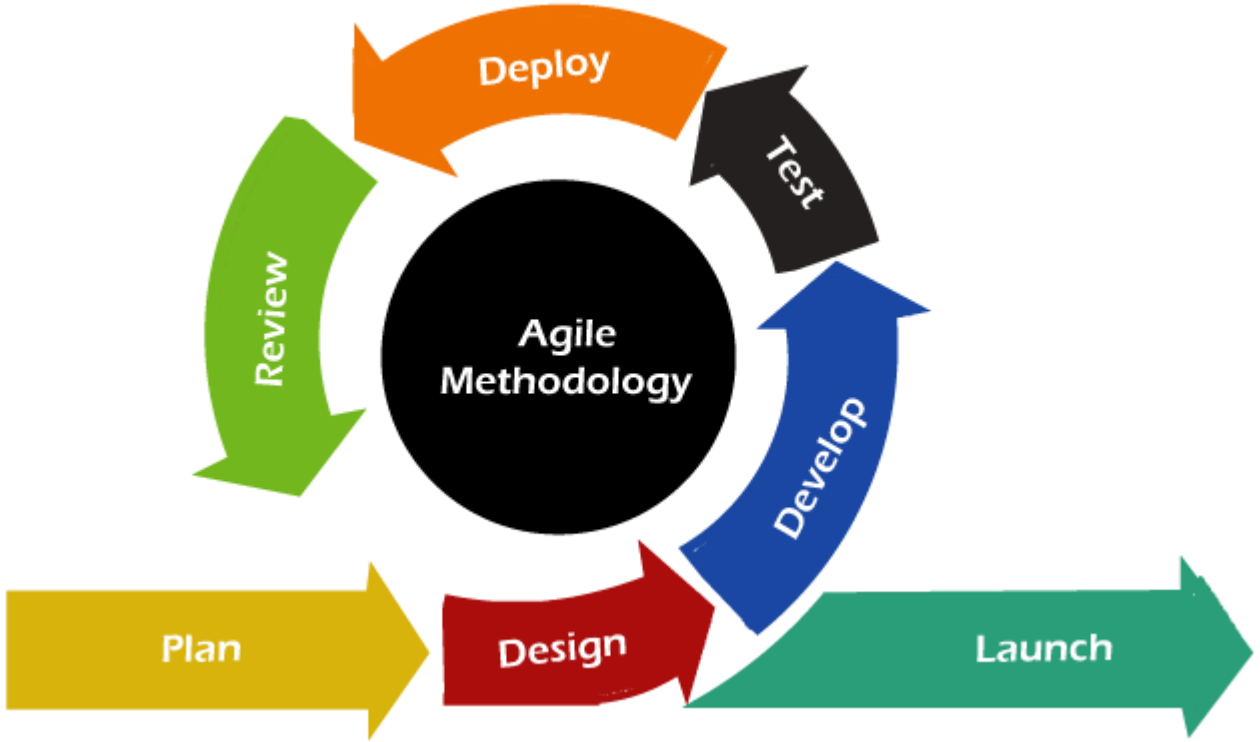
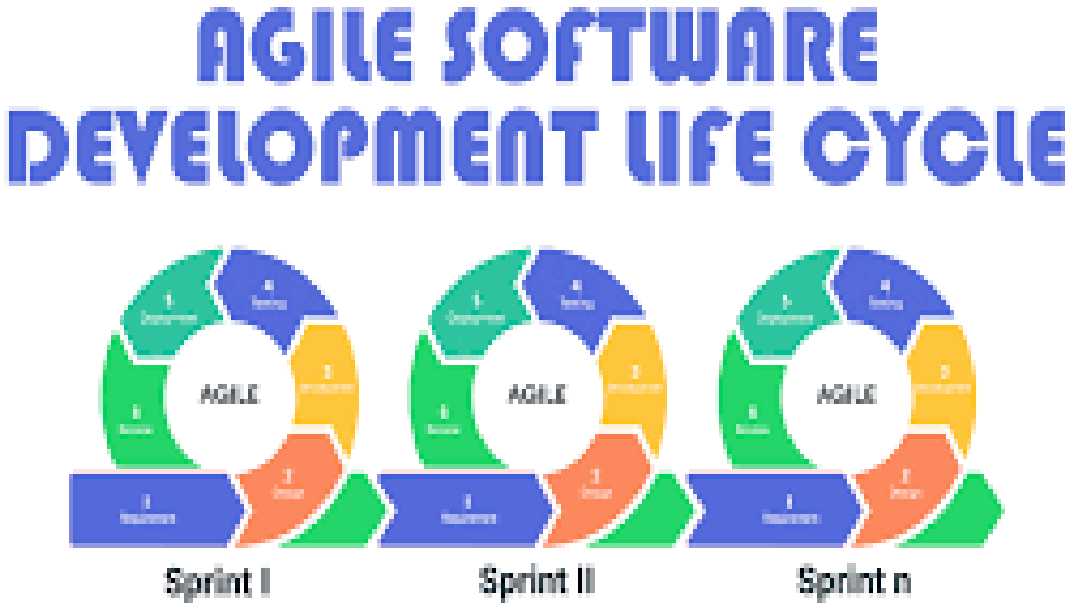
Agile Methodology

What is Agile?

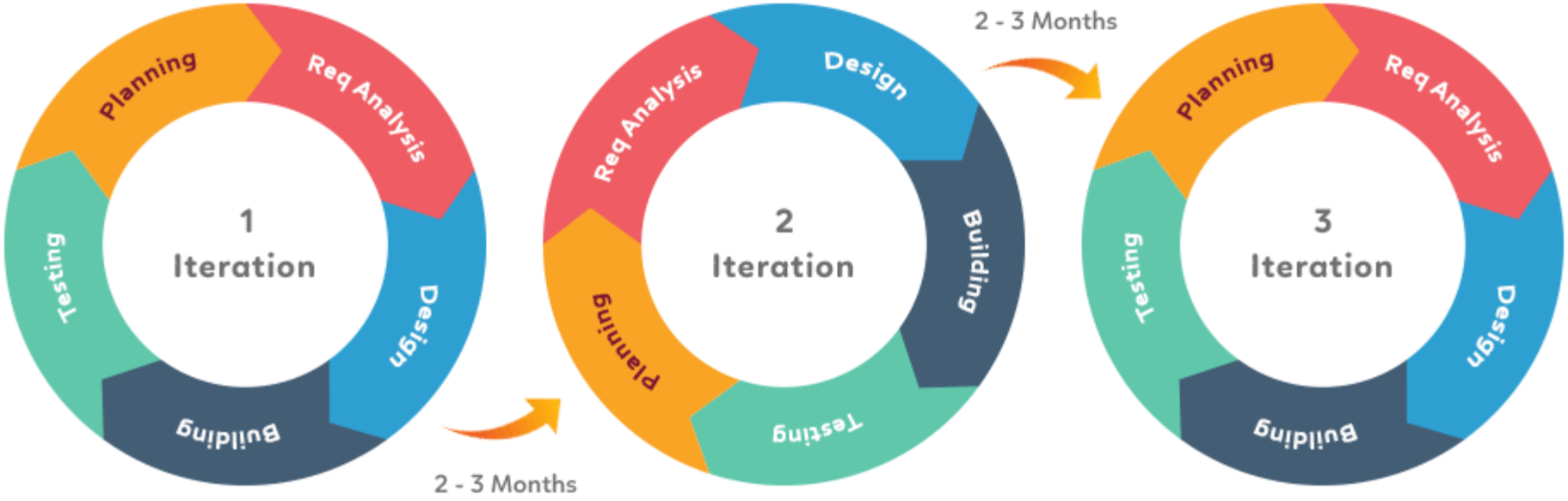
Agile is an approach to software development that seeks the **continuous delivery of working software** created in rapid iterations.



In practical terms, agile software development methodologies are all about delivering small pieces of working software quickly to improve customer satisfaction.



Usually, agile software development consists of small, self-organizing teams of software developers and business representatives regularly meeting in-person throughout the software development life cycle.



Agile favors a lightweight approach to software documentation and accepts willingly —rather than resists—changes at any stage of the life cycle.

As simple, the agile principles are all about the iterative and incremental approach with self-organizing and functional teams.

Instead of attempting to tackle the whole problem in one massive cycle, work gets split into manageable chunks, often supposed to be delivered in 2-week cycles.

At the end of the period, the team shows progress, allowing early visibility on what is being developed.

