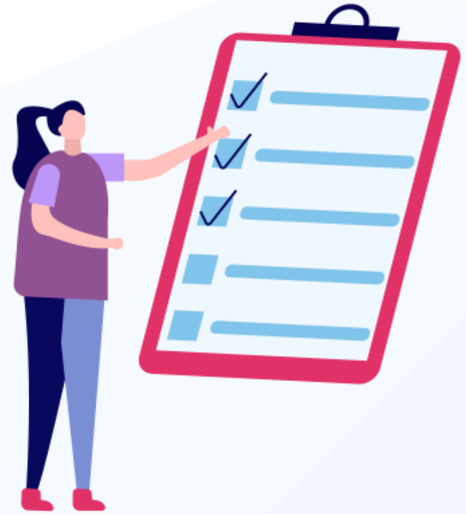
Complex Projects

Learning Objectives

By the end of this lesson, you will be able to:

- Explain how to handle a complex project in Agile to enable flexibility, iterative progress, and continuous feedback
- Describe the concept of scaling in Agile to illustrate how Agile principles are adapted for managing larger teams in complex projects
- Outline the Nexus framework to identify how it coordinates multiple Scrum teams through added roles, events, and artifacts
- Compare various agile tools to determine how they support agile practices and improve team efficiency



Understanding Agile for Complex Projects

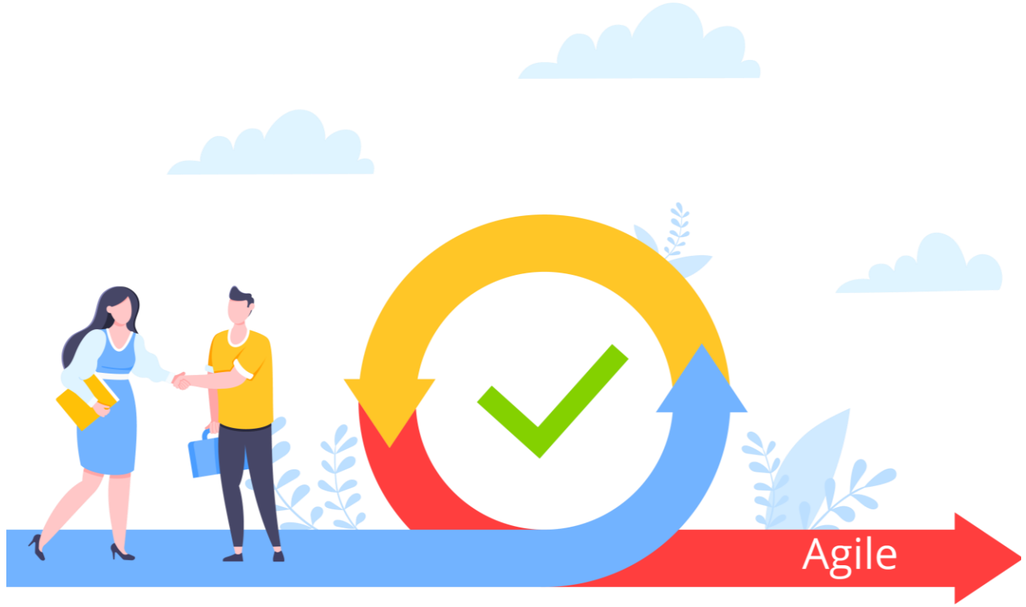
What Is a Complex Project?

It is characterized by high uncertainty, interdependent tasks, and dynamic environments.

These projects often involve multiple stakeholders, technological advancements, and changing requirements.

Why Agile for Complex Projects?

Traditional project management methodologies, like waterfall, struggle to cope with the complexities of modern projects.



Agile, with its iterative and incremental approach, is better suited to handle the dynamic nature of complex projects.

Advantages of Agile in Complex Projects

Allows adaptation to changing requirements and unforeseen challenges

Emphasizes teamwork and communication, which are essential for complex projects

Prioritizes delivering valuable features early and often

Promotes early identification and mitigation of risks

Involves the customer throughout the project, leading to higher satisfaction

Scaling Agile

What Is Scaling Agile?

Scaling Agile involves extending Agile methodologies to larger and more complex projects.

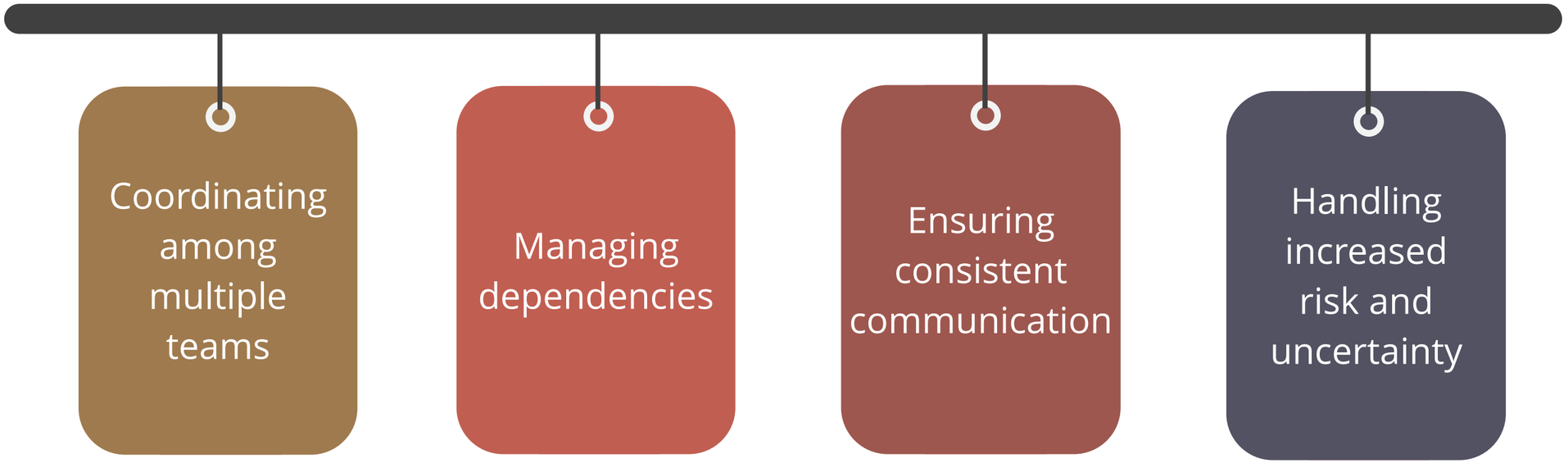
It requires adapting Agile principles to manage increased scope, larger teams, and more intricate dependencies.

It ensures that Agile practices remain effective as project complexity grows.

Why Scaling Agile Is Required?

Scaling Agile helps address challenges faced in complex projects.

Some of the challenges in complex projects are:



Addressing these challenges through scaling is essential for successful project delivery.

Using the Product Backlog in a Scaled Environment

Introduction to Product Backlog in Scaling

The product backlog lists everything the project needs, from features to bug fixes.

When scaling Agile, the product backlog must adapt to the increased scope and complexity.

This adjustment ensures teams can access the necessary information and work toward common goals.

Scaling the Product Backlog

It refers to managing and organizing the product backlog when a project involves multiple teams working on the same product or when the product is large and complex.

Prioritizing features that deliver the most value

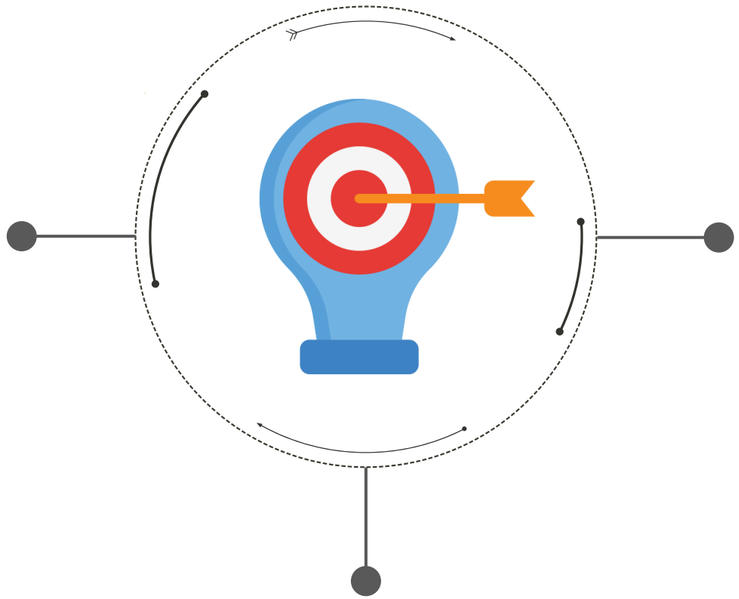
It involves:

Ensuring clarity and detail in backlog items

Regularly refining and updating the backlog to reflect changing priorities

Strategies to Scale a Product Backlog

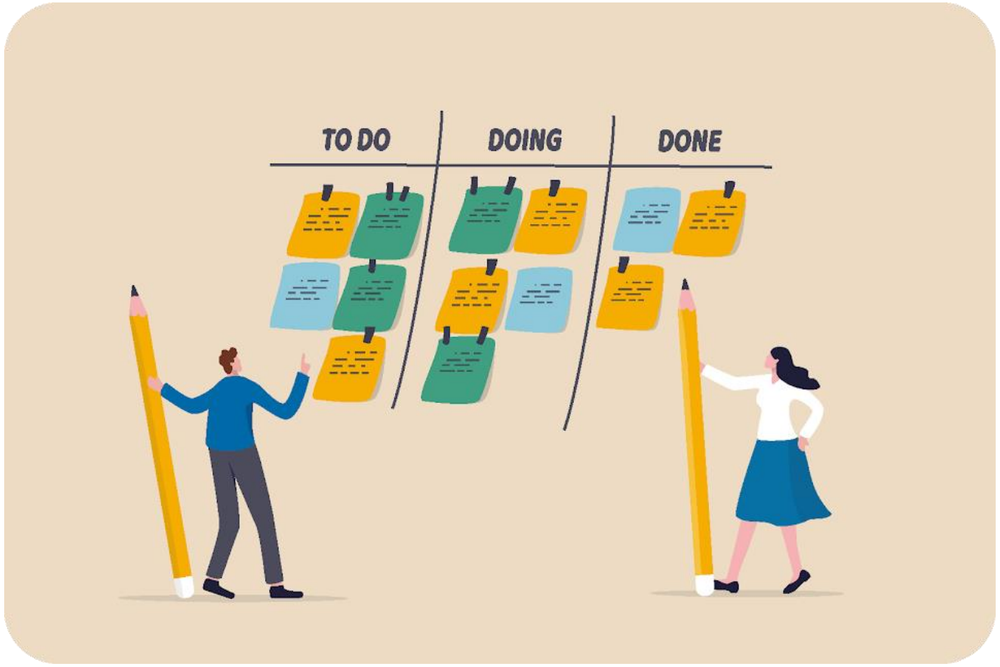
The three main strategies for scaling a product backlog are:



- Combining backlog items into epics
- Structuring product owner teams
- Creating different views of the product backlog

