expectations which can help when developing an effective Project Vision Statement. Applying Scrum practices involves engaging and collaborating with all business representatives to get their buy-in for the project and to deliver greater value.

#### 8.1.2.2 JAD Sessions

A Joint Application Design (JAD) session is a requirement gathering technique. It is a highly structured facilitated workshop which hastens the *Create Project Vision* process as it enables the business stakeholder(s) and other decision makers to come to a consensus on the high-level scope, high-level objectives, and other specifications of the project.

JAD sessions consist of methods for increasing user participation, speeding development, and improving specifications. Relevant program and portfolio Product Owners and Scrum Masters could meet to outline and analyze the desired business outcomes and to discuss their vision for the upcoming Scrum project.

#### 8.1.2.3 SWOT Analysis

A SWOT Analysis is a structured approach to project planning that helps evaluate the strengths, weaknesses, opportunities, and threats related to a project. This type of analysis helps identify both the internal and the external factors that could impact the project. Strengths and weaknesses are internal factors, whereas opportunities and threats are external factors. Identification of these factors helps business stakeholders and decision makers finalize the processes, tools, and techniques to be used to achieve project objectives. Conducting a SWOT Analysis allows for early identification of priorities, potential changes, and risks.



#### 8.1.2.4 Gap Analysis

A Gap Analysis is a technique used to compare the current, actual state with some desired state. In an organization, it involves determining and documenting the difference between current business capabilities and the final desired set of capabilities. A project is normally initiated to bring an organization to the desired state, so conducting a Gap Analysis would help decision makers determine the need for the project.

Figure 8-5 explains the gap analysis process. It helps identify disparities between current performance and develop outcomes to plan corrective actions or improvements.

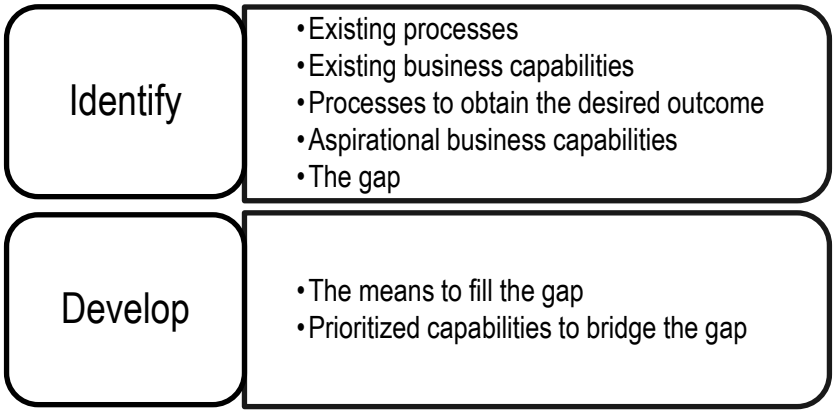


**Figure 8-5: The Gap Analysis Process**

## 8.1.3 Outputs

### 8.1.3.1 Identified Product Owner*

One of the outputs of this process is the identification of the Product Owner. The Product Owner is the person responsible for achieving maximum business value for the project. He or she is also responsible for articulating customer requirements and maintaining business justification for the project. The Product Owner represents the Voice of the Customer.

The Product Owner role is described in more detail in section 3.4.

### 8.1.3.2 Project Vision Statement*

The key output of the *Create Project Vision* process is a well-defined and structured Project Vision Statement. A good Project Vision statement explains the business need(s) the project is intended to meet (rather than how it will meet those needs).

The Project Vision Statement should not be too specific and should have room for flexibility. It is 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 statement is flexible enough to accommodate these changes. The project vision should focus on the problem rather than the solution.

### 8.1.3.3 Project Charter

A Project Charter is an official statement of the desired objectives and outcomes of the project. In most organizations, the Project Charter is the document that officially and formally authorizes the project, providing the team with written authority to begin the project work.

### 8.1.3.4 Project Budget

The project budget is a financial document that includes the cost of people, materials, and other related expenses in a project. The project budget is typically approved and signed off by the sponsor(s) to ensure that sufficient funds are available to complete the project. Once signed off, the Product Owner and the Scrum Master manage the project budget on a regular basis and also ensure that the people and other resources required for project activities are available.

### 8.1.3.5 AI-powered Scrum Project Tool

The selection of an AI-powered Scrum tool must be grounded in a thorough understanding of both agile methodologies and the capabilities of artificial intelligence. A well-executed selection process not only enhances Scrum effectiveness but also fosters a culture of data-driven decision-making and continuous improvement. The selection of an AI-powered Scrum project tool is a strategic decision that should align with the organization's agile maturity, operational needs, and long-term objectives. The following structured approach provides a formal framework for evaluating and selecting an appropriate tool.

**1. Assess Organizational Requirements**



- Begin by conducting a comprehensive assessment of the organization's project management needs, with a particular focus on how Scrum is currently implemented. This includes:
- The scale of Scrum adoption across teams.
- Current challenges in backlog management, sprint planning, or retrospectives.
- The technical sophistication of the teams and their readiness to adopt AI-driven processes.
- The desired level of automation, analytics, and predictive capabilities.

**2. Define Selection Criteria**

- Clearly defined criteria are essential for an objective comparison of available tools. Criteria should include both functional and non-functional requirements:
- Functional Criteria
- Support for all core Scrum components (Epics, User Stories, Tasks, Sprint Backlogs, Retrospectives).
- AI-driven features such as:
- Automated backlog refinement
- Recommended breakdown and classifications for projects, Epics, User Stories
- AI-powered reporting and Data Analytics
- Intelligent prioritization
- Predictive risk identification
- Sentiment analysis in retrospectives

**Non-Functional Criteria**

- Ability to integrate with teams using other methodologies such as Kanban Workflows, DevOps, OKRs etc.
- Ease of integration with existing tools (e.g., Vabro, Git, Jira, Slack, CI/CD pipelines).
- Scalability to accommodate multiple teams or departments.
- Compliance with data security and privacy standards (e.g., GDPR, SOC 2).
- Usability and learning curve for end-users.

**3. Market Research and Tool Shortlisting**

Conduct thorough research on available AI-powered Scrum tools in the market. Consider industry benchmarks, analyst reports, and peer reviews. Based on the initial assessment and criteria, shortlist 3–5 tools for deeper evaluation. Examples of prominent tools in this space include:

- Vabro with Vabro Genie AI
- Jira with Atlassian Intelligence
- ClickUp AI
- Forecast.ai
- Monday.com (AI integrations)

Figure 8-6 depicts a screenshot from the Vabro tool used in Scrum projects. It highlights functionalities like backlog tracking, team collaboration, and project planning.

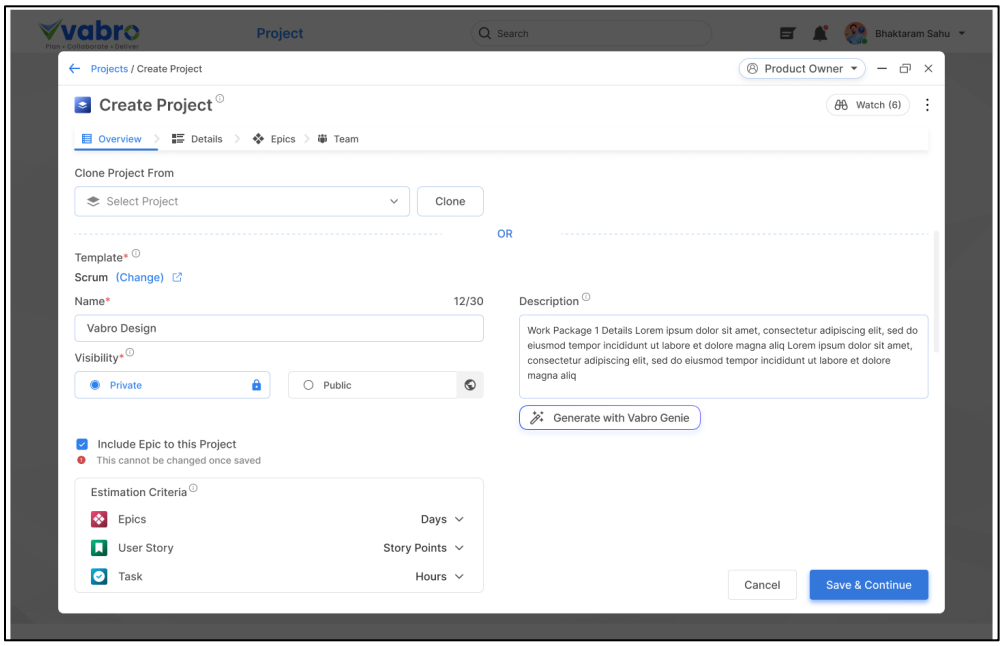


**Figure 8-6: Scrum Project Tools – Vabro**

Figure 8-7 shows Jira's Scrum interface for project management. It includes tools for sprint planning, backlog grooming, and tracking progress through agile workflows.

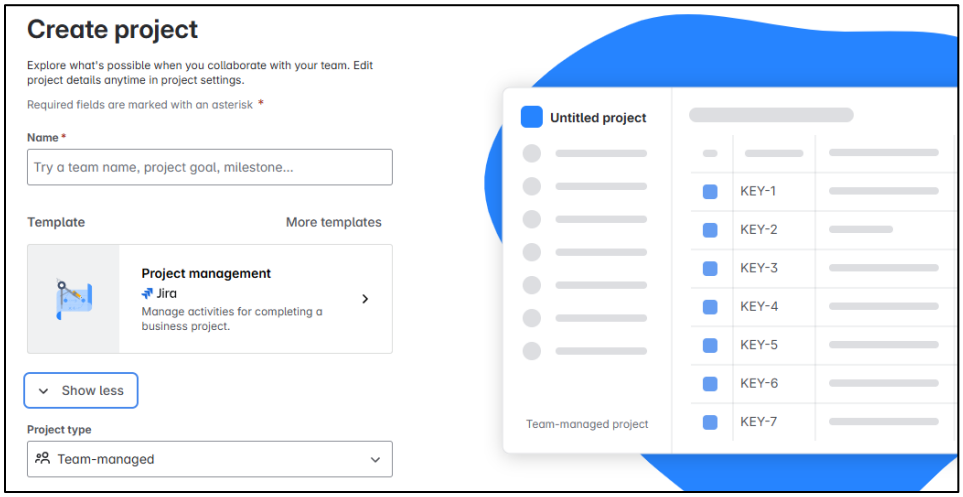


**Figure 8-7: Scrum Project Tools - Jira**

**4. Conduct Pilot Evaluations**

Organize pilot programs for each shortlisted tool using real project data and team workflows. During the pilot phase, measure the following:

- Accuracy and relevance of AI recommendations.
- Improvement in team productivity and planning efficiency.
- User adoption and satisfaction.
- Technical performance and reliability.
- Feedback should be collected from all key Scrum roles: Product Owners, Scrum Masters, and Development Team members.



**5. Analyze ROI and Total Cost of Ownership**

Evaluate the total cost of ownership (TCO), including licensing fees, training, support, and implementation time. Estimate the return on investment (ROI) based on projected gains in productivity, reduced cycle times, and improved delivery accuracy due to AI-enhanced decision-making.

**6. Final Selection and Implementation Planning**

Based on the evaluation outcomes, select the tool that best meets organizational goals. Develop a phased implementation plan that includes:

- Training sessions for Scrum teams
- Integration with existing systems
- Governance and change management strategies
- Ongoing performance monitoring and feedback loops

## 8.2 Identify Scrum Master and Business Stakeholder(s)

In this process, the Scrum Master is identified using specific selection criteria that can effectively assess the soft skills and Scrum knowledge needed for this key role. Additionally, business stakeholders are also identified during this process.

Figure 8-8 shows all the inputs, tools, and outputs for the *Identify Scrum Master and Business Stakeholder(s)* process.

