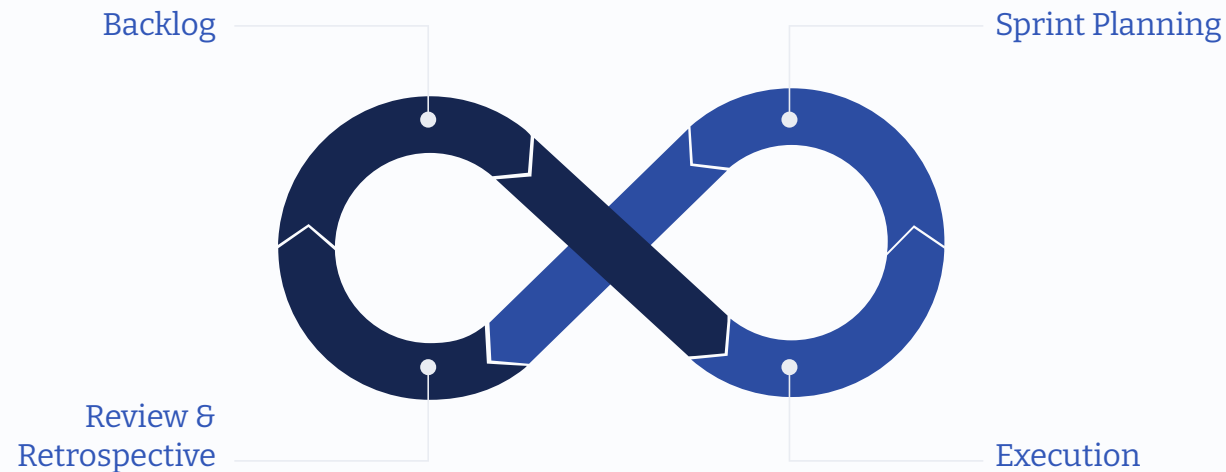




# The Agile Playbook

Welcome to your essential guide for implementing Agile methodology. This presentation provides a concise reference for product managers, scrum masters, and development teams looking to enhance their Agile practices. We'll cover the core concepts, benefits, and practical applications to help you deliver value faster.

# Agile Flow & Benefits



## Why Agile Works



### Faster Value

Deliver working increments quickly



### Adaptability

Respond to change effectively



### Collaboration

Enhanced transparency



### Improvement

Continuous learning cycles



### Satisfaction

Happier customers & teams

# Agile in a Nutshell

Agile is an **iterative & incremental** approach to delivering value through short cycles called **Sprints** (typically 1-4 weeks).

## Key Components:

### Roles

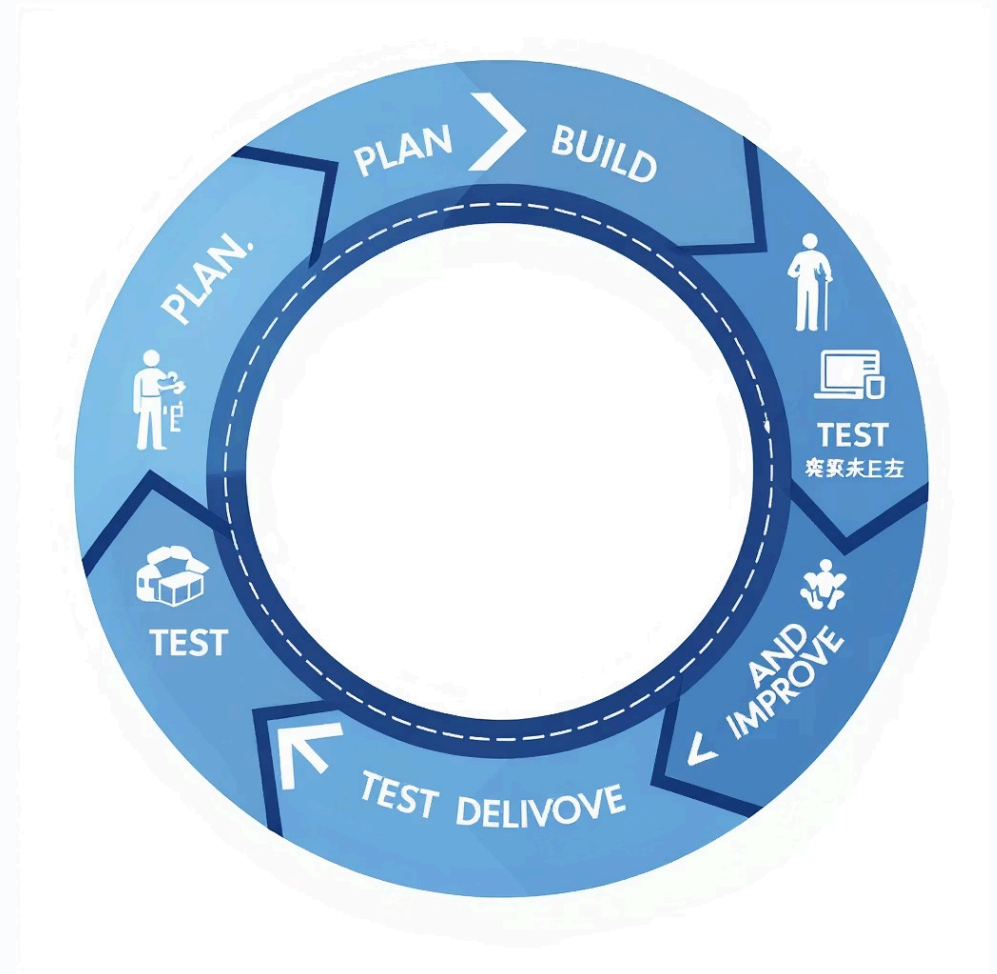
Product Owner (WHAT), Scrum Master (PROCESS),  
Development Team (HOW)

