Modern Software
Engineering with DevOps

nucamp

Software Development Methodologies:

Scrum and Kanban

Scrum Methodology

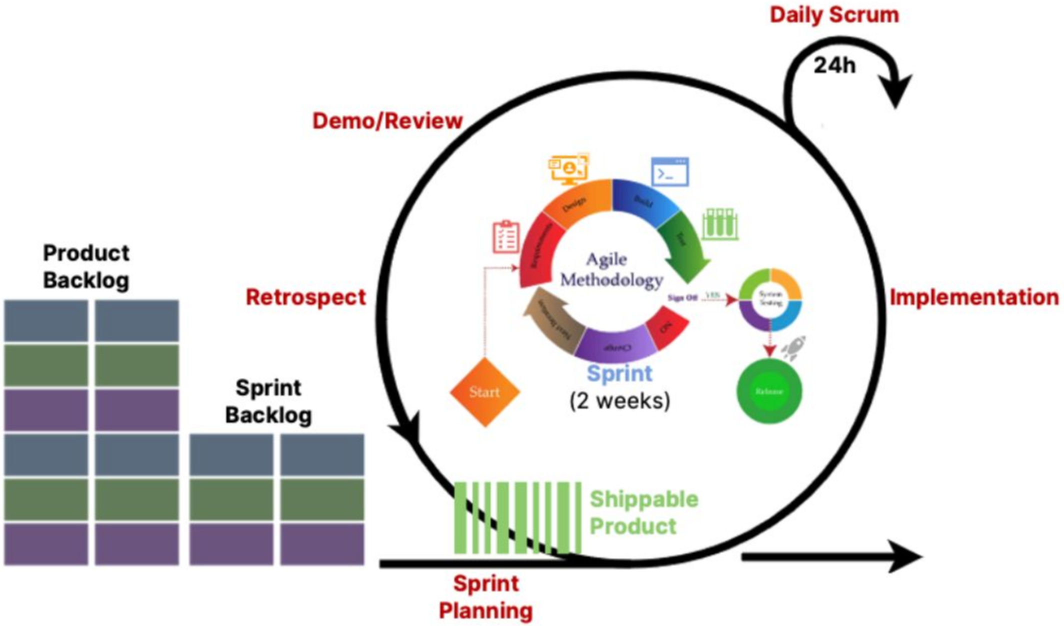
Agile framework with **Scrum rituals** (team meetings):

Sprint planning

Daily Scrums

Demo/Review

Retrospectives



Scrum pros/cons

Pros

Released product every 2 weeks (or at end of each Sprint)

Handles incomplete requirements

Improved product management

Higher product quality

Scrum pros/cons

Cons

Requires high client engagement

Not suited for inexperienced teams

Not suited for larger teams

When to use Scrum

Best for quick MVP (Minimum Viable Product) development

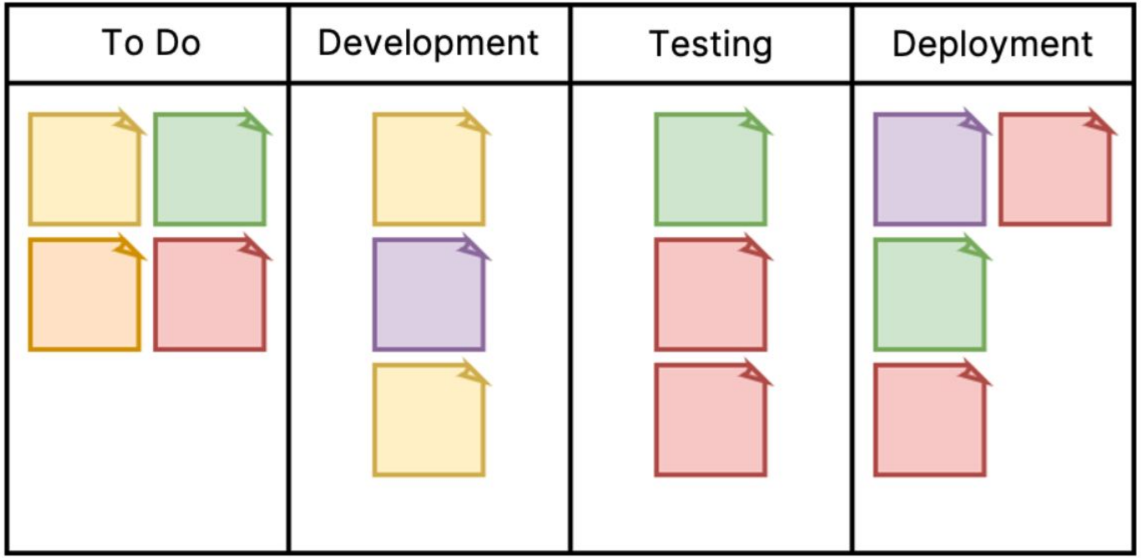
Best for regular improvements

Kanban methodology

Another Agile framework, considered more flexible than Scrum

Also includes daily meetings, demos for clients, retrospectives

Kanban board



To Do

Development

Testing

Deployment

Uses a “Kanban board” to quickly move tasks between stages (with software such as Trello, or physical sticky notes)

