introduction to Agile methods

Soumia LAYACHI

December 2025

1 Why Agile Methods?

Before Agile, teams mainly used traditional models such as the V-model or the waterfall model, where:

- requirements are defined at the beginning,
- development takes place afterwards,
- testing is performed at the end.

Problem: if requirements change (which happens frequently), much of the work must be redone.

Simple Example

A team must develop a *Library Management* application. In the V-model:

- requirements are fixed from the start,
- if the library changes its needs (e.g., adding digital books), it becomes difficult.

Agile methods allow the project to adapt continuously.

2 Definition of Agile Methods

Agile methods are a set of practices that allow teams to:

- develop rapidly,
- adapt to changes,
- involve the customer continuously,
- deliver a progressive product.

3 The Agile Principles

Agile has 12 official principles; here are the most essential:

- deliver quickly and frequently,
- embrace change at any time,
- maintain continuous collaboration with the customer,
- prioritize direct communication (short meetings),
- measure progress mainly by functional software,
- sustain a constant development rhythm,
- continuously improve processes and teamwork.

4 The Main Agile Methods

4.1 SCRUM

4.1.1 Roles in Scrum

Product Owner (PO) Represents the customer and defines what must be developed.

Scrum Master (SM) Ensures that the Scrum method is followed. He is not a manager; he supports and facilitates the team.

Development Team Codes, designs, and tests the product. Autonomous and multidisciplinary.

4.1.2 Scrum Events

Sprint A fixed work cycle (1 to 4 weeks). At the end, a functional product increment is delivered.

Daily Scrum A 15-minute daily meeting where each member answers:

1. What did I do yesterday?
2. What will I do today?
3. What obstacles did I encounter?

Sprint Review A meeting to present the work done:

- demonstration of the increment,
- feedback from the Product Owner,
- preparation for the next Sprint.

Sprint Retrospective An internal team meeting to:

- analyze what worked well,
- identify what should be improved,
- decide on an improvement action.

4.1.3 Scrum Artifacts

Product Backlog A prioritized list of all product features. Contains User Stories. Managed by the PO.

Sprint Backlog List of Product Backlog Items selected for the Sprint + tasks.

Burndown Chart A graph showing:

- remaining work,
- Sprint days,

used to check progress.

4.2 Example of Scrum: Student Management Application

Sprint 1 (2 weeks):

- Login page
- Create account
- Simple student interface

Sprint 2:

- Teacher grade import
- Display grades

Each Sprint ends with a working version.

4.3 Extreme Programming (XP)

4.3.1 Introduction

XP is an Agile method oriented toward software development. It emphasizes:

- code quality,
- quick reactions to change,
- strong collaboration with the customer.

4.3.2 Main Objectives

- improve software quality,
- reduce risks linked to late changes,
- encourage continuous communication,
- deliver working increments frequently.

4.3.3 Core Values of XP

1. Communication
2. Simplicity
3. Feedback
4. Courage
5. Respect

4.3.4 Key XP Practices

Test-Driven Development (TDD) Write tests before code: *Red → Green → Refactor*.

Pair Programming Two developers work together:

- one writes code,
- the other reviews and suggests improvements.

Continuous Integration Code is integrated and tested several times a day.

Refactoring Continuous improvement of code structure.

Collective Code Ownership Any team member can modify any part of the code.

Coding Standards All developers follow the same conventions.

Frequent Deliveries Small increments delivered every 1–3 weeks.

On-site Customer A customer representative is available permanently.

4.3.5 XP Development Cycle

1. Listen
2. Plan (User Stories, priorities)
3. Design / Code (TDD, pair programming, refactoring)
4. Test (unit + acceptance)
5. Deliver
6. Improve (retrospective)

4.3.6 Simple XP Example: Authentication Module

- User Story: "As a user, I want to log in with my email and password."
- TDD: write test `TestLogin`.
- Minimal code to pass the test.
- Refactor.
- Deliver.
- Feedback from customer.

4.4 Kanban

Visual method based on a board with 3 columns:

To Do	In Progress	Done

Work in Progress is limited to avoid overload.

Example: Computer Club Website

- Create homepage → In Progress
- Add gallery → To Do
- Add contact form → Done

5 Agile Project Lifecycle

1. General expression of need
   Example: “An app to manage student registrations.”
2. Product Backlog creation
3. Sprint Planning (1–4 weeks)
4. Development (small increments)
5. Daily Scrum (15 minutes)
6. Sprint Delivery: present working version
7. Retrospective: analyze improvements

6 Conclusion

Agile methods enable:

- flexible project management,
- better team communication,
- fast and continuous delivery,
- reduced risks.

SCRUM is today the most widespread Agile method in the industry.