Combining Backlog Items into Epics

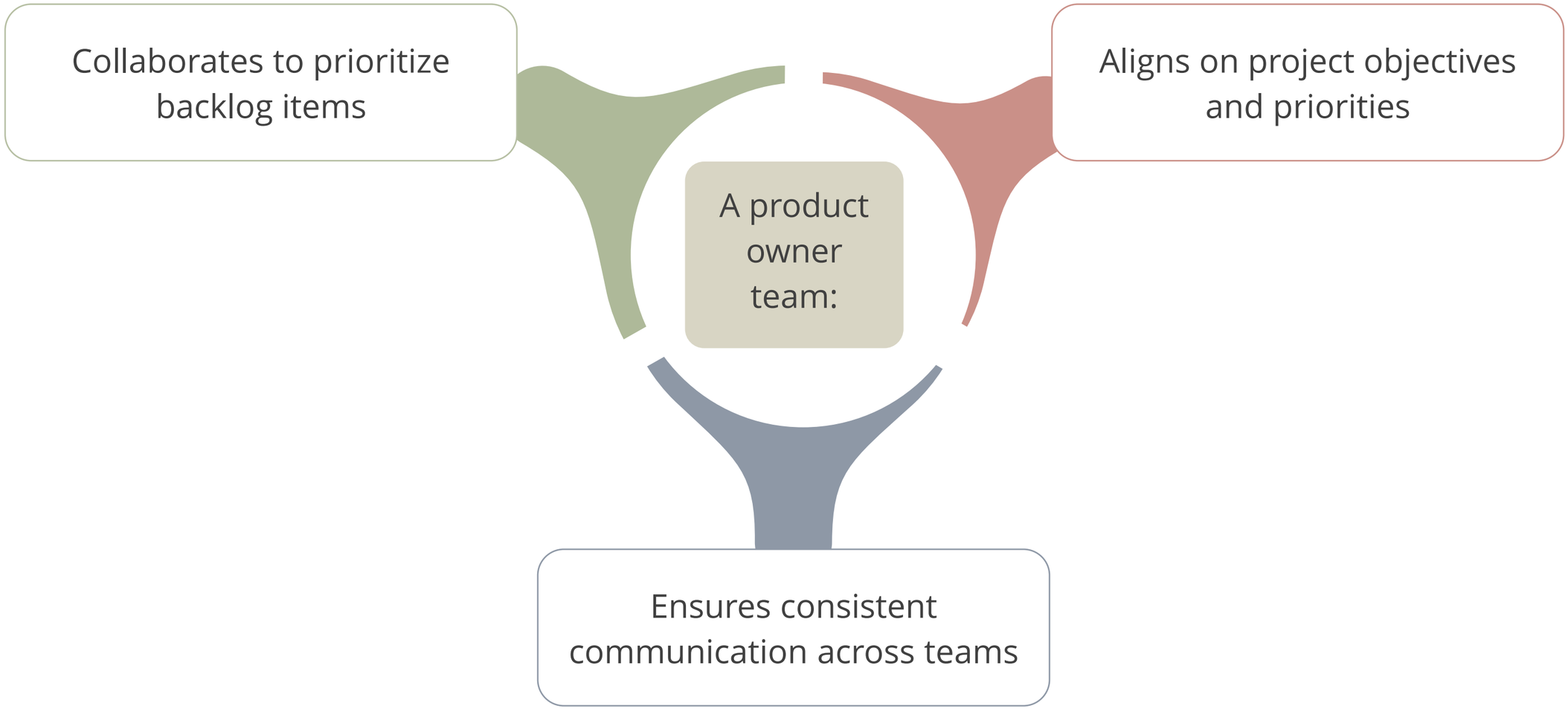
To simplify large and complex backlogs, combine backlog items into epics



- When scaling a product backlog, a single product owner may not be sufficient to manage all the requirements.
- Creating a product owner team allows the distribution of responsibilities among several product owners.

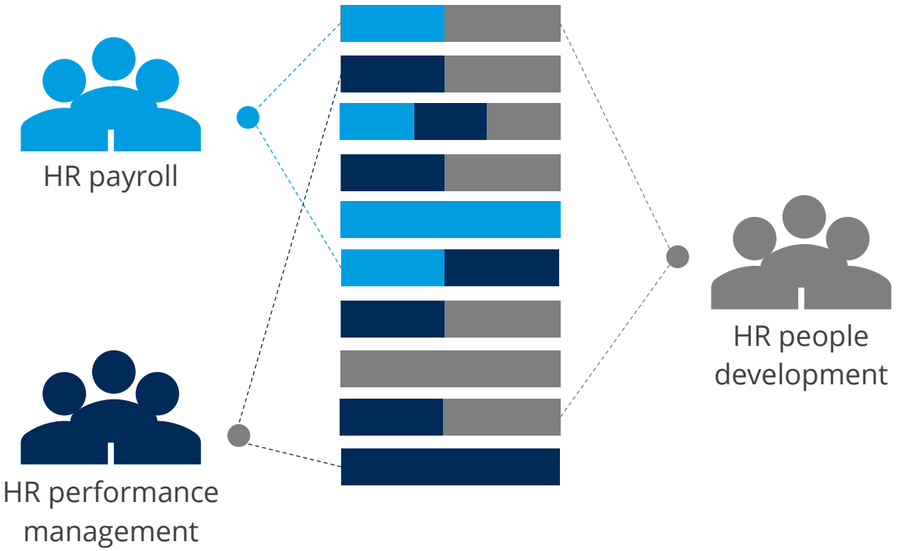
Combining Backlog Items into Epics

A product owner team, consisting of product owners and stakeholders, ensures that the backlog reflects the project's overall goals.