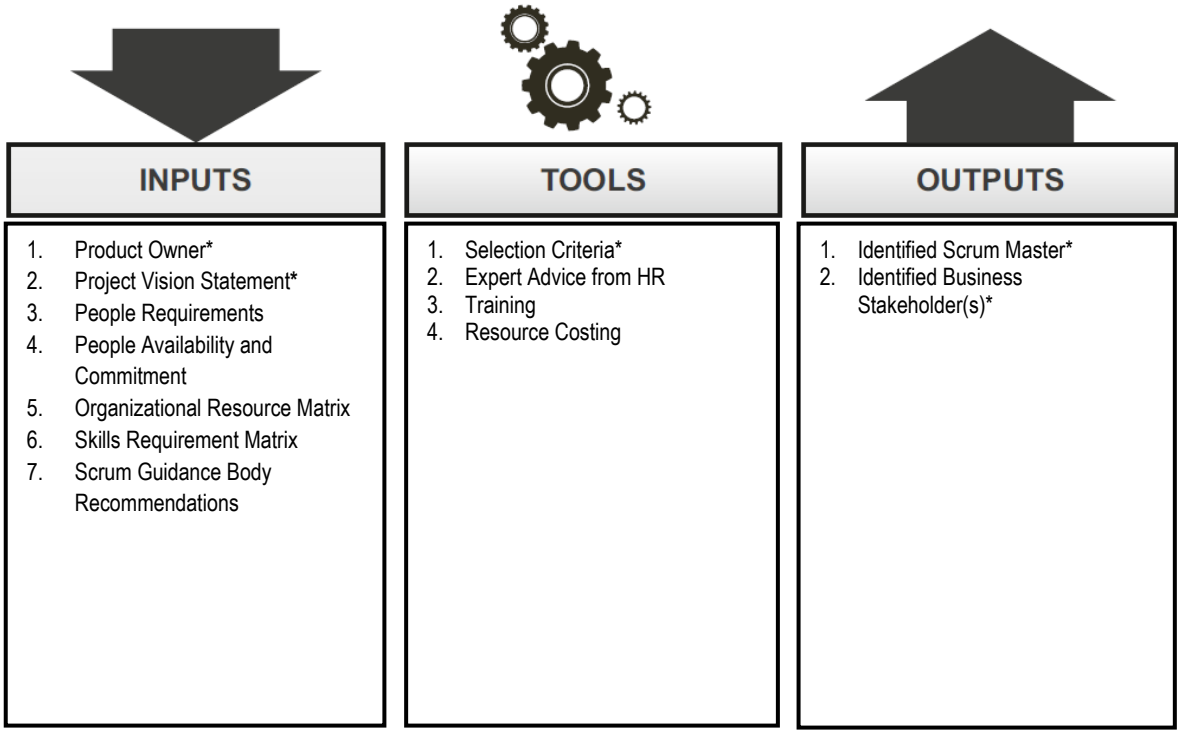


**Figure 8-8: Identify Scrum Master and Business Stakeholder(s)—Inputs, Tools, and Outputs**

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

Figure 8-9 illustrates the data flow for Identifying the Scrum Master and Business Stakeholders. It maps how selection criteria and organizational needs inform the identification process.

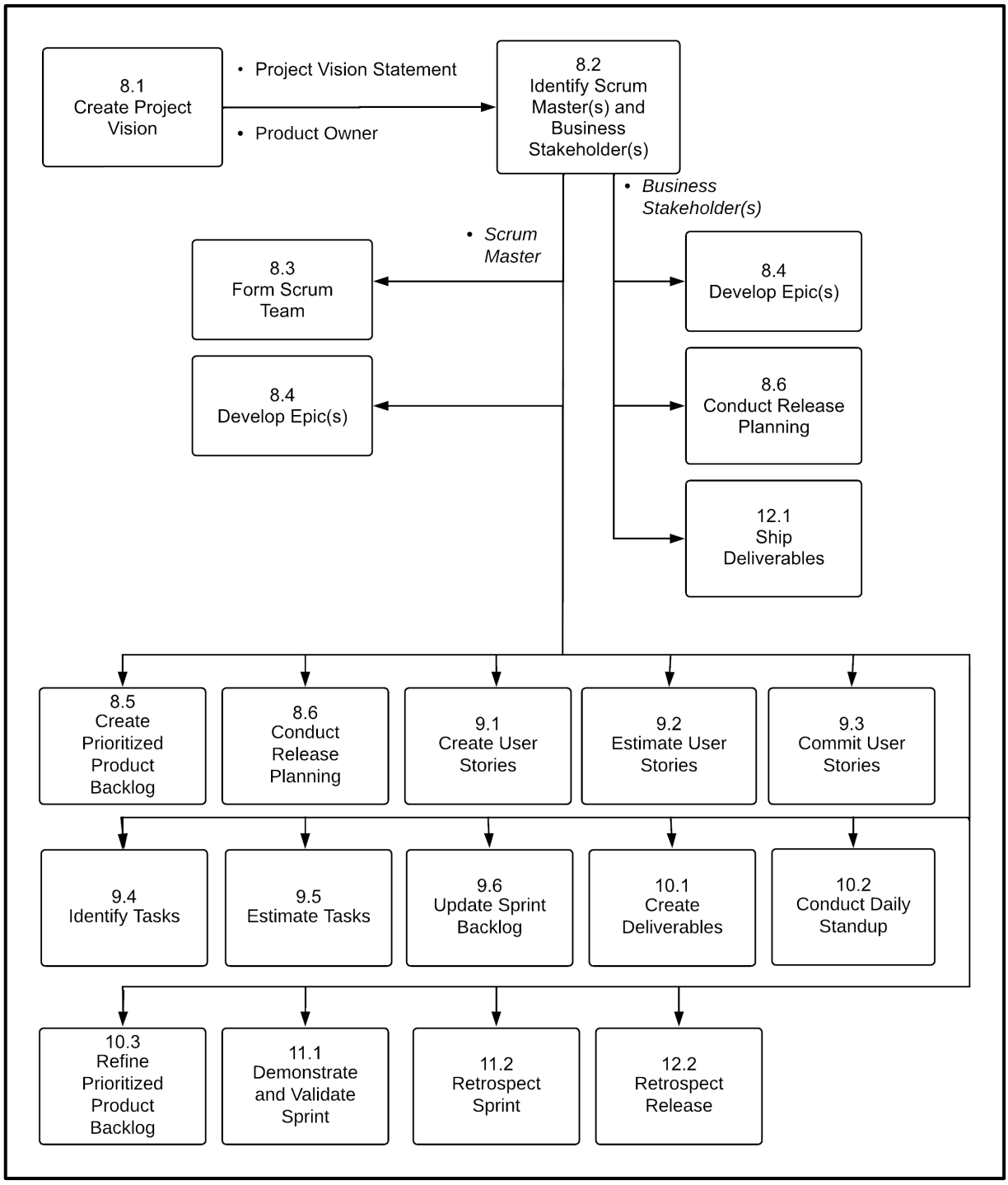


**Figure 8-9: Identify Scrum Master and Business Stakeholder(s)—Data Flow Diagram**

## 8.2.1 Inputs

### 8.2.1.1 Product Owner*

Described in section 8.1.3.1.

### 8.2.1.2 Project Vision Statement*

Described in section 8.1.3.2.

### 8.2.1.3 People Requirements

Identifying the required skill set of the Scrum Master is one of the initial steps to be performed before a Scrum Team can be fully formed (see process 8.3). It is important to document the role and responsibilities of the Scrum Master position, and the desired knowledge and skills needed to ensure a project's success. Usually, the Product Owner works with the Human Resource department or other relevant internal stakeholders to determine and finalize the Scrum Master's role description.

Key business stakeholders are also identified and their role in the project is determined.

### 8.2.1.4 People Availability and Commitment

Prior to selecting the Scrum Master and business stakeholder(s), their availability and commitment must be confirmed. Only Scrum Masters and business stakeholders who are available and can fully commit to the project should be considered. Availability is commonly depicted in the form of calendars showing if, and when human resources are available to work throughout the duration of the project. Commitment can then be gained once the availability of the desired Scrum Master (and business stakeholders) is confirmed. The Human Resource department can assist the Product Owner in selection of the appropriate candidates.

### 8.2.1.5 Organizational Resource Matrix

The Organizational Resource Matrix is a hierarchical depiction of a combination of a functional organizational structure and a projectized organizational structure. Matrix organizations bring together project team members from different functional departments, such as information technology, finance, marketing, sales, manufacturing, and other departments, to create cross-functional teams. Team members in a matrix organization fulfill two objectives—functional and project. Team members are directed by the Product Owner with respect to project-related activities, while the functional managers perform managerial activities related to their departments, such as performance appraisals and approving leaves.

#### 8.2.1.6 Skills Requirement Matrix

The Skills Requirement Matrix, also known as a competency framework, is used to assess skill gaps and training requirements for team members. A skills matrix maps the skills, capabilities, and interest level of team members in using those skills and capabilities required for a project. Using this matrix, the organization can assess any skill gaps in team members and identify the employees who will need further training in a particular area or competency.

#### 8.2.1.7 Scrum Guidance Body Recommendations

Described in section 8.1.1.7.



### 8.2.2 Tools

#### 8.2.2.1 Selection Criteria*

Selecting the appropriate Scrum Master and identifying relevant business stakeholders(s) is crucial to the success of any project. In some projects, some roles (such as Scrum Master) may already be preassigned prior to project initiation. When there is flexibility in choosing a Scrum Master, the following are important selection criteria:

1. **Problem-solving skills**—This is one of the primary criteria to be considered while selecting a Scrum Master. The Scrum Master should have the necessary skills and experience to help remove any impediments for the Scrum Team.
2. **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.
3. **Commitment**—The Scrum Master should be highly committed to ensure that the Scrum Team is provided with a conducive work environment to ensure successful delivery of Scrum projects.
4. **Supporting Leadership Style** (see section 3.9.4.1).

When identifying the business stakeholders, it is important to remember that business 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 creation of the project's products. The business stakeholders influence the project throughout its lifecycle.

#### 8.2.2.2 Expert Advice from HR

Expert advice from Human Resource personnel can be valuable in identifying the Scrum Master and the business stakeholders. The HR team members possess knowledge about the skill sets of employees within the organization and they also have the specialized knowledge and experience of various techniques that might help in identifying the Scrum Master and business stakeholder(s).

#### 8.2.2.3 Training

Scrum is a radically different framework from traditional methods of project management. Team members may not yet possess the required knowledge or skills needed for working in a Scrum environment. The Product Owner should evaluate the training needs of potential team members and facilitate training to bridge any knowledge gap in the team. The Product Owner is normally responsible for evaluating and selecting team members, but often does this in consultation with the Scrum Master who may have additional knowledge of the resources from working with them on other projects.

Appropriate training should be provided to the Scrum Team members both prior to the commencement of work and also while they are working on their projects. Scrum Team members should be ready to learn from each other and from more experienced persons in the team.

#### 8.2.2.4 Resource Costing

One of the primary considerations when selecting the appropriate Scrum Master for the project has to do with the trade-offs related to experience versus salary. There are other people-related factors impacting cost that may also need to be considered. Ideally, the Scrum Master, team members, and business stakeholder(s) should be colocated, so that they can communicate frequently and easily. If colocation is not possible and the team is distributed, additional resources may be needed to facilitate communications, understand cultural differences, synchronize work, and foster knowledge sharing.

### 8.2.3 Outputs

#### 8.2.3.1 Identified Scrum Master*

A Scrum Master is a facilitator and supporting 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. It is primarily the responsibility of the Product Owner to identify the Scrum Master for a Scrum project. The Scrum Master role is described in more detail in section 3.5.

#### 8.2.3.2 Identified Business Stakeholder(s)*

Business stakeholder(s) is a collective term that includes customers, users, and sponsors who frequently interface with the Scrum Core Team and influence the project throughout the product development process. It is for the business stakeholders that the project produces collaborative benefits. The business stakeholder(s) role is described in section 3.2.2.

## 8.3 Form Scrum Team

In this process, Scrum Team members are identified based on the skills required to complete the project deliverables, as well as considerations for the availability, costs, and soft skills important for members of a Scrum Team. Normally the Product Owner has the primary responsibility of selecting team members but often does so in collaboration with the Scrum Master.

Figure 8-10 shows all the inputs, tools, and outputs for the *Form Scrum Team* process.

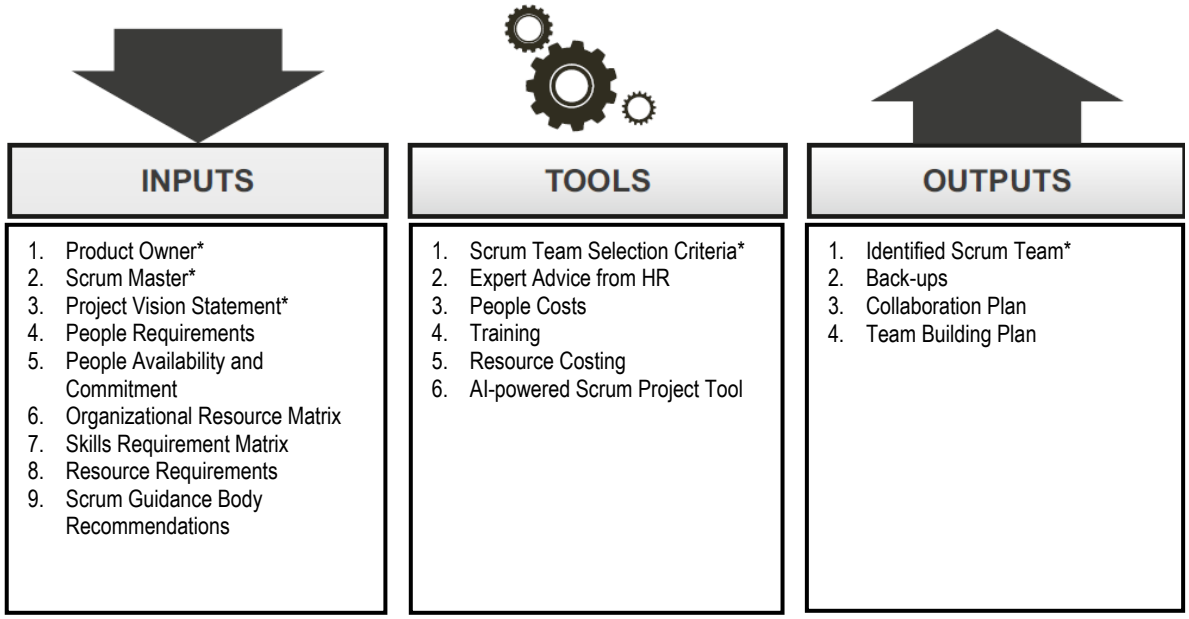


**Figure 8-10: Form Scrum Team—Inputs, Tools, and Outputs**

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

Figure 8-11 explains the data flow for Form Scrum team. It charts the process from identifying team needs to assembling a cohesive, functional team unit.

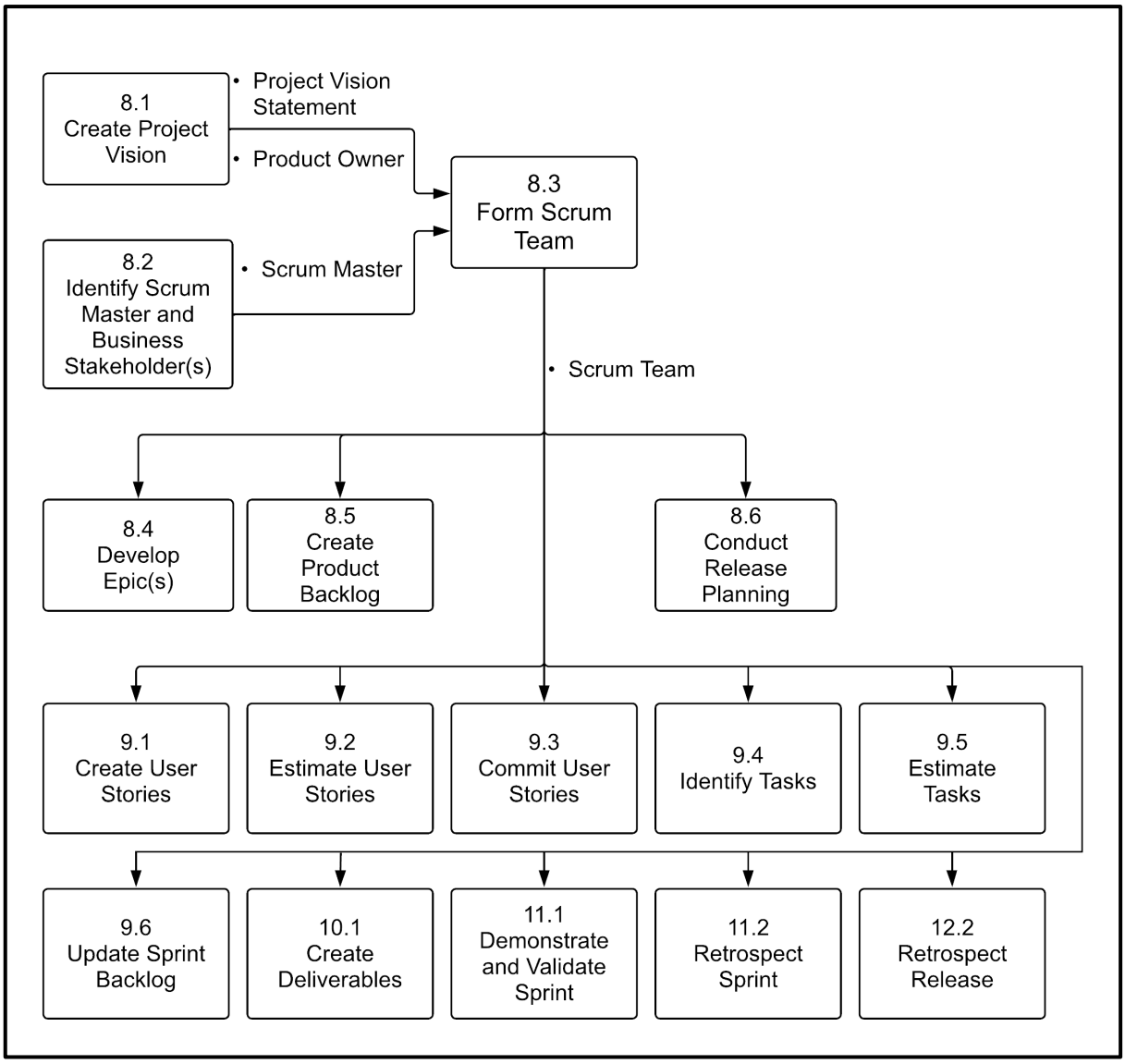


**Figure 8-11: Form Scrum Team—Data Flow Diagram**

## 8.3.1 Inputs

### 8.3.1.1 Product Owner*

Described in section 8.1.3.1.

### 8.3.1.2 Scrum Master*

Described in section 8.2.3.1.



### 8.3.1.3 Project Vision Statement*

Described in section 8.1.3.2.

### 8.3.1.4 People Requirements

It is important to identify and document the roles and responsibilities of the Scrum Team members and the desired knowledge and skill sets needed to ensure a project's success. Usually, the Product Owner works with the Human Resource department or other relevant people in the company to determine and finalize the roles and responsibilities of Scrum Team members.

### 8.3.1.5 People Availability and Commitment

Prior to selecting Scrum Team members, their availability and commitment must be confirmed. Availability is commonly depicted in the form of resource calendars showing if, and when human resources are available to work throughout the duration of the project. Commitment can then be gained once the availability of the desired team members is confirmed. Only team members who are available and can fully commit to the project should be selected as Scrum Team members. The Human Resource department can assist the Product Owner in selection of the appropriate candidates.

To be effective, Scrum Teams should ideally have six to ten members. Changing team members is not advisable in Scrum Core Teams. So, it is important to have Scrum Core Team members who are fully available and committed to the project.

### 8.3.1.6 Organizational Resource Matrix

Described in section 8.2.1.5.

#### 8.3.1.7 Skills Requirement Matrix

Described in section 8.2.1.6.

#### 8.3.1.8 Resource Requirements

These requirements include all the resources—other than people—required for the Scrum Team to function effectively. Examples of these resources include office infrastructure, meeting space, work equipment, and the appropriate tools needed to apply Scrum effectively (e.g., Scrumboard, Burndown Charts, index cards, estimation cards, etc.). In the case of virtual teams, additional resources such as collaboration tools, video conferencing, shared document repositories, and translation services must be considered.

#### 8.3.1.9 Scrum Guidance Body Recommendations

Described in section 8.1.1.7.

### 8.3.2 Tools

#### 8.3.2.1 Scrum Team Selection Criteria*

The Scrum Team is the core of any Scrum project and getting the appropriate team members is important for the successful delivery of a Scrum project. Scrum Team members are expected to be 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 a 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.

#### 8.3.2.2 Expert Advice from HR

Expert advice from Human Resource (HR) personnel can be valuable while forming a Scrum Team. The HR team members possess knowledge about the skill sets of employees in the organization and they also have the specialized knowledge and experience of various techniques that might help Product Owners, Scrum Masters, and sponsors identify the right team members.

#### 8.3.2.3 People Costs

All costs associated with people requirements need to be assessed, analyzed, approved, and budgeted for.

#### 8.3.2.4 Training

Team members may not possess all the required skill sets or knowledge to carry out specialized tasks. The Product Owner and/or Scrum Master should evaluate the training needs of potential team members. Once candidates are selected, the appropriate training should be provided for any skill or knowledge gaps found. For a truly effective Scrum implementation, there must be a significant level of awareness within the organization of Scrum principles and values. This awareness aids in the successful execution of Scrum. The Scrum Team has to be sensitized and trained in the practices of Scrum and the Scrum Master should play the role of a coach for the team. Because planning Sprints is a major success factor, training will help teams understand how to discuss and identify achievable Sprint goals. The Scrum Master needs to bring out the best from the Scrum Team members by motivating them and facilitating the development process. By training and coaching team members, the Scrum Master can help them articulate any issues and challenges they may face. Normally any issues or conflicts experienced within the team are solved by the team with coaching and assistance from the Scrum Master as required. The Scrum Master should address issues such as low morale or lack of coordination within the team. He or she is responsible for removing impediments for the team. When required, the Scrum Master can escalate external issues and impediments to management for resolution or removal.

Training is also discussed in the *Identify Scrum Master and Business Stakeholder(s)* process in section 8.2.2.3.

#### 8.3.2.5 Resource Costing

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.

#### 8.3.2.6 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 8-12 depicts the Vabro interface used for creating Scrum teams. It features roles, permissions, and collaboration tools within a Scrum project environment.

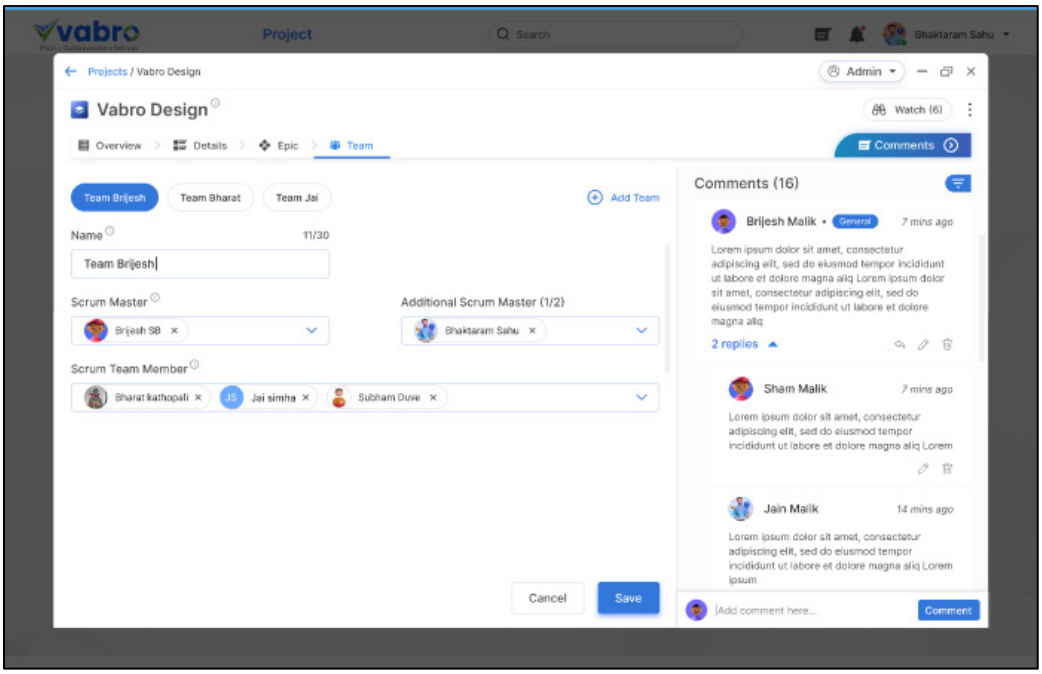


**Figure 8-12: Creating a Team (Source: Vabro)**

Figure 8-13 shows the team setup screen in Jira. It allows team member configuration, permissions assignment, and integration into ongoing Scrum workflows.

