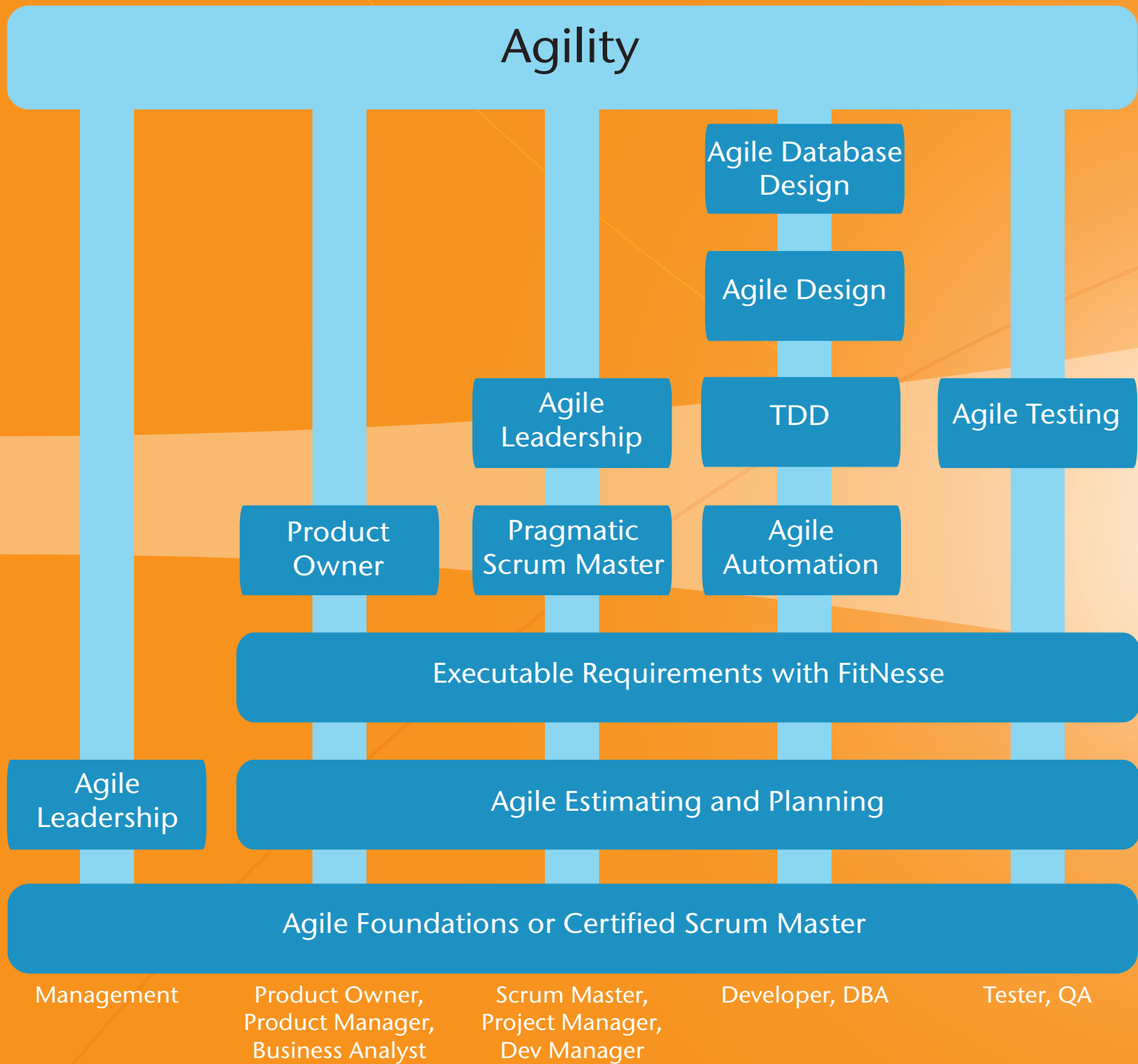


Training Catalog 2009-2010

New Classes
Added!

Curriculum Map



CURRICULUM

AGILE FOUNDATIONS.....3

A foundation for your entire team to learn the principles of agility.

AGILE LEADERSHIP.....3

Learn the skills that it takes to transform an organization into an agile organization.

EXECUTABLE REQUIREMENTS W/ FITNESS.....4

Learn how to tightly integrate the roles of analysis and testing to remove the pain of ‘testing after the fact’.

AGILE ESTIMATION AND PLANNING.....5

Advanced techniques for backlog management, task breakdown and more.

AGILE PRODUCT OWNER.....5

Learn how to proactively manage backlogs and efficiently communicate requirement needs in this intensive hands-on class.

AGILE DATABASE DESIGN.....6

Bring agility to the database concerns in your software development process with these techniques.

AGILE TEST DRIVEN DEVELOPMENT.....7

More than just xUnit training. We teach you to think and program with a test-driven mind set.

AGILE AUTOMATION.....8

Continuous Integration, automated deployment and proper SCM usage will have your team performing at their best

AGILE TESTING.....8

Learn how testing works in an agile environment, the importance of testing, and how testers become first-class citizens in the development team.

AGILE DESIGN.....9

Apply solid OO practices and evolutionary design principles to solve complex real world problems.

CERTIFIED SCRUM MASTER.....9

Prepare your scrum team for the realities of working in a highly collaborative and effective software development environment and earn the Scrum Master Certificate from Scrum Alliance.

PRAGMATIC SCRUM MASTER.....10

Offer your team the real world techniques they will need in order to successfully participate in a Scrum based project.

Agile Foundations

This course prepares all members of an agile team for the realities of working in a highly collaborative and effective software development environment. Over the course of an intense three days, Agile Foundations provides the breadth and depth of knowledge team members need to begin working in short iterations to produce working software incrementally. Using hands-on exercises, participants learn how to write effective user stories, estimate in relative units, track velocity and plan releases, create and interpret a variety of information radiators, participate in daily stand-ups, and much, much more. Towards the end of the course, participants learn to recognize and correct problems that may arise when making the transition to an agile process.

Topics Include:

- Incremental and Iterative Processes Overview
- Agile Team Roles
- Agile Planning
- Applying Just-In-Time Principles
- Defining and Estimating User Stories
- Prioritizing Backlogs
- Release and Iteration Planning
- Daily Standup Meeting
- Velocity
- Tracking and Reporting
- Agile Anti-patterns
- Agile Leadership Patterns

Prerequisites: A desire to become an agile organization

Audience: All roles involved in project delivery

Duration: 3 Days

Lab to Lecture Ratio: 40/60

Agile Leadership

A leader's job is to create an environment where others can succeed! This course is designed specifically to help senior and business management come to grips with agile software development and its impact on company processes and culture. Understand how agile software development changes the relationship between management and product delivery teams. Learn the deeper leadership philosophies of agile. Learn how to develop a management culture that builds and supports self-organizing agile teams and enables organizational change. Gain insight on how to apply agile thinking to your entire company. Then discover how to inspect your value stream to identify opportunities where agile practices will increase business efficiencies. This course is a 70/30 ratio of lecture to labs.

Topics Include:

- Origins of Agile – is it really new?
- Agile philosophies applied to product delivery
- Agile leadership, a new way of thinking and doing
- Applying Agile company wide

Continued...

For more information...us.training@valtech.com or 1 800 249 6595

www.valtech.us

(Agile Leadership continued...)

- Aspects of leadership: organizing, collaboration, ideation, motivation
- