# DAY 11:Agile + Jira with Agile

DAY-11.pptx

## Learning Objectives

By the end of this lesson, you will:

Understand the key concepts of Agile methodology.
Learn the features of JIRA, a popular tool for managing Agile projects.
Explore how Agile principles and practices are implemented using JIRA.
Gain hands-on knowledge of JIRA for creating and tracking Agile workflows.

## Why ,where?

Agile

For faster delivery, better collaboration, and adaptability.

In software, product management, IT, and beyond.

## JIRA + Agile

For centralized tracking, collaboration, and Agile-specific features.

For sprint management, tracking, and reporting in Agile workflows.

---

Analogy
Agile is like building a house room by room rather than constructing the entire house all at once.

Iterative Development: Instead of waiting until the whole house is complete, you finish one room (a small functional piece) and show it to the homeowner for feedback. This ensures that the room meets expectations before moving to the next.
Collaboration: The homeowner (customer) works closely with the architect and builders (developers and testers) to make changes and improvements as needed.
Flexibility: If the homeowner wants to add a new feature, like a balcony, it can be incorporated into future iterations without disrupting the entire project

# Introduction

Agile is a project management methodology that focuses on delivering small, functional pieces of a project through iterative development. Instead of building everything at once, Agile promotes continuous feedback and improvements. Teams work in short cycles, called *sprints*, delivering features incrementally. Agile is highly collaborative, customer-focused, and adaptive to change

JIRA, developed by Atlassian, is a widely used project management tool that supports Agile frameworks such as Scrum and Kanban. It enables teams to plan, track, and manage their work while maintaining transparency and efficiency.

---

# Advantages of Agile

Flexibility: Agile allows for changes throughout the project based on feedback.
Customer Satisfaction: Frequent delivery of functional product features ensures continuous feedback from customers.
Improved Quality: Regular testing and iterations help catch and fix issues early.
Transparency: Teams and stakeholders collaborate closely, giving clear insight into progress.
Risk Management: Frequent releases minimize the chances of project failure.

---

# Disadvantages of Agile

Scope Creep: Without proper control, Agile's flexibility can lead to uncontrolled changes in the project scope.
Time Commitment: Agile requires constant involvement from both the development team and stakeholders.
Documentation Challenges: Since Agile prioritizes working software over comprehensive documentation, this can lead to issues with maintaining detailed records.
Less Predictability: Since Agile allows changes in scope, time and cost estimation can be difficult.

---

# Scrum Terminology

Scrum: A subset of Agile, Scrum is a framework that structures the project into small iterations (sprints) with clear roles, events, and artifacts.

Product Owner: The person responsible for defining the product's vision, prioritizing the backlog, and ensuring the team delivers value.
Scrum Master: A facilitator who ensures the team follows Scrum principles, removes obstacles, and improves team efficiency.
Development Team: A cross-functional group of professionals responsible for delivering product increments.

---

## User Story

A user story is a short, simple description of a feature or functionality from the perspective of the end user. It outlines what the user wants to achieve and why. Example format:

```
As a [type of user], I want [goal] so that [reason].
```

Example:

User Story: As a registered user, I want to reset my password so that I can regain access to my account.

---

## Epic

An epic is a large user story that can be broken down into smaller, manageable stories. It represents a broad feature or requirement that usually takes several sprints to complete.

Example:

An epic for an e-commerce platform could be "Implement the shopping cart feature."

---

## Product Backlog

The product backlog is a prioritized list of features, enhancements, and bug fixes that need to be addressed during the development process. It is managed by the Product Owner and is constantly updated to reflect changes in requirements or feedback.

---

## Sprint

A sprint is a time-boxed period (typically 1-4 weeks) during which a set of work must be completed and made ready for review. Each sprint delivers a functional increment of the product.

---

## Sprint Planning

Sprint planning is an event in which the Scrum team discusses what can be delivered in the upcoming sprint and how that work will be achieved. The product owner presents the prioritized product backlog items, and the team selects which items they will work on during the sprint.

---

## Sprint Backlog

The sprint backlog is a subset of the product backlog. It includes all the user stories, tasks, and improvements that the team commits to complete during the sprint. This list is created during the sprint planning meeting.

---

## Scrum Meeting (Daily Standup)

The daily standup is a short, time-boxed meeting (usually 15 minutes) where each team member answers three questions:

1. What did you do yesterday?
2. What will you do today?
3. Are there any blockers or impediments?

The purpose of this meeting is to ensure everyone is aligned and to quickly address any issues.

## Scrum Ceremonies/Meetings

1. Sprint Planning
   Purpose: Decide what work will be done during the sprint and how it will be achieved.
   Participants: Scrum Master, Product Owner, and Development Team.
   Outcome: Sprint backlog with prioritized user stories and tasks.
2. Daily Scrum (Stand-up Meeting)
   Purpose: Synchronize team progress and address blockers.

Format: Each member answers three questions:
    What did I do yesterday?
    What will I do today?
    Are there any impediments in my way?