Combining Backlog Items into Epics

Example:

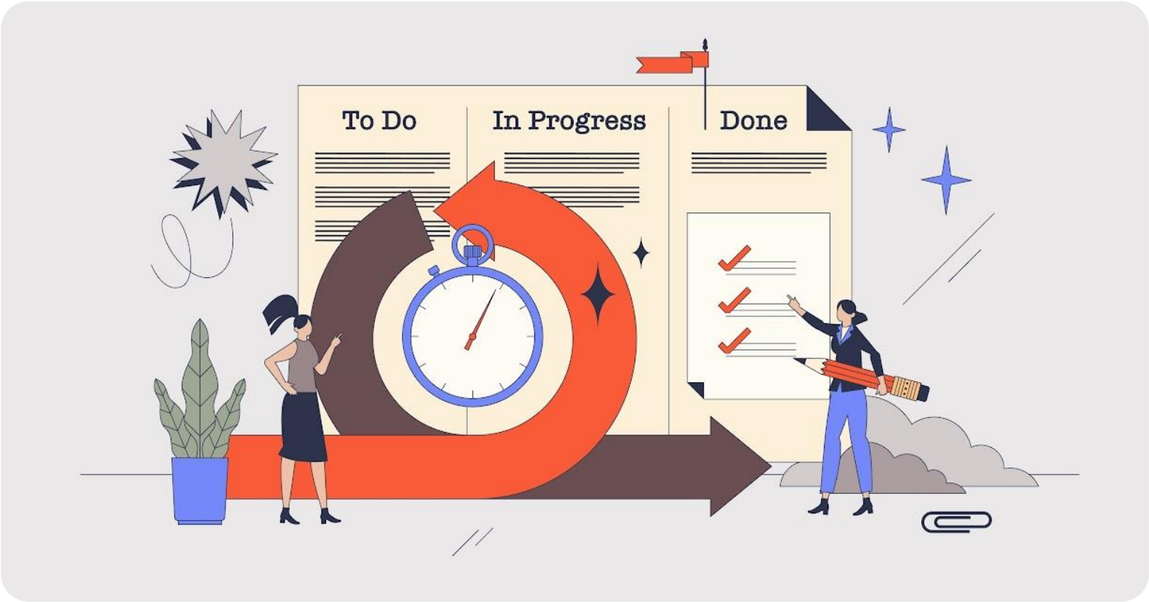


This image shows how different HR teams (**HR payroll**, **HR performance management**, and **HR people development**) work together to create a product backlog, which is a list of tasks or features needed for a project.

The lines connecting the teams to the backlog indicate that each team contributes to and influences the items in the backlog, ensuring that the final product meets the needs of all HR areas.

Creating Different Views of the Product Backlog

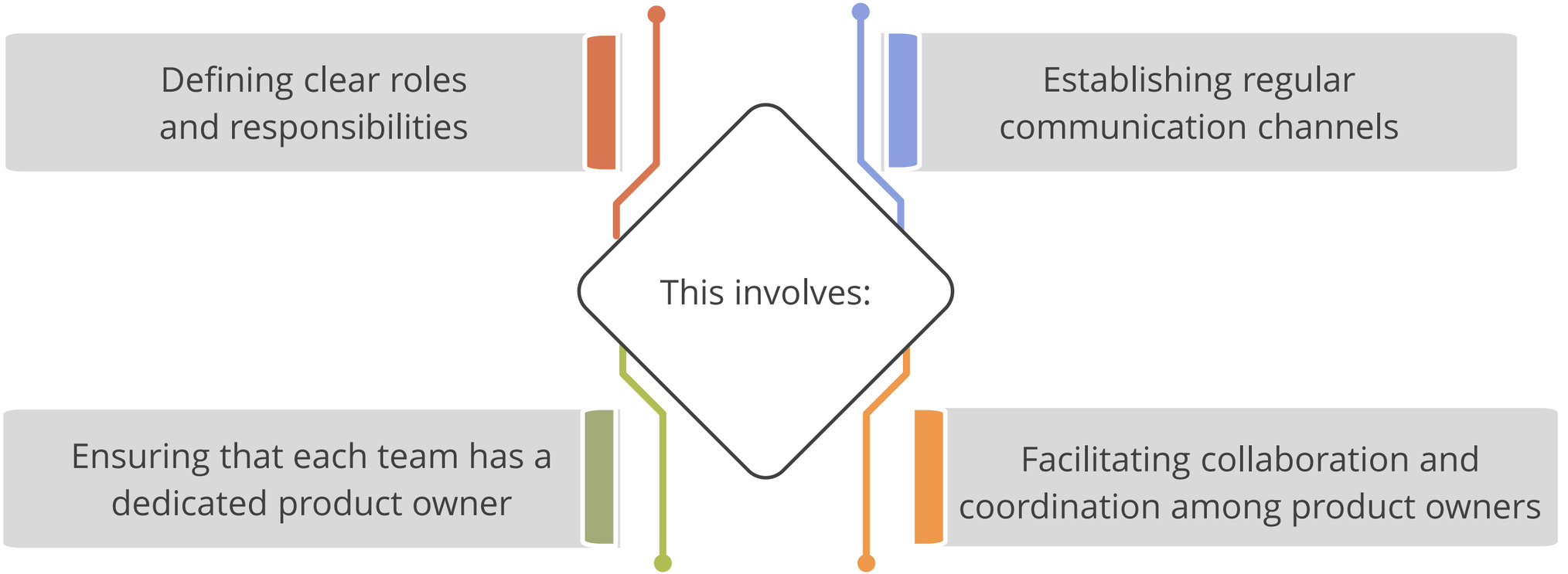
In many organizations, a single product may be viewed, used, and understood differently by various stakeholder groups.