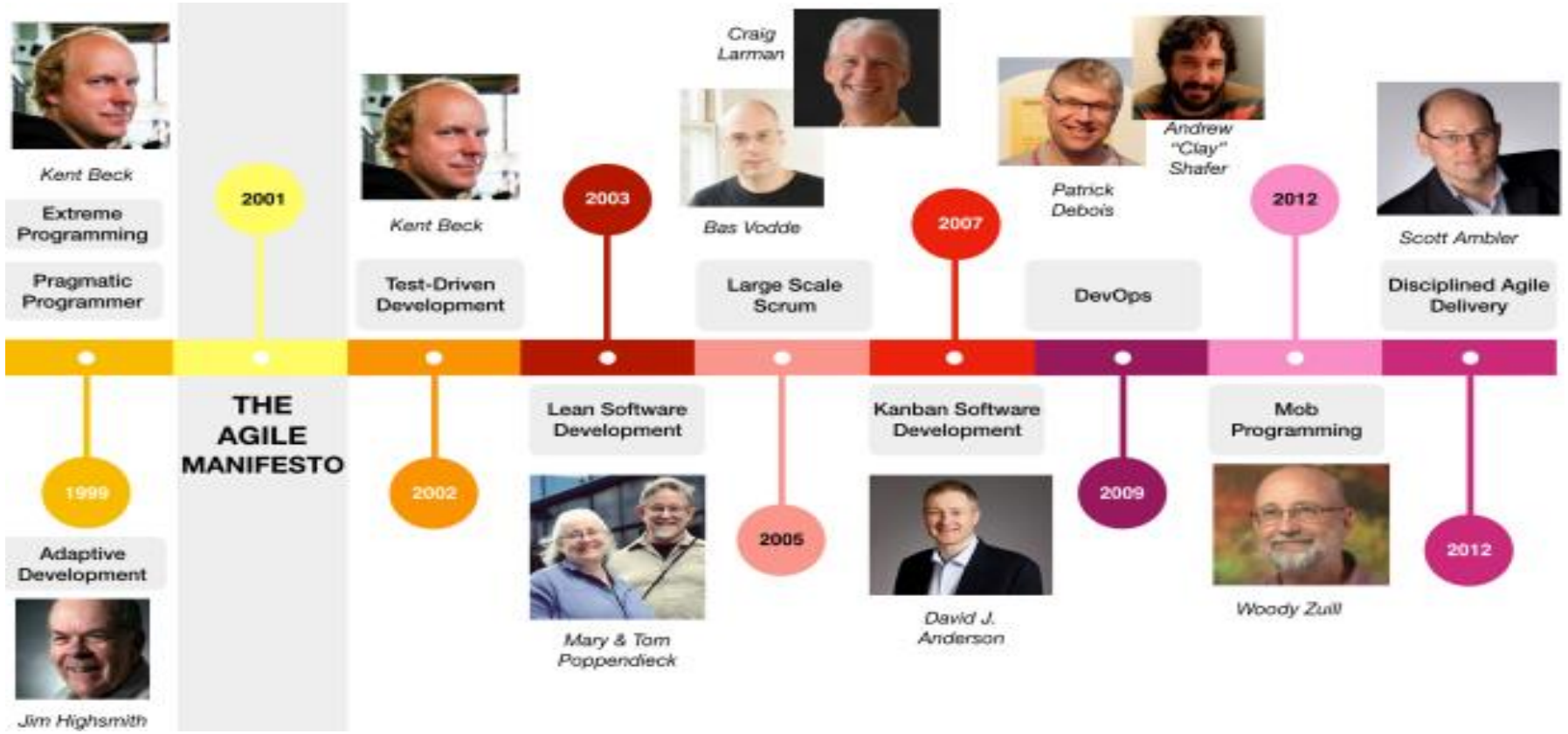
Changes can be made on the spot and requirements revised accordingly.

Agile Manifesto

It is a brief document built on 4 key values and 12 principles for agile software development.

Who Created the Agile Manifesto?

A group of software practitioners from various backgrounds gathered to form the Agile Alliance who created The Agile Manifesto. Here is the list of people, who signed the original Agile Manifesto back in 2001:




Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler
James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
2001 February + 'The Lodge' at Snowbird Ski Resort + 17 Thinkers = Agile Manifesto
Brian Marick
Bob Martin
Stephen Mellor
Jeff Sutherland
Ken Schwaber
Dave Thomas

The 4 Agile Values

1

Individuals and Interactions

over

Processes and Tools

2

Working Software

over

Comprehensive Documentation

3

Customer Collaboration

over

Contract Negotiation

4

Responding to Change

over

Following a Plan


AGILE MANIFESTO
INDIVIDUALS & INTERACTIONS
OVER
PROCESSES & TOOLS
WORKING SOFTWARE
OVER
COMPREHENSIVE DOCUMENTATION
CUSTOMER COLLABORATION
OVER
CONTRACT NEGOTIATION
RESPONDING TO CHANGE
OVER
FOLLOWING A PLAN