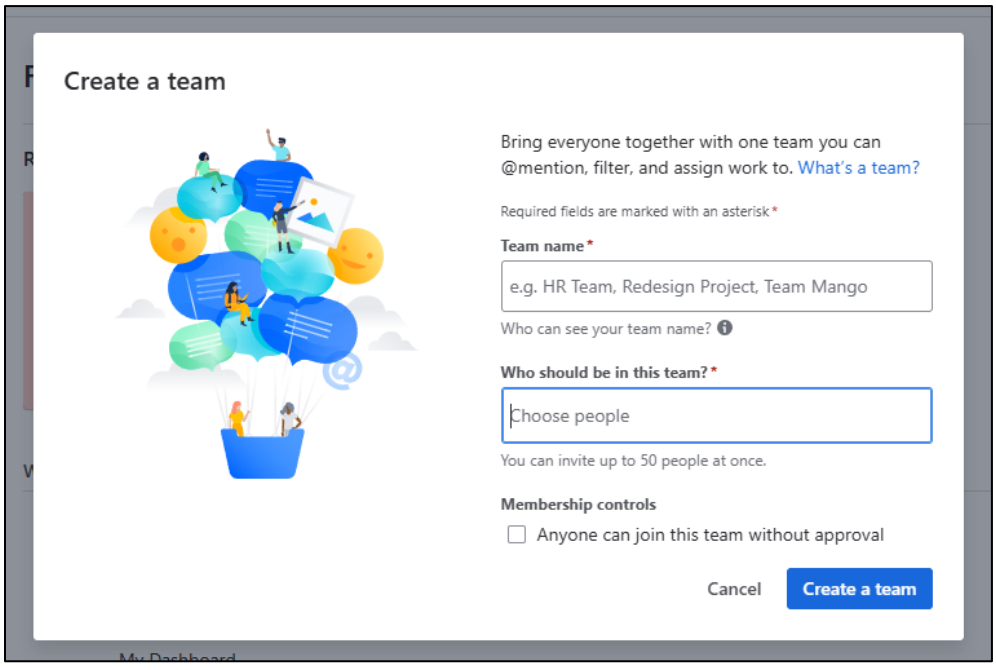


**Figure 8-13: Team Set-up (Source: Jira)**

## 8.3.3 Outputs

### 8.3.3.1 Identified Scrum Team*

The Scrum Team, sometimes referred to as the Development Team, is a group or team of people responsible for understanding the business requirements specified by the Product Owner, estimating User Stories, and creating the project deliverables. The Scrum Team consists of cross-functional team members who carry out all the work involved in creating potentially shippable deliverables, including work related to satisfying the desired quality assurance and quality control parameters of each deliverable. Scrum Teams are cross-functional and self-organizing. The team decides the amount of work to commit to in a Sprint and determines the best way to perform the work. Identifying the Scrum Team is the responsibility of the Product Owner, often in consultation with the Scrum Master. The Scrum Team role is described in more detail in section 3.6.



### 8.3.3.2 Back-ups

When selecting team members, it is important to identify backups for critical skills, preferably within the same Scrum Team. Although people availability and commitment are confirmed for team members in advance, issues may arise such as illness, family emergencies, or team member loss. Scrum Teams work in small groups of six to ten people. Having back-ups assigned to tasks ensures that there is no major decrease in productivity due to the loss of a team member.

### 8.3.3.3 Collaboration Plan

Collaboration is an important Scrum principle. Planning for how the various decision makers, business stakeholders, and team members engage and collaborate with each other is vital. The Collaboration Plan is an optional output that may be formal or informal. At times, it may simply be an oral understanding between the various business stakeholders and the Scrum Team (since creating unnecessary documentation should be avoided). However, for larger, more complex projects, especially those with distributed teams, a more formal agreement may need to be put in place. The plan may address how the Scrum Core Team members, business stakeholders(s) and others involved in the Scrum project will communicate and collaborate throughout the project and may also define specific tools or techniques to be used for that purpose. For example, in distributed teams, there may be a need for an agreement on when and how meetings will be conducted, what type of communication tools will be used, and who should be involved in each type of meeting.

### 8.3.3.4 Team Building Plan

Since a Scrum Team is cross-functional, each team member needs to participate actively in all aspects of the project. The Scrum Master identifies issues with team members and addresses them diligently in order to maintain an effective team. To build team cohesion, the Scrum Master should ensure that relationships among the team members are positive and that the team members are unified in achieving the overall project and organizational goals, thus leading to greater efficiency and increased productivity.

Section 3.9 discusses popular HR theories and their relevance to a Scrum project environment.

## 8.4 Develop Epic(s)

In this process, the Project Vision Statement serves as the basis for developing Epics, which define the high-level requirements for the project. The Product Owner may use User Group meetings and other tools to collect requirements from business stakeholders.

Figure 8-14 shows all the inputs, tools, and outputs for the *Develop Epic(s)* process.

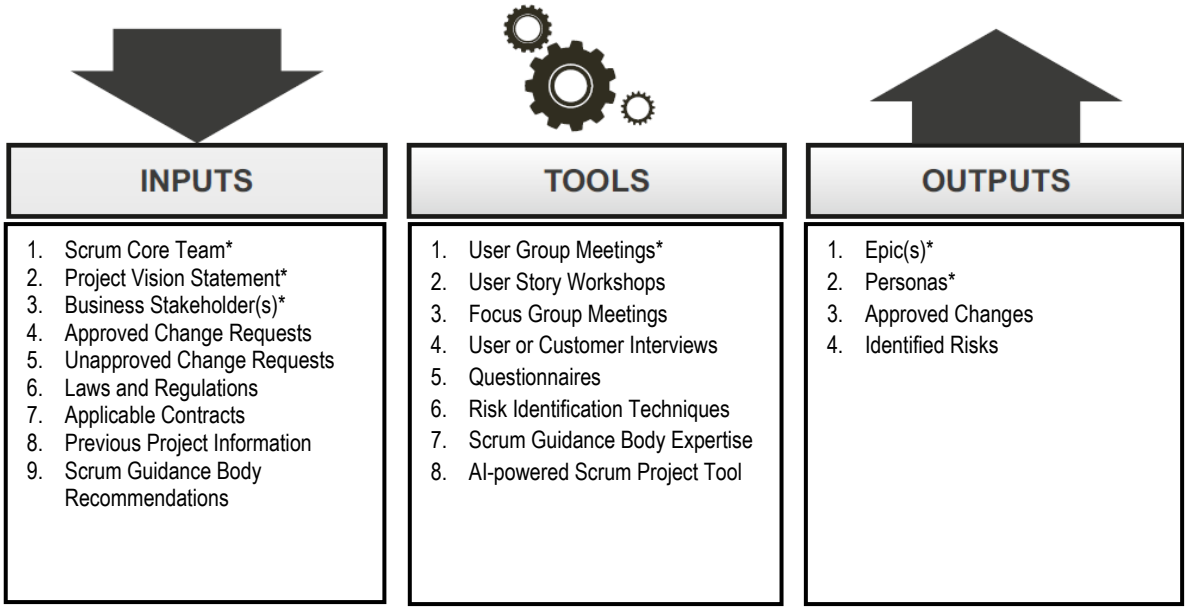


**Figure 8-14: Develop Epic(s)—Inputs, Tools, and Outputs**

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

Figure 8-15 explains the data flow in Develop Epics. It visualizes the transformation of business inputs into actionable backlog components using Scrum techniques.

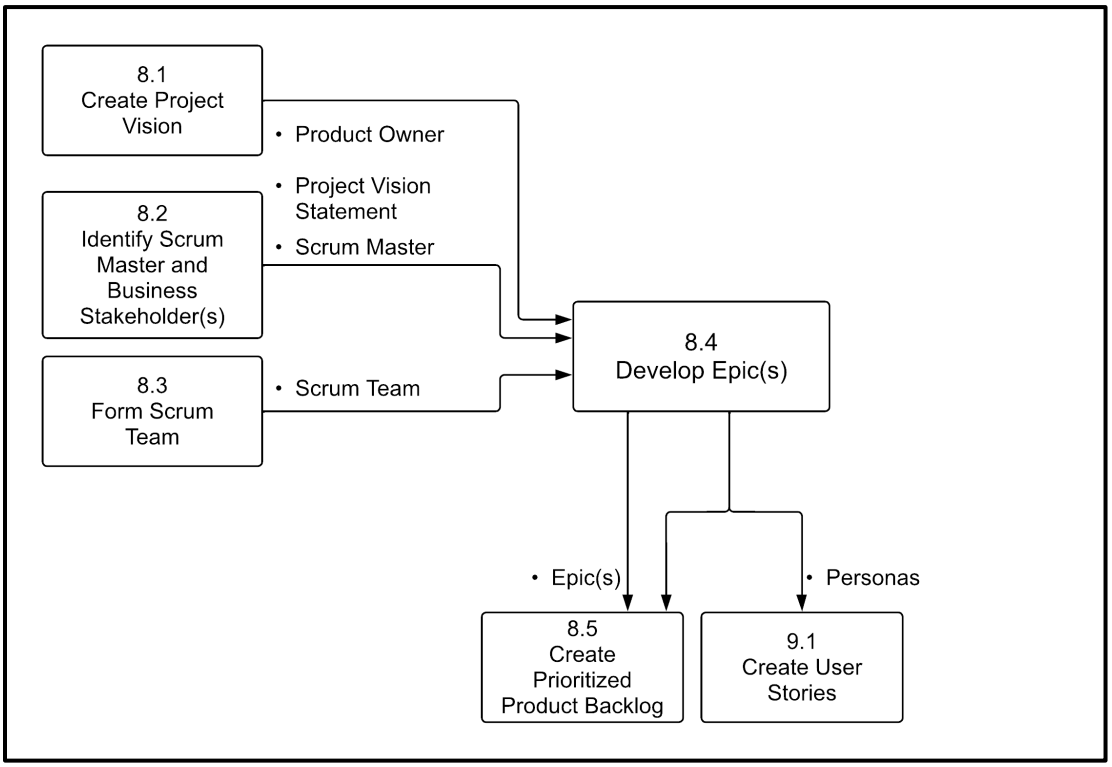


**Figure 8-15: Develop Epic(s)—Data Flow Diagram**

## 8.4.1 Inputs

### 8.4.1.1 Scrum Core Team*

The Scrum Core Team consists of the Scrum Team, the Scrum Master, and the Product Owner as described in section 3.2.1.

### 8.4.1.2 Project Vision Statement*

Described in section 8.1.3.2.

### 8.4.1.3 Business Stakeholder(s)*

Described in section 8.2.3.2.

#### 8.4.1.4 Approved Change Requests

Approved Change Requests originating from the business, or more specifically a program or portfolio, are inputs to be added to the list of approved project changes for implementation in future Sprints. Each change can require its own Epic or User Story and could become an input to the *Develop Epic(s)* process. Approved Change Requests could also result from other Scrum processes, where they are initially considered unapproved Change Requests (until they are discussed and approved).

Change Requests and Approved Change Requests are discussed in sections 6.3.1, 6.4.2.1, and 6.6.

#### 8.4.1.5 Unapproved Change Requests

Requests for changes are usually submitted as Change Requests and remain in an unapproved status until they are formally approved. Unapproved Change Requests in the *Develop Epic(s)* process could come from the *Create Deliverables*, *Conduct Daily Standup,* and sometimes other processes. Change Requests and Unapproved Change Requests are discussed in sections 6.3.1 and 6.4.2.1.

#### 8.4.1.6 Laws and Regulations

Depending on the project, there may be applicable laws and regulations imposed by governing bodies, which impact the planning and execution of the project. Laws are external to the organization and imposed by a governmental entity. Regulations can either be internal or external. Internal regulations are those that are applicable within the company, typically based on organizational policies. Internal regulations often pertain to company procedures and processes related to quality management, financial management, staffing management, and so on. External regulations are those relating to government-established standards, norms, and/or requirements. Laws and regulations must be considered while developing Epics. Although Epics are based on business requirements, to meet those requirements, the project team must abide by both applicable internal and external laws and regulations.

At times, some of the laws and regulations impacting multiple Scrum projects may be included as part of the Scrum Guidance Body Recommendations, as discussed in section 8.1.1.7.

#### 8.4.1.7 Applicable Contracts

If the entire project or portions of it are being completed under a contract, the contract defines the scope of work and the specific terms of the contract. The type of contract used influences project risk.

Some of the most common types of contracts used in Scrum projects are as follows:

- **Incremental Delivery Contract**—This contract mandates inspection points at regular intervals. It helps the customer and business stakeholders make decisions regarding product development periodically throughout the project at each inspection point.

The customer can either accept the current development of the product, decide to stop the development, or request product modifications.

- **Joint Venture Contract**—This contract is used when two or more parties partner to accomplish the work of a project. The parties involved in the project will both achieve some Return on Investment because the revenues or benefits generated will be shared between the parties.
- **Development in Phases Contract**—This contract makes funding available each month or each quarter after a release is successfully completed. It gives incentive to both customer and supplier and ensures that the monetary risk for the customer is limited to that particular time period since unsuccessful releases are not funded.
- **Incentive and Penalty Contract**—This contract is based on the agreement that the supplier will be rewarded with a financial incentive if the project's products are delivered on time but will incur financial penalties if the delivery is late.



Other popular contract types include paying by features contract, time and materials contract, fixed price and fixed scope contract, and fixed profit contract.

Epics should be developed keeping in mind the terms and conditions of the contract and the contract type being used.

#### 8.4.1.8 Previous Project Information

Information and insights gained from previous similar projects within the organization are valuable inputs for developing Epics and assessing risk. Previous project information may be reflected in previous Scrum Core Team members' notes, lessons learned documents, and stakeholder feedback. Best practices and other useful information may also be available through the Scrum Guidance Body recommendations.

#### 8.4.1.9 Scrum Guidance Body Recommendations

Scrum Guidance Body recommendations may include information on rules, regulations, standards, and best practices for developing Epics. For more information on the Scrum Guidance recommendations, see section 8.1.1.7.

## 8.4.2 Tools

### 8.4.2.1 User Group Meetings*

User Group Meetings provide the Product Owner with firsthand information about user expectations from relevant business stakeholders (primarily users and/or customers). Relevant members from the Scrum Core Team may also be present at User Group Meetings. These meetings help in formulating the Acceptance Criteria for the product and provide valuable insights for developing Epics. User Group meetings also promote buy-in for the project and create a mutual understanding among the Product Owner, Scrum Team, and relevant business stakeholders. User Group Meetings are vital in the prevention of expensive rework that may result from a lack of clarity regarding expectations and requirements.

### 8.4.2.2 User Story Workshops

User Story Workshops may be held as part of the *Develop Epic(s)* process. The Scrum Master facilitates these sessions with the entire Scrum Core Team, and at times, other business stakeholders. These workshops help the Product Owner to prioritize requirements and enable the Scrum Core Team to gain a shared perspective of the Acceptance Criteria. 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. They also facilitate preparation for Sprint planning. 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.

### 8.4.2.3 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 clarifications on particular subjects or concepts. Through questioning, constructive criticism, and feedback, focus groups lead to a better and higher 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 consensus.

Focus group sessions can help teams come up with innovative ideas, solve problems, and give suggestions for improvement. These meetings facilitate fact-finding and generate ideas and feedback from potential users and product developers. They are usually conducted to assist with planning, evaluating, or improving a product or service. Insights obtained from these meetings can also help develop Epics and User Stories. At times, Focus Group Meetings are conducted to resolve issues that may arise during the development of Epics.

#### 8.4.2.4 User or Customer Interviews

Engaging business stakeholders, including the sponsor(s), 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/or customers can help to ensure that the requirements of Epics align with the overall project vision, thereby delivering greater value.

These interviews help to:

- Identify and understand business stakeholders' needs and expectations
- Gather opinions and facts
- Understand business stakeholders' perspective of the end product
- Gather feedback about the iterated or partially developed product
- Get buy-in and commitment from users and/or customers



#### 8.4.2.5 Questionnaires

A cost-effective way to gain quantitative and qualitative statistical insight from a large number of users or customers is to use surveys or questionnaires. A questionnaire is a research or data collection tool that contains questions to be asked to a chosen set of individuals 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. While developing Epics, the Product Owner or the Scrum Master might conduct a survey to gather relevant information from business stakeholders or the Scrum Team.

#### 8.4.2.6 Risk Identification Techniques

Described in section 7.4.1.1.

#### 8.4.2.7 Scrum Guidance Body Expertise

While creating Epics, Scrum Guidance Body Expertise could refer to documented regulations, standards, and/or best practices for creating Epics. There may also be a team of subject matter experts who are available to assist the Product Owner in developing Epics. 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 a particular project, as they tend to move from project to project during the 'selling phase' or 'phase zero' which involves customers and/or users. For more information on the Scrum Guidance Body, see section 3.2.2.

### 8.4.2.8 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 8-16 shows Vabro's interface for epic development. It includes features for creating, managing, and organizing epics aligned with Scrum project goals.

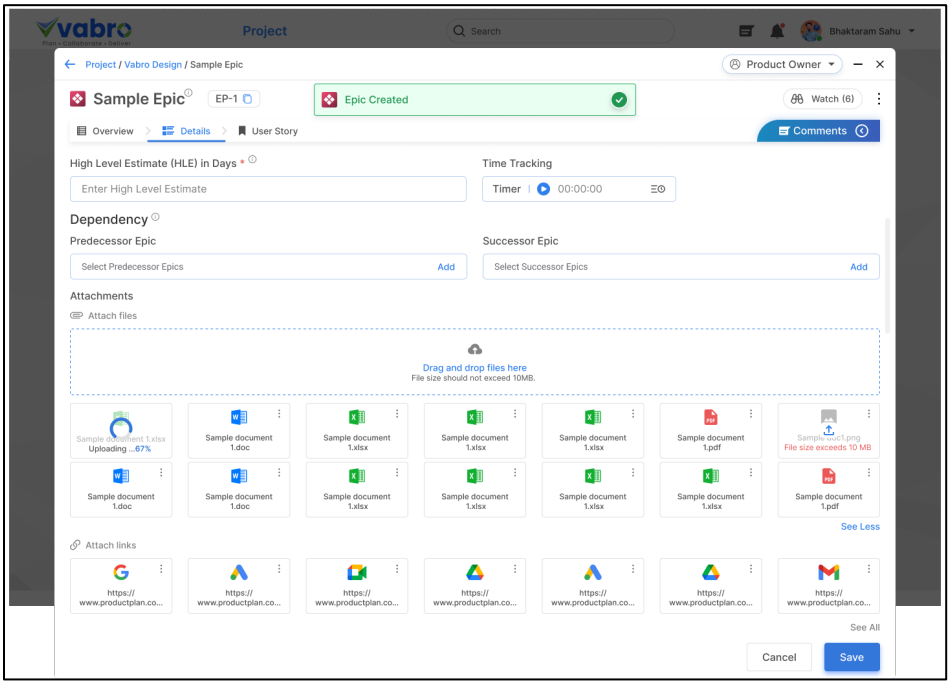


**Figure 8-16: Sample Scrum Project Tool – Vabro**

Figure 8-17 depicts the Jira tool used for managing epics. It supports backlog prioritization, user story linkage, and progress tracking within Scrum.

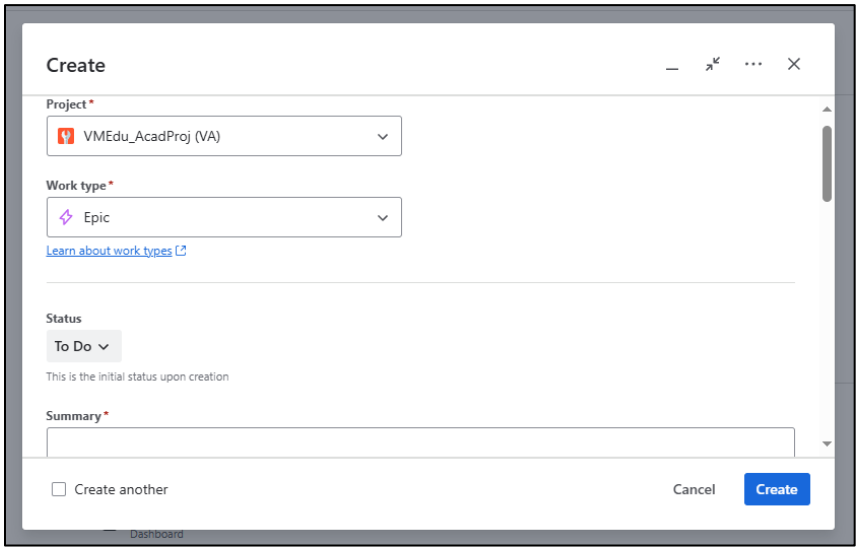


**Figure 8-17: Sample Scrum Project Tool – Jira**

## 8.4.3 Outputs

### 8.4.3.1 Epic(s)*

Epics are written in the initial stages of a project when most requirements are high-level functionalities or when product descriptions are broadly defined. Epics exist as large, unrefined User Stories in the Prioritized Product Backlog. Epics help the Product Owner and relevant business stakeholders in planning for releases, prioritizing high-level requirements, and establishing an overall roadmap for the project. As the Product Owner gets additional clarity on user requirements, these Epics are then broken down into smaller, more granular User Stories. So, Epics contain one or more User Stories. These smaller User Stories are simple, short, and easy-to-implement functionalities or blocks of tasks to be completed in a Sprint. User Stories should be defined by the Product Owner and need to satisfy the Definition of Ready, and should be estimated, before they come up in the Prioritized Product Backlog for addition in an upcoming Sprint.



### 8.4.3.2 Personas*

Personas are highly detailed fictional characters, representative of the majority of users and of other business stakeholders who may not directly use the end product. 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.

**Creating a Persona**—This involves assigning a fictional name and preferably a picture, like a stock image, to the character. The Persona will include highly specific attributes such as age, gender, education, environment, interests, and goals. A quote illustrating the Persona's requirements can also be included. Below is an example of a Persona for a travel website.

> *Example:* Vanessa is a 39-year-old resident of San Francisco. She is pursuing her passion for traveling after having a highly successful career as an attorney. Vanessa likes to have options while picking air travel and accommodations so that she can choose the best and the most affordable services. She gets easily frustrated with slow and cluttered websites.

### 8.4.3.3 Approved Changes

Unapproved Change Requests may be approved by the Product Owner during the *Develop Epic(s)* process, at times with suggestions and other input provided by relevant business stakeholders. Such changes are categorized as Approved Changes and can be prioritized and implemented in future Sprints. Change Requests and Approved Change Requests are discussed in sections 6.3.1 and 6.4.2.1.

### 8.4.3.4 Identified Risks

When creating Epics, new risks may be identified, and these identified risks form an important output of this process. Identified risks contribute to the development of the Prioritized Product Backlog (also be referred to as the Risk Adjusted Product Backlog). Risk Identification is described in section 7.4.1.

## 8.5 Create Prioritized Product Backlog

In this process, Epics are refined and elaborated, and most importantly, prioritized according to their respective business value to create a Prioritized Product Backlog for the project. Additionally, based on the Scrum Guidance Body recommendations, the Product Owner and the Scrum Team establish the Done Criteria for the project.

Figure 8-18 shows all the inputs, tools, and outputs for the *Create Prioritized Product Backlog* process.

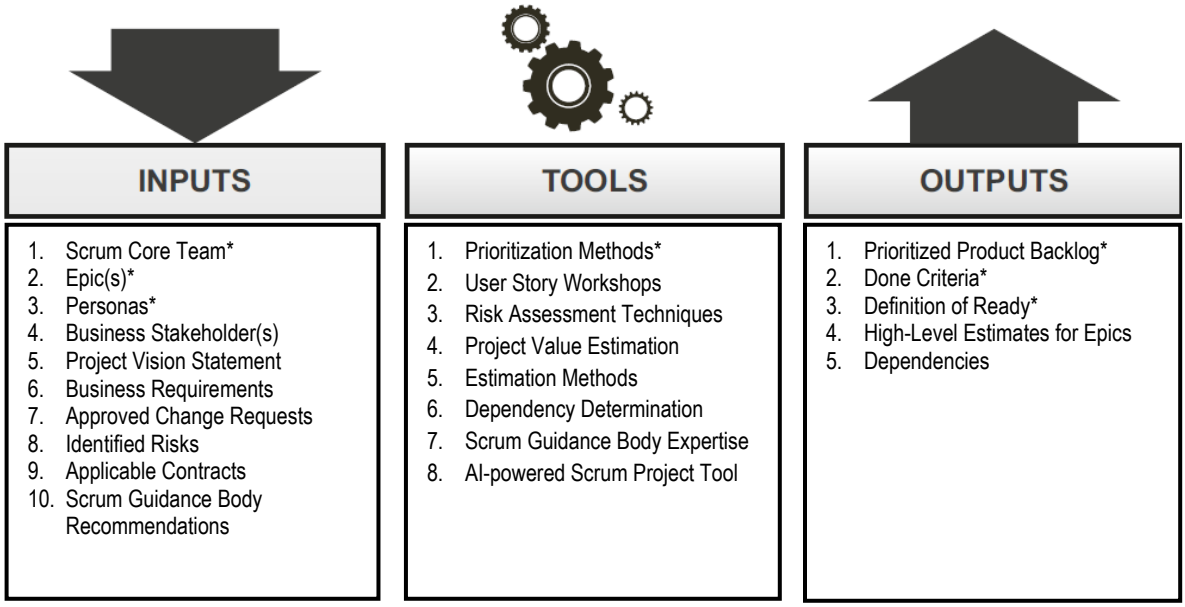


**Figure 8-18: Create Prioritized Product Backlog—Inputs, Tools, and Outputs**

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

Figure 8-19 shows the data flow for product backlog creation. It maps how inputs are processed and refined into a structured backlog ready for release planning.