To address these differences, create different views of the product backlog to align with each group’s specific needs and priorities

Structuring Product Owner Teams

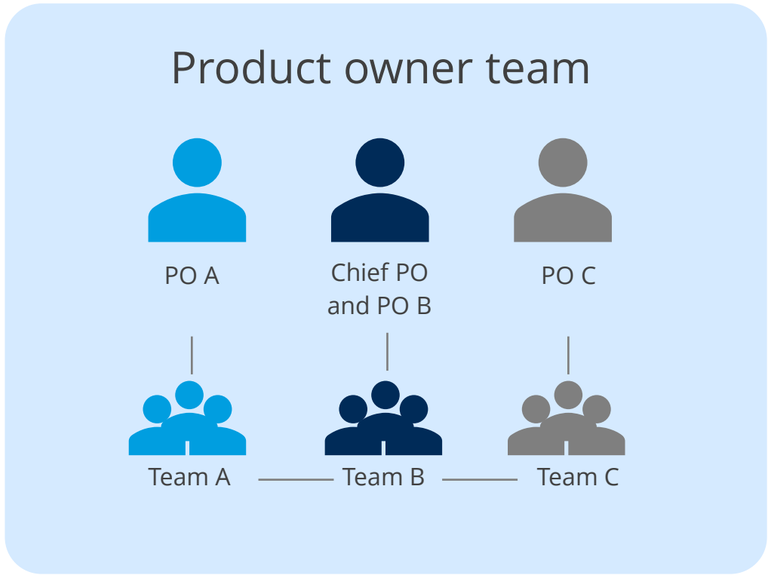
It refers to the organization and arrangement of product owners within a large or complex project, particularly in a scaled Agile environment where multiple teams are involved.



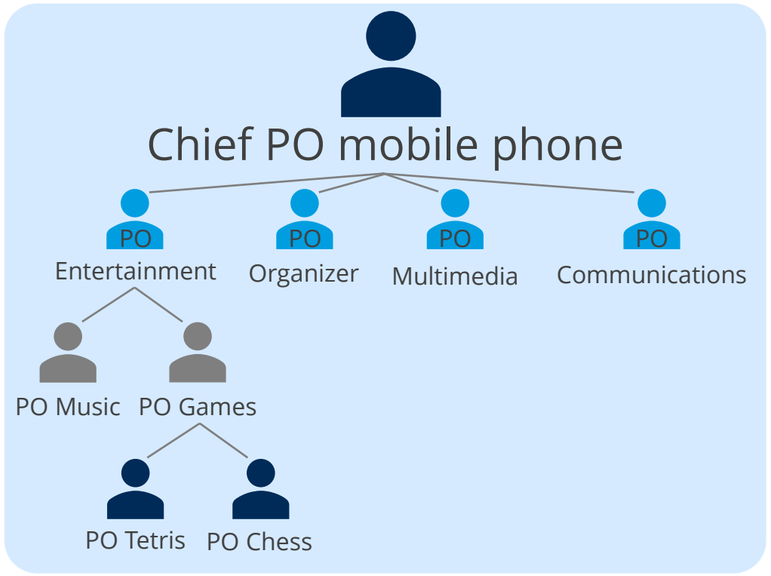
Structuring Product Owner Teams

This image shows two ways to organize product owner teams:

Simple product owner hierarchy



Complex product owner hierarchy



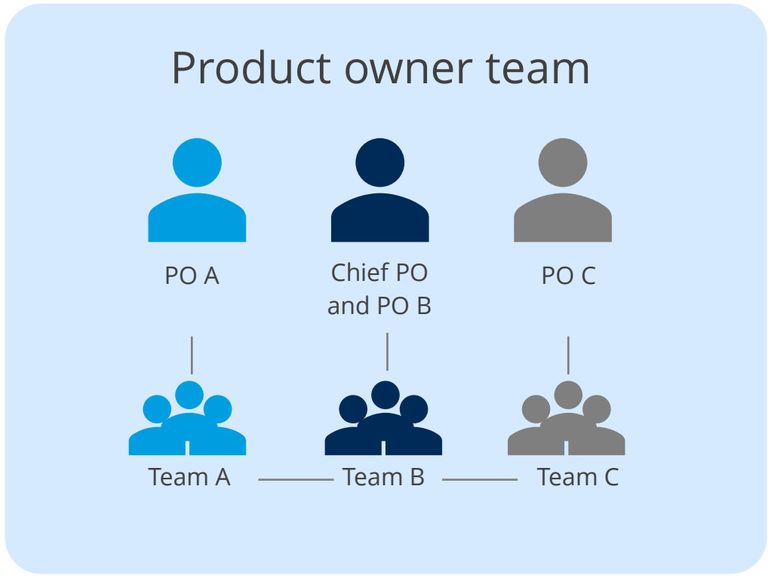
Simple product owner hierarchy:

Here, a chief product owner (**Chief PO**) works with a couple of other product owners. Each product owner guides their team (**Team A, Team B, and Team C**) to build different product parts.

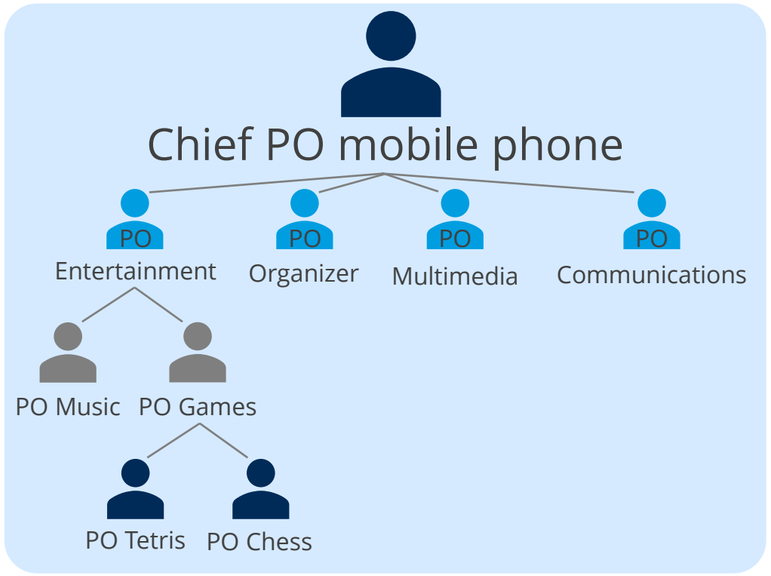
Structuring Product Owner Teams

This image shows two ways to organize product owner teams:

Simple product owner hierarchy



Complex product owner hierarchy



Complex product owner hierarchy:

Here, a chief product owner (**Chief PO**) oversees several product owners, each responsible for different parts or features of a more complex product (like a mobile phone). Some features are broken down further, with even more specialized product owners handling specific areas.

