Developer Essentials Training from To Be Agile

This immersive, 40-hour training program contains a set of two courses that provides the skills needed to become a successful software developer and a valuable member of a development team. It can be delivered online or in-person in half-day sessions or full-day sessions.

In our first training course, Agile Analysis and Design Patterns (AADP), we will explore the principles of Agile software development and how they support faster and simpler software development. We'll then take a deep dive into Agile design and discover a core set of design patterns that every developer should be familiar with, along with essential design principles and practices.

You'll be taken on a guided tour through essential developer practices, such as story writing, team planning, pair programming, and test-driven development. You'll learn how to discover patterns in problems and implement designs as needed. You'll also explore the principles behind the practices so you understand how to use them to make the best design choices, and you'll gain a powerful framework for encapsulating and abstracting virtually any problem for maximum flexibility without over-complicating the solution. By exploring the secrets of high-performing, cross-functional development teams you'll gain a shared design vocabulary for dramatically improving inter-team communication that can be applied equally well to new development as to maintaining or extending existing systems.

The training program concludes with the Hands-On: Extreme Programming Practices (HEPP) course, which provides sixteen hours of hands-on programming labs and four hours of additional advanced instruction. In this advanced, hands-on class, you'll learn how to write higher-quality code more quickly and with fewer defects using practices from Extreme Programming (XP) that include test-first development, refactoring, and emergent design, as well as use a variety of Agile problem-solving techniques and approaches to solve real-world problems.

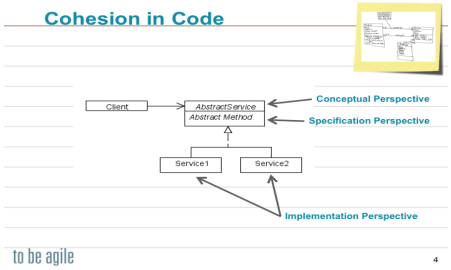
You'll then put theory into practice and apply your new skills by building an application's core using the expert-level techniques you've learned for rapidly writing quality software. By the end of this training program, you'll be armed with several new, effective tools and techniques for Scrum development that will make your software more robust, manageable, and easier to extend.

Who Should Take These Courses?

These software developer training courses are for all team members and have the greatest impact when the entire team attends. These courses will benefit Architects, DBAs, Designers, Developers, Development Managers, Directors, Product Managers, Programmers, QA Engineers, Software Engineers, Technical Analysts, Technical Leads, Technical Writers, and Testers. Familiarity with basic Object-Oriented (OO) concepts and terminology is recommended. The Advanced Developer Essentials course is for all technical team members and includes 20 hours of programming labs in Java or C Sharp.

Courses Benefits

Completing this training program will give you the knowledge and experience of using Extreme Programming to build quality software and enable you to rapidly:



- Write stories and build features in iterations
- Estimate development tasks more accurately
- Master test-first development to drive design
- Efficiently use TDD's red-green-refactor cycle
- Work effectively to refactor legacy code
- Diagnose and fix pathologies of poor code
- Exercise techniques to test untestable code
- Collaborate successfully with pairing and mobbing
- Employ acceptance tests to specify and document stories
- Avoid upfront overdesign and practice just-in-time development
- Distinguish between twelve design patterns by what they encapsulate
- Define a strategy for continuously integrating software as it is built
- Write software that supports an iterative process without excessive rework
- Support collaborative code ownership and embrace a common aesthetic
- Refactor to patterns and emerge designs in iterative development
- Share a common vocabulary for evaluating and communicating designs
- Implement techniques for recognizing and managing technical debt
- Quantify software qualities that make code easier to maintain and extend
- Recognize how test-driven development informs design decisions
- Appreciate the value of adopting shared coding standards
- And much more…

Agenda

Agile Analysis and Design Patterns (AADP): Covers essential elements of Agile for software developers: Agile principles, values, and framework; artifacts, story writing, collaboration, estimation and planning, coaching, and facilitation. We explore Agile principles and patterns: approaches to design, problem-solving techniques, seeing patterns by what they encapsulate, discovering patterns in problems, emergent design, and refactoring to patterns. Hands-on exercises include story writing, defining acceptance tests, and an in-depth design exercise and review. See the full course description for the Agile Analysis and Design Patterns training.

Hands-On: Extreme Programming Practices (HEPP): Covers essential practices from XP: continuous integration, pair programming, coding standards, test-first development (TDD), red-green refactoring, using TDD to inform design, code qualities, discovering design patterns, conducting code reviews, essential Scrum developer practices, writing testable code, advanced testing techniques; refactoring legacy code; emerging solutions. Includes twenty hours of hands-

on programming exercises in Java or C Sharp. See the full Hands-On: Extreme Programming Practices training course description.

Your Instructor, David Bernstein

My continuing passion for software design and construction has led me to train more than 10,000 professional software developers for clients that have included Fortune 500 firms such as Microsoft, IBM, Yahoo!, Boeing, AT&T, Sprint, Medtronic, SunGard, State Farm, MetLife, and Weyerhaeuser. As a longtime IBM consultant, I trained software engineers around the globe, giving them the skills to write the next generation of applications and operating system software while earning one of the highest satisfaction ratings in the history of IBM education. Since 2006, I've devoted my consulting practice to providing organizations with technical training and coaching for software developers and teams transitioning to Agile and Scrum.

Praise for David's Training

> "Have you ever felt the fear of starting a project, not knowing what design to start? Ever felt the fear of making a small change and having everything crash down? Take this class and learn a better way to live and develop without fear."
> –D.J. Hagberg, Software Architect

> "Take the course. It will change how you approach software development."
> –Eric Huber, Senior Software Developer

> "One of the most interesting and interactive trainings that I have ever taken part in."
> –Brandon Whitaker, Software Engineer

> "Go to it! This course takes Scrum out of the conceptual and puts it into the practical."
> —Kevin Hallquist, Software Design Engineer, Certified ScrumMaster

> "Take it; have your team take it together. Make time for this class."
> —Edward J. Newton, Software Development Manager, Certified ScrumMaster

Become a more proficient developer—**Register for the next session or contact me to arrange a private, onsite session for your team.** For more information visit http://ToBeAgile.com.