### Events

Planning, Daily Scrum, Review, Retrospective

### Artifacts

Product Backlog, Sprint Backlog, Increment



Each sprint follows the same pattern, creating a rhythm for the team

# Agile Sprint Cycle

## Sprint Planning

Select user stories and define sprint goal

Team commits to deliverables

## Sprint Retrospective

Reflect on process & team dynamics

Identify improvements for next sprint



## Daily Scrum

15-minute team sync

What was done, what's next, blockers?

## Development & Testing

Team works on stories & tasks

Continuous integration & testing

## Sprint Review

Demo working software to stakeholders

Gather feedback for improvement



## Agile Roles & Ownership



### Product Owner

Owns the [Product Backlog](#)

- Defines WHAT to build & WHY
- Prioritizes stories by value
- Makes business decisions



### Scrum Master

Owns the [Agile Process](#)

- Ensures framework is followed
- Removes impediments/blockers
- Facilitates events & growth

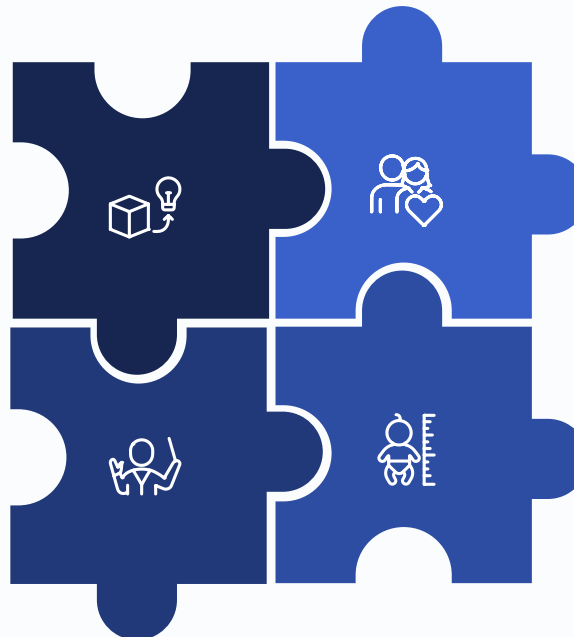


### Development Team

Owns the [Sprint Backlog & Delivery](#)

- Decides HOW to build solutions
- Self-organizes to deliver
- Collaborates cross-functionally

Product Owner



Shared Goals

Scrum Master

Development Team



# Work Breakdown & Estimation

Agile teams break down large initiatives into smaller, manageable pieces to facilitate planning, development, and delivery. This hierarchical structure ensures clarity and focus at every level.



## Epic

A big, overarching goal or project.

**Example:** Build Online Learning Portal



## Feature

A specific capability or function within an Epic.

**Example:** Student Registration & Login



## Story

A user-centric description of a desired functionality or need.

**Example:** As a student, I want to sign up with email



## Task

Detailed, step-by-step work required to implement a Story.

**Example:** Design form, Write database code

This breakdown provides a clear path from high-level objectives to executable work, improving transparency and facilitating accurate estimation.



# Work Breakdown & Estimation

## From Epics to Tasks



Example: "Build Online Learning Portal" → "Student Registration" → "Sign up with email" → "Design form"

## Story Points Scale

Points measure [relative effort](#), not hours

| Point | Size         | ~Time     |
|-------|--------------|-----------|
| 1     | Very Small   | 2-3 hours |
| 2     | Small        | 4-6 hours |
| 3     | Medium       | ~1 day    |
| 5     | Large        | 2-3 days  |
| 8     | Complex      | ~1 week   |
| 13    | Very Complex | >1 week   |