Scaling Scrum with Nexus

Nexus Framework

It extends Scrum by adding roles, events, and artifacts specifically aimed at addressing the challenges of large-scale product development, such as integration and cross-team dependencies.

The framework ensures that all teams work together effectively to produce a cohesive, integrated product increment at the end of each sprint.

Differences between Scrum and Nexus

Scrum

- Focuses on a single team working on a product or project
- Consists of three roles such as product owner, Scrum Master, and development team
- Requires each team to work on its own increment, which is integrated at the end of the sprint

Nexus

- Extends Scrum to coordinate the work of three to nine Scrum teams working together on a single product
- Introduces the Nexus Integration Team (NIT), which includes representatives from each Scrum team, such as Scrum Masters and senior developers
- Ensures continuous integration of work from all teams, facilitated by the NIT, to produce a cohesive product increment

Differences between Scrum and Nexus

Scrum

- Includes five events, such as sprint, sprint planning, daily scrum, sprint review, and sprint retrospective
- Includes artifacts, such as backlog and product increment

Nexus

- Includes four events, such as Nexus sprint planning, Nexus daily scrum, Nexus sprint review, and Nexus sprint retrospective
- Includes artifacts, such as the Nexus Sprint Backlog, Nexus Integrated Increment, and Nexus Definition of Done

Key Components of the Nexus Framework

Nexus Integration Team (NIT)

Is responsible for coaching, guiding, coordinating, and integrating the work of multiple Scrum teams

Nexus Sprint

Is a synchronized iteration that takes place across all Nexus teams

Integrated increment

Is a potentially shippable product increment that is delivered at the end of a Nexus Sprint

Nexus Integration Team (NIT)

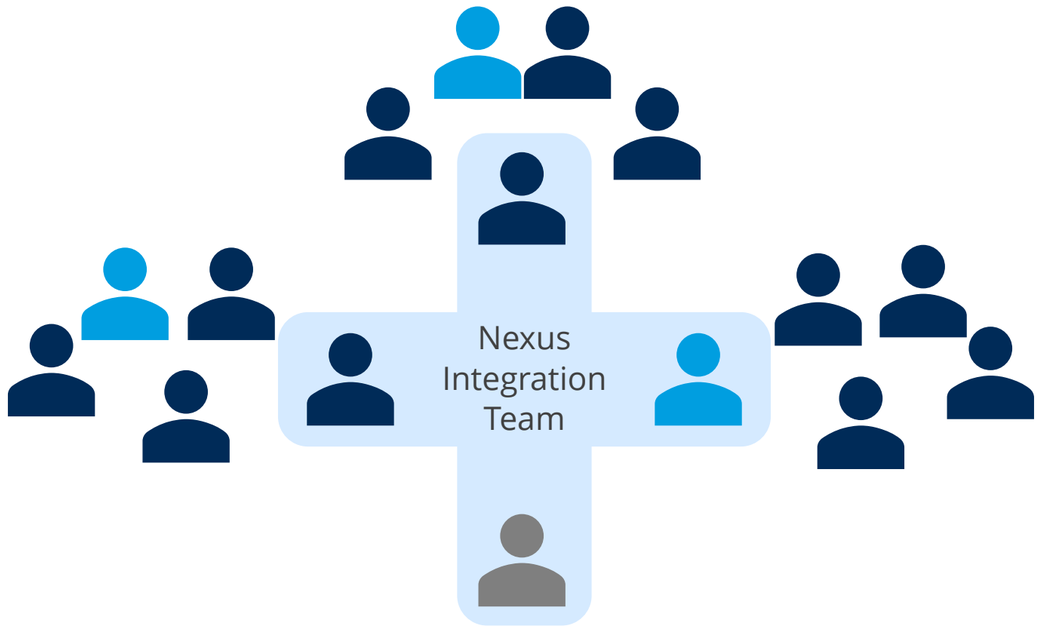
Identifies and resolves dependencies

Ensures alignment with project goals

Facilitates communication and collaboration among teams

Nexus Integration Team (NIT)

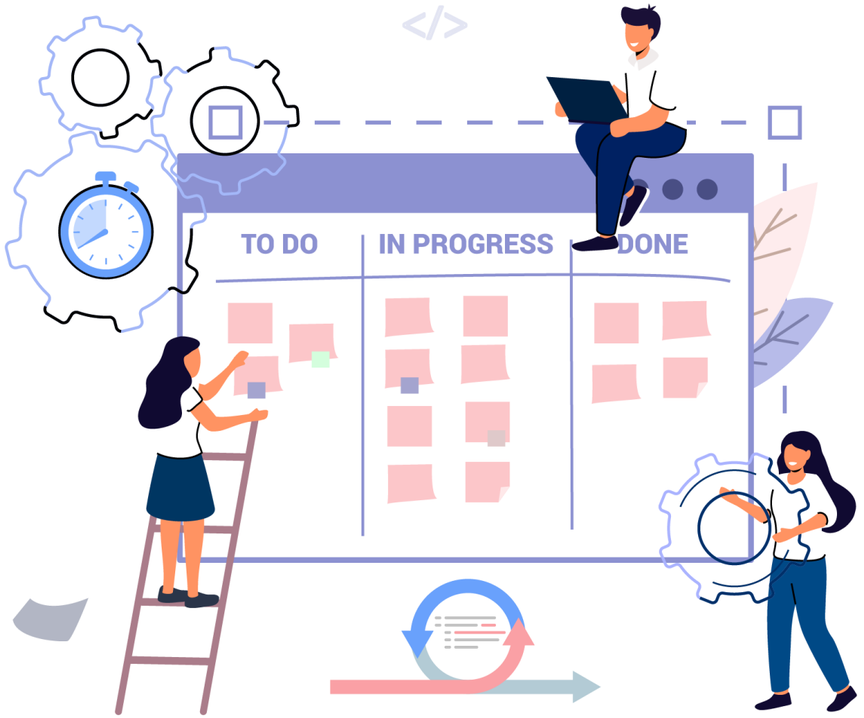
The Nexus consists of three to nine Scrum teams, all working with a shared product owner and a single product backlog.