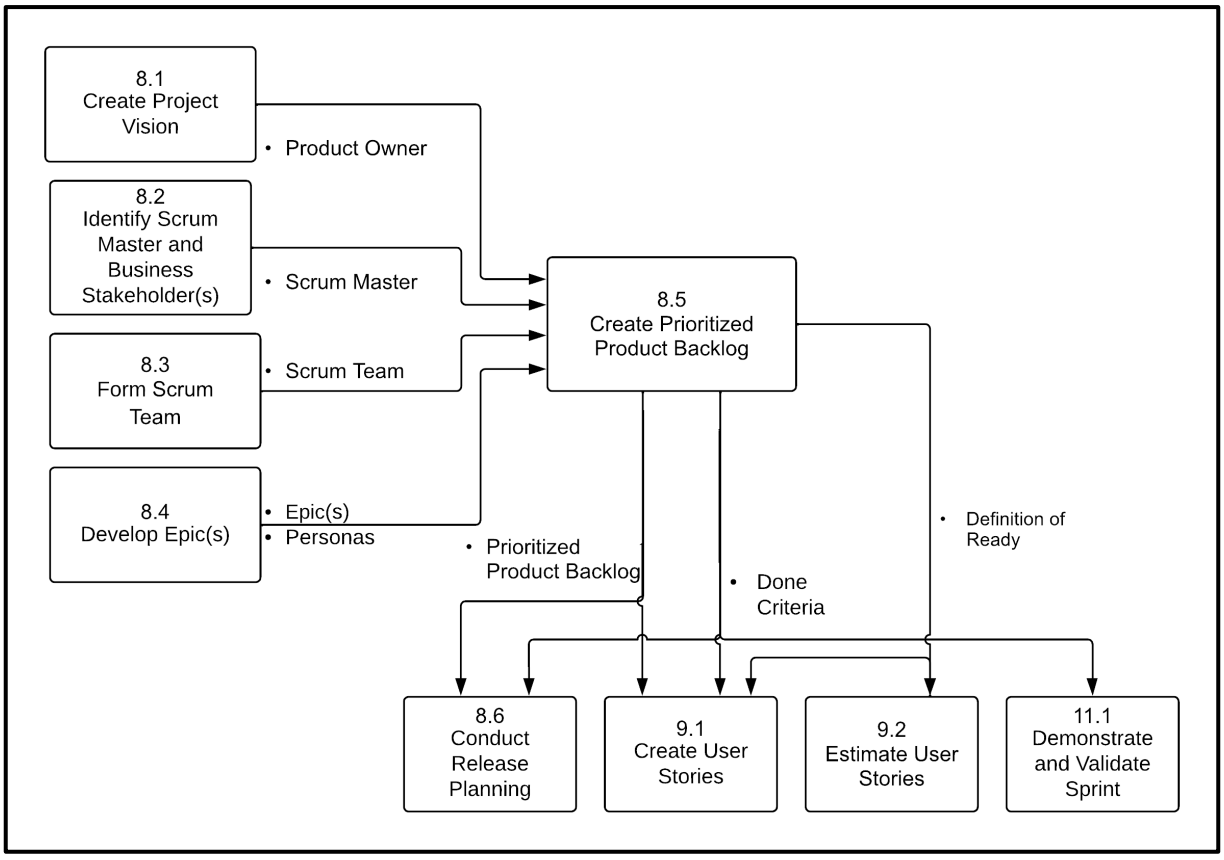


**Figure 8-19: Create Prioritized Product Backlog—Data Flow Diagram**

## 8.5.1 Inputs

### 8.5.1.1 Scrum Core Team*

Described in section 3.2.1.

### 8.5.1.2 Epic(s)*

Described in section 8.4.3.1.

### 8.5.1.3 Personas*

Described in section 8.4.3.2.

#### 8.5.1.4 Business Stakeholder(s)

Described in section 8.2.3.2.

#### 8.5.1.5 Project Vision Statement

Described in section 8.1.3.2.

#### 8.5.1.6 Business Requirements

The insights gained through various tools such as user or customer interviews, questionnaires, JAD sessions, Gap Analysis, SWOT Analysis, and other meetings, help develop a better perspective about the business requirements and aids in creating the Prioritized Product Backlog.

#### 8.5.1.7 Approved Change Requests

Described in section 8.4.3.3.

#### 8.5.1.8 Identified Risks

Described in section 8.4.3.4.

#### 8.5.1.9 Applicable Contracts

Described in section 8.4.1.7.

#### 8.5.1.10 Scrum Guidance Body Recommendations

While creating the Prioritized Product Backlog, Scrum Guidance Body recommendations may include information on rules, regulations, standards, and best practices for developing the Prioritized Product Backlog. The Scrum Guidance Body may also provide recommendations for the Definition of Ready (the criteria that a User Story has to satisfy before being considered for estimation or inclusion into a Sprint) and the Definition of Done (the set of rules that are applicable to the User Stories in a given Sprint). Further details about Scrum Guidance Body recommendations can be found in section 8.1.1.7.

## 8.5.2 Tools

### 8.5.2.1 Prioritization Methods*

Some techniques used to prioritize the Epics, User Stories, or requirements in the Prioritized Product Backlog, based on business value are presented below:

- **MoSCoW Prioritization Scheme**—The MoSCoW prioritization scheme derives its name from the first letters of the phrases "Must have," "Should have," "Could have," and "Won't have." This prioritization method is more effective than simple schemes. The labels are in decreasing order of priority with "Must have" features or functionalities being those without which the product will have no value and "Won't have" functionalities being those that, although would be nice to have, are not necessary to be included.
- **Paired Comparison**—In this technique, a list of all the Epics/User Stories in the Prioritized Product Backlog is prepared. Next, each Epic/User Story is taken individually and compared with the others in the list, one at a time. Each time two Epics or User Stories are compared, a decision is made regarding which of the two is more important. Through this process, a prioritized list of Epics or User Stories can be generated.
- **100-Point Method**—Dean Leffingwell and Don Widrig (2003) developed the 100-Point Method. It involves giving the customers 100 points they can use to vote for the Epics or User Stories that are most important. The objective is to give more weight to the Epics/User Stories that are of higher priority when compared to the others. Each group member allocates points to the various User Stories, giving more points to those they feel are more important. On completion of the voting process, prioritization is determined by calculating the total points allocated to each Epic or User Story.
- **Kano Analysis**—Described in section 4.5.2.

### 8.5.2.2 User Story Workshops

Described in section 8.4.2.2.

### 8.5.2.3 Risk Assessment Techniques

Described in section 7.4.2.1.

### 8.5.2.4 Project Value Estimation

Described in section 4.5.1.

#### 8.5.2.5 Estimation Methods

It might be useful to formulate some very high-level time estimates for Epics as these estimates can help the Product Owner plan appropriately for releases and can also aid in the prioritization process. However, since Epics planned at this stage are typically not yet broken down into well-defined User Stories, and there is also a possibility that some of the initially developed Epics may not be implemented, the team should not spend too much time at this point on estimating Epics.

This tool is used differently in the *Estimate User Stories* process (as described in section 9.2.2.1), where the Scrum Team is very involved with estimating well-defined User Stories. The Product Owner, Scrum Master, and relevant business stakeholders should understand that any high-level estimates provided in this process are used for guidance only, and that there may be significant deviations from the Scrum Team estimates for User Stories, once User Stories are properly defined and estimated.

Some specific tools that can be used to provide high-level estimates of Epics are:

- **Pre-existing Epic Estimates**—If similar Epics were developed in the past for the same project or other related projects, these prior estimates can be used to derive detailed estimates in this process.
- **Expert Opinion**—Experts who have implemented similar project functionalities in the past may be able to provide high-level effort estimates for Epics.
- **5-10-Minute Estimates**—Relevant team members can be provided with an overview of the general requirements for each Epic and asked to provide a rough estimate (e.g., in hours, days, or weeks) within five to ten minutes. This estimation method should be an option when only a small number of Epics need to be estimated, otherwise it could become time-consuming.
- **Affinity Estimation**—Affinity Estimation, such as T-shirt sizing, is a technique used to quickly create a time estimate for a large number of Epics. This is described in detail in section 9.2.2.1.

#### 8.5.2.6 Dependency Determination

Properly documenting dependencies helps the Scrum Team to determine the relative order in which Epics (and User Stories) should be executed to create the Sprint deliverables. Dependencies highlight the relationship and interaction between Epics (and between User Stories), both within the Scrum Team working on a given Sprint and with other Scrum Teams working on the project. As the Prioritized Product Backlog is created, the Product Owner identifies any dependencies and relationships between Epics (and between User Stories), including any technical dependencies and dependencies related to the availability of people, as these dependencies will impact the order and priority of work to be done on the project.

Dependencies can be mandatory or discretionary, internal, or external, or some combination of these. For example, a dependency may be both mandatory and external to the project.

The following is a description of each type of dependency:

- **Mandatory Dependencies**—Mandatory dependencies are dependencies that dictate the sequence of the work to be completed in the project and are therefore important to consider when initially creating the Prioritized Product Backlog. These dependencies may be mandatory because of the nature of the work (such as a physical limitation on work sequence) or may exist due to contractual obligations or legal requirements. Mandatory dependencies are also commonly described as hard logic.
- **Discretionary Dependencies**—Discretionary dependencies are dependencies that are placed into the workflow based on past experiences or best practices in a particular field or domain. For example, the team may decide to complete one Epic (or User Story) before another because this flow of completing the work has worked well on past projects.
- **External Dependencies**—External dependencies are dependencies pertaining to activities or factors that are outside the scope of the work to be executed by the Scrum Team but are needed to complete a project task or create a project deliverable. External dependencies are usually outside the Scrum Team's control and therefore can produce greater risk to a project.
- **Internal Dependencies**—Internal dependencies are those factors within the project that affect the sequence of work to be done. These factors are usually under the control of the Scrum Team.

There are diverse ways to identify, define, and present project tasks and their dependencies.



#### 8.5.2.7 Scrum Guidance Body Expertise

While creating the Prioritized Product Backlog, Scrum Guidance Body expertise could relate to documented rules and regulations or standards and best practices for prioritizing the product backlog. There may also be a team of subject matter experts who are available to assist the Product Owner in this process. This team could include business analysts, lead architects, senior developers, Scrum experts, and/or other experienced people. This expert group is usually not the same team that will stay on and work on this project, as they tend to move from project to project during the 'selling phase' or 'phase zero' which involves customers and/or users. For more information on the Scrum Guidance Body see section 8.4.2.7.

#### 8.5.2.8 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 8-20 depicts the Vabro tool interface used to create a prioritized product backlog. It supports task hierarchy, backlog grooming, and estimation tracking.

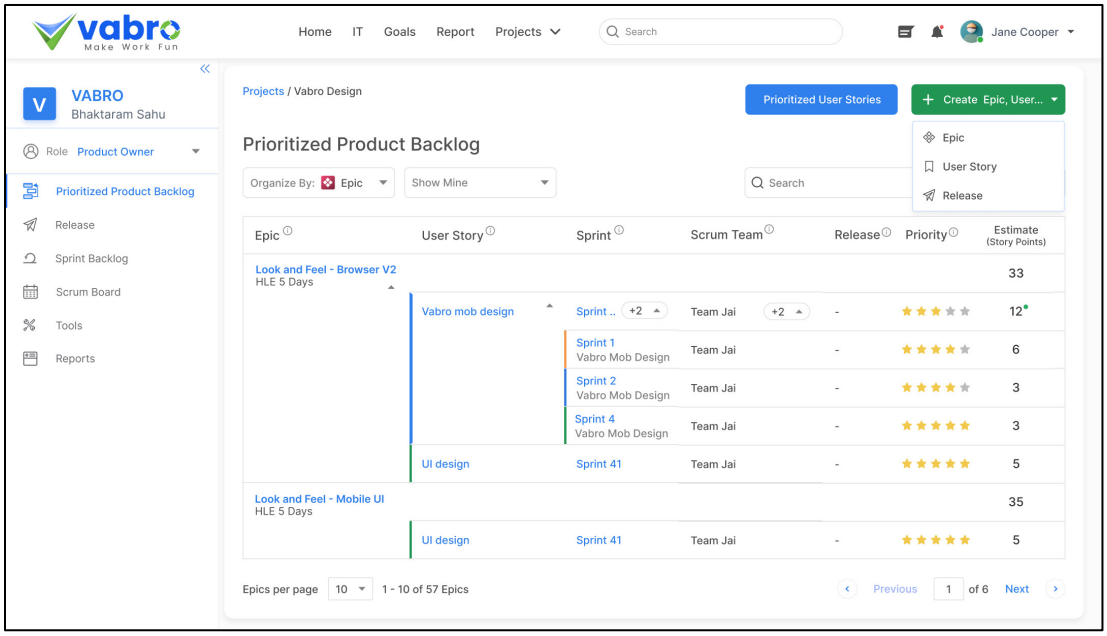


**Figure 8-20: Creation of Prioritized Product Backlog (Source: Vabro)**

Figure 8-21 shows how Jira manages the prioritized product backlog. It enables users to organize, prioritize, and visualize user stories and epics for development.

