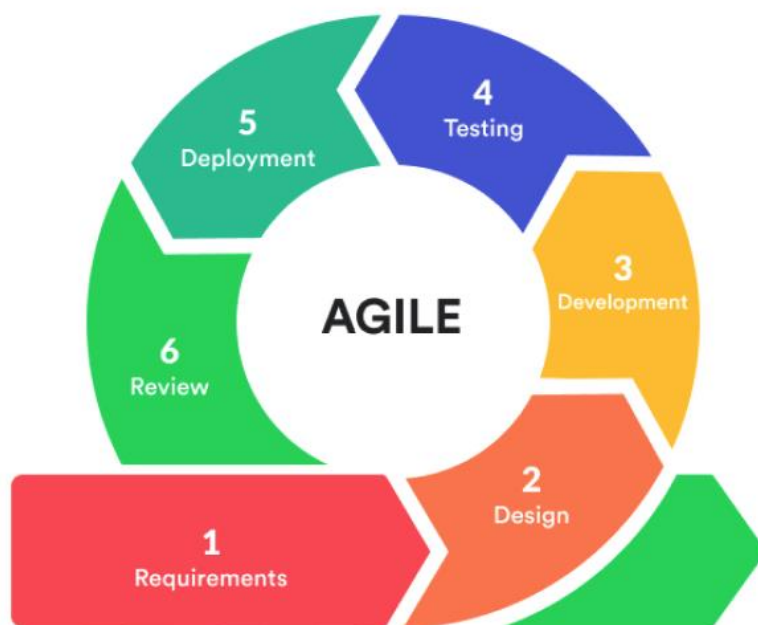

1. Overview of Agile Principles and Values

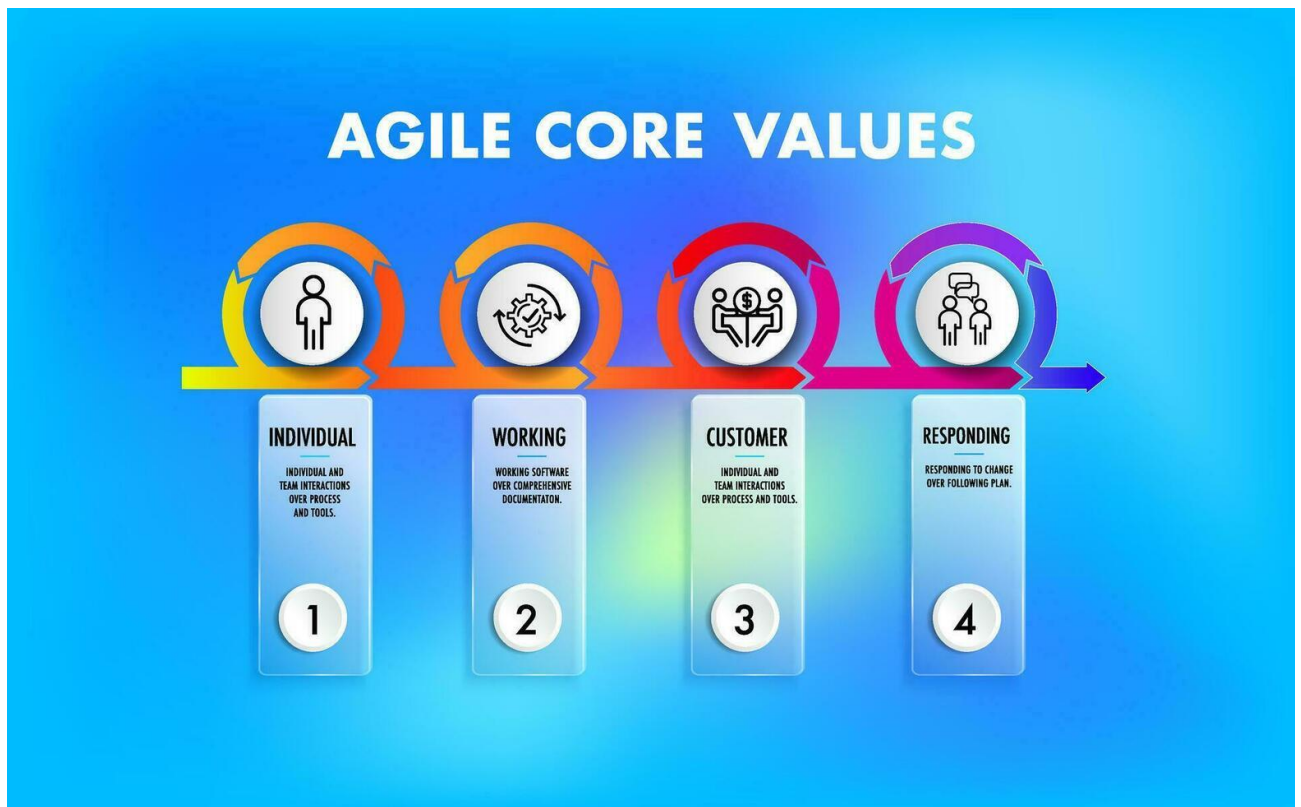
Agile is a flexible, iterative approach to project management and software development that helps teams deliver value to customers faster.