The NIT consists of the product owner, a Scrum Master, and Nexus Integration Team members from the teams that are part of the Nexus.

Nexus Sprint

It is the iterative cycle within the Nexus framework, similar to a Scrum sprint but scaled for multiple teams.



It encompasses planning, daily coordination, review, and retrospective activities, all designed to manage the complexity of coordinating work across several teams.

Nexus: Events

Nexus sprint planning	This event involves representatives from each Scrum team and the NIT planning the work for the upcoming sprint, focusing on integration and dependencies.
Nexus daily scrum	This is a daily meeting where representatives from each team synchronize their work, address issues, and adjust plans as needed.

Nexus: Events

Nexus sprint review	During this event, all teams present their integrated increment to stakeholders. The NIT gathers feedback and ensures that the increment meets the Definition of Done (DoD).
Nexus sprint retrospective	In this meeting, all teams reflect on the sprint, identify improvements, and develop plans for enhancing future sprints.

Integrated Increment

It is the combined output of all Scrum teams working together during a sprint.

It represents the integrated work that meets the Definition of Done (DoD) and is potentially releasable.

Identification of Dependencies across Nexus

Identifying dependencies is crucial in a scaled environment because unmanaged dependencies can lead to delays and integration issues.

Early identification allows teams to plan and coordinate their work effectively, minimizing risks and ensuring smoother project progress.

Identification of Dependencies across Nexus

Identifying dependencies involves:

- Mapping out interdependencies among teams
- Prioritizing and addressing critical dependencies
- Ensuring that all teams are aware of and manage their dependencies effectively

Tools for Tracking Dependencies

Tools like Jira, Trello, and Asana offer features for tracking and managing dependencies.

These tools provide visibility into how dependencies impact the project and assist teams in coordinating their efforts effectively.

Suitability of Agile for Different Types of Projects

When to Use Agile?

Agile is suitable for projects with:

- High uncertainty and complexity
- Frequent changes in requirements
- A need for close customer collaboration

When Not to Use Agile?

Agile may not be ideal for projects with:

- Fixed requirements and scope
- Strict regulatory or compliance constraints
- Minimal customer interaction

Why Is a Small Team Beneficial for Agile?

Small teams offer several benefits, including:

- Enhanced communication and collaboration
- Greater flexibility and responsiveness
- Easier coordination and management
- Increased accountability and ownership

Activity: Applying Agile and Scaling Concepts to a Complex Project

Applying Agile and Scaling Concepts to a Complex Project

Scenario:
Your project involves multiple teams working on a complex product with high uncertainty and frequent changes in requirements. The teams are struggling to coordinate their efforts, manage dependencies, and deliver an integrated product increment. This has led to delays and a lack of cohesion in the final product.

Task:

1. Create a product owner team
2. Structure the product owner team
3. Conduct meetings regularly
4. Implement Nexus

Agile Administration in Tooling and Tool Integration

Tools for Agile Adoption

Several tools can help teams adopt agile practices and improve efficiency, such as:

Jira	A comprehensive tool for project management, bug tracking, and agile planning
Trello	A visual and flexible tool for task management and collaboration
Asana	A tool that focuses on work management and team collaboration
Monday.com	A highly customizable platform for project management and workflow

Jira

It helps teams plan, track, and manage work efficiently.

It supports various Agile methodologies, including Scrum and Kanban, making it a versatile choice for different project needs.

Jira: Key Features

Allows teams to create and manage a product backlog, prioritize user stories, and plan sprints

Helps teams to plan sprints by selecting user stories from the backlog and estimating effort using story points

Offers comprehensive reporting tools, including burndown charts, velocity charts, and sprint reports to monitor progress

Integrates with various other tools, such as Confluence, Bitbucket, and Slack, enhancing collaboration and information sharing

Trello

It is a user-friendly project management tool known for its visual approach.

It uses boards, lists, and cards to help teams organize tasks and collaborate effectively.

Trello: Key Features

Organizes work into boards containing lists of tasks represented as cards, creating a visual system that makes it easy to see the project's status at a glance

Helps teams to create cards for tasks, assign members, set due dates, and add checklists, labels, and attachments

Supports real-time collaboration, allowing team members to comment on cards, mention colleagues, and share updates

Integrates with numerous other tools and services, including Slack, Google Drive, and Jira, to streamline workflows and enhance productivity

Asana

It is a tool designed to help teams coordinate and manage their work.

It supports a range of project management methodologies and offers features that enhance planning, tracking, and collaboration.

Asana: Key Features

Allows teams to create tasks, set deadlines, assign team members, and track progress through various views, such as lists, boards, and timelines

Supports collaboration through comments, mentions, and project updates, ensuring everyone stays informed

Provides advanced reporting tools, including project status updates, workload management, and custom dashboards to monitor progress

Integrates with multiple tools, such as Slack, Google Workspace, and Microsoft Teams, enhancing workflow and productivity

Monday.com

It is a versatile project management tool that supports a wide range of workflows and methodologies.

