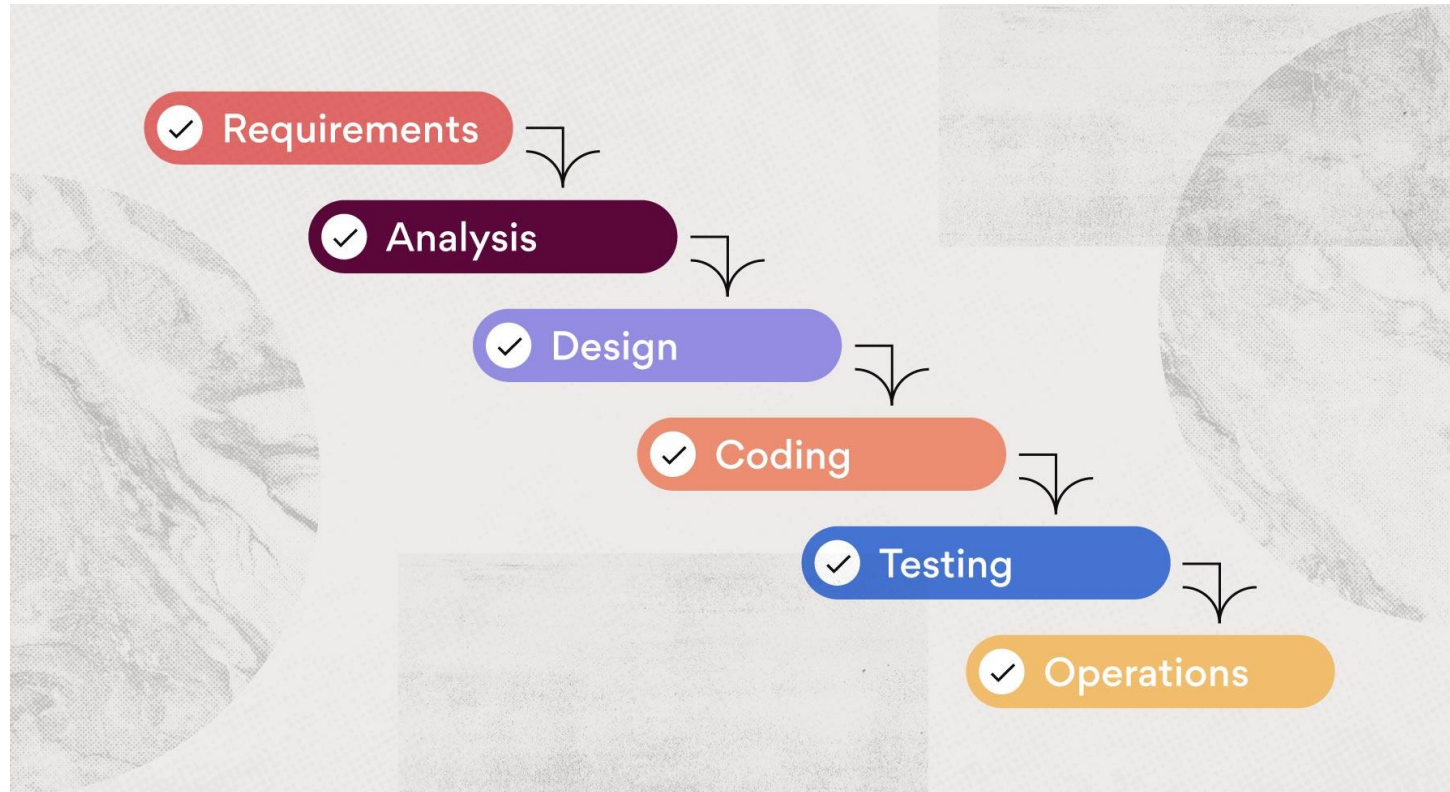


A white board with a grid of colorful circular holes, some containing colored discs. The holes are arranged in a grid pattern, and the colors of the discs include red, yellow, blue, green, and orange. The board is slightly angled, and the lighting creates soft shadows.

Introduction to Agile Project Management

ARCTECH INFO PRIVATE LTD

Traditional Waterfall Model



Traditional PM versus Agile Methods

Traditional PM Approach

- Concentrates on thorough, upfront planning of the entire project.
- Requires a high degree of predictability to be effective.

Agile Project Management (Agile PM)

- Relies on incremental, iterative development cycles to complete less-predictable projects.
- Is ideal for exploratory projects in which requirements need to be discovered and new technology tested.
- Focuses on active collaboration between the project team and customer representatives.

Traditional Project Management versus Agile Project Management

TRADITIONAL

Design up front

Fixed Scope

Deliverables

Freeze Design as early as possible

Low uncertainty

Avoid change

Low customer interaction

Conventional project teams

AGILE

Continuous Design

Flexible

Features / Requirements

Freeze Design as late as possible

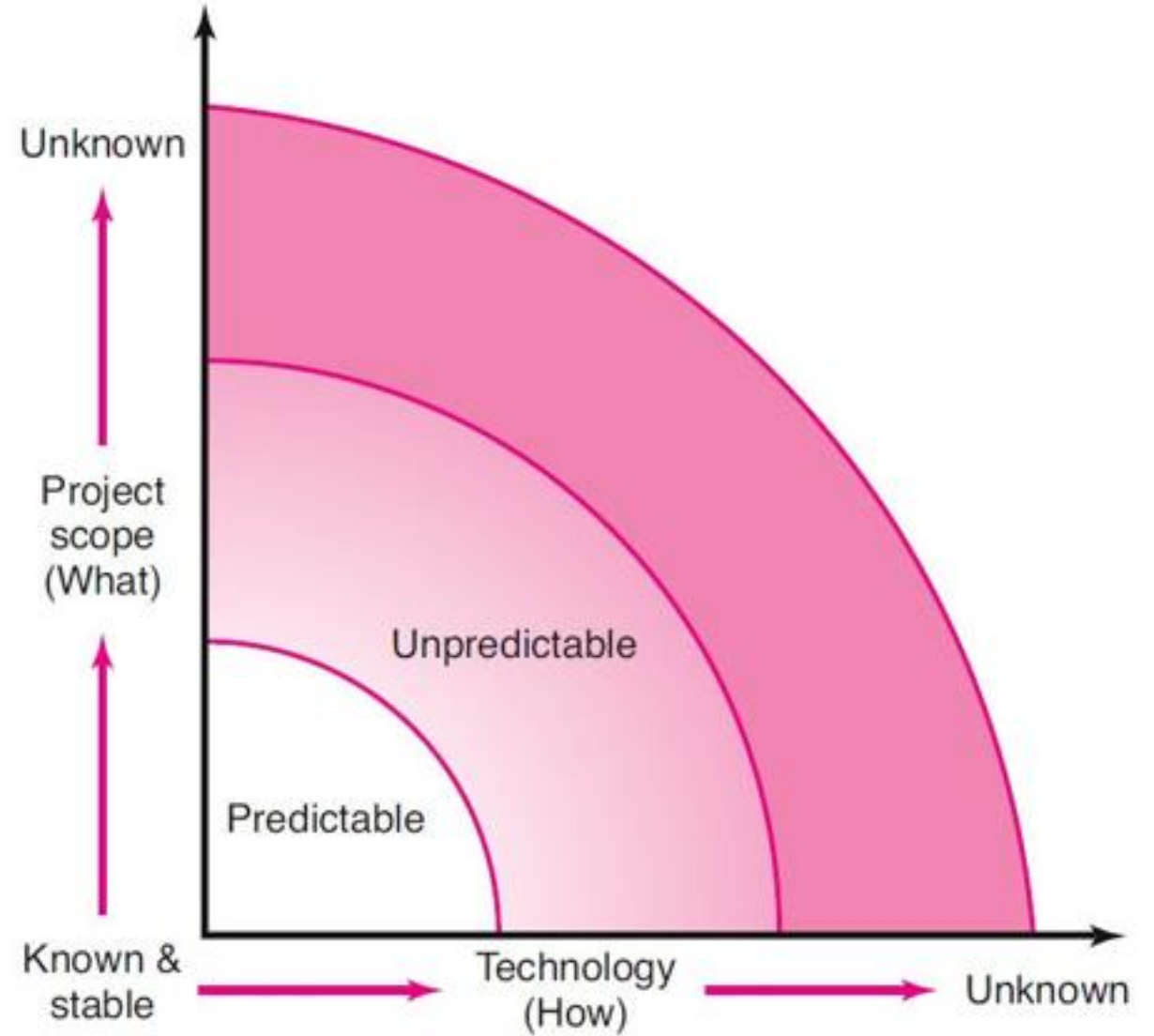
High uncertainty

Embrace change

High customer interaction

Self-organized project teams

Project Uncertainty



Agile Project Management

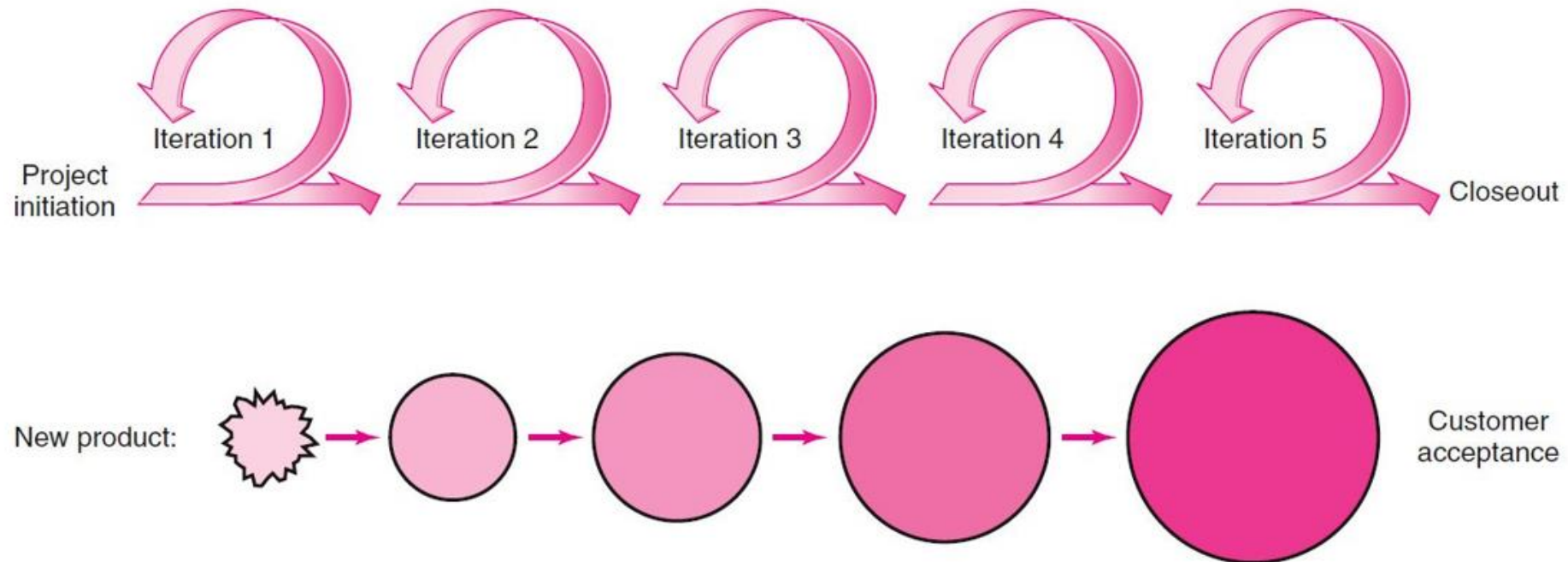
Is related to the rolling wave planning and scheduling project methodology.

