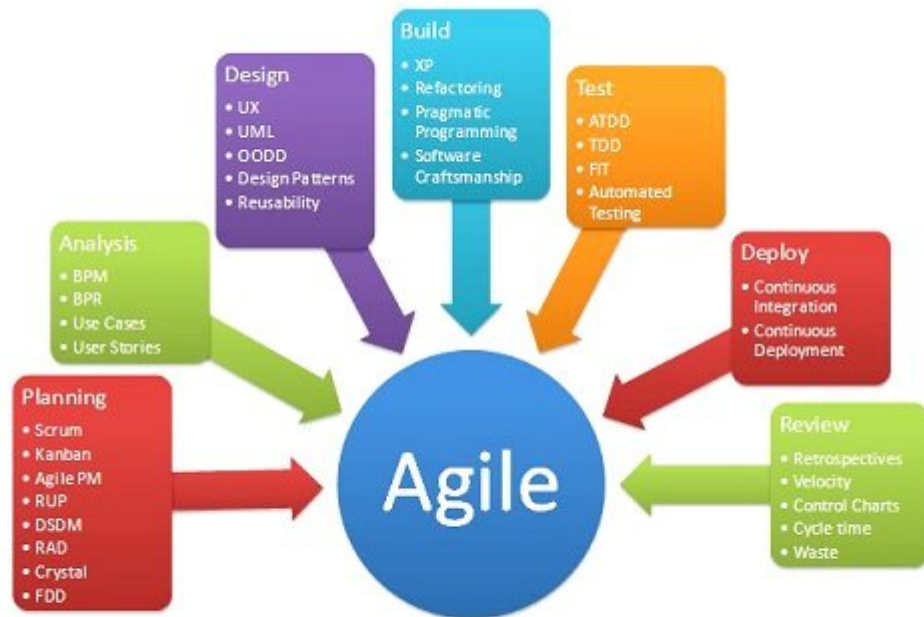


[illegible][illegible][illegible]

