

Esoon Ko
IT Högskolan

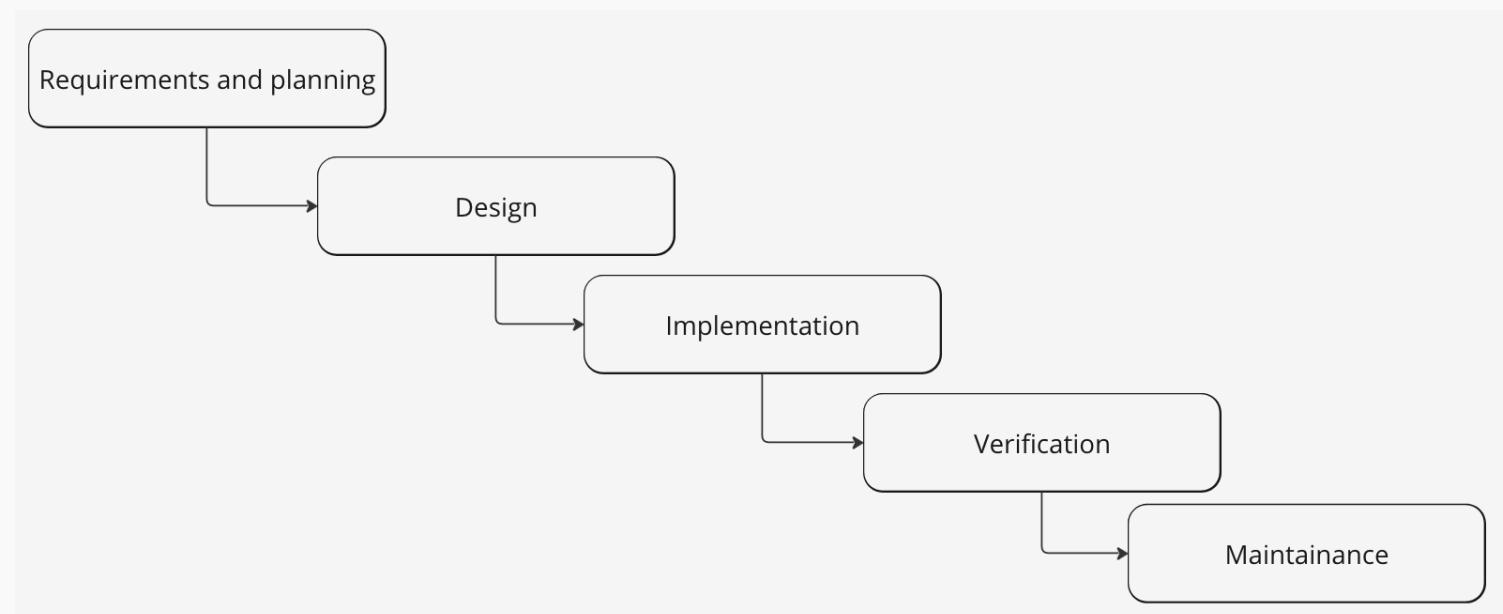
Project Management and Agile Development



Traditional project management in Software Development

The **Waterfall model** is usually used as an example of traditional project management in software development.

The **Waterfall model** is a **linear and sequential approach** where each phase depends on the deliverables of the previous one. This model is named "Waterfall" because it suggests that progress flows downward (like a waterfall) through phases.



Traditional project management in Software Development

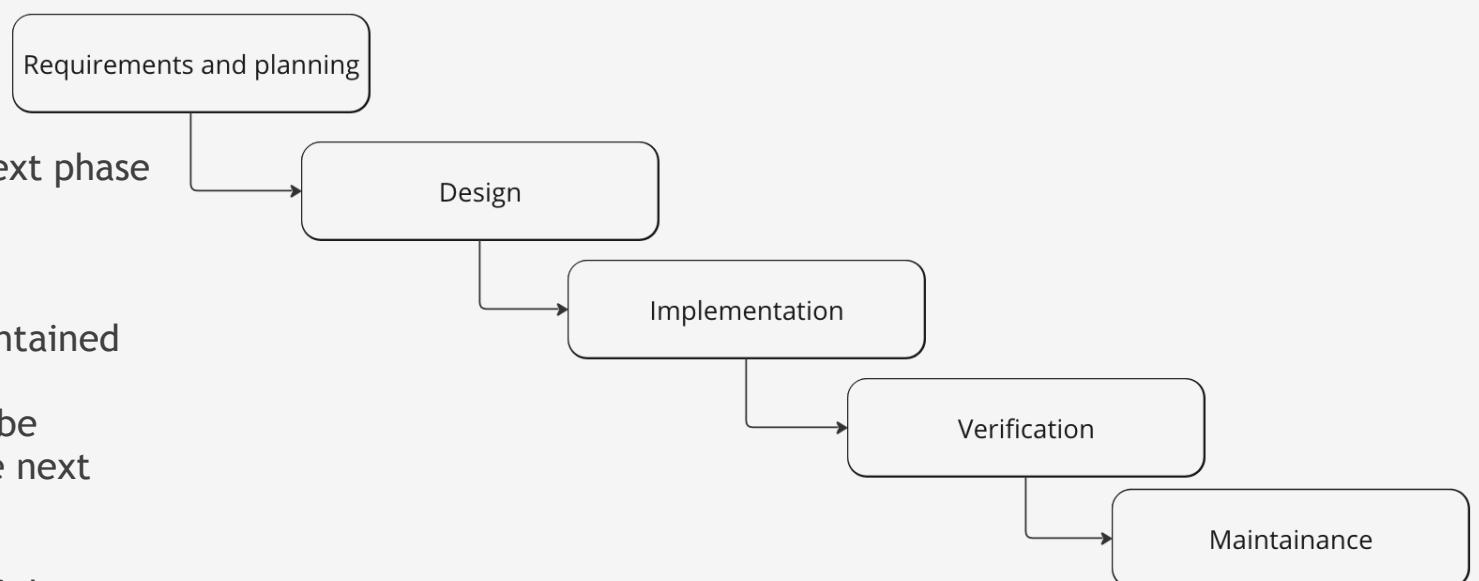
The **Waterfall model** is usually used as an example of traditional project management in software development.