Uses iterations (“time boxes”) to develop a workable product that satisfies the customer and other key stakeholders.

Stakeholders and customers review progress and re-evaluate priorities to ensure alignment with customer needs and company goals.

Adjustments are made and a different iterative cycle begins that subsumes the work of the previous iterations and adds new capabilities to the evolving product.

Iterative, Incremental Product Development



Advantages of Agile PM

Useful in developing critical breakthrough technology or defining essential features

Continuous integration, verification, and validation of the evolving product.

Frequent demonstration of progress to increase the likelihood that the end product will satisfy customer needs.

Early detection of defects and problems.

Agile PM Principles

Focus on customer value

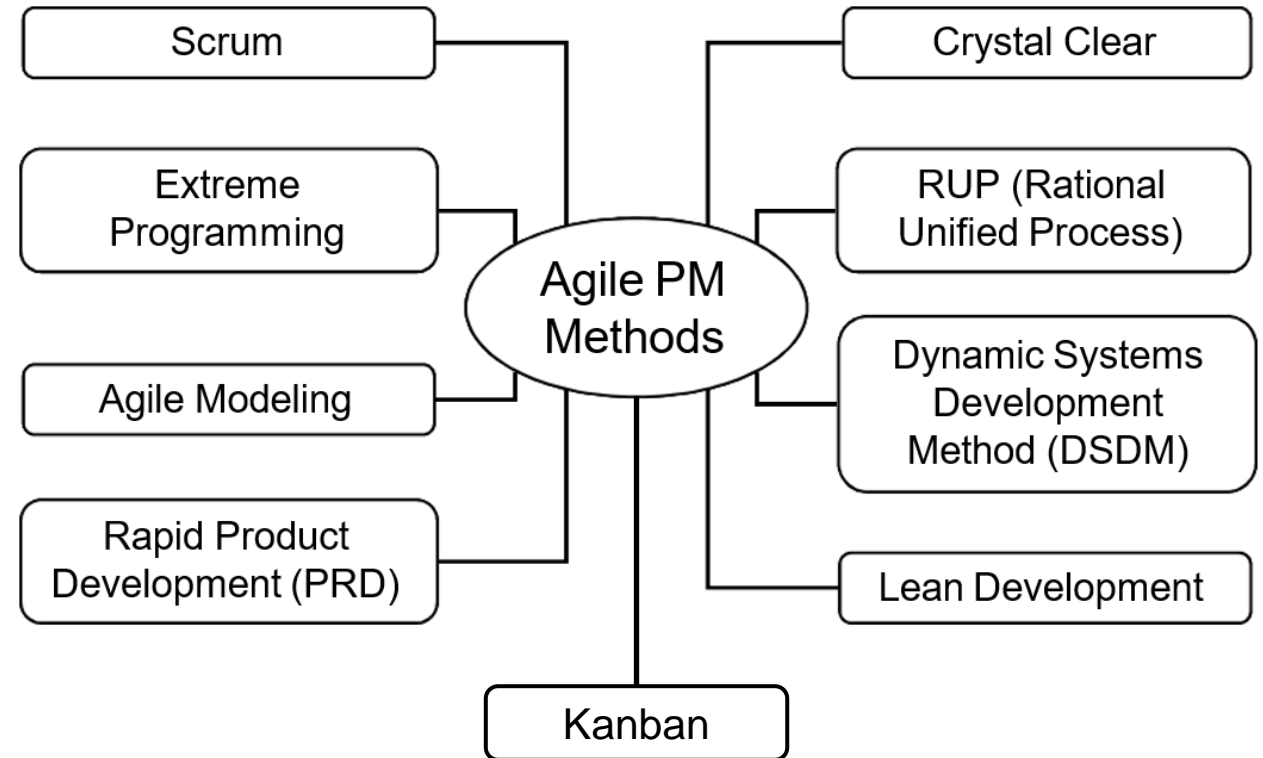
Iterative and incremental delivery

Experimentation and adaptation

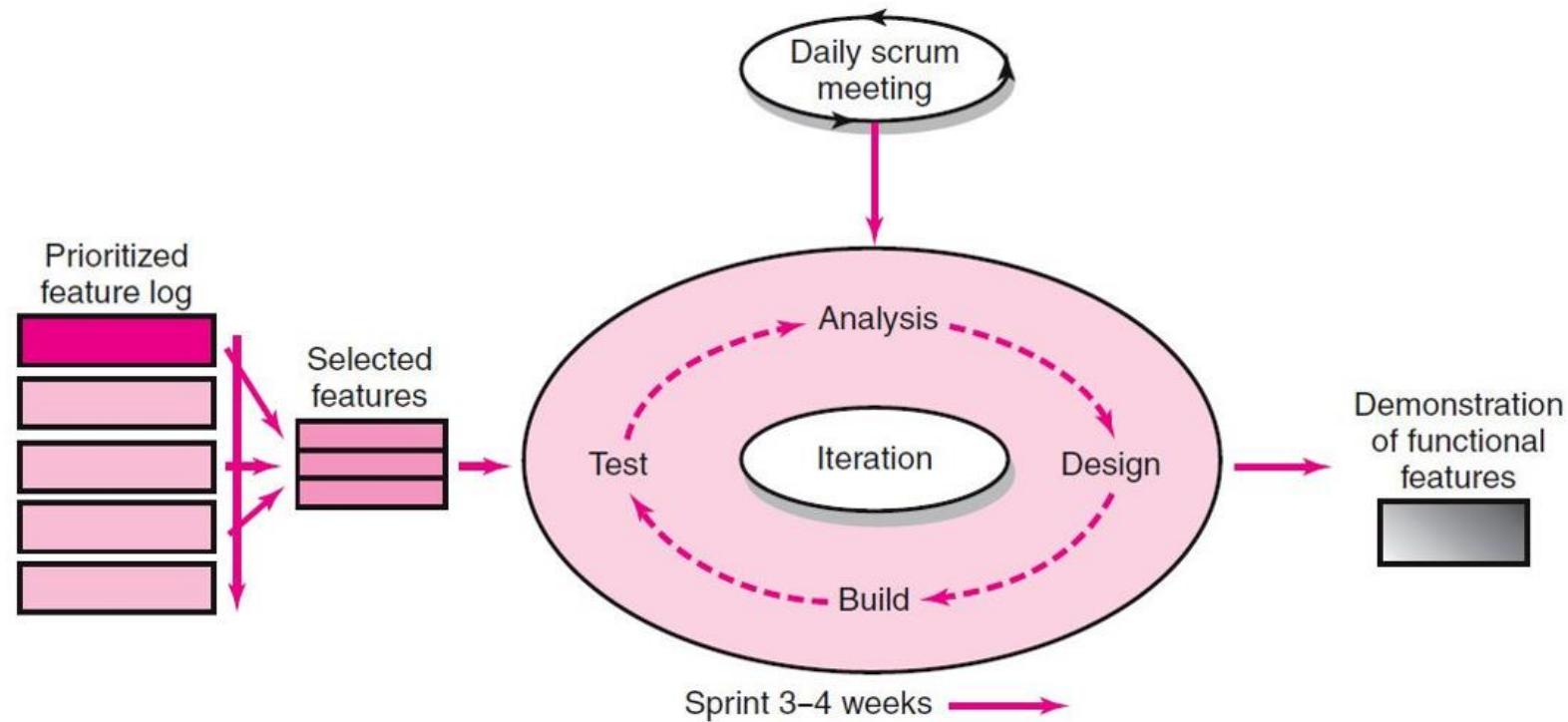
Self-organization

Continuous improvement

Popular Agile PM Methods



Scrum Development Process



Extreme Programming Key Practices

01

Pair
Programming

02

Planning
Game

03

Continuous
Process.

04

Coding
Standards.

05

Sustainable
Pace.