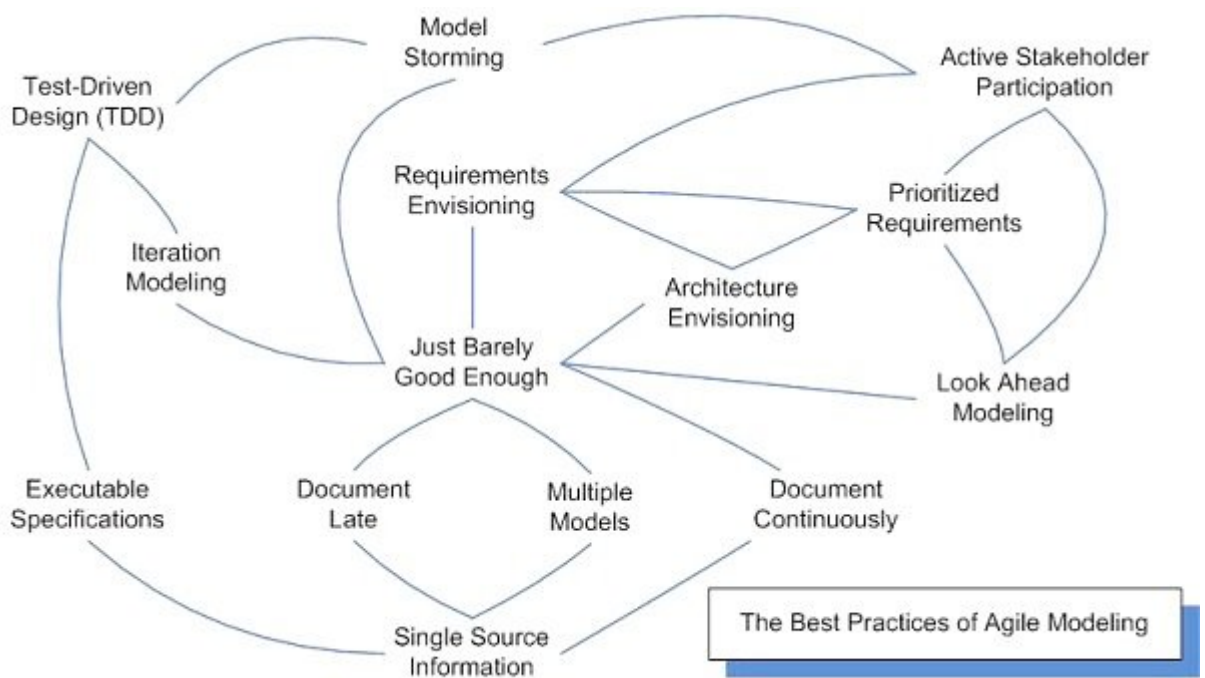
(Extreme programming) □□□□ □□□□□□□□□□

90 1994 2001 90 1 (1)



XP 프로그래밍 방법

이 책은 1996년에 이 방법론을 창시한 스티브 크로커 (Steve Krug) (EXtreme Programming 방법론) XP 이 방법론을 창시한 스티브 크로커 (Steve Krug)가 저술했다. 이 책은 XP 방법론의 핵심 원칙과 실천 방법을 상세히 설명하고 있다. XP는 다른 애자일 방법론과 달리, 고객과의 긴밀한 협력과 빠른 피드백을 강조한다. 이 책은 XP의 12가지 원칙을 소개하고, 각 원칙이 어떻게 프로젝트에 적용되는지 설명한다. 이 책은 XP의 성공적인 적용을 위한 실용적인 조언을 제공한다. 이 책은 XP의 철학과 가치를 이해하고, 이를 프로젝트에 적용하는 데 도움이 될 것이다. (2판)



Pair Programming

Pair Programming (PP) is a software development practice where two programmers work together at a single computer. One programmer, the **Driver**, writes the code, while the other, the **Observer**, reviews the code as it is written, catching errors and suggesting improvements. This practice is often used in agile development frameworks like Scrum and XP. The Observer can also take over the role of the Driver when needed, allowing for continuous learning and knowledge sharing. This collaborative approach can lead to higher code quality, faster problem-solving, and reduced errors.

Observer



V-model

[V-model diagram showing the relationship between development and testing phases.](#)

Code Review

Code Review is a systematic process of checking and evaluating source code to identify and correct errors, improve code quality, and share knowledge. It involves a developer (the author) presenting their code to one or more reviewers. The reviewers provide feedback on the code's structure, logic, and adherence to coding standards. This process can be done informally or through formal checklists. Code reviews are a critical part of software development, helping to catch bugs early and ensure the code is maintainable and reliable.

Code Review

XP (eXtreme Programming) is an agile development framework that emphasizes frequent releases, customer collaboration, and continuous communication. It includes practices like Pair Programming, Test-Driven Development (TDD), and frequent releases. XP is designed to be lightweight and adaptable, focusing on the core principles of agile development.

XP

XP is a software development methodology that emphasizes frequent releases, customer collaboration, and continuous communication. It includes practices like Pair Programming, Test-Driven Development (TDD), and frequent releases. XP is designed to be lightweight and adaptable, focusing on the core principles of agile development.

XP (eXtreme Programming)

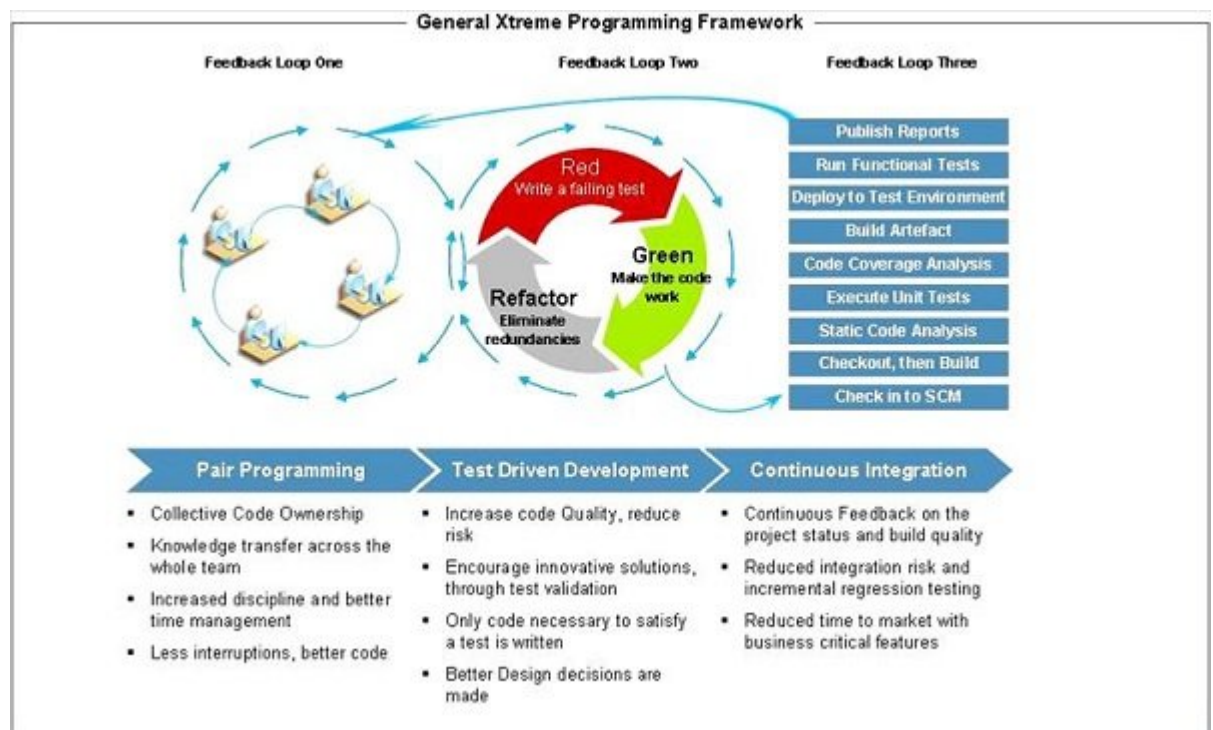


XP (eXtreme Programming) is a software development methodology.

XP (eXtreme Programming) is a software development methodology.

XP (eXtreme Programming) is a software development methodology.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. (3 points). XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.



XP (eXtreme Programming) is a software development methodology.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **User Stories** are a type of requirement that is written from the perspective of the user. They are a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Iteration Plan** is a document that outlines the goals and tasks for a specific iteration. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Task** is a specific piece of work that needs to be completed. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Task** is a specific piece of work that needs to be completed. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Metaphore** is a technique used to describe a complex concept in a simple and relatable way. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Metaphore** is a technique used to describe a complex concept in a simple and relatable way. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.

XP (eXtreme Programming) is a software development methodology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Spike** is a technique used to explore a complex problem or to gather information about a new technology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. **Spike** is a technique used to explore a complex problem or to gather information about a new technology. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development. It is a lightweight, iterative, and collaborative approach to software development.

