

# Certified Scrum Developer Essentials

This immersive, forty-hour training program contains a set of three comprehensive courses that provide the knowledge and skill to become a successful Scrum developer and a valuable member of a Scrum team.

Our first course is devoted to an [Introduction to Scrum for Software Developers](#), in which you will explore the basics of the Scrum Framework and how it supports faster and simpler software development. The second course covers [Design Pattern Developer Essentials](#), where you will discover a core set of design patterns that every developer should know, and their underlying forces. This training program concludes with the popular [Scrum Software Developer Essentials](#) course, in which you'll learn how to write higher-quality code more rapidly and with fewer defects using test-first development, refactoring, and a variety of Agile problem-solving techniques.

You'll be taken on a guided tour through essential practices from Scrum and XP, such as story writing, sprint planning, pair programming and test driven development, and you'll learn how to discover patterns in problems and to emerge designs as needed. You'll 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.

You'll learn the secrets of high-performance, cross-functional development teams, as well as a shared design vocabulary for dramatically improving inter-team communication that can be applied equally well to new development and to maintaining or extending existing systems. You'll then put theory into practice and apply your new skills by building the core of an application 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.

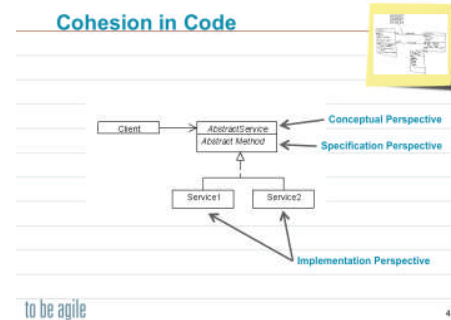
## Who Should Take This Course

This training is for all team members, and has the greatest impact when the entire team can attend. This set of courses will benefit architects; designers and developers; development and product managers; software engineers/programmers; testers and QA engineers; and technical analysts, leads, and writers. Familiarity with basic Object-Oriented (OO) concepts and terminology is recommended. Participants who successfully complete the programming exercises in the [Scrum Software Developer Essentials](#) training are eligible to become a [Certified Scrum Developer](#), and require the ability to write simple programs in Java using Eclipse or in C Sharp using Visual Studio.

## Course Benefits

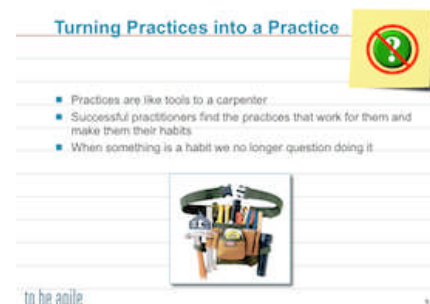
Completing this week-long training will give you the knowledge and experience of using Scrum to rapidly build quality software, and enable you to:

- Accurately estimate development tasks
- Write stories and build features in sprints
- Participate in each type of Scrum meeting
- Use test-first development to drive design
- Work effectively with legacy code
- Employ techniques that reduce defects
- Quickly implement effective design alternatives
- Collaborate more smoothly with other team members
- Understand the Scrum framework and its roles, artifacts and ceremonies
- Recognize the key differences between Scrum and traditional waterfall development
- Define an effective strategy for continuously integrating software as it is built
- Lead a Scrum team to plan, estimate, and prioritize their backlog
- Establish standards and practices for rapidly building quality software
- Avoid upfront overdesign and practice just-in-time development
- Identify and correct potential impediments to Scrum success
- Use object-oriented techniques to write more extensible software
- Support collaboration, pair programming, and cross-functional teams
- Write software that supports an iterative process without excessive rework
- Support collaborative code ownership and embrace a common aesthetic
- Implement better design alternatives and avoid upfront overdesign



## Agenda

**Introduction to Scrum for Software Developers** — Covers essential elements of Scrum for software developers: Scrum principles, values, and framework; artifacts; story writing; collaboration; estimation and planning; coaching and facilitation. Hands-on exercises include story writing, defining acceptance tests, estimation and sprint planning. See the full course description for [Introduction to Scrum for Software Developers](#).



**Design Pattern Developer Essentials** — Covers Agile principles and patterns: approaches to design; key principles; problem-solving techniques; seeing patterns by what they encapsulate; discovering patterns in problems; design exercise; what patterns teach us; emergent design and

refactoring to patterns. Includes an in-depth design exercise and debrief. See the full course description for [Design Pattern Essentials for Developers](#).

**Scrum Software Developer Essentials** — Covers essential practices from Scrum and XP: continuous integration; pair programming; coding standards; test-first development (TDD); red green refactor; using TDD to inform design, code qualities; OOAD revisited and revised; discovering design patterns; conducting code reviews; essential Scrum developer practices; writing testable code; advanced testing techniques; refactoring legacy code; emerging solutions. Includes six hands-on programming exercises in Java and C Sharp. See the full course description for [Scrum Software Developer Essentials](#).

## Your Instructor



David Bernstein's continuing passion for software design and construction has led him to train more than 7,500 developers in the last 23 years 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, David 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, he has devoted his consulting practice to providing organizations with training and coaching for software developers and teams transitioning to Agile and Scrum.

## Certification

The three courses of this forty-hour training (which together comprise Our Certified Scrum Developer Essentials Training Week) satisfy all the training requirements for becoming a Certified Scrum Developer (CSD) through the Scrum Alliance. These courses count for 40 Professional Development Units (PDUs). See <http://ToBeAgile.com/faq> for more details.

## 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

Become a more proficient developer—**Register for the next session or contact me for more information** at <http://ToBeAgile.com>.