The **Waterfall model** is a **linear and sequential approach** where each phase depends on the deliverables of the previous one. This model is named "Waterfall" because it suggests that progress flows downward (like a waterfall) through phases.

Key Characteristics:

Sequential Progression:

- Each phase must be completed before the next phase begins.
- There is no overlap between phases.



Documentation:

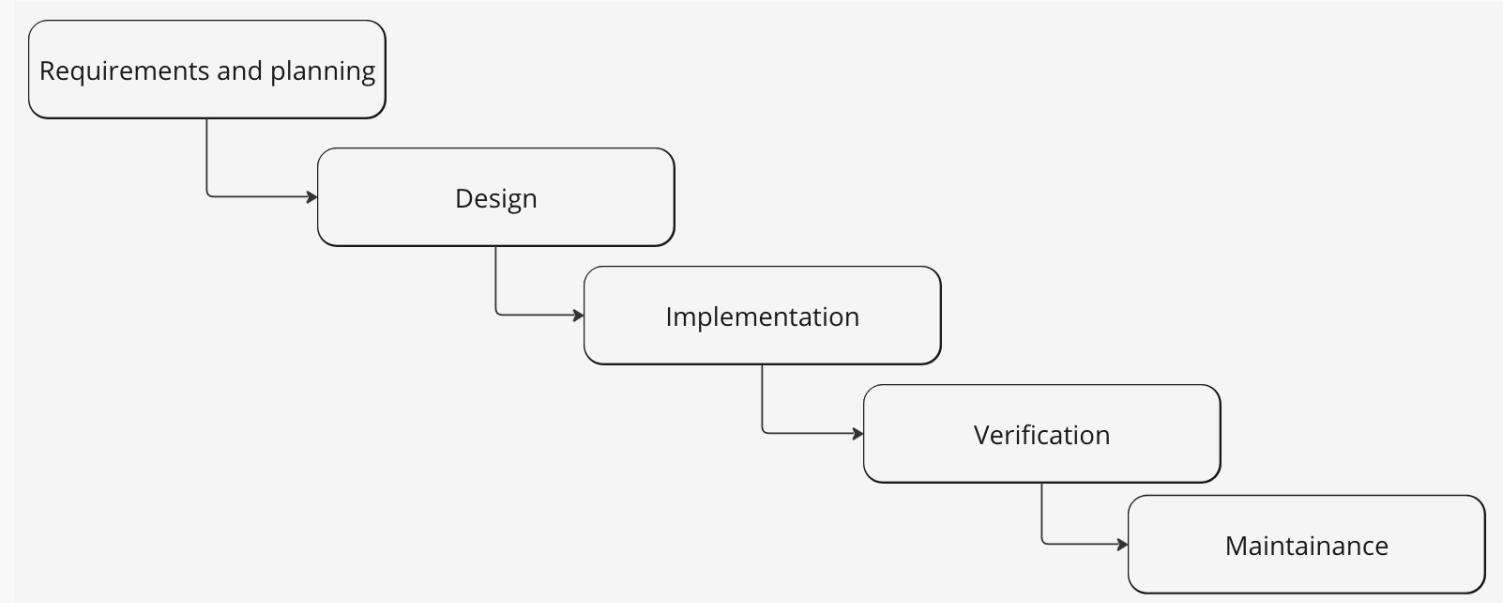
- Extensive documentation is created and maintained throughout the project lifecycle.
- Each phase produces deliverables that must be reviewed and approved before moving to the next phase.

Upfront Planning:

- Detailed planning is done at the beginning of the project.
- Schedules, costs, and resource allocation are estimated and documented upfront.

Advantages and Drawbacks of traditional project planning/management

What advantages and disadvantages do you see?



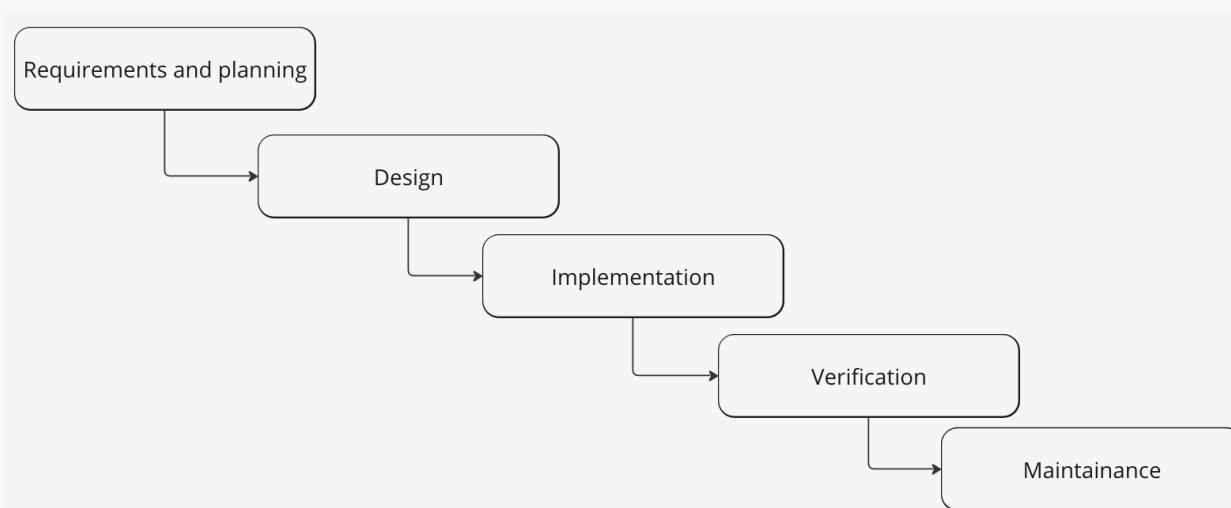
Advantages and Drawbacks of traditional project planning/management

Advantages:

- Structured Approach:
 - Clear and straightforward with defined stages and deliverables.
- Easy to Manage:
 - Simplicity and the sequential nature make it easier to manage and track progress.
- Clear Documentation:
 - Extensive documentation provides a clear trail of the project's history and decisions.

Disadvantages:

- Inflexibility:
 - Changes are difficult to incorporate once the project is underway.
 - Any change requires going back and modifying documents and phases already completed.
- Late Testing:
 - Testing occurs after the implementation phase, making it harder to identify and fix issues early.
- Assumption of Perfect Requirements:
 - Assumes that all requirements can be gathered and understood upfront, which is often not possible in dynamic and complex projects.



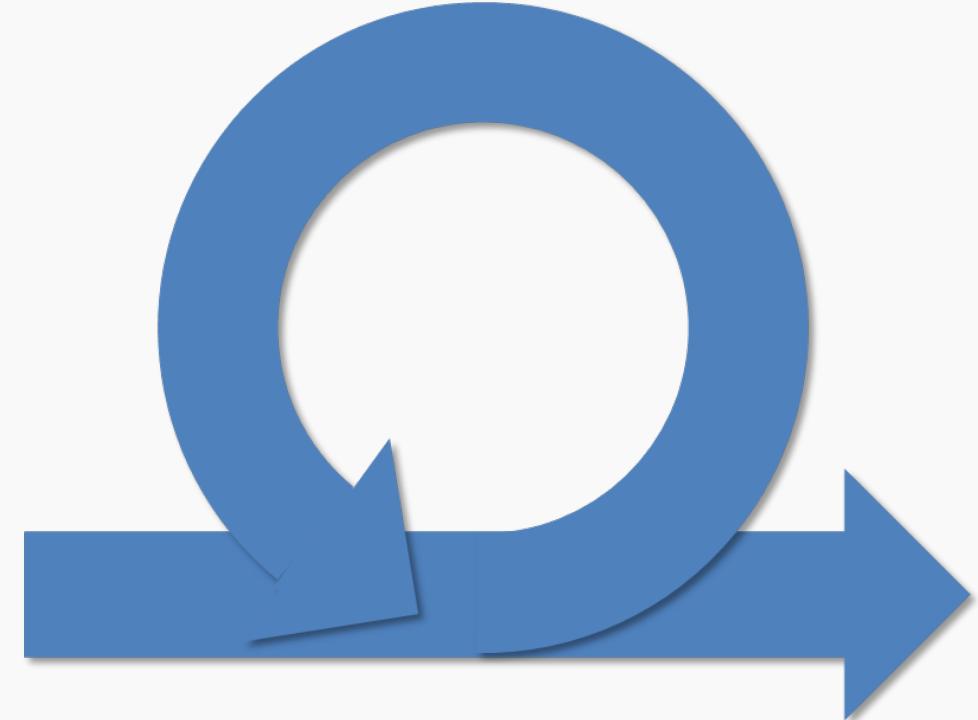
Agile Theory as an alternative

Agile theory represents a set of principles and practices designed to enhance **flexibility, collaboration, and customer satisfaction** in software development. It is often seen as a response to the rigidity of traditional project management methodologies like Waterfall.

Definition: Agile is an iterative and incremental approach to software development that emphasizes flexibility, collaboration, customer feedback, and continuous improvement.

The goal is to deliver value to customers more frequently and adapt quickly to changing requirements.

In other words: Agile is not a instruction manual but rather an approach.



Agile Theory Core Concepts

1. Iterative and Incremental Development:

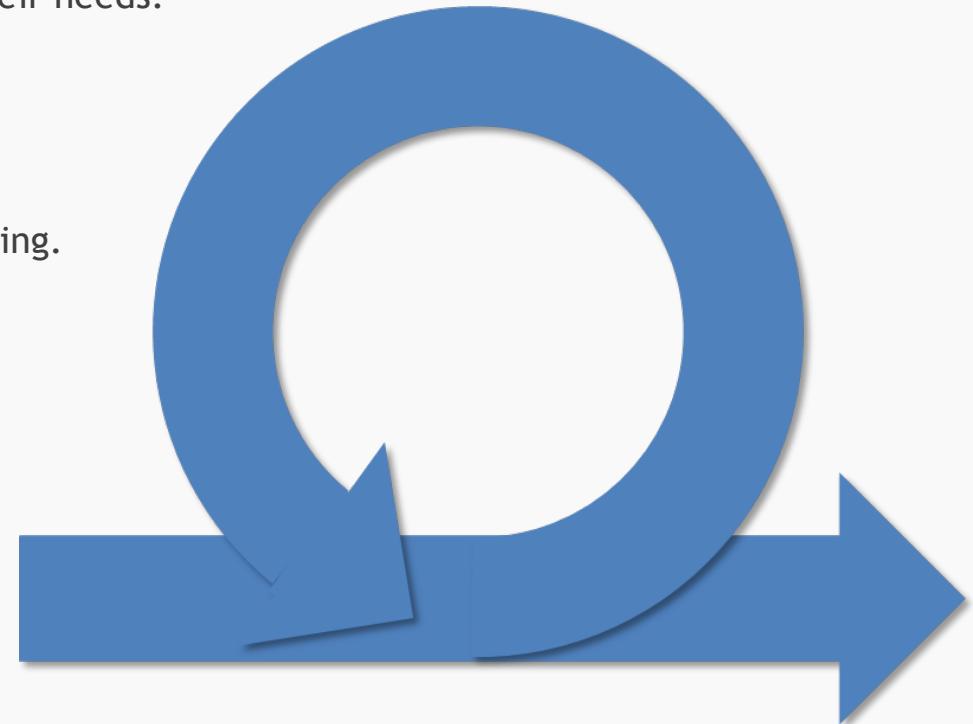
- **Iterative:** Repeating cycles to refine and improve the product.
- **Incremental:** Delivering the product in small, functional pieces.

2. Continuous Feedback and Improvement:

- Regular feedback loops with stakeholders to ensure the product meets their needs.
- Continuous improvement through retrospectives and adapting processes.

3. Collaboration and Communication:

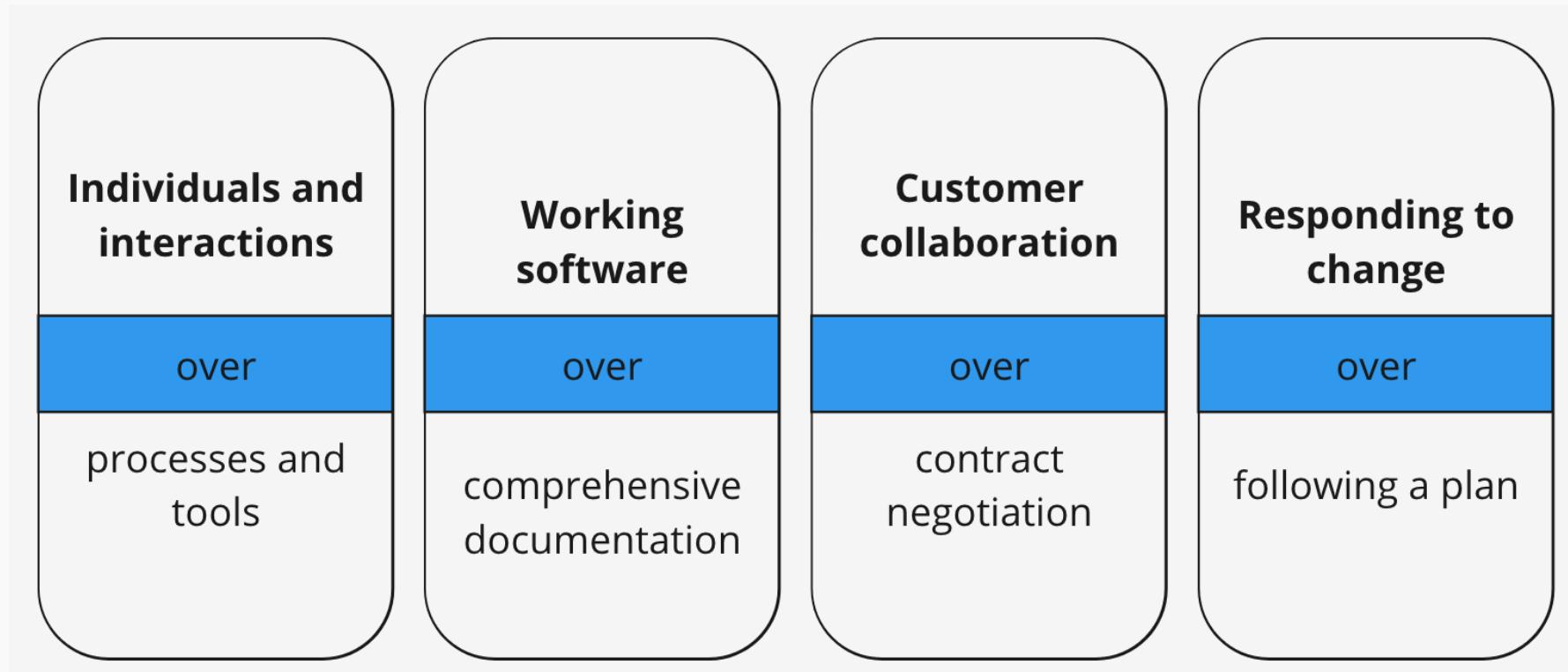
- Close collaboration between cross-functional teams and stakeholders.
- Face-to-face communication preferred for clarity and quick decision-making.
-



Agile Manifesto

The **Agile Manifesto**, created in 2001 by a group of 17 software developers, is a formal proclamation of four key values and twelve principles to guide iterative and people-centric software development processes. It emerged as a response to traditional, rigid methodologies like Waterfall, emphasizing flexibility, collaboration, and customer satisfaction.

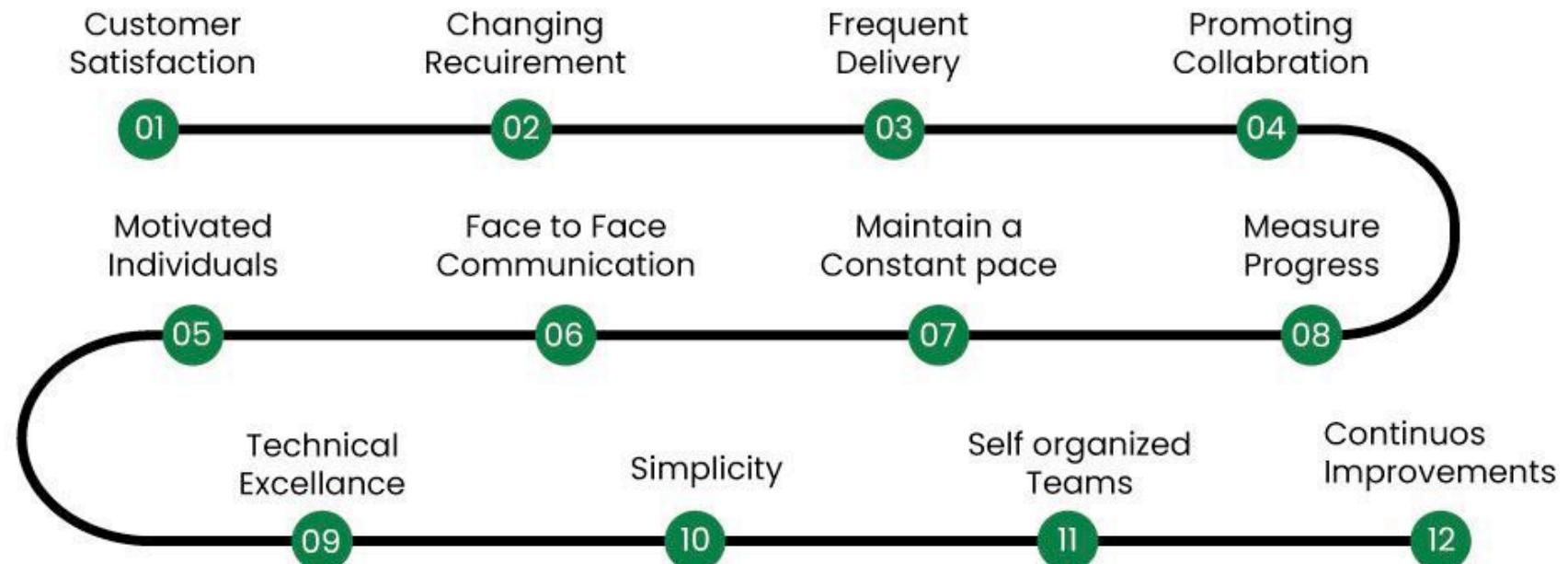
The four key values:



Agile Manifesto

The Agile Manifesto, created in 2001 by a group of 17 software developers, is a formal proclamation of four key values and twelve principles to guide iterative and people-centric software development processes. It emerged as a response to traditional, rigid methodologies like Waterfall, emphasizing flexibility, collaboration, and customer satisfaction.

The twelve key principles:

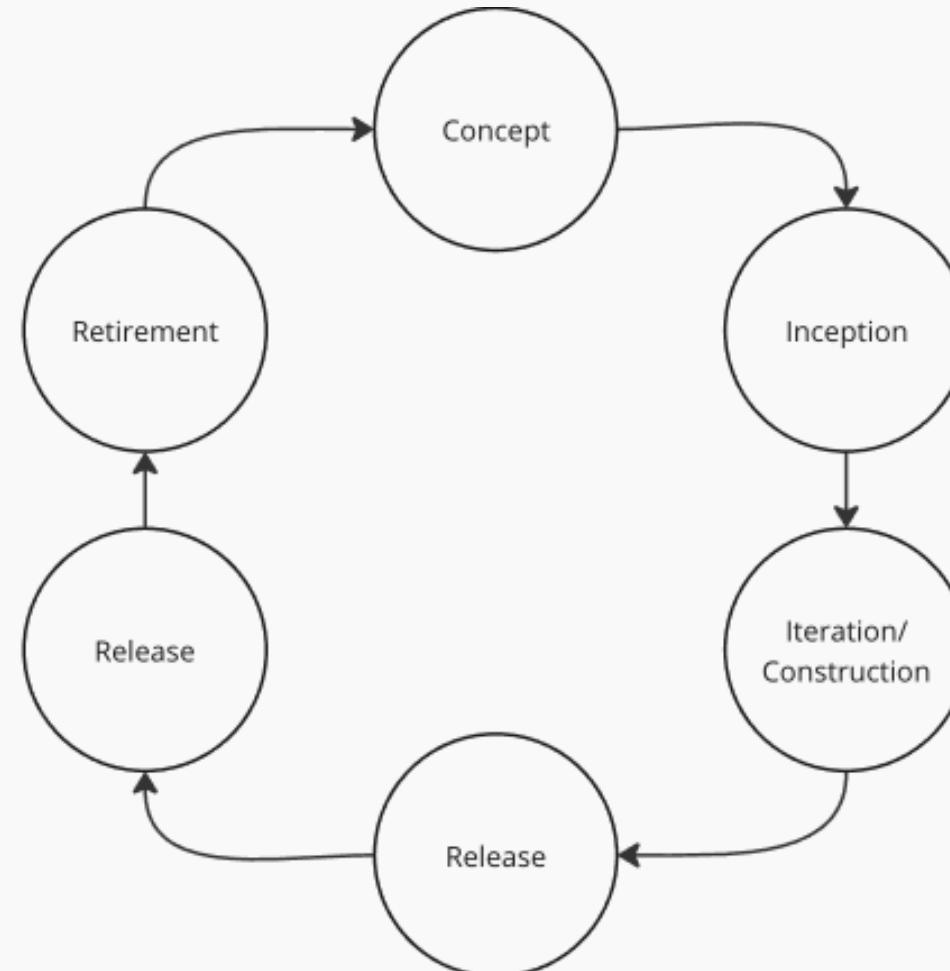


Source:
Geeksforgeeks
<https://www.geeksforgeeks.org/software-engineering-agile-software-development/>

Agile in practice

The manifesto only outlines the basis of Agile. In practice the development cycle takes on the form of six phases:

- Concept
- Inception
- Iteration/construction
- Release
- Maintenance
- Retirement



Fail fast and often!

Agile Frameworks and practices

Agile frameworks and practices provide structured approaches for implementing Agile principles in software development. Each framework has its own set of practices and rules but aligns with the core values and principles of the Agile Manifesto.

Some of them are:

- Scrum
- Kanban
- Scaled Agile Framework (SAFe)
- Continuous Integration
- Test-Driven Development
- Pair Programming

These frameworks and practices can be used together!

Scrum

One of the extensively used agile frameworks to implement agile theory into project management.

Centers around **Sprints** that occur between 1-4 weeks with events being a formal opportunity to inspect and adapt Scrum artifacts.

Daily Scrum meetings occur daily for maximum of 15 min to inspect progress toward the sprint goal.

Artifacts represent work or value

The **Scrum Team** makes sure the sprints go on.

Sprint planning outlines the **sprint goal** which explains how the sprint brings value. The planning takes maximum of 8 hours.