Comparison with Waterfall Methodology:

Feature	Agile	Waterfall
Approach	Iterative and incremental	Sequential and linear
Flexibility	Adapts to changes easily	Changes are difficult once the project starts
Deliverables	Continuous small increments	Delivered at the end of the project
Customer Involvement	High throughout the project	Involvement mainly in the beginning and end
Testing	Ongoing throughout development	Done in a final testing phase

2. Agile Manifesto: Four Core Values



- 1. Individuals and interactions over processes and tools**
 - People and communication are more important than rigid processes.
- 2. Working software over comprehensive documentation**
 - Delivering functional products is prioritized over excessive documentation.
- 3. Customer collaboration over contract negotiation**
 - Continuous feedback and collaboration with customers are essential.
- 4. Responding to change over following a plan**
 - Agile welcomes changes to requirements even late in development.

3. Agile's 12 Principles

12 agile principles

in software development



Customer
satisfactions



Changing
requirements



Frequent
delivery



Communicate
regularly



Support
team member



Face-to-face
communication



Measure
work progress



Development
process



Good
design



Measure
progress



Continue
seeking result



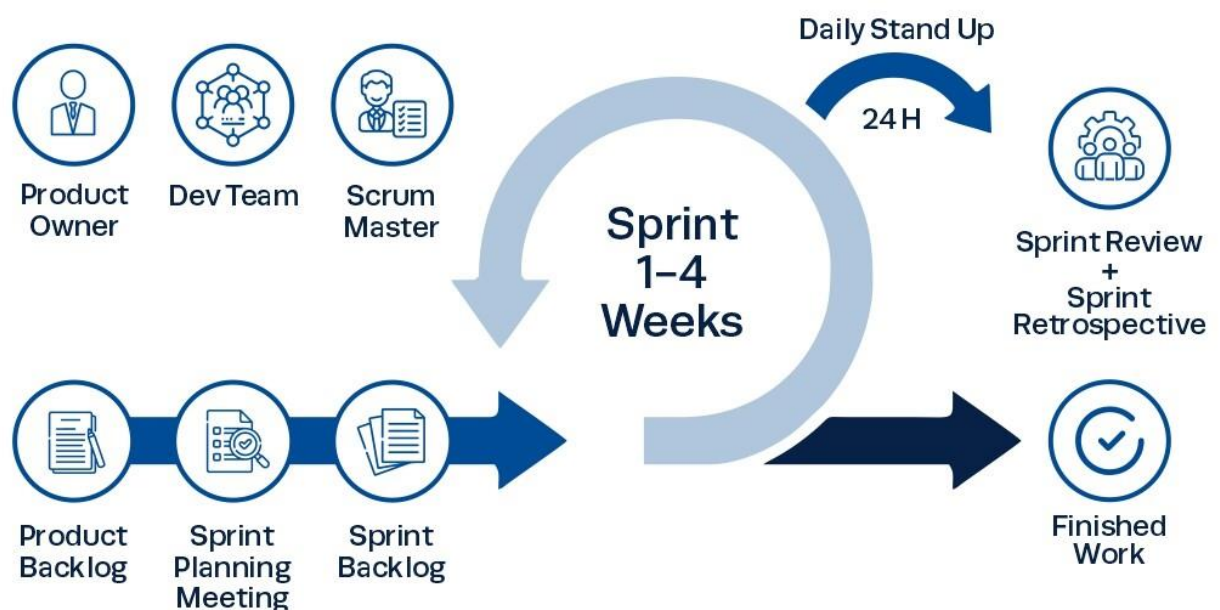
Reflect and
adjust regularly

1. **Customer satisfaction through early and continuous delivery.**
2. **Welcome changing requirements, even late in development.**
3. **Deliver working software frequently (weeks rather than months).**
4. **Close collaboration between business and developers.**
5. **Support and trust motivated individuals.**
6. **Face-to-face communication is the best way to convey information.**
7. **Working software is the primary measure of progress.**
8. **Sustainable development pace.**
9. **Continuous focus on technical excellence and good design.**
10. **Simplicity – maximizing the amount of work not done – is essential.**

11. **Self-organizing teams produce the best results.**
 12. **Regular reflection and adjustments to improve efficiency.**
-

4. Scrum Framework

Scrum is a popular Agile framework used for software development.



Key Scrum Roles:

- **Product Owner (PO):** Defines the product vision, prioritizes backlog items, and ensures value delivery.
- **Scrum Master:** Facilitates the Scrum process, removes blockers, and ensures the team follows Agile best practices.
- **Development Team:** A cross-functional group that builds the product increment.

Scrum Ceremonies:

1. **Sprint Planning:** Team selects backlog items and plans the sprint goals.
2. **Daily Stand-up:** A short daily meeting to discuss progress and challenges.
3. **Sprint Review:** Team showcases completed work to stakeholders.
4. **Sprint Retrospective:** Reflection on the sprint to identify improvements.

Scrum Artifacts:

- **Product Backlog:** A prioritized list of work items (user stories, features, bugs).
- **Sprint Backlog:** Selected tasks from the Product Backlog for a sprint.
- **Increment:** The usable, working product delivered at the end of a sprint.

5. Agile Estimation & Planning Techniques

Agile Estimation Techniques:



1. **Story Points:** A relative estimation method based on complexity and effort.
2. **Planning Poker:** A collaborative game where team members estimate story points using numbered cards.

Agile Planning Techniques:

1. **Release Planning:** High-level roadmap for delivering product increments.
 2. **Iteration (Sprint) Planning:** Breakdown of tasks for a specific sprint.
-

6. Writing Effective User Stories

User stories describe features from an end-user perspective in this format:

"As a [user], I want [feature] so that [benefit]."

Example:

"As a customer, I want to reset my password so that I can regain access to my account."

Acceptance Criteria for User Stories

- Define conditions that must be met for the story to be considered "done."
 - Example for a password reset feature:
 - User receives a reset link via email.
 - Link expires after 24 hours.
 - User is prompted to set a new password.
-

7. Key Agile Metrics & Reporting

Agile Metrics:

1. **Velocity:** Measures the amount of work completed in a sprint (measured in story points).
2. **Burndown Chart:** Tracks work completed versus work remaining in a sprint.

Reporting Progress in Agile Projects:

- **Sprint Reports:** Show what was planned vs. completed.
 - **Release Reports:** Track overall project progress.
 - **Daily Stand-ups:** Provide quick status updates.
-