12 Principles of Agile

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

02 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

03 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

04 Business people and developers must work together daily throughout the project.

05 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

06 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

07 Working software is the primary measure of progress.

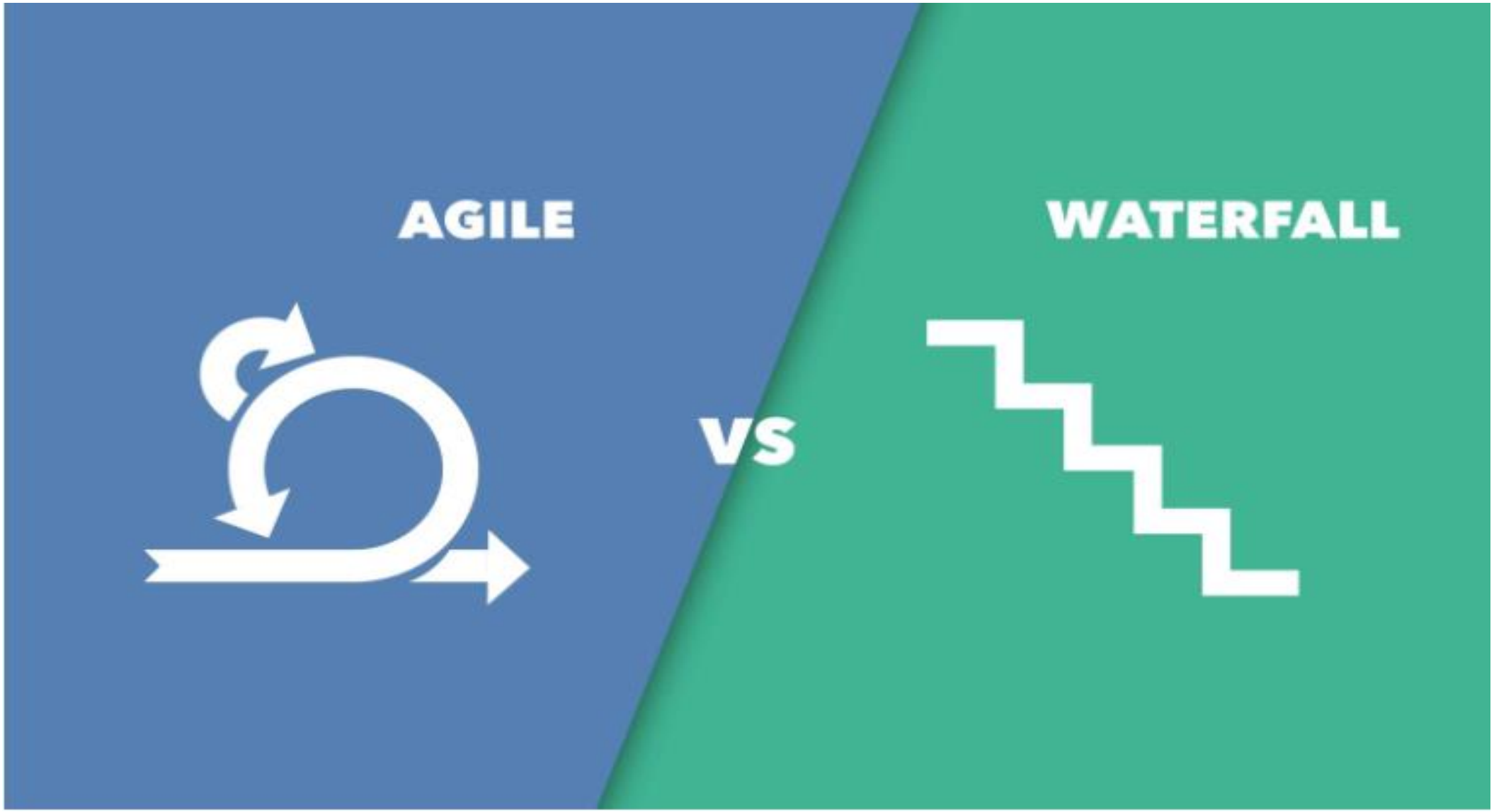
08 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

09 Continuous attention to technical excellence and good design enhances agility.

10 Simplicity—the art of maximizing the amount of work not done—is essential.

11 The best architectures, requirements, and designs emerge from self-organizing teams.

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


AGILE
VS
WATERFALL

Agile
Project Management

VS

Waterfall
Project Management

Project Scope

Changes can be made well in advance with time and within the budget. Agile works well even if the Scope is not defined in advance

Waterfall performs well when the Scope is well known in advance and contract terms limit changes

Team

Agile intends small or mid-sized dedicated teams with high coordination

Waterfall involves large teams. It decreases coordination among team members

Customers	
 Agile allows customers to be available throughout the project	 Waterfall requires customers to be available only at milestones
Feature Prioritization	
 Features are prioritized and issues are resolved according to priorities. It increases funding efficiency and evades complete failures	 Features are not prioritized. It leads to either complete success or complete failure

Feasibility

Agile project management looks better when it is feasible

NOT

Waterfall project management does not depend on its feasibility

Funding

Agile works extremely well by increasing funding efficiency

Waterfall works well by reducing fixed funding through up-front contracts

WATERFALL

- The Waterfall development process is divided into distinct stages.
- Software development is completed as one single project.
- The method is a sequential design process.
- This is a structured software development approach so most times it can be quite rigid.
- There is no scope for changing the requirements once the project development starts.

AGILE

- Agile breaks the project development lifecycle into sprints.
- Agile can be considered as a set of many different projects.
- The methodology follows an incremental approach.
- Flexibility is what makes Agile different.
- The approach is quite flexible that allows changes in the project development requirements even if the initial planning has been completed.

WATERFALL	AGILE
Waterfall demonstrates a project mindset and places its focus completely on the project accomplishing.	Agile is a mindset where the software product satisfies the needs of the end clients and changes itself as per the client's demands.
All the project development phases are completed once.	The method follows an iterative approach. Different phases may appear more than once.
The testing plan is rarely discussed during the test phase.	The testing plan is reviewed after each sprint.
This approach looks ideal for projects that have definite requirements and changes not at all expected.	According to Agile, the requirements are expected to change and evolve.
The testing phase comes after the build phase.	In Agile, testing is performed concurrently with software development.

WATERFALL	AGILE
Due to the getting risk agreement at the beginning of the process, Waterfall reduces risks in the firm fixed-price contracts.	Agile works exceptionally well with time and materials or non-fixed funding.
A detailed description needs to implement Waterfall.	You may change the description of project details anytime during the SDLC process.
The process is always straightforward so, the project manager plays an essential role during every SDLC stage.	Agile team members are interchangeable, so they work faster. There is no need for project managers as the projects are managed by the entire team.
Team coordination/synchronization looks rather limited.	The method implies small but dedicated teams with a high degree of coordination and synchronization.
Business analysis prepares requirements before the project beginning.	A product owner with a team prepares requirements every day during a project.

Making the Choice Between Agile and Waterfall

Your ultimate choice between these two methodologies depends much on several factors.

Waterfall may be the best choice if there is no (or limited) access to a customer to provide constant feedback. It will also suit projects with a dispersed team, fixed budget, and scope.

Agile looks preferable for more complex and larger projects, where there is easy access to customer feedback. Its flexibility makes the technique more suitable for projects with constantly changing requirements.

Agile Umbrella

LIGHTWEIGHT APPROACHES

FULLER APPROACHES

AGILE

LEAN SOFTWARE DEVELOPMENT

EXTREME PROGRAMMING

CONTINUOUS INTEGRATION

CONTINUOUS DELIVERY

KANBAN (PROCESS + METHOD)

FEATURE DRIVEN DEVELOPMENT

TEST DRIVEN DEVELOPMENT

SCRUM

CRYSTAL CLEAR

SCRUM-OF-SCRUMS

SCRUM AT SCALE

LARGE-SCALE SCRUM

DYNAMIC SYTEMS DEVELOPMENT METHOD

SCALED AGILE FRAMEWORK

DISCIPLINED AGILE DELIVERY

AGILE PROJECT MANAGEMENT

OPEN UNIFIED PROCESS

AGILE UNIFIED PROCESS

Design @kasiashilina

Bee Agile @ainaalive


Enjoyable
Higher Revenue
Increased Productivity
Happy Stakeholder
Right Product
Quality
Fun
Benefits Of AGILE
Visibility
guidance
Time-to-Market

Any Questions