# Origin

In rugby, a scrummage (or *scrum* for short) was the method used to restart play in a match after a foul. Visually, it's eight players from each team packed together with heads down, all trying to take possession of the ball. So, not exactly the poster child for project management, but with a little imagination, it makes sense.

On a project team, the goal is to get the project done. Historically, **in traditional methods, we plan and design the whole project at the beginning and stick to that plan with no variation**. However, modern project work is completely unpredictable. It's impossible to know at the beginning exactly how a project will unfold and how to best meet its unique challenges.

The Agile Project Management comes out an idea to adapt in real time to the changing circumstances, and this is where the rugby team comes in. In rugby, the object is to move the ball down the field, one possession at a time, so why couldn't projects do the same thing? Why not change the focus from just winning the whole game to winning each and every milestone and deliverable?

The scrum reflects this approach – **breaking the deliverables and milestones into smaller pieces and gets the whole team together to focus on just that one goal until it's done**.

# Agile Overview

## Manifesto

* **Individuals and interactions** over processes and tools
* **Working software** over comprehensive documentation
* **Customer collaboration** over contract negotiation
* **Responding to change** over following a plan

## Principles

1. Our highest priority is to **satisfy the customer through early and continuous delivery** of valuable software.

*We not just show up at the beginning to describe what they want, then show up at the end and tell us how we missed the mark. We need direct, ongoing interaction.*

1. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
2. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
3. Business people and developers must work together daily throughout the project.
4. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
5. The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
6. **Working software** is the primary measure of progress.

*We no longer want to measure success using milestones and phases. We want working software to tell everyone how we're doing, and we want to hear feedback the whole time.*

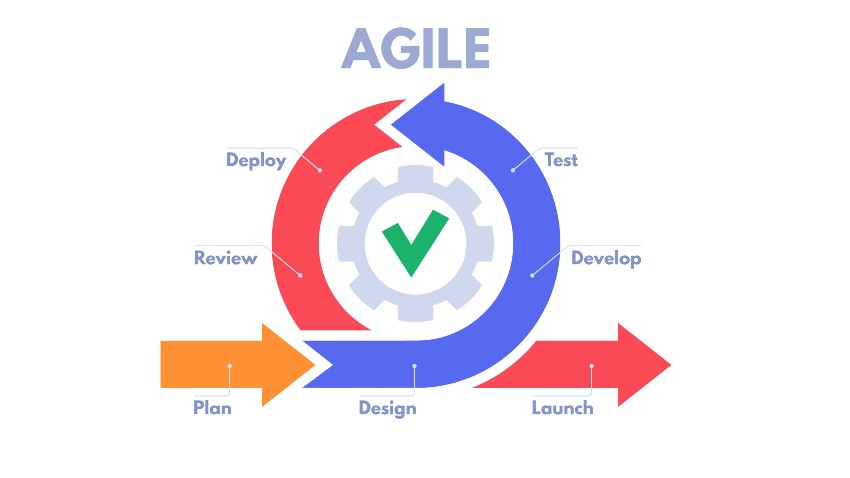
1. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
2. Continuous attention to technical excellence and good design enhances agility.
3. **Simplicity** – the art of maximizing the amount of work not done – is essential.
4. The best architectures, requirements, and designs emerge from **self-organizing teams**.

*Allow teams to self-organize. They'll do a much better job doing the design and tests from the ground level than any upfront plan could do.*

1. At regular intervals, the team reflects on how to become more effective, then tunes and **adjusts its behavior accordingly**.

## Workflow

A spiral model:



## Agile Frameworks

Scrum, Kanban, Scrumban, EXtreme Programming (XP), Crystal, Dynamic Systems Development Method (DSDM), Feature Driven Development (FDD), Adaptive Software Development (ASD)