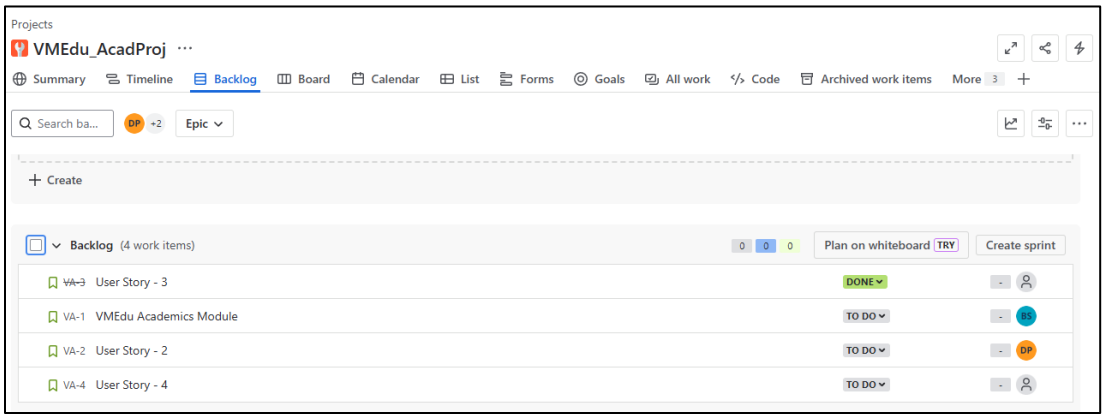


**Figure 8-21: Use of Prioritized Product Backlog (Source: Jira)**

## 8.5.3 Outputs

### 8.5.3.1 Prioritized Product Backlog*

The Product Owner develops a Prioritized Product Backlog which contains a prioritized list of business and project requirements written in the form of Epics, which are high-level User Stories. The Prioritized Product Backlog is based primarily on three factors—value, risk or uncertainty, and dependencies. It may also be referred to as the Risk Adjusted Product Backlog since it includes identified and assessed risks related to the project.

The Prioritized Product Backlog also encompasses all the approved changes that can be appropriately prioritized (as described in section 6.3.1).



Below is a description of the key factors that are considered when prioritizing items in the Prioritized Product Backlog.

- **Value—**The Product Owner is responsibility to ensure the delivery of Epics or those product increments or functionalities that provide the highest level of business value first. Even an extremely valuable product increment may not be part of the first release if there are other product increments of even higher value that are sufficient for a first release.
- **Risk and Uncertainty—**The more uncertainty that exists, the riskier a project will be. Therefore, it is important that riskier Epics in the Prioritized Product Backlog are given higher priority. Requirements carrying a higher level of risk will require risk mitigation actions. When risks and their corresponding risk mitigation actions are considered and prioritized against other backlog items, the result is a Risk Adjusted Product Backlog. Dealing with risks early in the project does not guarantee that a project will be successful, but it does enhance the team's ability to successfully deal with those risks. The topic of risks is further described in section 7.4.3.
- **Dependencies—**Most projects will inherently have dependencies between some of the Epics or User Stories. These dependencies should be taken into consideration while creating the Prioritized Product Backlog. Functional requirements often depend on other functional and even non-functional requirements. These dependencies can impact how the Epics (and User Stories) in the Prioritized Product Backlog are prioritized. Two of the most common ways to resolve dependencies are to either split a single Epic (or User Story) into multiple Epics (or User Stories) or combine the interdependent portions.
- **Estimates—**High-level estimates for Epics generated from the estimation methods are also available in the Prioritized Product Backlog.

It is important to note that the prioritization of Epics may be different from that of its underlying User Stories. For example, even if an Epic is categorized as high priority, some User Stories contained in the Epic may be categorized as low priority while other User Stories in the Epic may be categorized as high priority.

#### 8.5.3.2 Done Criteria*

The Done Criteria are a set of rules that are applicable to all User Stories. A clear definition of Done is critical because it removes ambiguity from requirements and helps the team adhere to mandatory quality norms. This definition is used to create the agreed-upon Done Criteria that will be used to determine when User Stories are complete. 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 Done Criteria may initially be determined and documented by the Scrum Guidance Body. However, there are typically project-specific Done Criteria that need to be incorporated during this process. Further details on the Done Criteria can be found in section 5.4.3.

#### 8.5.3.3 Definition of Ready*

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

The Definition of Ready may initially be determined and documented by the Scrum Guidance Body. However, there are typically project-specific criteria that need to be incorporated during this process. Further details on the Definition of Ready can be found in section 5.4.2.

#### 8.5.3.4 High-Level Estimates for Epics

High-level estimates for Epics may be initiated by the Product Owner using various estimation methods (as described in section 8.5.2.5). These estimates will help the Product Owner get an approximate idea of how much time and effort it will take to complete each Epic, which in turn will help with the prioritization of Epics in the Prioritized Product Backlog and plan project releases.

#### 8.5.3.5 Dependencies

Dependencies describe the relationship and interaction between different Epics (or User Stories) in a project. Dependencies can be classified as mandatory or discretionary; internal or external; or some combination of these (as discussed in section 8.5.2.6). Dependencies will affect the relative order in which Epics (and User Stories) will be executed to create the Sprint deliverables, and will therefore affect their priority, as documented in the Prioritized Product Backlog.

## 8.6 Conduct Release Planning

In this process, the Product Owner, with inputs from business stakeholders and members of the Scrum Team, develops the initial Release Planning Schedule, which is communicated to, and shared with, all business stakeholders and Scrum Team Members. It is understood that the iterative nature of Scrum may necessitate future adjustments to the release schedule. The length of each Sprint is also determined in this process.

Figure 8-22 shows all the inputs, tools, and outputs for the *Conduct Release Planning* process.

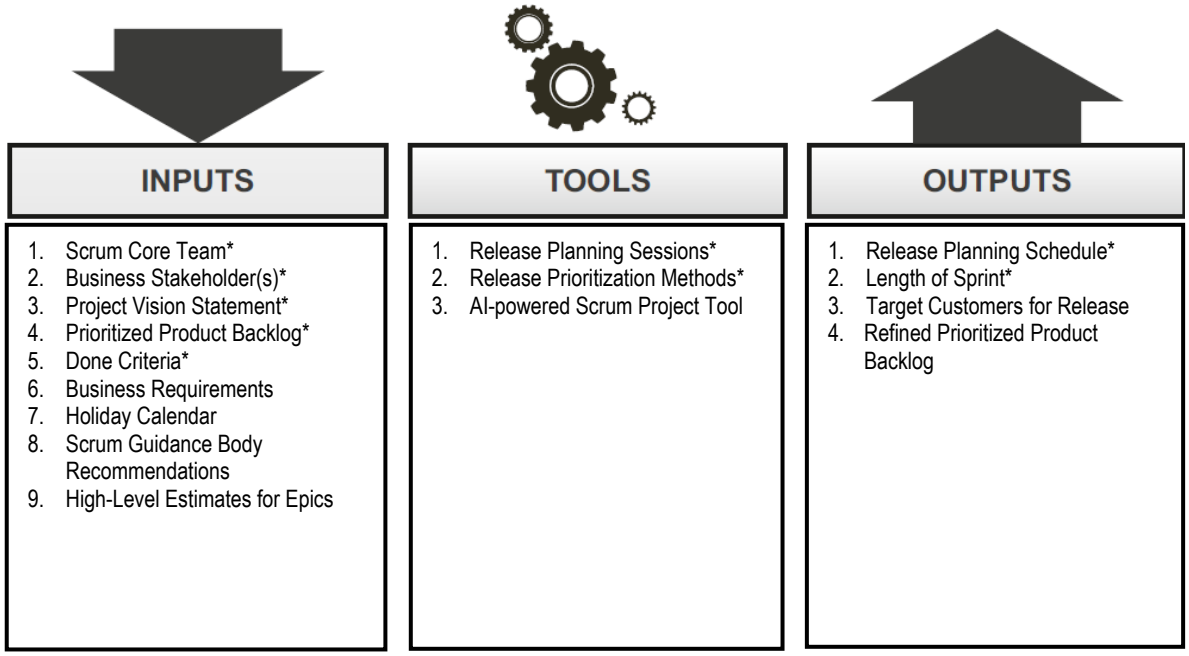


**Figure 8-22: Conduct Release Planning—Inputs, Tools, and Outputs**

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

Figure 8-23 illustrates the data flow during release planning. It shows how backlog items and estimates are translated into detailed release schedules and goals.

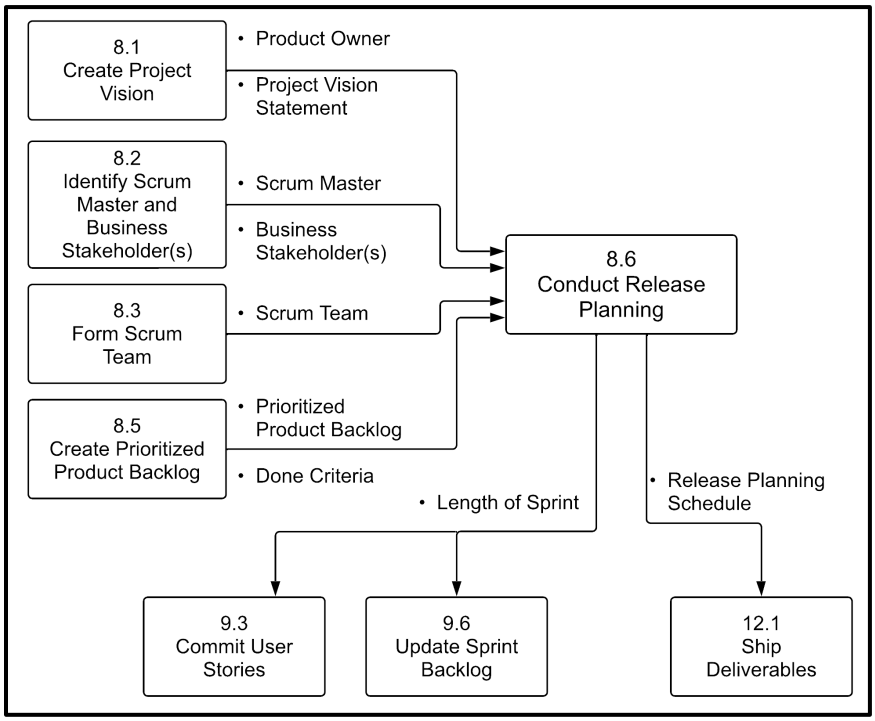


**Figure 8-23: Conduct Release Planning—Data Flow Diagram**

## 8.6.1 Inputs

### 8.6.1.1 Scrum Core Team*

Described in section 3.2.1.

### 8.6.1.2 Business Stakeholder(s)*

Described in section 8.2.3.2.

### 8.6.1.3 Project Vision Statement*

Described in section 8.1.3.2.

#### 8.6.1.4 Prioritized Product Backlog*

Described in section 8.5.3.1.

#### 8.6.1.5 Done Criteria*

Described in section 8.5.3.2.



#### 8.6.1.6 Business Requirements

Described in section 8.5.1.6.

#### 8.6.1.7 Holiday Calendar

It is important for the Scrum Team to keep track of key dates and availability of each team member. This can be accomplished through the use of a shared calendar that provides information on official holidays, leaves, travel plans, events, and so on that could impact project scheduling. This calendar will help the team in planning and executing Sprints.

#### 8.6.1.8 Scrum Guidance Body Recommendations

In the *Conduct Release Planning* process, Scrum Guidance Body recommendations may pertain to rules, regulations, standards, and best practices for developing the Release Planning Schedule. The Scrum Guidance Body may be the best authority for defining guidelines related to business value, release expectations, deployment strategies, quality, and security. For more information on the Scrum Guidance Body see section 3.2.2.

#### 8.6.1.9 High-Level Estimates for Epics

In release planning sessions, high-level estimates for Epics are used to plan releases using either a schedule-driven approach or a feature-driven approach. For more information on these estimates, see sections 8.5.2.5 and 8.5.3.4.