06

Test Driven
Development
(TDD)

Agile Modeling

KEY VALUES

Communication

Simplicity

Feedback

Courage

Humility

CORE PRINCIPLES

Model With a Purpose

Adopt Simplicity

Embrace Change

Enable the Next Effort (Secondary goal)

Incremental Change

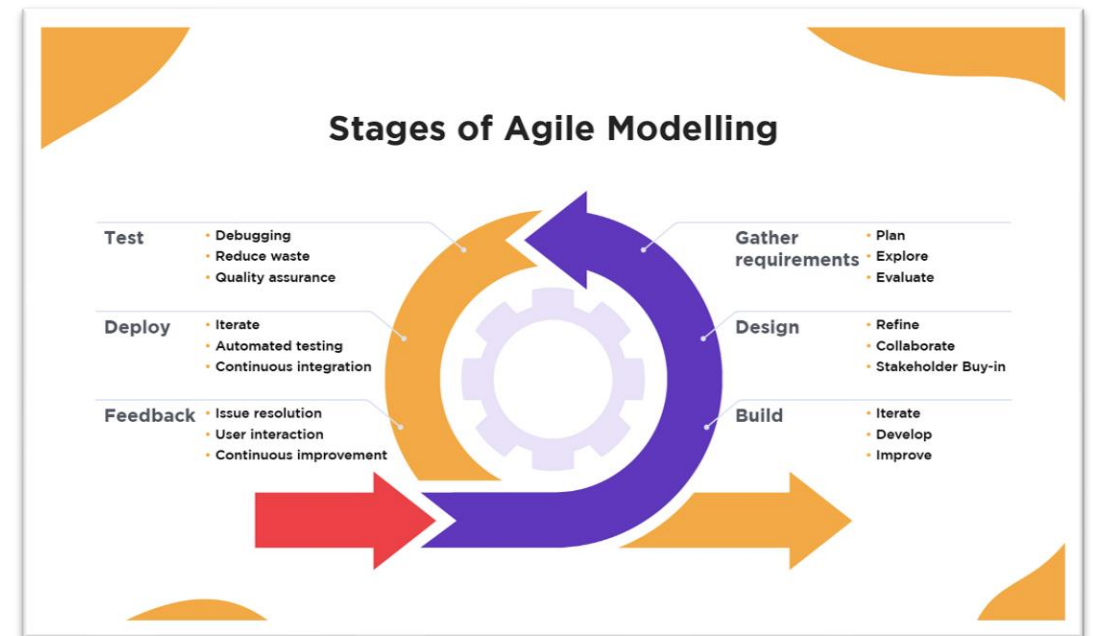
Produce Quality Work

Provide Rapid Feedback.

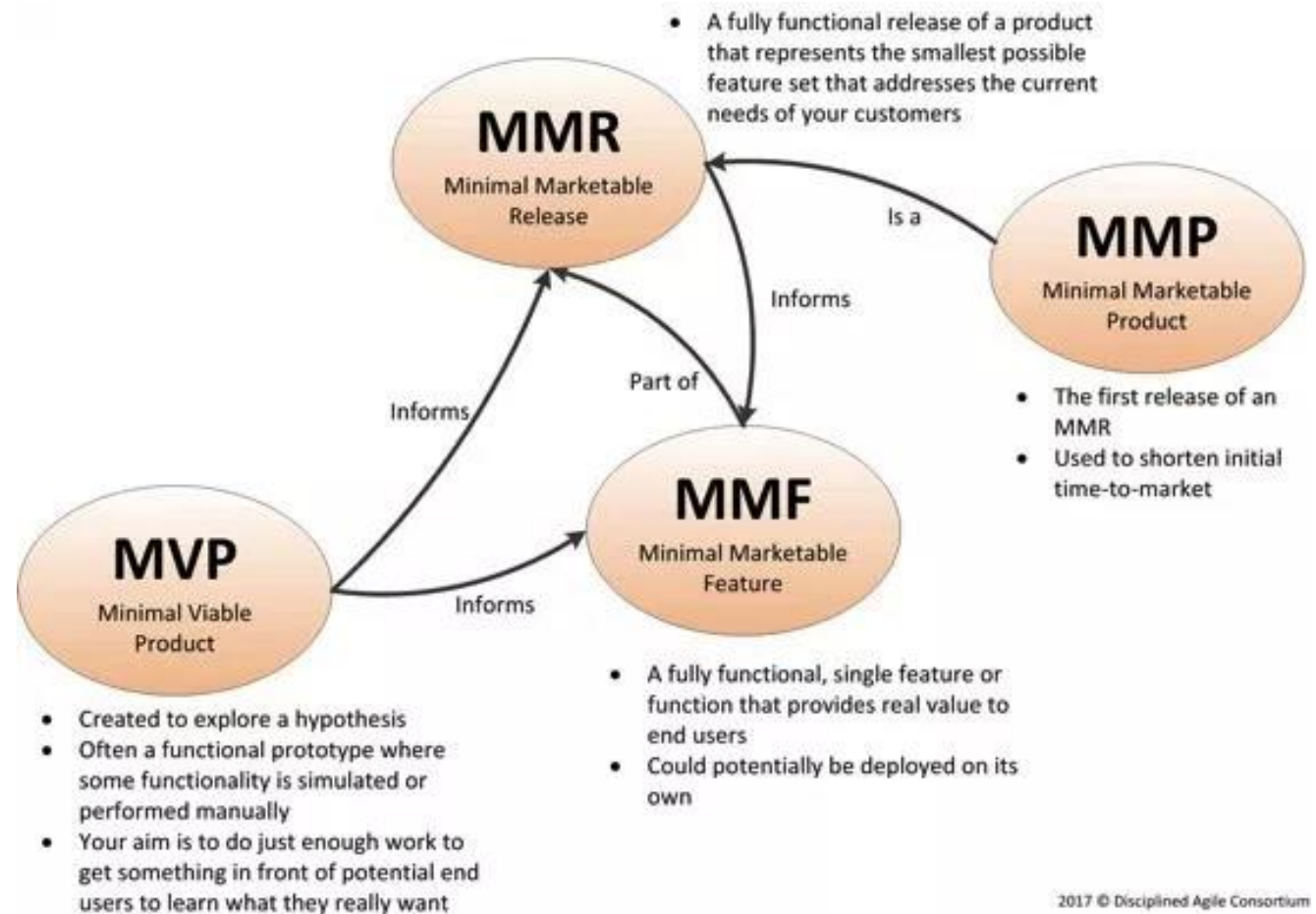
Make Working Software (Primary Goal)

Travel Light

PHASES OF THE AGILE MODEL



Rapid Product Development (RPD)



Properties of Crystal Clear methodology

Frequent Delivery

Reflective Improvement

Osmotic communication

Personal safety

Focus

Easy access to expert users

Technical environment

Conclusion

- Development teams do best in an environment that is safe, from a personal and emotional point of view, and is free from personal attacks.
- A key Crystal Clear concept is to have constructive but not vindictive criticism.
- Ron Jeffries closely characterized Crystal Clear as:
 - Bring a few developers together in peace, love, and harmony; shipping code every other month, and good software will emerge

Rational Unified Process (RUP)

Inception

- Communication and planning

Elaboration

- Planning and modeling

Construction

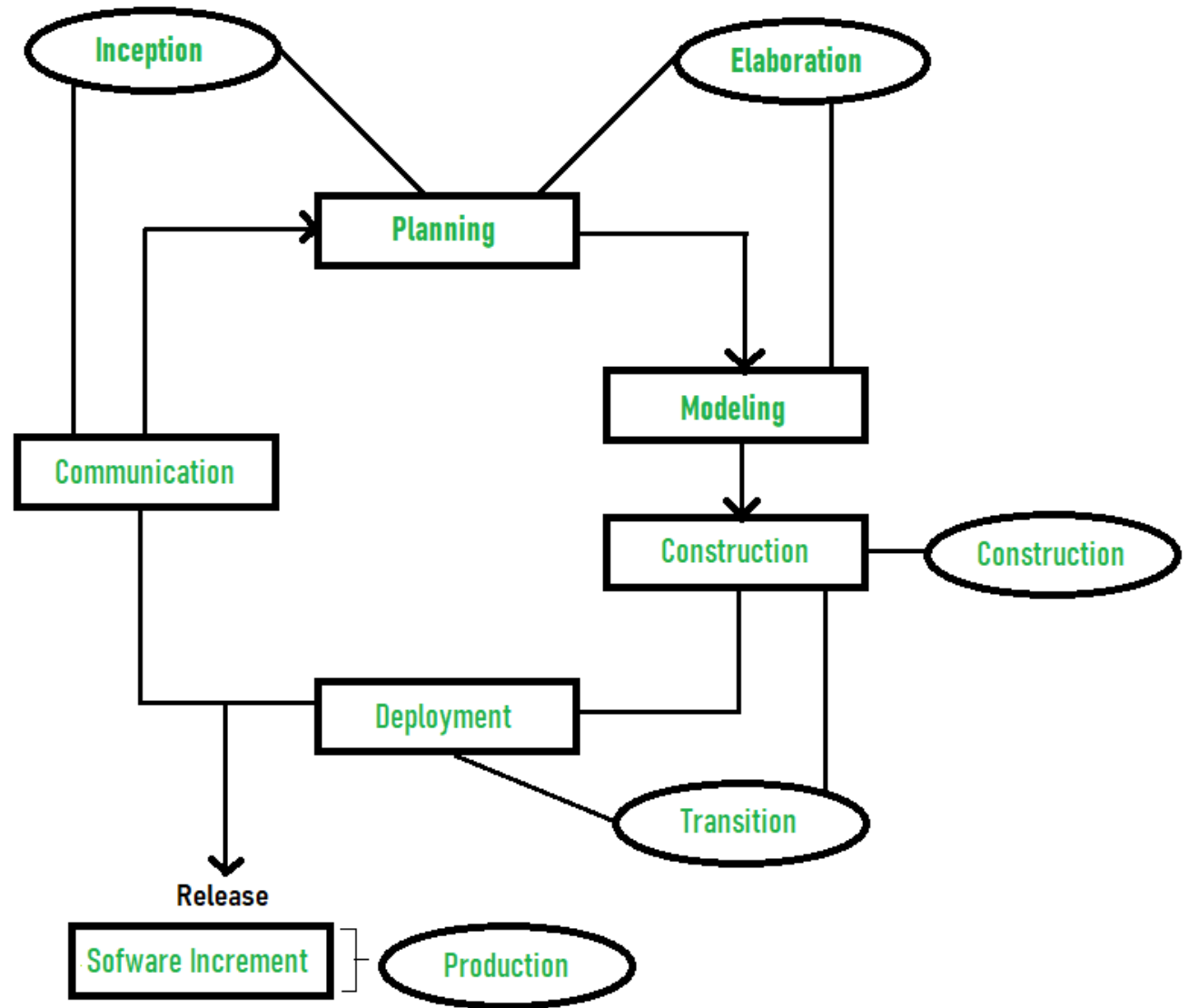
- Project is developed and completed

Transition

- Final project is released to public

Production

- Project is handed to operational team



Principles of Dynamic Systems Development Method

DSDM is an Agile method that focuses on the full project lifecycle

DSDM (formally known as Dynamic System Development Method) was created in 1994, after project managers using RAD (Rapid Application Development) sought more governance and discipline to this new iterative way of working.

Focus on the business need

Deliver on time

Collaborate

Never compromise quality

Build incrementally from firm foundations

Develop iteratively

Communicate continuously and clearly

Demonstrate control

Lean Development Methodology



Improve efficiency by eliminating waste.



