

Practices of eXtreme Programming

by Randy Ballew

Table of contents

1 Practices of eXtreme Programming.....	2
1.1 Whole Team.....	2
1.2 Planning Game.....	2
1.3 Customer Tests.....	2
1.4 Simple Design.....	2
1.5 Pair Programming.....	2
1.6 Test-Driven Development.....	2
1.7 Design Improvement.....	2
1.8 Continuous Integration.....	3
1.9 Collective Code Ownership.....	3
1.10 Coding Standard.....	3
1.11 Metaphor.....	3
1.12 Sustainable Pace.....	3

1. Practices of eXtreme Programming

Summarized from [BECK] XP explained.

1.1. Whole Team

All the contributors to an XP project -- developers, business analysts, testers, etc. -- work together in the same open space, members of one team. The walls of this space are littered with big visible charts and other evidences of their progress.

If geography and facilities make this impossible, the team should build a virtual open space, and paper its virtual walls with artifacts of progress.

1.2. Planning Game

Planning is continuous. Every two weeks, for the next two weeks, developers estimate the cost of the candidate features, and customers select those features to be implemented based upon cost and business value.

1.3. Customer Tests

As part of selecting each desired feature, the customers define automated acceptance tests to show that the feature is working.

1.4. Simple Design

The team keeps the design exactly suited for the current functionality of the system. It passes all the tests, contains no duplication, expresses everything the authors want expressed, and contains as little code as possible.

1.5. Pair Programming

All production software is built by two programmers, sitting side by side, at the same machine.

1.6. Test-Driven Development

The programmers work in very short cycles, adding a failing test, then making it work.

1.7. Design Improvement

Don't let the sun set on bad code. Keep the code as clean and expressive as possible. Refactor ruthlessly, and often.

1.8. Continuous Integration

The team keeps the system fully integrated at all times.

1.9. Collective Code Ownership

Any pair of programmers can improve any code at any time.

1.10. Coding Standard

All the code in the system looks as if it was written by a single -- very competent -- individual.

1.11. Metaphor

The team develops a common vision of how the system works.

1.12. Sustainable Pace

The team is in it for the long term. They work hard, at a pace that can be sustained indefinitely. They conserve their energy, treating the project as a marathon rather than a sprint.