Rakshitha Devi J

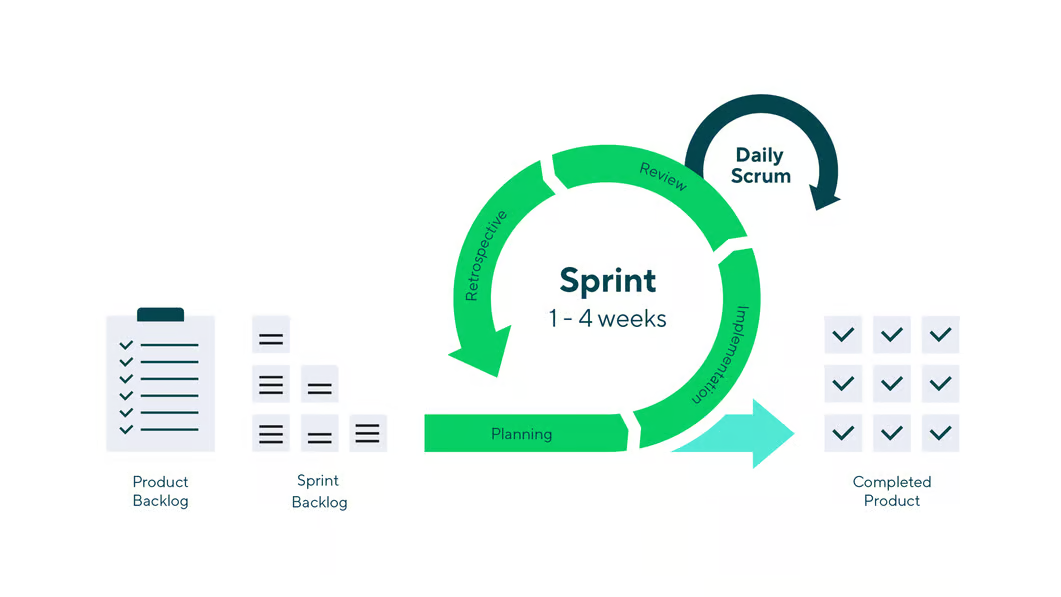
Agile Methodology

The **Agile methodology** is a proper way of managing the project with breaking them into smaller phases which is iteration. It basically focus on flexibility of the project which we can change and improve the team work regularly as per requirements.

Agile is built on four main values from the [Agile Manifesto](http://agilemanifesto.org/):

* **People over processes**: Teams work together closely, solving problems through open conversations.
* **Working solutions over detailed documentation**: Agile teams prioritize building and testing usable products over writing lengthy reports.
* **Customer collaboration over rigid contracts**: Agile encourages ongoing discussions with customers to adjust work as needed.
* **Adapting to change over following a strict plan**: Agile teams stay flexible, ready to adjust to new information, customer needs, or shifting priorities.

By responding to customer needs and adapting to change more efficiently, these values help drive a development process that reliably delivers quality products and happy customers.



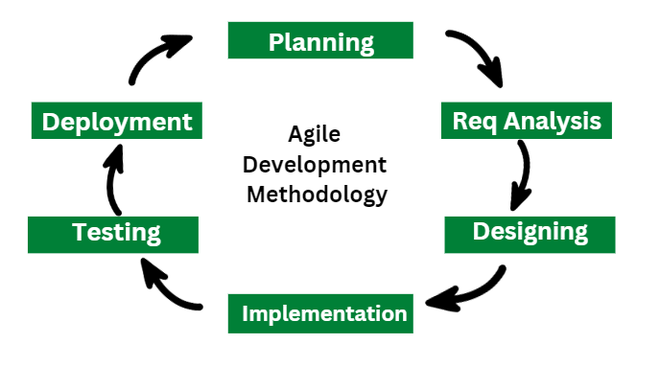
### **12 Principles of Agile**

1. Satisfy the customer through early and continuous delivery of valuable software.
2. Welcome and harness changes for the customer's competitive advantage, even late in development.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for shorter timescales.
4. Have daily collaboration between business people and developers throughout the project.
5. Build projects around motivated individuals. Create the environment and support developers need, and trust them to get the job done.
6. Prioritize face-to-face conversation as the most efficient and effective method of conveying information to and within a development team.
7. Measure progress by the amount of working software completed.
8. Maintain a constant and sustainable pace of development indefinitely.
9. Enhance agility through continuous attention to technical excellence and good design.
10. Keep it simple. Simplicity—the art of maximizing the amount of work not done—is essential.
11. Recognize that the best architectures, requirements, and designs emerge from self-organizing teams.
12. Regularly reflect and adapt behavior for continual improvement.

These Agile values and principles represent an umbrella philosophy that can (and has been) applied to numerous frameworks and methodologies in both software development and other project management processes

## **Life cycle of Agile Methodology**

The [**Agile Software Development Life Cycle**](https://www.geeksforgeeks.org/agile-sdlc-software-development-life-cycle/) helps you break down each project you take on into six simple stages:



### **1. Requirement Gathering**

* In this stage, the project team identifies and documents the needs and expectations of various stakeholders, including clients, users, and subject matter experts.
* It involves defining the [**Project's Scope**](https://www.geeksforgeeks.org/what-is-scope-in-project-management/), objectives, and requirements.
* Establishing a budget and schedule.
* Creating a project plan and allocating resources.

### **2. Design**

* Developing a high-level system architecture.
* Creating detailed specifications, which include data structures, algorithms, and interfaces.
* Planning for the software's user interface.

### **3. Development (Coding)**

* Writing the actual code for the software.
* Conducting unit testing to verify the functionality of individual components.

### **4. Testing**

This phase involves several types of testing:

* [**Integration Testing**](https://www.geeksforgeeks.org/software-engineering-integration-testing/)**:** Ensuring that different components work together.
* [**System Testing**](https://www.geeksforgeeks.org/system-testing/)**:** Testing the entire system as a whole.
* [**User Acceptance Testing**](https://www.geeksforgeeks.org/user-acceptance-testing-uat/): Confirming that the software meets user requirements.
* [**Performance Testing**](https://www.geeksforgeeks.org/performance-testing-software-testing/)**:** Assessing the system's speed, scalability, and stability.

### **5. Deployment**

* Deploying the software to a production environment.
* Put the software into the real world where people can use it.
* Make sure it works smoothly in the real world.
* Providing training and support for end-users.

### **6. Review (Maintenance)**

* Addressing and resolving any issues that may arise after deployment.
* Releasing updates and patches to enhance the software and address problems.

## 

## 

## **Benefits of Agile development methodology**

* **Flexibility and Adaptability**
* **Improved Collaboration**
* **Faster Delivery**
* **Enhanced Quality and Customer Satisfaction**
* **Iterative Development**
* **Transparency**
* **Quality Assurance**