Traditional, waterfall project management, which dictates a set plan laid out by a project manager



Lean agile strives to reduce all tasks and activities that don't provide real value.



This helps ensure everyone involved in a project or product development can work at optimal efficiency.

Value

- Identify Value

Value stream

- Map the value stream

Flow

- Create flow

Pull

- Establish a pull system

Perfection

- Seek perfection

Kanban

Kanban is a subsect of the Agile methodology developed in 1940s

The Kanban Agile philosophy is all about

- adaptive planning,
- early delivery, and
- continuous improvement

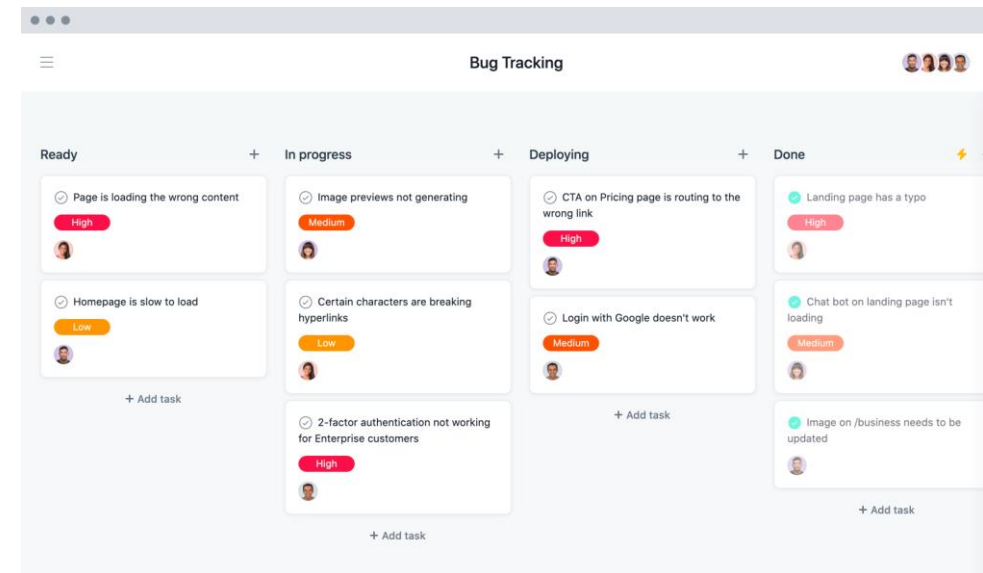
When someone speaks of Kanban in project management, they're most commonly referring to Kanban boards.

A Kanban board represents stages of work with columns that hold the individual tasks for each stage—but more on that in a little bit.

The Kanban framework is very flexible and can help your team become more dynamic and agile over time.

Kanban boards are one of the most popular forms of visual project management.

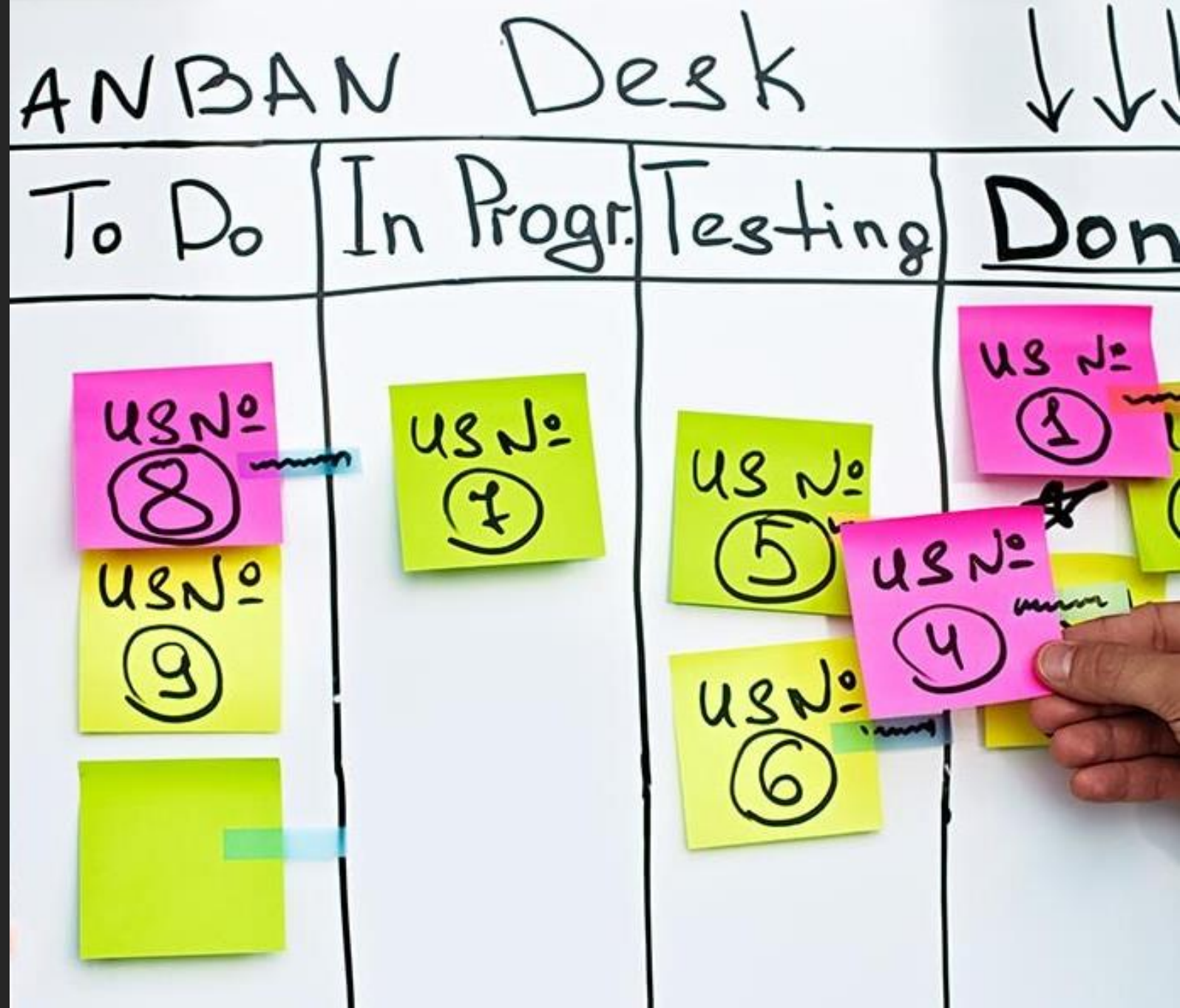
They're most effective for providing easy, at-a-glance insight into a project.



Benefits of Kanban boards

When you use a Kanban board for visual project management, you provide your team at-a-glance information, including but not limited to:

- Tasks or deliverables
- Task assignee
- Due dates
- Relevant tags, like priority or task type
- Task details
- Context
- Relevant files



What is Scrum

Scrum is one of the most popular Agile frameworks.

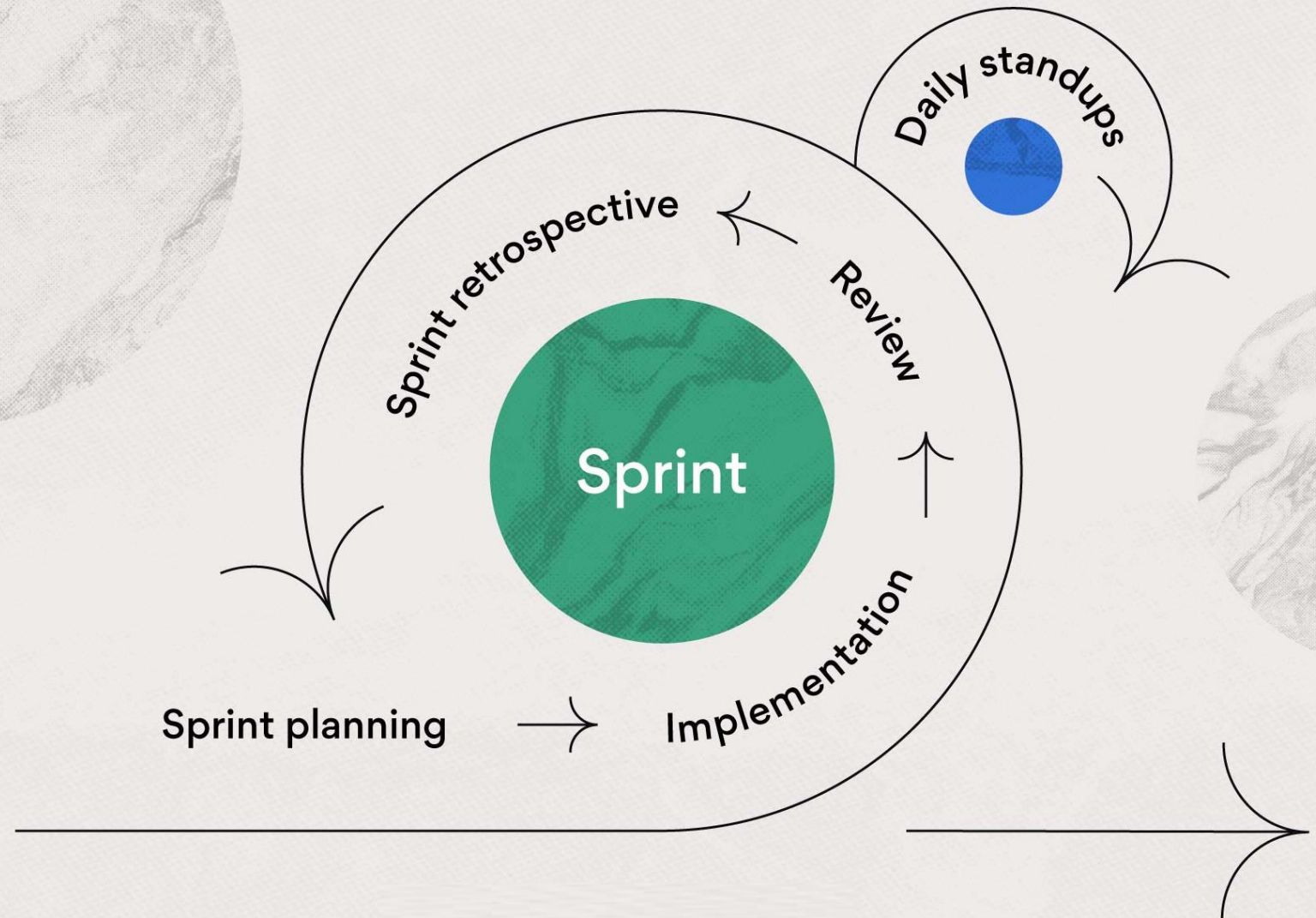
Kanban, is generally used as a tool to visualize work

Scrum is a full framework and you can “run teams” on Scrum.