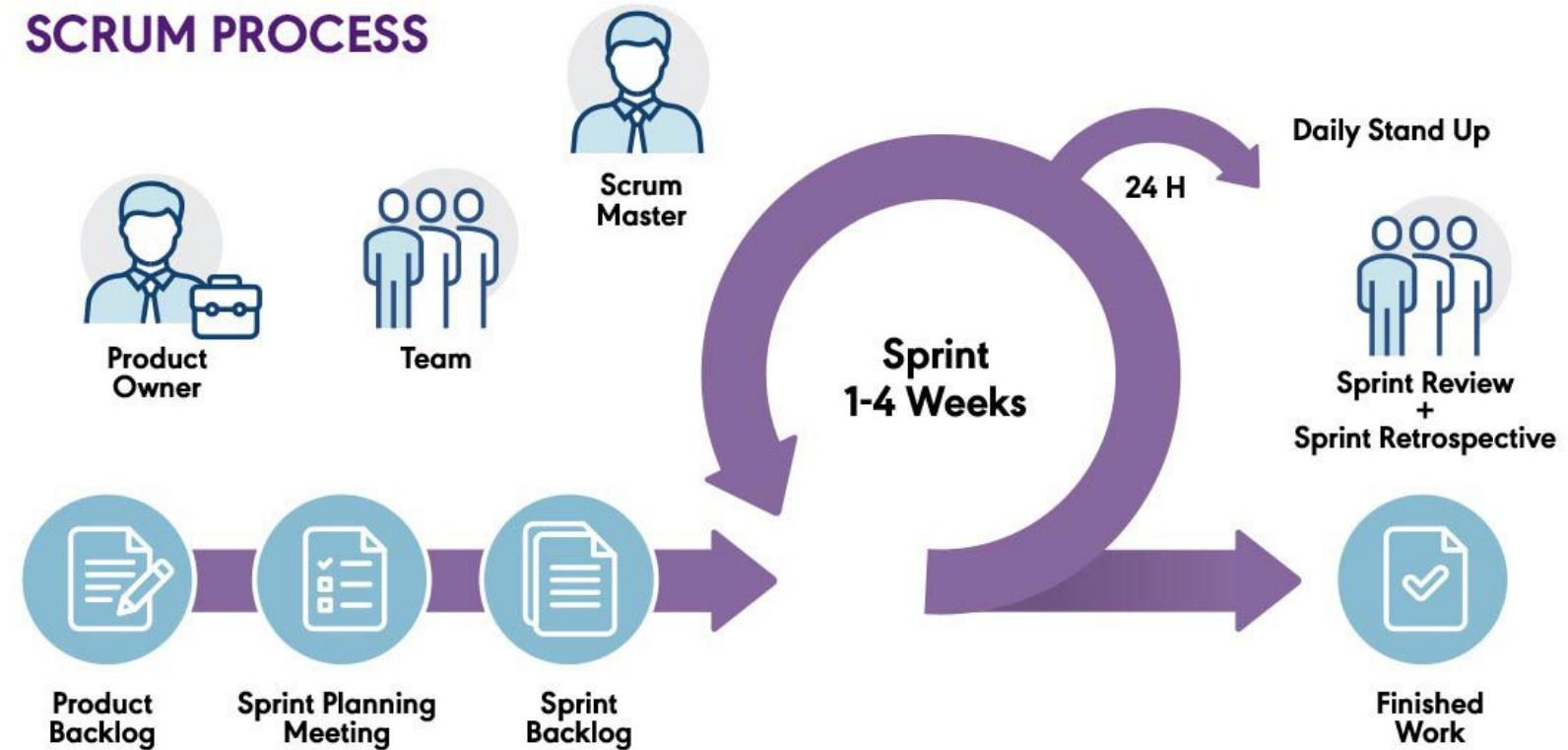


Image Source:: PM-Partners
<https://www.pm-partners.com.au/insights/the-agile-journey-a-scrum-overview/>

Scrum Team

The Scrum team is at the heart of Scrum and is composed of **three roles**:

- Product Owner
 - Represents the customer's voice
- Scrum Master
 - Accountable for establishing Scrum and the team's effectiveness
- Developers
 - Developers committed to delivering increments



Image Source:
PM-Partners
<https://www.pm-partners.com.au/insights/the-agile-journey-a-scrum-overview/>

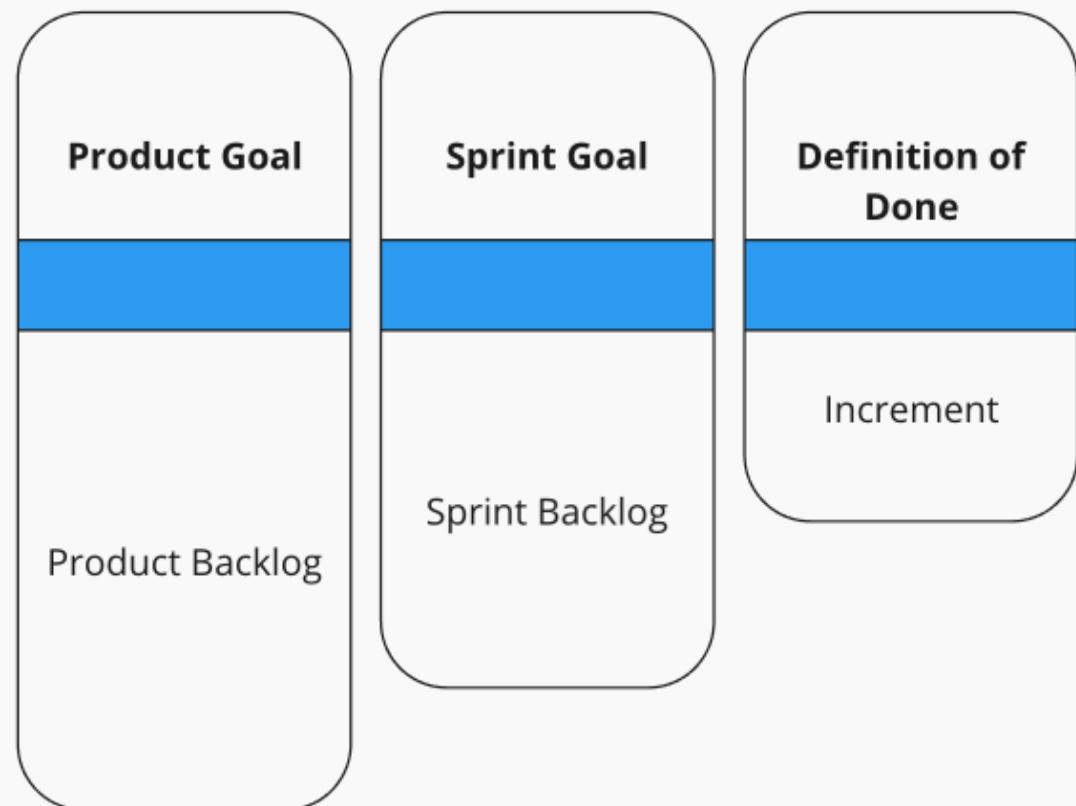
Scrum Artifacts

Artifacts represent work or value:

- **Product Backlog:** Ordered list of what is needed to improve the product
- **Sprint Backlog:** Sprint goal and set of product backlog items related to the sprint
- **Increment:** Represents the latest functional product version after a Sprint

Each **Artifact** has a **Commitment:**

- **Product Goal:** Describes a future state of the product which can serve as a target for the Scrum Team to plan against.
- **Sprint Goal:** The Sprint Goal is the single objective for the Sprint.
- **Definition of Done:** Formal description of the state of the Increment when it meets the quality measures required for the product.



Scrum Three pillars

Implements three pillars:

- **Transparency**

- All aspects of the process that affect the outcome must be visible to those managing the outcomes. This includes visibility into the work process, progress, and any obstacles.

- **Inspection**

- Regularly checking Scrum artifacts and progress toward the Sprint Goal to detect any variances or problems.

- **Adaptation**

- If any aspects of the process deviate outside acceptable limits, adjustments must be made as soon as possible to minimize further deviation.

How is it done?