## 8.6.2 Tools

### 8.6.2.1 Release Planning Sessions*

Release planning sessions are conducted with the goal of developing a Release Planning Schedule for the project. This schedule defines when various sets of usable functionality or product increments will be delivered to the customer. A major objective of a Release Planning Meeting or session is to enable the Scrum Team to have an overview of the planned releases and delivery schedule for the product they are developing. This needs to align with the expectations of the Product Owner and relevant business stakeholders (primarily the project sponsor). Many organizations have a strategy regarding the release of projects. Some organizations prefer continuous deployment, where there is a release after the creation of specified usable product functionality.

Other organizations prefer phased deployment, where releases are made at predefined intervals. Depending on the organization's strategy, these planning sessions may be driven by functionality—in which the objective is to deliver working deliverables (of product increments) once a set of predetermined functionalities has been developed—or by date, where releases occur on predefined dates. Since the Scrum framework promotes information-based iterative-decision-making over the detailed upfront planning practiced in traditional waterfall-style project management, release planning sessions need not produce a detailed Release Planning Schedule for the entire project. The plan is expected to be updated continually as relevant and more detailed information becomes available.

### 8.6.2.2 Release Prioritization Methods*

Release prioritization methods are used to develop the Release Planning Schedule. These methods are usually industry- or organization-specific and are usually determined by the organization's senior management.

### 8.6.2.3 AI-powered Scrum Project Tool

Described in section 2.4.4.

Figure 8-24 shows release planning in Vabro. It provides visual tools for scheduling, tracking dependencies, and aligning releases with business priorities.

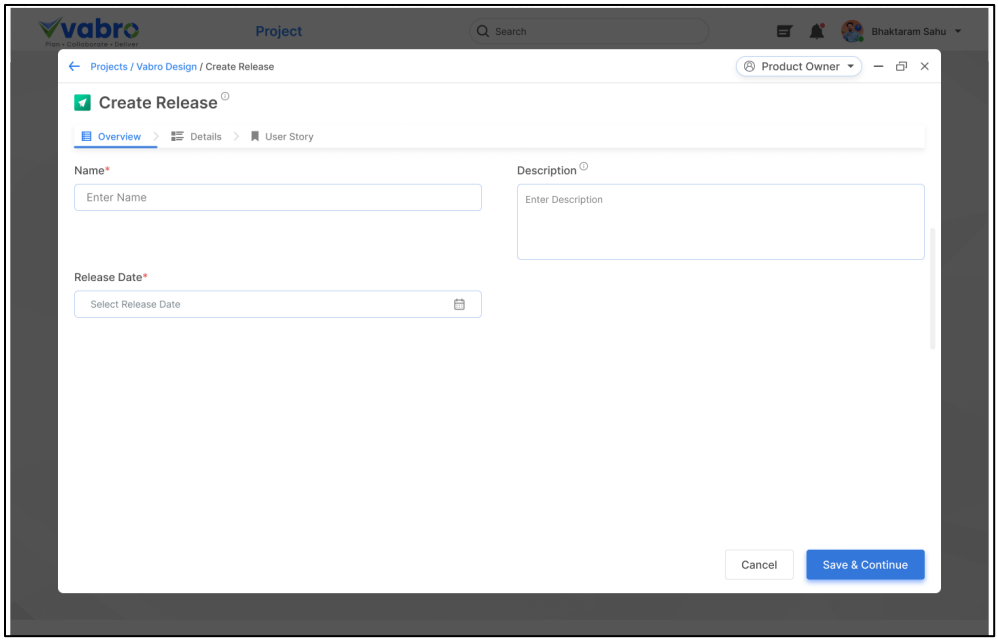


**Figure 8-24: Release Planning in Scrum (Source: Vabro)**

Figure 8-25 depicts release and configuration management in Jira. It highlights features for managing software versions, release readiness, and deployment tracking.

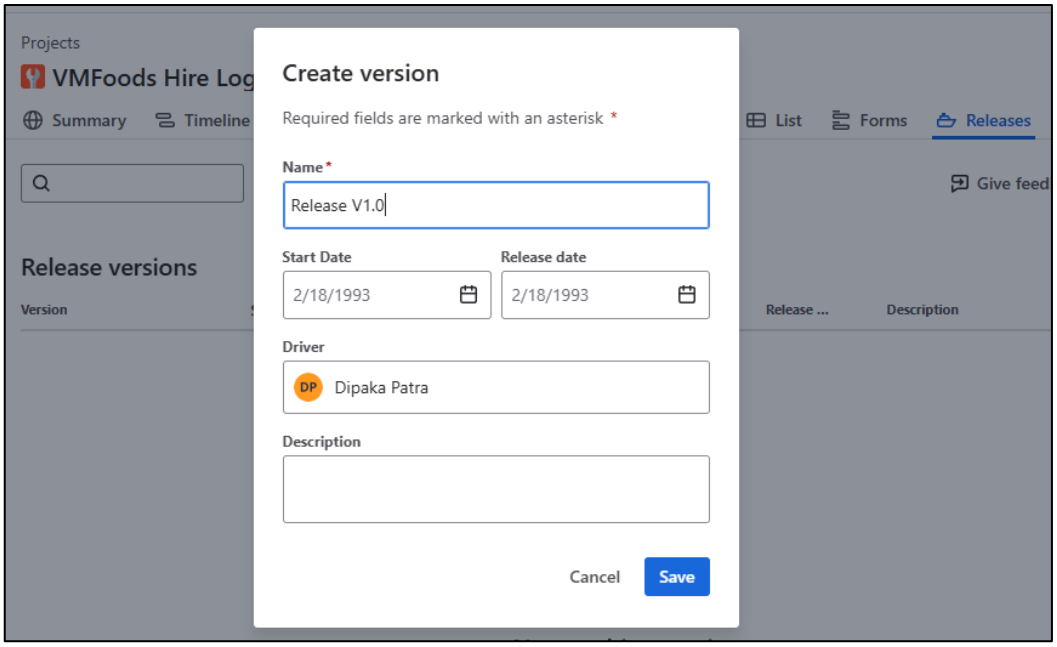


**Figure 8-25: Release and Configuration Management in Scrum (Source: Jira)**

## 8.6.3 Outputs

### 8.6.3.1 Release Planning Schedule*

A Release Planning Schedule is a key output of the *Conduct Release Planning* process. The Release Planning Schedule states which deliverables are to be released to the customers, along with planned intervals, and release dates. A release typically includes a group of User Stories in the Prioritized Product Backlog, which should be completed and shipped together as part of the release. It is important to note, however, that User Stories in a release do not always provide the complete functionality of Epics. Some Epics may only be partially completed since not all User Stories in an Epic are necessarily of a high enough priority to be part of that particular release. 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 may be driven by functionality—in which the objective is to deliver working deliverables once a set of predetermined functionalities has been developed—or by date, where releases occur on predefined dates. Each deliverable should only be released when it offers sufficient business value to the customer. During the *Conduct Release Planning* process, it may be useful to consider the Sprint Length and the assumptions for Team Velocity in order to derive a better Release Planning Schedule.

### 8.6.3.2 Length of Sprint*

Based on the various inputs, the business requirements, and details of the Release Planning Schedule, the Product Owner and the Scrum Team decide on the appropriate Sprint length for the project. Once determined, the length of each Sprint typically remains the same throughout the entire project. However, the duration of Sprints can be changed if the Product Owner and the Scrum Team have justification for the change. For example, early in the project, the team may still be experimenting to find the best Sprint length. Later in the project, a reduction in the Sprint length can occur due to improvements in the project environment. To get maximum benefits from a Scrum project and to provide maximum flexibility for change, the length of a Sprint should be as short as possible. A Sprint is typically Time-boxed with a duration of one to four weeks.

Most Scrum projects typically have Sprints Time-boxed at a duration of two or three weeks. However, for projects with very stable requirements, Sprints can extend to six weeks, if justified. Because changes are not allowed during a Sprint, the impact and frequency of any changes expected may have an impact on the decision related to the Length of Sprint. The impact of expected change on the Length of Sprint is described in section 6.5.1.

### 8.6.3.3 Target Customers for Release

Not every release will target all business stakeholders or users. The business stakeholder(s) may choose to limit certain releases to a subset of customers and/or users. The Release Planning Schedule should specify the target customers and/or users for each release.

#### 8.6.3.4 Refined Prioritized Product Backlog

The Prioritized Product Backlog, initially developed in the *Create Prioritized Product Backlog* process, may be refined in this process. For example, there may be additional clarity about the User Stories after the Scrum Core Team conducts release planning sessions with business stakeholders.

## 8.7 Initiate Phase Data Flow Diagram

Figure 8-26 illustrates the data flow for the Initiate phase. It outlines how stakeholder inputs and tools combine to produce critical outputs for initiating a Scrum project.

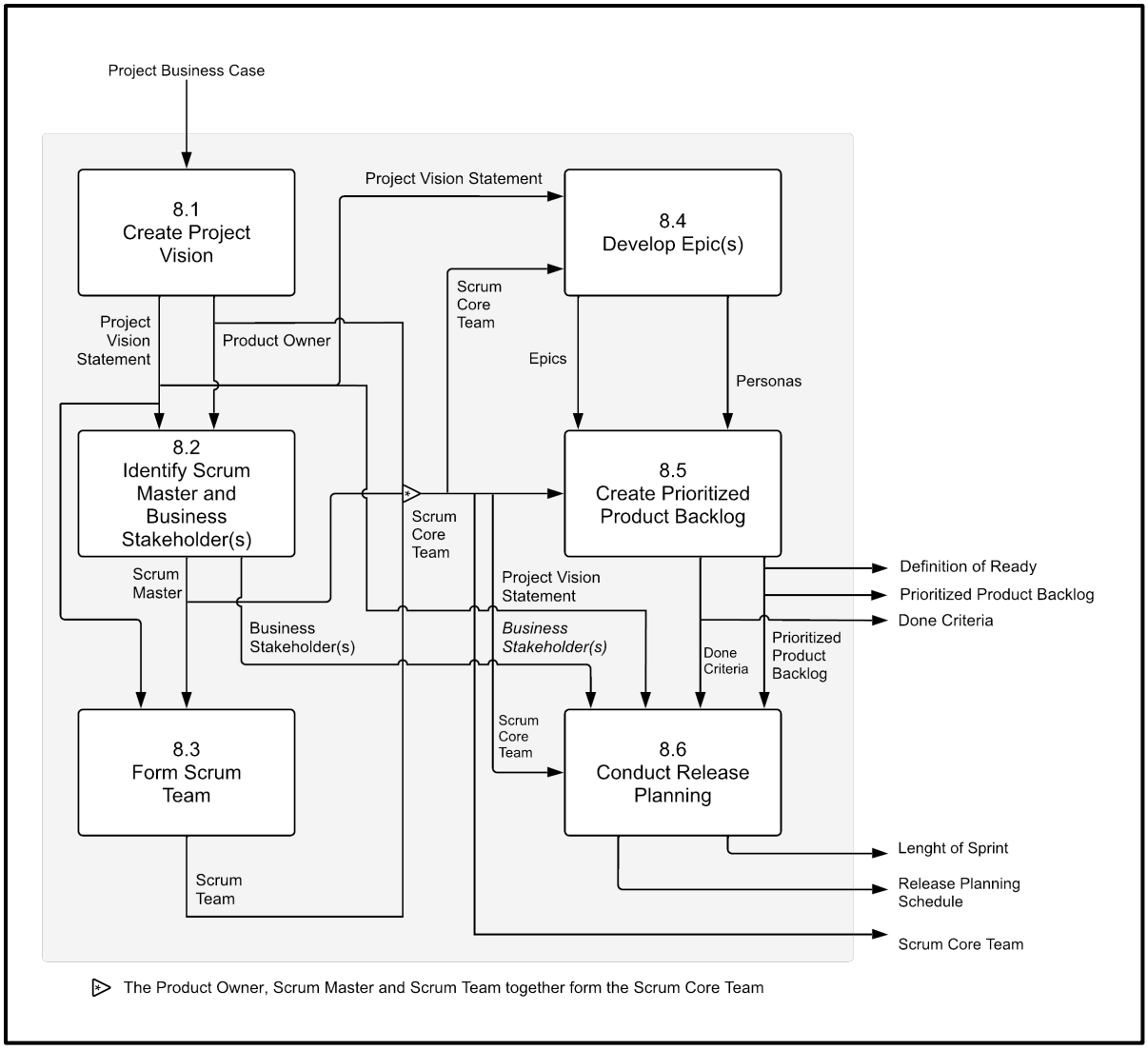


**Figure 8-26: Initiate Phase—Data Flow Diagram**

## 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™ is a brand of VMEdu, Inc.