The framework was pioneered by Taiichi Ohno and provides a blueprint of values, guidelines, and roles to help your team focus on continuous improvement and iteration.

It's much less flexible than Kanban but a great way for Agile teams to collaborate and get high-impact work done.

Agile and Scrum



Agile PM in Action: Scrum Methodology

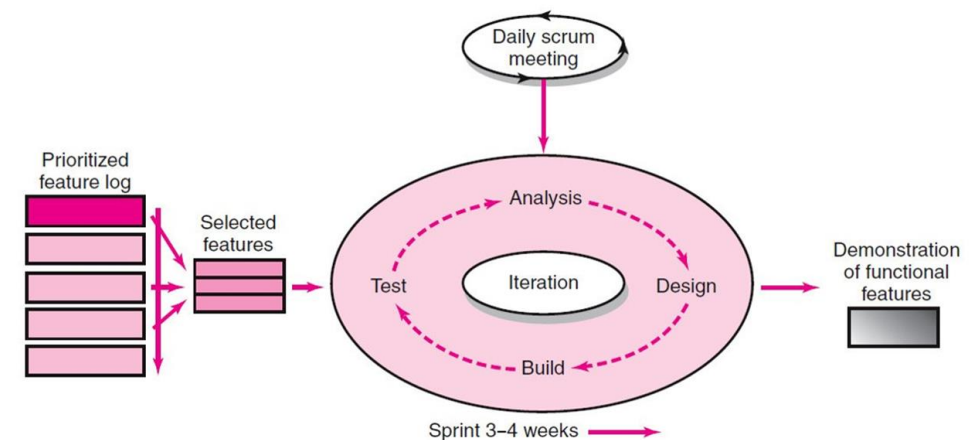
Is a holistic approach for use by a cross-functional team collaborating to develop a new product.

Defines product features as deliverables and prioritizes them by their perceived highest value to the customer.

Re-evaluates priorities after each iteration (sprint) to produce fully functional features.

Has four phases: analysis, design, build, test

Scrum is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems



More on Scrum

While Scrum, like Agile, was originally created for software development teams, industries like product, engineering, and others now run Scrum to execute their work faster and more effectively.

To run a Scrum, teams typically assign a Scrum master, who is in charge of running the three distinct Scrum phases and keeping everyone on track.

The Scrum master can be your team lead, project manager, product owner, or the person most interested in running Scrum.

The Scrum master is responsible for implementing the three traditional Scrum phases:

Phase 1

Sprint planning.

- A Scrum sprint is usually two weeks long, though teams can run faster or shorter sprints.
- During the sprint planning phase, the Scrum master and team take a look at the team's product backlog and select work to accomplish during the sprint.

Phase 2

Daily Scrum standups.

- Over the course of the Scrum (also known as the Scrum "cycle time"), teams traditionally meet for 15 minutes every day to check in on progress and make sure the amount of assigned work is appropriate.

Phase 3

Sprint retrospective.

- When the Scrum is over, the Scrum master hosts a sprint retrospective meeting to evaluate

Key Roles and Responsibilities in the Scrum Process

Product Owner

- Acts on behalf of customers to represent their interests.

Development Team

- Is a team of five-nine people with cross-functional skill sets is responsible for delivering the product.

Scrum Master (aka Project Manager)

- Facilitates scrum process and resolves impediments at the team and organization level by acting as a buffer between the team and outside interference



Product Backlog in SCRUM

A product backlog is a prioritized list of work for the development team

It is derived from the project roadmap and its requirements.

The most important items are shown at the top of the product backlog.

The development team doesn't work through the backlog at the product owner's pace

The product owner isn't pushing work to the development team.

Instead, the development team pulls work from the product backlog as there is capacity for it, by iteration (scrum).

Product Backlog Items



A Product Backlog Item (PBI) is a single element of work that exists in the product backlog.



PBIs can include user stories, epics, specifications, bugs, or change requirements.



The Product Owner of an Agile team compiles and prioritizes the product backlog



The most urgent or important PBIs are put at the top.



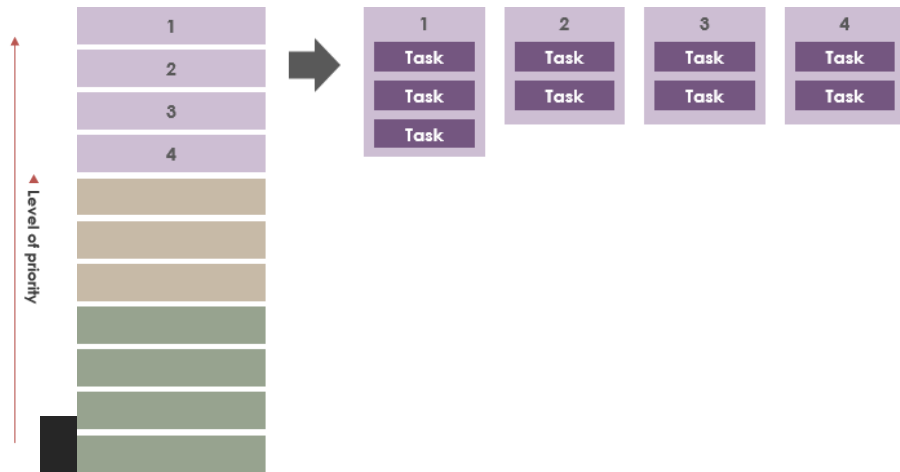
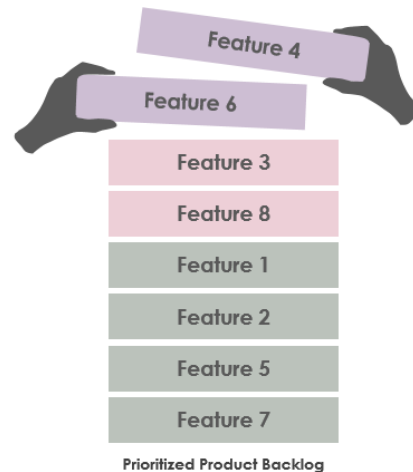
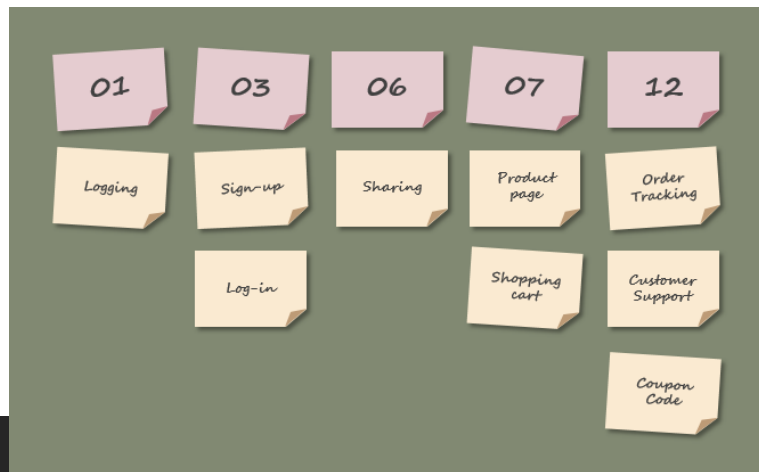
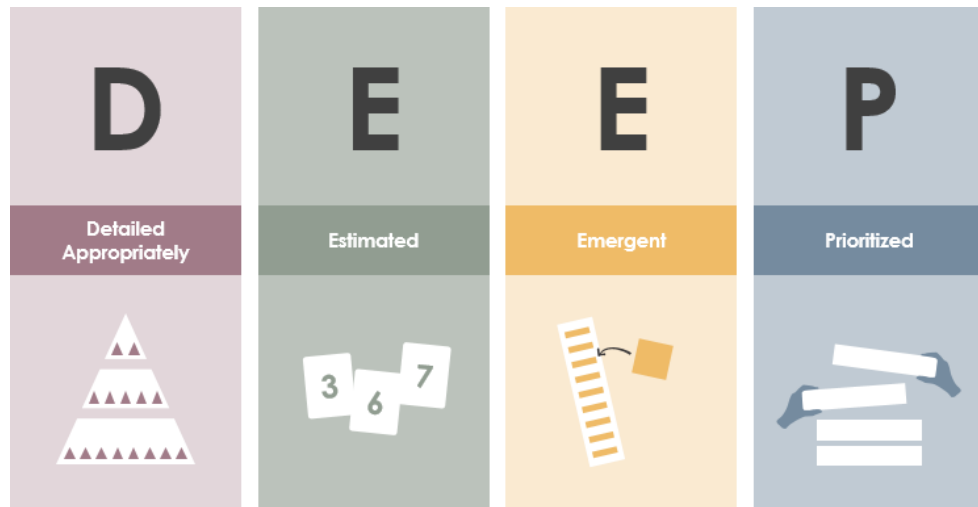
PBIs comprise tasks that need to be completed during a Scrum sprint

a PBI must be a small enough increment of work to be completed during a single sprint.



As PBIs move up to a higher priority in the product backlog, they are broken down into user stories.