- **Artifacts:** Product backlog, Sprint Backlog and Increments must be transparent

- **Meetings:** Regular meetings provide platform for sharing openly

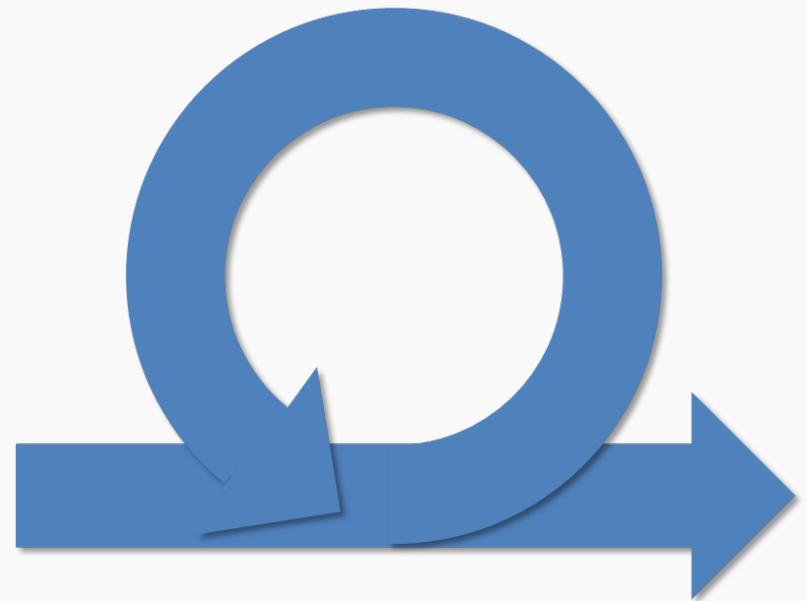
- **Daily Scrum:** Short, daily meeting of 15 minutes where the team inspects progress and adjusts plan.

- **Sprint Review:** At the end of each sprint inspection and discussion occurs where feedback from stakeholders can lead to adjustments in product backlog or sprint backlog.

- **Sprint Retrospective:** The team reflects on the past sprint, discussing what went well and what didn't. Action items identified for improvement are implemented in the next sprint.

Agile Benefits/Drawbacks

What benefits and drawbacks are there? Discuss!



Agile Benefits/Drawbacks

Advantages:

- Deployment of software is quicker with frequent deliverables which help build trust to the customer.
- Can better adapt to rapidly changing requirements and respond faster.
- People and interactions are given a higher priority than processes and tools.
- Increased collaboration and communication.

Disadvantages:

- Can be hard to asses effort required for a product in the beginning phases of development.
- Heavily dependent on inputs from customers. If the customer has ambiguity in their vision or “unsure” of what they want the project quickly snowball.
- Can be hard to predict project outcomes, timelines and budget.
- Risk of team burnout.
- Lack of structure and governance.



Kanban

Kanban is a popular framework within Agile methodology, used for managing and improving work across human systems. It helps teams visualize their work, maximize efficiency, and improve continuously.

Comes from Toyota and means “Visual Signal” in Japanese

Nowadays it is used as a development framework that involves the use of a **Kanban Board** to visualize the workflow of the entire project.

A Kanban Board is simply a work board that is divided into several columns, each representing a workflow phase.



Image Source: Wikipedia

[https://sv.wikipedia.org/wiki/Kanban#/media/
File:Simple-kanban-board-.jpg](https://sv.wikipedia.org/wiki/Kanban#/media/File:Simple-kanban-board-.jpg)

Kanban Workflow

Usually workflow in Kanban has 4 phases:

- **To do:** Tasks that needs to be done
- **In progress:** Tasks in progress
- **Validation:** Tasks that need to be reviewed and tested
- **Done:** Tasks that are done

To Do	In Progress	Validation	Done

Kanban Principles

Four principles in Kanban:

- Start with what you do know

- Value the existing workflows and practices that work well instead of setting up the environment from scratch in the beginning.

- Agree to continue evolutionary and incremental changes

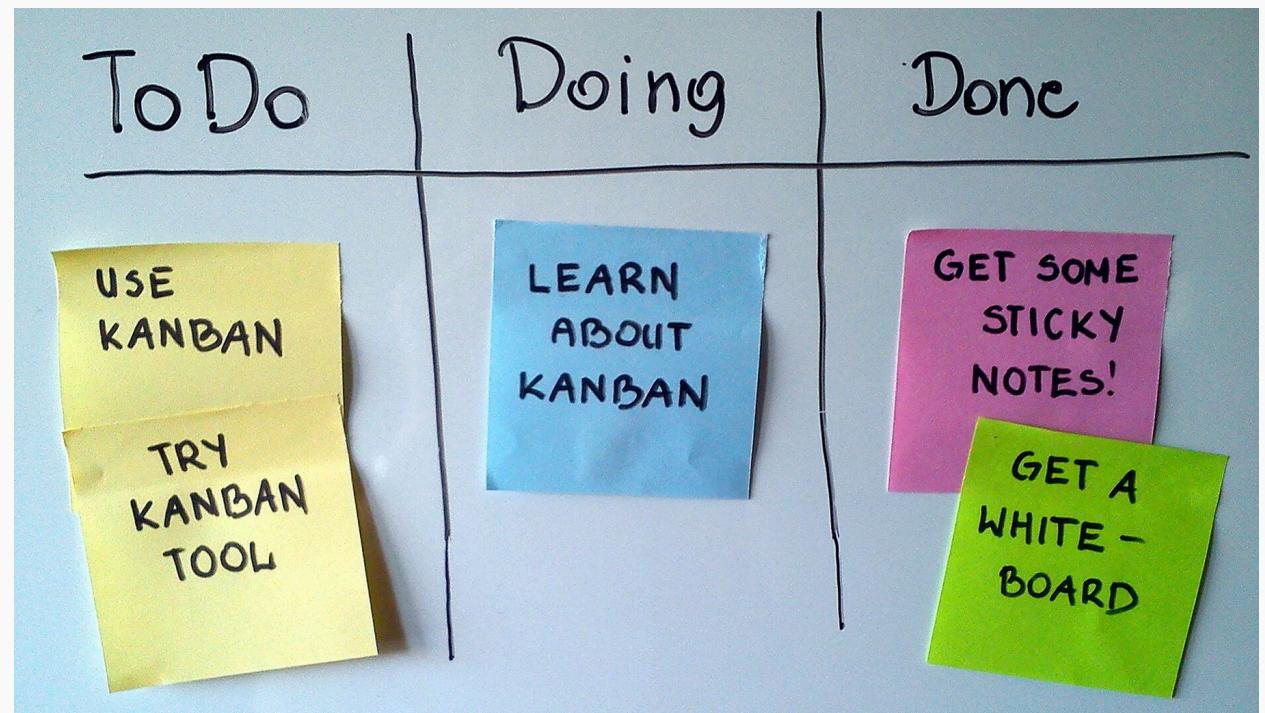
- Focus on small and manageable changes. Massive changes can overwhelm teams and introduce unforeseen challenges.