3. Sprint Review
    Purpose: Present completed work to stakeholders and gather feedback.
    Outcome: Stakeholder feedback for future iterations.
4. Sprint Retrospective
    Purpose: Reflect on what went well, what didn't, and identify actionable improvements for the next sprint.
5. Bug Triage Meeting
    Purpose: Review and prioritize bugs for resolution.
    Outcome: Updated backlog with prioritized bugs.
6. Release Retrospect Meeting
    Purpose: Evaluate the release process and gather insights to improve future releases.
7. Project Closure Meeting
    Purpose: Assess overall project performance, document lessons learned, and celebrate successes.

---

## 12. Sprint Retrospective Meeting

The sprint retrospective is held at the end of each sprint to reflect on what went well, what didn’t, and how processes can be improved for future sprints. It is a continuous improvement process where teams discuss lessons learned.

---

## 13. Story Point

Story points are units of measurement used in Agile to estimate the relative complexity or effort required to implement a user story. They do not correlate directly with hours but instead reflect the difficulty of the task.

Example:

> A simple user story might be assigned 2 points, while a more complex one might be assigned 8 points.

---

## 14. Burndown Chart

A burndown chart is a visual tool used in Agile to track the progress of work over time. It shows the amount of work remaining versus the amount of time left in the sprint. Teams use it to ensure they are on track to meet their sprint goals.

## 15.Burnup Chart:

Displays completed work over time, highlighting progress toward sprint goals.

## 16. Capacity:

The total available effort (often in hours) of the team for a sprint after accounting for other commitments.

## 17. Velocity:

The average number of story points a team completes in a sprint, used for forecasting future work.

## What is Agile Methodology?

Agile is a software development methodology focused on iterative development, where requirements and solutions evolve through collaboration between cross-functional teams. It emphasizes flexibility, customer feedback, and rapid delivery of small, functional increments of a product.

## Core Components of Agile Methodology

1. Iterative Development
   Work is divided into iterations (sprints), delivering incremental value in each cycle.
2. Continuous Feedback
   Regular collaboration with customers and stakeholders ensures the product meets user needs.
3. Collaboration
   Cross-functional teams work together, including developers, testers, product owners, and business analysts.
4. Flexibility
   Plans can be adjusted based on feedback and changing requirements.

## Introduction to Jira

---

Jira is a widely used tool for bug tracking, issue tracking, and project management, commonly applied in Agile methodologies. It is designed to help teams manage various aspects of software development, testing, and other project management activities.

Key features relevant to manual testing in Jira include:

Issues: These are the fundamental units in Jira and can represent anything from a bug to a task, a user story, or other types of project-related work.
Projects: A collection of related issues that are managed and tracked together.
Workflows: Defines the lifecycle of an issue. It represents the statuses (e.g., "To Do," "In Progress," "Done") and transitions (e.g., moving from "In Progress" to "Done") that an issue goes through.
Dashboards: Customizable views that summarize project activities. These dashboards can include various gadgets to provide insights into project status and performance.
Reports: Jira offers different reports to track project progress and analyze team performance over time, such as burndown charts, velocity reports, and issue distribution.

# Using JIRA for Agile

# A. Setting Up a JIRA Project

1. Log into JIRA and create a new project.
2. Choose a template:
   Scrum: For sprint-based development.
   Kanban: For continuous workflows.

# B. Managing Backlogs

1. Navigate to the Backlog view.
2. Add issues to the backlog:
   Issue Types: Stories, Tasks, Bugs, Epics.
   Add details like title, description, priority, and assignee.

# C. Planning Sprints

1. Select backlog items and move them to a sprint.
2. Start the sprint, defining the duration (e.g., 2 weeks).

# D. Tracking Progress

1. Use the Scrum Board or Kanban Board:
   Columns represent stages (e.g., To Do, In Progress, Done).

Move issues across columns as they progress.

2. Monitor sprint progress using reports:
   Burndown Chart: Tracks work completed vs. work remaining.
   Velocity Chart: Shows the amount of work completed in past sprints.

## E. Closing a Sprint

1. Complete or move incomplete items back to the backlog.
2. Conduct a sprint review and retrospective.

---

# Examples

## 1. Creating a User Story in JIRA

Scenario: As a user, I want to reset my password so that I can regain account access.

Steps in JIRA:

1. Navigate to the backlog and click "Create Issue."
2. Select the issue type: Story.
3. Add the title: *Reset Password Feature*.
4. Add details:
   Description: Steps for resetting passwords.
   Priority: High.
   Assignee: Team member responsible.

---

## Tracking Sprint Progress

Example: During a 2-week sprint, monitor the burndown chart daily to ensure work is on track. If tasks remain stagnant in the "In Progress" column, address blockers during the standup.

# Summary

Agile methodology emphasizes iterative development, collaboration, and adaptability. JIRA is a powerful tool for managing Agile projects, offering features like backlog management, sprint tracking, and reporting.

By combining Agile principles with JIRA, teams can streamline workflows and deliver high-quality software efficiently.