Characteristics of Product Backlog

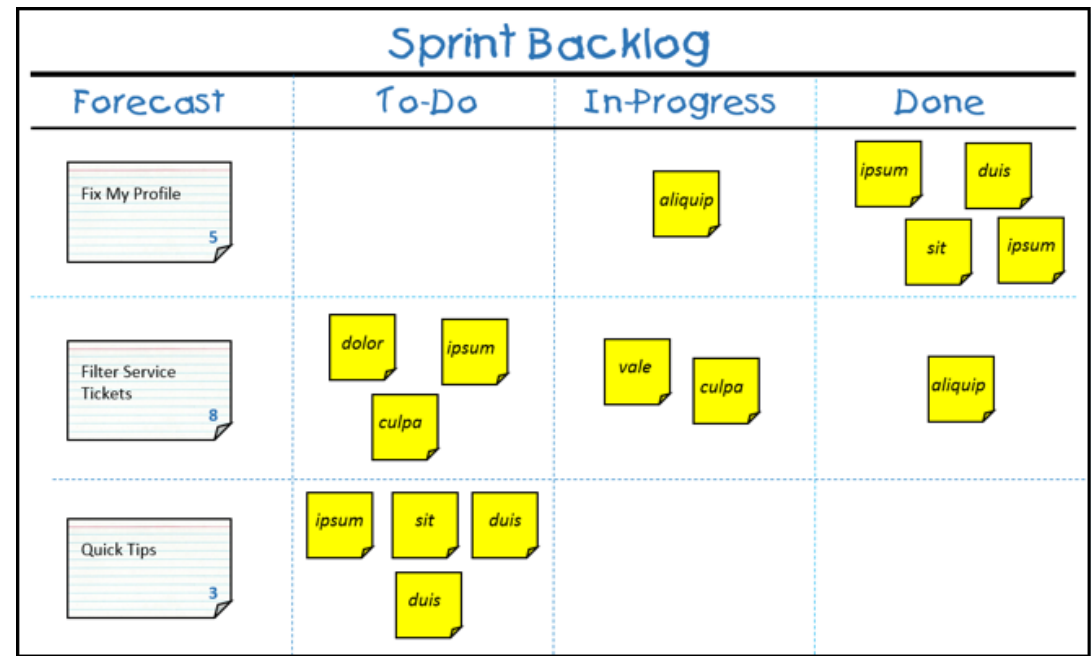


Sprint Backlog

Is a subset of the product backlog

Sprint backlog has a limited scope

It contains PBIs which are part of the upcoming sprint.



Tasks, Stories, epics, and initiatives

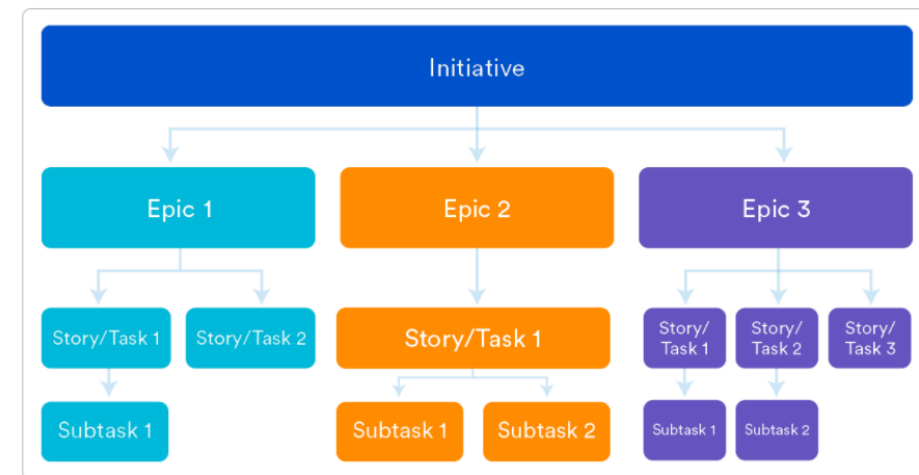
Initiatives are collections of epics that drive toward a common goal.

Epics are large bodies of work that can be broken down into a number of smaller tasks (called stories).

Stories, also called “user stories,” are short requirements or requests written from the perspective of an end user.

Tasks, are work that needs to be done to build the feature.

SubTasks are used to split the issue into smaller chunks or to allow various aspects of an issue to be assigned to different people



Scrum Meetings or Ceremonies

Sprint planning meeting

- Before your team begins a Scrum sprint, you need to know where you're going. ...

Daily standup meeting

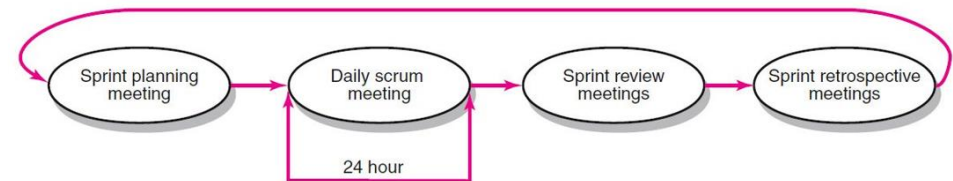
- most frequently held Agile Scrum meetings, daily standup meetings.
- They're short, to the point, and, as the name suggests, held each day

Sprint review meeting.

- Sprint review meetings are held at the end of each sprint.
- This meeting is an opportunity for you and your team to demonstrate what you've accomplished

Sprint retrospective meeting.

- Just like review meetings, a sprint retrospective meeting is held at the end of each sprint
- primarily for the benefit of your Scrum team—there's usually no need to get outside players involved.



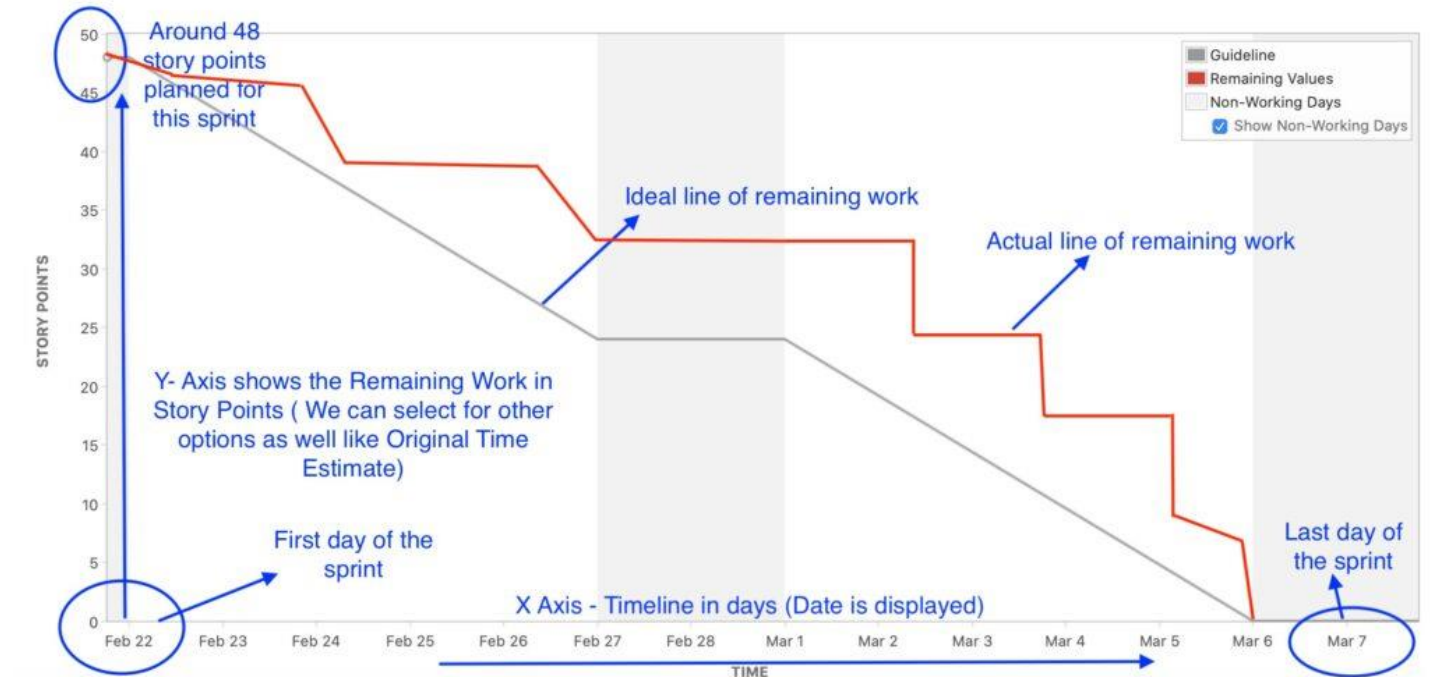
Burndown chart

A burndown chart is a graphical representation of work left to do versus time.

The outstanding work (or backlog) is often on the vertical axis, with time along the horizontal.

Burn down charts are a run-rate chart of outstanding work.

It is useful for predicting when all of the work will be completed.



Benefits of Scrum



Teams that run Scrum have clearly established rules, rituals, and responsibilities.



Additionally, your daily Scrum meetings, combined with sprint planning and sprint review (or “retrospective” meetings), help teams continuously check in and improve on current processes.