- Respect current roles, processes, responsibilities & titles

- Respect the organizational processes, roles, and titles and aim to enhance rather than disrupt the natural order of operations. By honoring the current structure, Kanban reduces resistance to change and allows for quick implementation because the company does not need to restructure before starting with Kanban.

- Encourage Leadership at all levels

- Decisions should not always be proposed by big leaders only but even the freshest team member can have creative ideas that should be appreciated, embraced and acknowledged.



Implementing Kanban

1. Map your current workflow

Identify the steps in your current workflow. Talk through how work travels from “To Do” to “Done”. Try mapping it out on a whiteboard.

2. Visualize your work

Visualize the work process by creating a board with cards for each work. Do not feel pressure to fill out specific details.

3. Focus on flow

A good flow is how smoothly work moves through your system without much delay or queue.

4. Limit your WIP

Deliberately control how much work is in the system in order to reduce work in progress. In Kanban this means keeping the amount of cards to a minimum.

5. Measure and Improve

Your Kanban board should be ever evolving. Be flexible and open to improvements and change. Measure and analyze your performance as a team (such as lead time).



Agile tools

There are several agile tools you can use to help with your agile development:

- Jira
- Monday
- Notion
- And more

You will notice many combines aspects of both Scrum and Kanban

Jira

Offers Scrum with Kanban support.

The screenshot shows the Jira interface with the 'Board' view selected. On the left, there's a sidebar with project navigation links like 'Your work', 'Projects', 'Filters', 'Dashboards', 'People', 'Plans', 'Apps', 'Create', 'Search', and user icons. The main area is titled 'Board' and displays a Kanban board with four columns: 'TO DO 5', 'IN PROGRESS 5', 'CODE REVIEW 2', and 'DONE 8'. Each column contains several items represented as cards. The 'TO DO' column has two cards: 'Engage Jupiter Express for outer solar system travel' and 'Create 90 day plans for all departments in the Mars Office'. The 'IN PROGRESS' column has three cards: 'Requesting available flights is now taking > 5 seconds', 'Engage Saturn Shuttle Lines for group tours', and 'Enable Speedy SpaceCraft as the preferred'. The 'CODE REVIEW' column has one card: 'Register with the Mars Ministry of Revenue'. The 'DONE' column has five cards: 'Homepage footer uses an inline style - should use a class', 'Draft network plan for Mars Office', 'Engage JetShuttle SpaceWays for travel', 'Engage Saturn Shuttle Lines for group tours', and 'Establish a catering vendor to provide meal service'. Each card includes details like assignee, priority, and due date (e.g., TIS-25, TIS-8, TIS-11, TIS-12, TIS-15, TIS-23, TIS-15). The sidebar also lists 'Active sprints' (which is currently selected), 'Roadmap', 'Backlog', 'Reports', 'Issues', 'Components', 'Releases', 'Project pages', 'Add item', and 'Project settings'.

Monday

Allows you to add any board of your liking with board templates.

The screenshot displays two main sections of the Monday.com platform:

Main Table View: The top section shows a table titled "monday 2.0 launch". The columns are: Item, Person, Status, Date, Hours, and Location. The data includes:

Item	Person	Status	Date	Hours	Location
Departmental and Overall rebrand budget	(User icon)	Stuck	Jul 16, 2020		
Logo Redesign	(User icon)	Working on it	Jul 15, 2020		
Copyright Procedures	(User icon)	Up Next	Jul 17, 2020		
Benefits Overhaul	(User icon)	Done!	Jul 17, 2020		
Website Redesign	(User icon)	In progress	Jul 18, 2020		
Proofreading	(User icon)	Working on it	Jul 13, 2020		
+ Add Item					

Kanban Inventory View: The bottom section shows a Kanban board titled "Mad Hatter Inventory". It has five columns: Time to Order / 3, Fully Stocked / 6, Out / 2, Well Stocked / 3, Running Low / 2, and Empty / 0. The "Kanban Column" settings panel on the right is highlighted with a red box, showing options for Stock, Color, and Stock.

Time to Order / 3	Fully Stocked / 6	Out / 2	Well Stocked / 3	Running Low / 2	Empty / 0
Samuel - Wool	Clarita	Hudson - Felt	Hudson - Felt	Frederich - 13"	
Marvin - 10"	Jack - 8"	Jerry	Beach Style	Charlie	
Cooper - Canvas	Genevieve	+ Add	Loretta - Wool	+ Add	
+ Add	Loretta - Straw		Allie	+ Add	

Notion

Combines notebook style page editors with board templates.

The screenshot shows a Notion Kanban board with the following structure:

- Not started:** 3 items
- In development:** 4 items
 - Create marketing materials (Nate Martins, Design, December 30, 2022)
 - Finalize visuals for feature (Nate Martins, Design, December 24, 2022)
 - Create visuals for feature (Ben Lang, Design, December 24, 2022)
 - User feedback on feature (Ben Lang, Design, December 24, 2022)
- Testing:** 3 items
- Reviewing:** 3 items
- Done:** 5 items
 - Create iconography for feature (Sohrab Amin, Design, December 22, 2022)
 - User testing of feature (Sohrab Amin, Design, December 22, 2022)
 - Design icons and buttons (Ben Lang, Design, December 19, 2022)
 - Brainstorm feature ideas (Sohrab Amin, Design, December 19, 2022)
 - Create wireframes of feature (Nate Martins, Design, December 19, 2022)
- Engineering:** 9 items

Another good tool for whiteboarding: Miro

Allows you to collaborate on white boarding

