Agile Design Pattern Essentials

This in-depth two-day training presents the same core set of design patterns as our one-day Design Pattern Developer Essentials along with key Agile principles, Scrum practices, and an expanded group design exercise to help you better understand iterative development.

You'll come to know patterns as more than just "reusable solutions to common problems within a given context" and recognize them as a collection of forces that help you penetrate deeper into problems, discovering elegantly simple solutions that make your software more robust and easier to maintain and extend. The expanded view of patterns you'll gain from this course will help you encapsulate and abstract virtually any problem with maximum flexibility and without over-complicating the solution. You'll learn to make better coding choices and master a shared vocabulary for talking about design that will dramatically improve inter-team communication.

By the end of this training, you'll be armed with several new, effective tools for solving design problems that will empower you to produce immediate improvements in the quality of the software you design and build.

Course Benefits

Completing this course will give you a deeper understanding of Design Pattern development in an Agile context, and enable you to:

- See how the Agile principles relate to development
- Use XP and Scrum practices in collaborative coding
- Explain what patterns are and know when to use them
- Employ Agile principles to create higher-quality code
- Use a common vocabulary for communicating designs
- Adopt simple methods to find patterns in problems
- Understand patterns by what they encapsulate
- Apply patterns just-in-time to avoid over-design
- Appreciate the value of shared coding standards
- Contribute to design reviews and evaluate others' designs
- Gain several techniques for doing effective analysis and design
- Master techniques for emerging designs in iterative development
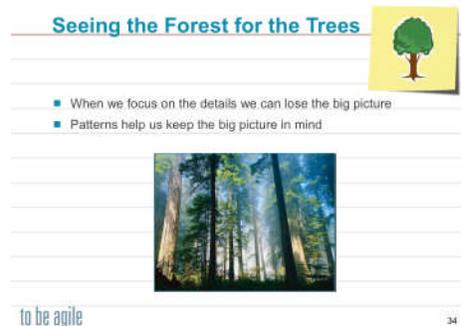
Who Should Take This Course

This training will benefit all team members including architects; business analysts; DBAs; designers and developers; development managers; directors; documentation specialists; operations and support staffers; product and project managers; software engineers/programmers; testers and QA engineers; and technical writers, analysts and leads.

Agenda

- **Day 1 - Analysis**
    - **Introduction**: Purpose and objectives
    - **Agile Principles**: What and why go Agile?
    - **Scrum Practices**: Scrum process, framework
    - **Paradigms and Wisdom**: Advice from GoF
    - **Principles and Perspectives**: Agile design
    - **Software Patterns**: Misconceptions; forces
    - **Encapsulating Varying Behavior**: Strategy, Template-Method, State, and Bridge
- **Day 2 - Design**
    - **Review**: Integrate and deepen what we covered
    - **Encapsulating Foreignness**: Adapter and Façade
    - **Encapsulating Construction**: Factory-Method, Abstract Factory, and Singleton
    - **Design Exercise**: Set up and group design exercise to be solved using patterns
    - **Exercise Review and Debrief**: Design review and discussion of design tradeoffs
    - **Encapsulating Sequence, Cardinality and Selection**: Proxy, Decorator, CoR
    - **Case Study Revisited**: Solve the initial design problem by refactoring to patterns
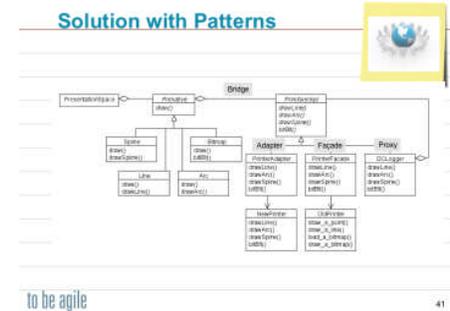
Your Instructor

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. In the past five years, he has focused on providing organizations with training and coaching for software developers and teams transitioning to Agile and Scrum.

Praise for this Training

"This course helped me to understand how to do quality software in a sustainable manner and support our software development organization to great results." –*Mikko Ala-Fossi, Dev. Mgr.*

"If you are interested in learning techniques that guarantee high quality code—take this class— the payback will start immediately and be significant." –*Jonathan Lister, SOA Architect/Platform Product Owner*

"I thought I was an ok developer. Taking this class has made me realize that there is a better path to follow, which will not only improve the way I work but improve the software I create... I would say take the class and let it change your life." – *Stephen Jones, GIS Architect*