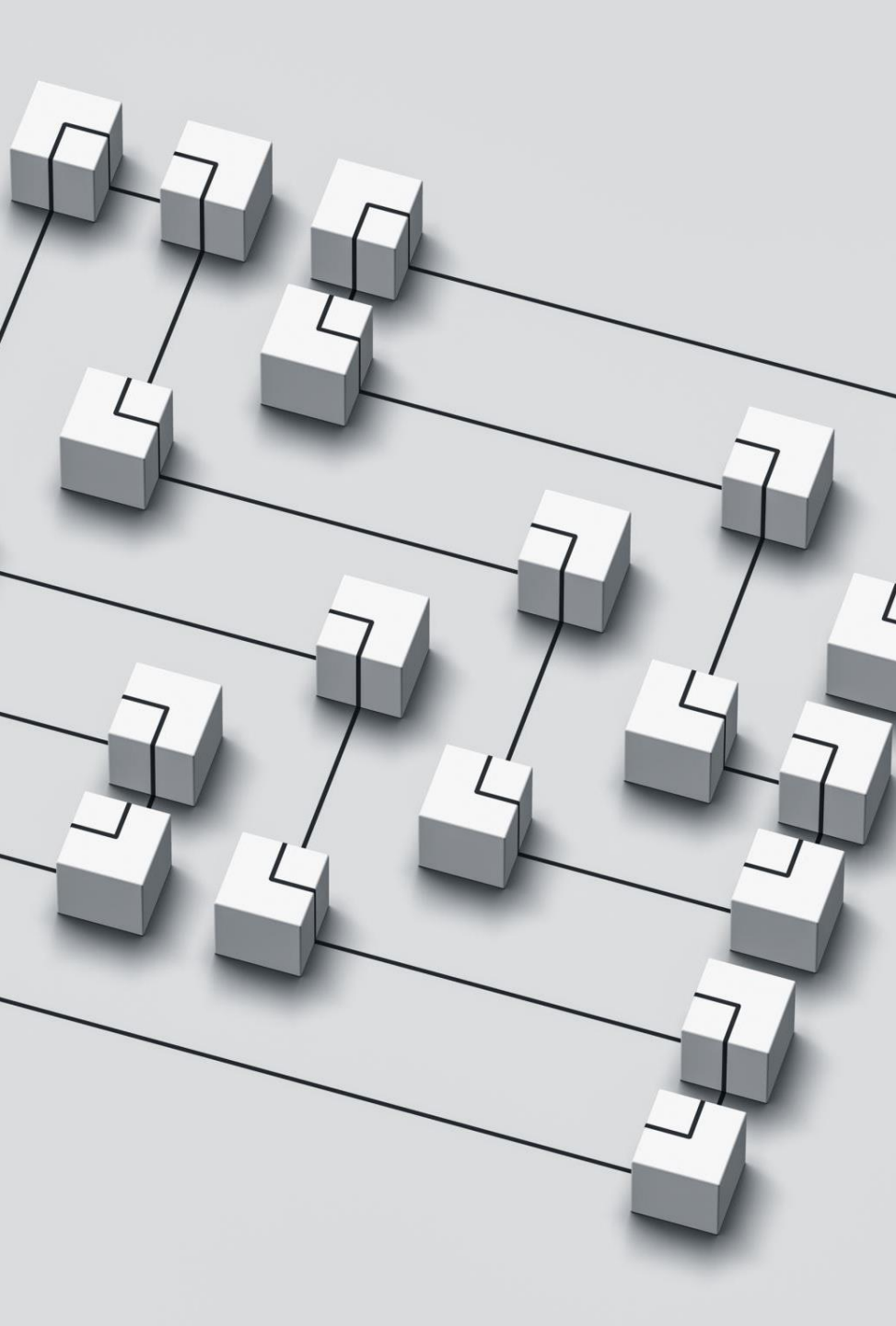
Because it draws from a backlog of work, and begins with a sprint planning meeting, Scrum offers an easy, built-in structure for team leads or product owners to manage and support their team’s most important work.



During a Scrum, your team has a pre-set and limited amount of work and time for each sprint.



This level of built-in prioritization is combined with clearly defined responsibilities ensuring that everyone knows what they’re responsible for at all times.



Introduction to Jira

A web-based project management system and issue tracking system

Can be customized to support any type of workflow or business process.

JIRA consists of

- Jira Software
 - Built for every member of an agile team to plan, track, and ship world-class software.
- Jira Service Management
 - An easy way for customers to ask for help and your agents a faster way to deliver it.
- Jira Work Management
- Jira Align

Jira Software

Jira Software is an edition of JIRA that incorporates the Scrum framework,

- an agile methodology for software development teams.

Jira Software enables member of your software team to

Plan

- Create user stories and issues, plan sprints, and distribute tasks across your software team.

Track

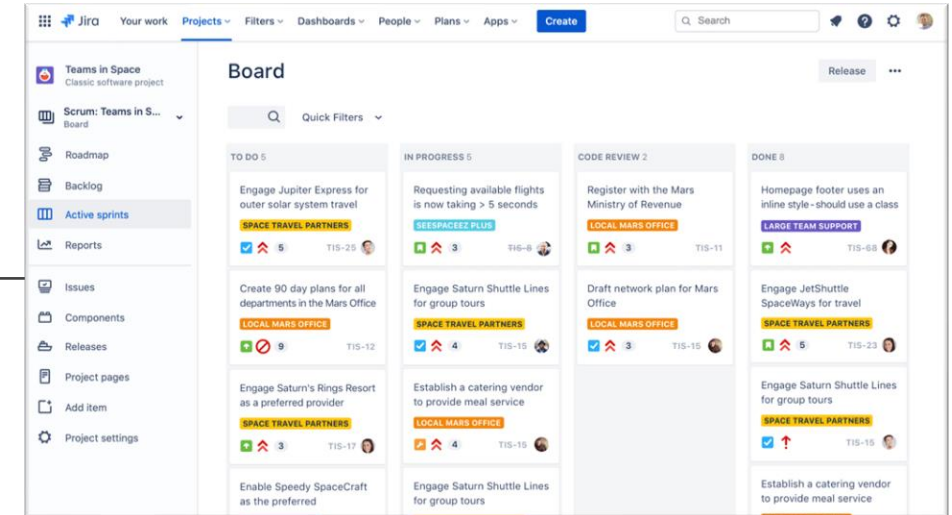
- Prioritize and discuss your team's work in full context with complete visibility.

Release

- Ship with confidence and sanity knowing the information you have is always up-to-date.

Report

- Improve team performance based on real-time, visual data that your team can put to use.



The screenshot shows the Jira Releases view for the same project. It displays a table of releases with the following data:

| Version | Status | Progress | Start date | Release date | Description | Actions |
|-------------|-------------|------------------------|------------|--------------|---------------|---------|
| Version 4.0 | IN PROGRESS | <div><div></div></div> | 03/01/2021 | -- | Awesome... | ... |
| Version 3.0 | UNRELEASED | <div><div></div></div> | 02/15/2021 | -- | Website to... | ... |
| Version 2.0 | UNRELEASED | <div><div></div></div> | 02/01/2021 | -- | -- | ... |
| Version 1.8 | RELEASED | <div><div></div></div> | 01/04/2021 | 02/14/2021 | Version 1.8 | ... |
| Version 1.5 | RELEASED | <div><div></div></div> | 11/30/2020 | 12/18/2020 | Version 1.5 | ... |
| Version 1.3 | RELEASED | <div><div></div></div> | 11/16/2020 | 12/01/2020 | Version 1.3 | ... |
| Version 1.2 | RELEASED | <div><div></div></div> | 10/12/2020 | 11/13/2020 | Version 1.2 | ... |
| Version 1.0 | RELEASED | <div><div></div></div> | 09/21/2020 | 11/06/2020 | Version 1.0 | ... |

At the bottom, there is a form to add a new release with fields for 'Version name', 'Start date', 'Release date', 'Description', and an 'Add' button.

Issue

- Issue is a unit of work within Jira that will be traced through a workflow, from creation to completion.

An Issue can be

- Epic
 - a large body of work requirement
- Story
 - software system requirement that is expressed in a few short sentences
 - ideally using non-technical language.
- Task
 - add your own development related actions.
- Sub-Task
 - single to-dos, assigned to one employee and planned in a specific sprint
- Bug
 - description of problems found in released code

Backlog

- list of the outstanding user stories, bugs and features for a product or sprint.

Story

A story is an informal, general explanation of a software feature

It is written from the perspective of the end user or customer

The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer

The 3 parts of a Story are Who, What, Why

“As a [person], I [want to], [so that].”

- E.g., As system admin, I want to login with maximum rights, so that I can add/edit/delete other users.

It consists of

- Summary, Scope, Pre-requisite, Acceptance Criteria, UI/UX, Attachment, etc.



Tasks are being used to plan specific tasks which should not take more than 1 working day.



They could be part of a bigger project / story or planned by themselves.



Tasks can be separated into subtasks.



Tasks can be assigned to a responsible employee (assignee)



Tasks are part of a sprint or scrum.



They are being worked on actively, can be forwarded to someone else for feedback and can be marked as done when finished.



Tasks can be cross-linked and can block each other.

Project Backlog

One of the key components of a scrum project in Jira Software is the backlog.

The backlog is a list of defined sprints and outstanding issues not assigned to a sprint.

In Jira, “issue” is just a general term for a user story, task, or bug.

These components of the project can have zero or more sub-tasks.

- smaller tasks that are required for the overall issue to be completed.

In terms of functionality for Jira Software, user stories, tasks, and bugs are treated the same.

You can create them in the backlog of the project and move them to a sprint as you see fit.

Jira Backlog

The screenshot displays the Jira Backlog interface for the 'SeniorDesign' project. The top navigation bar includes 'Jira Software', 'Dashboards', 'Projects', 'Issues', 'Boards', and a 'Create' button. A search bar is located on the right. The left sidebar shows the project's navigation menu with options like 'SD board', 'Backlog', 'Active sprints', 'Releases', 'Reports', 'Issues', and 'Components'. Below this is a 'PROJECT SHORTCUTS' section with an 'Add link' button. The main content area is titled 'Backlog' and features a search bar and 'QUICK FILTERS' for 'Only My Issues' and 'Recently Updated'. It lists two sprints: 'SD Sprint 1' (7 issues, ACTIVE) and 'SD Sprint 2' (1 issue). The tasks in SD Sprint 1 include 'SD-1 Play with Jira Software', 'SD-2 Determine my project goals', 'SD-3 Define project scope and direction', 'SD-4 Create project tasks', 'SD-5 Assign tasks to project members', 'SD-6 Weekly sprint review', and 'SD-7 Sprint retrospective'. SD Sprint 2 contains 'SD-8 Consolidate duplicate tasks'. A right-hand panel shows details for issue 'SD-1', including a 'Comments' section (no comments), an 'Attachments' section (drop files to attach), and a 'Sub-Tasks' section with 'SD-9 Create Jira account' and 'SD-10 Login to Jira'.

Jira Software Dashboards ▾ Projects ▾ Issues ▾ Boards ▾ **Create**

Search 🔍 🔊 ⓘ ⚙️ 🖨️

SeniorDesign

SD board ▾

Backlog

Active sprints

Releases

Reports

Issues

Components

PROJECT SHORTCUTS

Add a link to useful information for your whole team to see.

+ Add link

Project settings ⚙️

Backlog

Board ▾ ↗️

🔍 QUICK FILTERS: Only My Issues Recently Updated

SD Sprint 1 7 issues **ACTIVE**

📅 👤 ⋮ 25/Sep/18 1:58 PM • 28/Sep/18 1:58 PM

Get started with Jira

- ✓ ↑ SD-1 Play with Jira Software 👤
- ✓ ↑ SD-2 Determine my project goals 🖨️
- ✓ ↑ SD-3 Define project scope and direction 👤
- ✓ ↑ SD-4 Create project tasks 👤
- ✓ ↑ SD-5 Assign tasks to project members 👤
- ✓ ↑ SD-6 Weekly sprint review
- ✓ ↑ SD-7 Sprint retrospective 🖨️

SD Sprint 2 1 issue **Start Sprint** ⋮

- 🔴 ↑ SD-8 Consolidate duplicate tasks

+ Create issue

SeniorDesign / SD-1

Play with Jira Software

Comments ↑

There are no comments yet on this issue.