Its customizable platform allows teams to plan, track, and manage their work in a way that suits their specific needs.

Monday.com: Key Features

Enables teams to create custom workflows, set up boards, and define columns and statuses to match their processes

Helps teams to create tasks, assign team members, set deadlines, and track progress through visual boards and timelines

Offers automation capabilities to streamline repetitive tasks, such as updating statuses, sending notifications, and creating new tasks based on triggers

Integrates with numerous tools and services, including Slack, Microsoft Teams, and Google Workspace, enhancing workflow efficiency

Creating Dashboards and Setting up Jira Project

Duration: 15 minutes

Problem statement:

You have been assigned a task to demonstrate the process of creating a Jira account and dashboards and setting up customized projects

ASSISTED PRACTICE

Assisted Practice: Guidelines

Duration: 15 minutes

Steps to be followed:

1. Sign up for a free-tier Jira account
2. Create a Jira dashboard
3. Create a custom project

ASSISTED PRACTICE

Creating User Stories and Sprint

Duration: 15 minutes

Problem statement:

You have been assigned a task to utilize Jira for effective task and timeline management by creating user stories and organizing them into sprints.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Duration: 15 minutes

Steps to be followed:

1. Create user stories
2. Import issues from the CSV file
3. Create a sprint

ASSISTED PRACTICE

Creating Epics, Tasks, and Subtasks

Duration: 15 minutes

Problem statement:

You have been assigned a task to create epics, tasks, and subtasks in Jira and link tasks to epics for effective project management.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Duration: 15 minutes

Steps to be followed:

1. Create an epic in Jira
2. Create a task
3. Create a subtask
4. Add an epic to tasks

ASSISTED PRACTICE

Prioritizing Backlog Items

Duration: 15 minutes

Problem statement:

You have been assigned a task to demonstrate the process of prioritizing backlog items in Jira by configuring the priority field.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Duration: 15 minutes

Steps to be followed:

1. Prioritize backlog issues

ASSISTED PRACTICE

Generating the Agile Reports

Duration: 20 minutes

Problem statement:

You have been assigned a task to demonstrate the process of generating various agile reports in Jira, including burnup, burndown, velocity, and cumulative flow diagrams.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Duration: 20 minutes

Steps to be followed:

1. Enable reports view in Jira
2. Generate burnup report
3. Generate burndown chart
4. Generate velocity report
5. Generate a cumulative flow diagram

ASSISTED PRACTICE

Key Takeaways

- A complex project is characterized by high uncertainty, interdependence of tasks, and dynamic environments.
- Scaling Agile involves extending Agile methodologies to larger and more complex projects.
- The product backlog lists everything the project needs, from features to bug fixes.
- The three main strategies for scaling a product backlog are combining backlog items into epics, creating different views of the product backlog, and structuring product owner teams.
- The Nexus framework is designed to coordinate multiple Scrum teams working on a single product.
- Jira helps teams plan, track, and manage their work efficiently.

Knowledge Check

Knowledge Check

1

Which of the following is responsible for coordinating and integrating the work of multiple Scrum teams in the Nexus framework?

A. The product owner

B. The Scrum Master

C. The Nexus Integration Team (NIT)

D. The development team

Knowledge Check 1

Which of the following is responsible for coordinating and integrating the work of multiple Scrum teams in the Nexus framework?

A. The product owner

B. The Scrum Master

C. The Nexus Integration Team (NIT)

D. The development team

The correct answer is **C**

In the Nexus framework, the Nexus Integration Team (NIT) is specifically responsible for coordinating and integrating the work of multiple Scrum teams.

Knowledge Check

2

What is the main purpose of the Nexus framework?

A. To increase the speed of individual Scrum teams

B. To coordinate multiple Scrum teams working on separate products

C. To manage and align multiple Scrum teams working on a single product

D. To replace Scrum in large organizations

Knowledge Check 2

What is the main purpose of the Nexus framework?

A. To increase the speed of individual Scrum teams

B. To coordinate multiple Scrum teams working on separate products

C. To manage and align multiple Scrum teams working on a single product

D. To replace Scrum in large organizations

The correct answer is **C**

The Nexus framework is specifically designed to coordinate multiple Scrum teams working together on a single product. It adds roles, events, and artifacts to address the challenges.

Knowledge Check 3

How many Scrum teams typically make up a Nexus, and who are the key members of the Nexus Integration Team (NIT)?

A. Two to four Scrum teams; product owner and development team members

B. Five to seven Scrum teams; product owner, Scrum Master, and stakeholders

C. Three to nine Scrum teams; product owner, Scrum Master, and NIT members

D. Ten to fifteen Scrum teams; Scrum Master and development team members

Knowledge Check 3

How many Scrum teams typically make up a Nexus, and who are the key members of the Nexus Integration Team (NIT)?

A. Two to four Scrum teams; product owner and development team members

B. Five to seven Scrum teams; product owner, Scrum Master, and stakeholders

C. Three to nine Scrum teams; product owner, Scrum Master, and NIT members

D. Ten to fifteen Scrum teams; Scrum Master and development team members

The correct answer is **C**

A Nexus is made up of three to nine Scrum teams and consists of the product owner, a Scrum Master, and NIT members.

Knowledge Check

4

Which of the following best describes the main features of Jira?

A. A tool for designing user interfaces

B. A comprehensive tool for project management, bug tracking, and agile planning

C. A software for creating charts and reports

D. A platform for social media marketing

Knowledge Check 4

Which of the following best describes the main features of Jira?

A. A tool for designing user interfaces

B. A comprehensive tool for project management, bug tracking, and agile planning

C. A software for creating charts and reports

D. A platform for social media marketing

The correct answer is **B**

Jira is a comprehensive tool that supports project management, bug tracking, and agile planning. It is specifically designed to help teams efficiently plan, track, and manage their work.