Kanban board

Each sticky note represents a task, or a **user story**

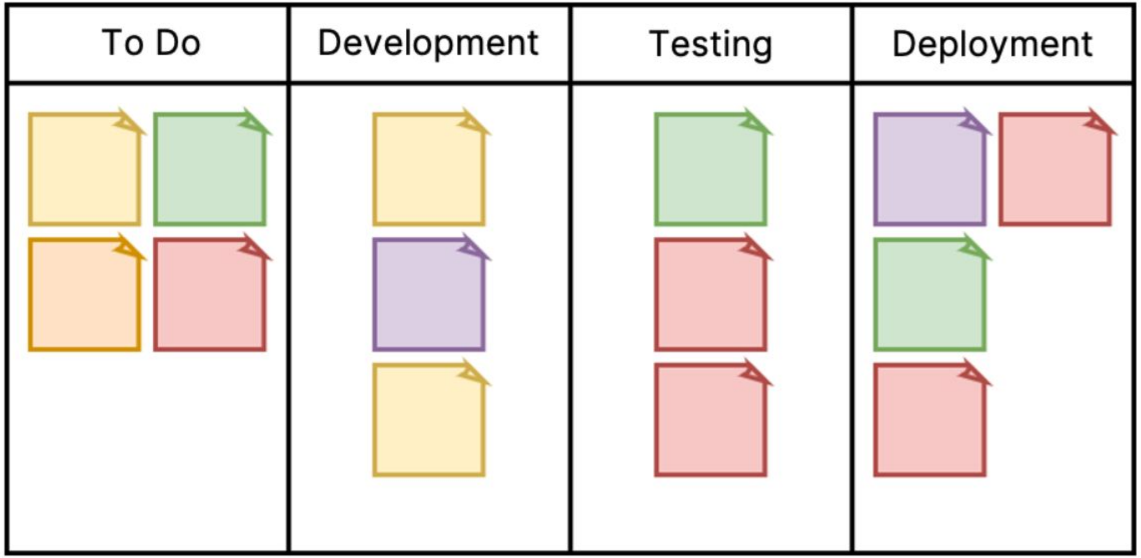
User story: description of a feature from a user's point of view

Includes: type of user, desired feature, motivation

Example

"As a customer, I want to receive an email when I make a purchase,
so that I can have my receipt and order details."

Kanban board



To Do

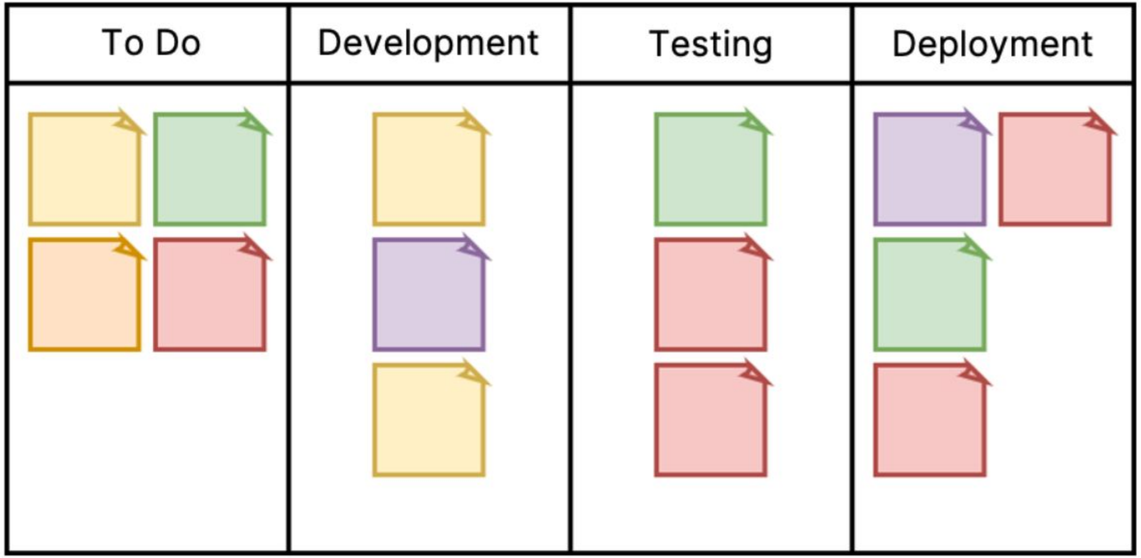
Development

Testing

Deployment

Board divided into columns for different stages of work

Kanban board



To Do

Development

Testing

Deployment

Board is limited in size, ensures quick iterations and releases

Kanban pros/cons

Pros

Simple task management approach

Very flexible

Used by small and large teams

Doesn't need re-planning

Kanban pros/cons

Cons

Easy to lose direction

Requires self-management skills

Lower developer productivity

Hard to control scope

When to use the Kanban methodology

For project in a support phase

When there is a constant flow of change requests

Incomplete requirements with frequent changes