🔍 Comment

Attachments ⋮

📎 Drop files to attach, or browse.

Sub-Tasks +

- SD-9 Create Jira account 📄 TO DO ✎️ ✕
- SD-10 Login to Jira 📄 TO DO ✎️ ✕

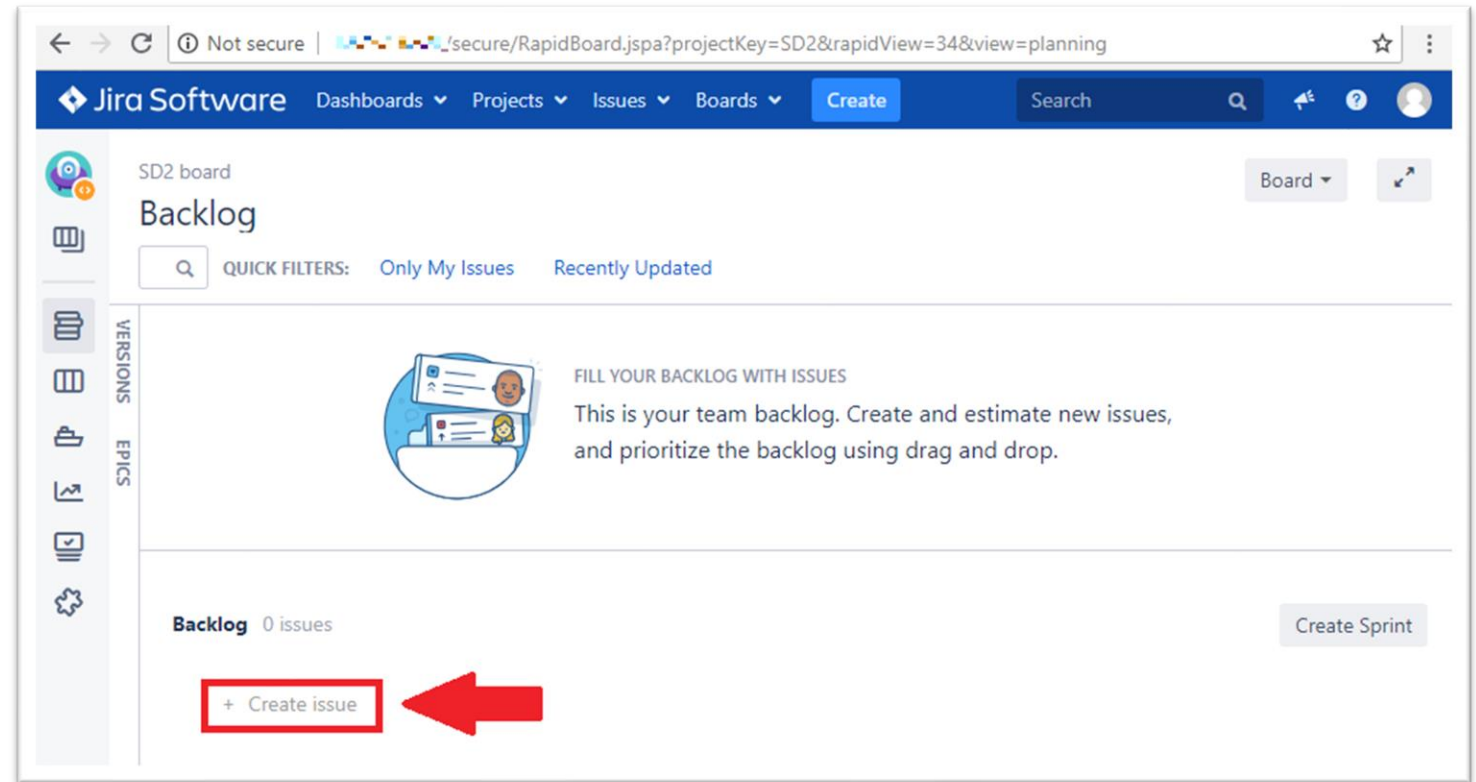
Jira Issues

In Jira Software an issue is a general term used to refer to

- user stories,
- tasks, or
- bugs.

They can be created by navigating to the Backlog of a project

- or by using the Create button on the main navigation bar of the site.



Jira Issues

Enter the issue details, making sure to include

- the project the issue belongs to
- the type of the issue, and
- the summary (title) of the issue.

When finished entering details, click the Create button.

Create Issue

⚙️ Configure Fields ▼

➡️

Project*

Senior Design 2 (SD2) ▼

➡️

Issue Type*

Story ▼ ?

➡️


Summary*

Component/s **None**

Description

Style ▼

B *I* U A ▼ ~~A~~ ▼

🔗 ▼  ▼

☰ ☷

😊 ▼ + ▼

⬆

☐ Create another

Create

Cancel

Scrum Master responsibilities



Create the software project in Jira

Project start/end date (high level)
Roadmap, Backlog, Board



Add stories to the backlog

This is based on the current understanding of the requirements and could change



Add tasks and/or sub-tasks to the backlog

Tasks can be linked to Stories



Do Sprint planning.

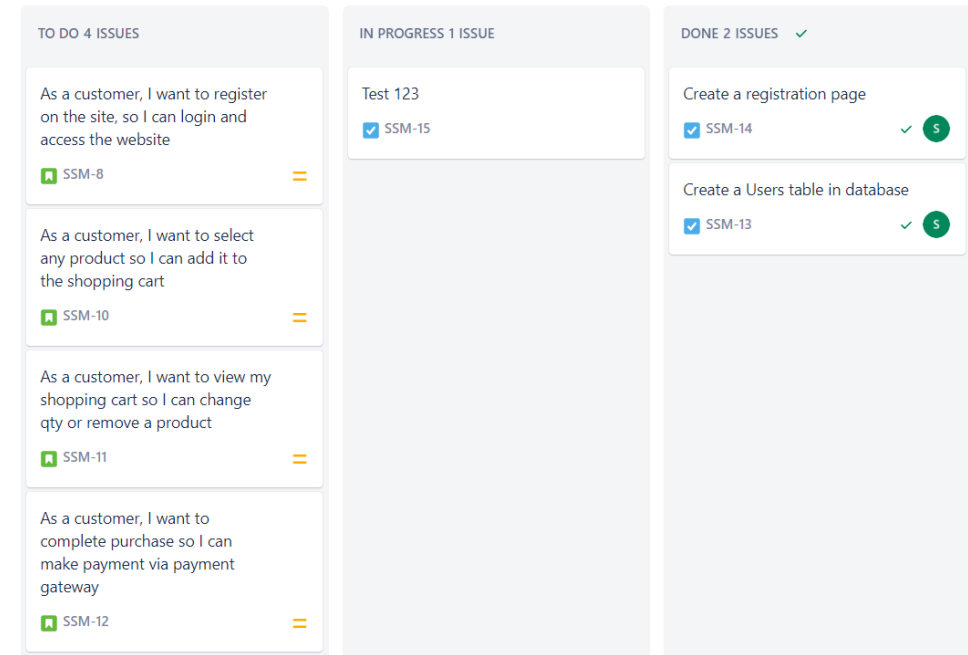
Number of sprints and duration of each sprint (high level)
The 1st one or two sprints with definite start and end dates
Add issues from project backlog to the sprint backlog

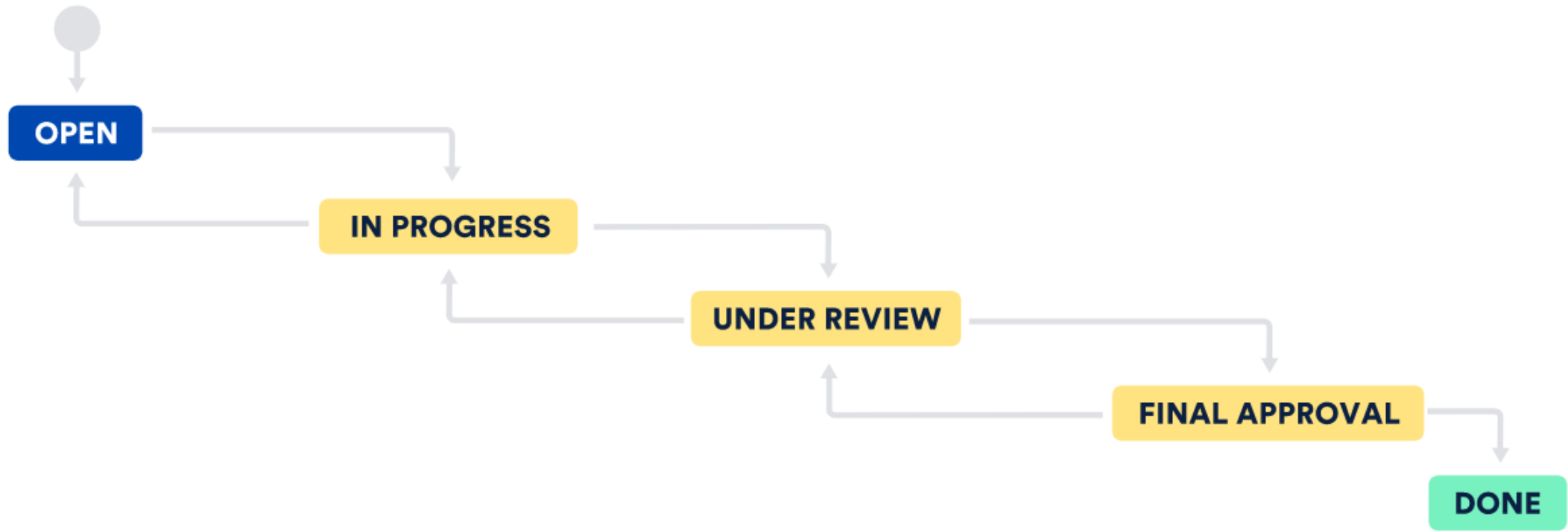
Developer responsibility

When Scrum Master assigns a Task to any developer, an email is sent to the developer.

Developer can then work on the task by changing workflow state from **TO DO** to **IN PROGRESS**

Once work is complete, developer can change the workflow state from **IN PROGRESS** to **DONE**





Choose a workflow, or make your own

Every team has a unique process for shipping software.

Use an out-of-the-box workflow

Or create one to match the way your team works.