XP 12 個の原則は、開発者と顧客との密なコミュニケーションを促進し、高品質なソフトウェアを迅速に提供することを目的としています。

Planning Game: 開発者と顧客が共同で、短期間の計画を立てるプロセスです。顧客は常にチームと一緒に作業し、要求の変更を迅速に対応します。

Simple Design: 単純な設計を優先し、複雑さを避ける原則です。必要に応じて機能を追加しますが、最初から完璧な設計を追求しません。

Testing: テストは開発プロセスの一部であり、コードを書く前にテストケースを作成します。自動化されたテストを使用します。

Refactoring: コードの品質を向上させるために、頻りにコードのリファクタリングを行います。機能の追加とコードの改善は同時に進みます。

Pair Programming: 2人の開発者が1つの作業を行うことで、品質を向上させ、知識を共有します。

Collective Ownership: コードはチーム全体で所有し、誰でも変更できるという原則です。

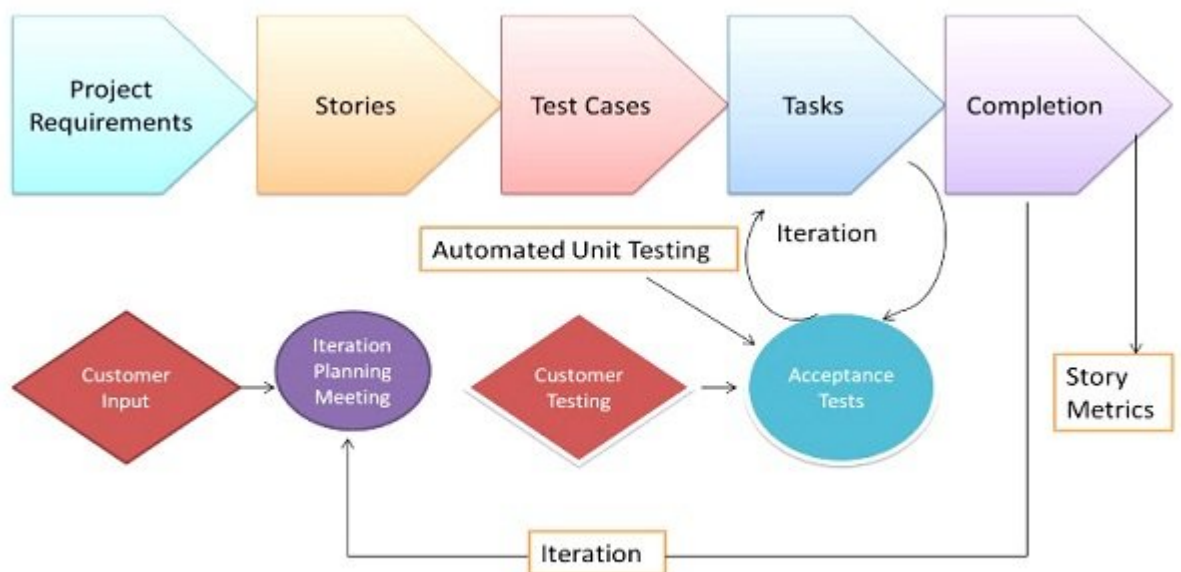
Continuous Integration: 頻りにコードを統合し、ビルドとテストを実行することで、問題を早期に発見します。

40 Hour Week: 週40時間の作業を推奨し、過度な労働を防止し、持続可能な開発環境を築きます。

On-Site Customer: 顧客は開発チームと一緒に作業し、要求の変更を迅速に対応します。

Coding Standards: コーディング規約を定め、コードの品質を向上させ、メンテナンス性を高めます。

XPの原則は、開発者と顧客との密なコミュニケーションを促進し、高品質なソフトウェアを迅速に提供することを目的としています。



Extreme Programming (XP)

□ □ □ □ □ □ □ □

[illegible]

•

A horizontal number line with arrows at both ends. It is marked with numbers from 0 to 100 in increments of 10: 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100. There are vertical tick marks at each of these numbers.

•

[agilealliance](#)

neomobile

cis.upenn

cs.usfca

[cis.upenn](http://cis.upenn.edu)

[extremeprogramming](#)

java-bugs

guru99

• □ □ □ □ □ □ □ □

[illegible]

• □ □ □ □ □ □ □ □ □ □

11:20 - 29/08/1397

•

[Scrum](#) - [Kanban](#) - [Lean](#) - [Agile](#) - [DevOps](#) - [XP](#) - [TDD](#) - [BDD](#) - [CI/CD](#) - [Git](#) - [Docker](#) - [Kubernetes](#)

11

<https://www.shabakeh-mag.com/workshop/programming/13808/%DA%A9%DB%8C%D9%81%D:>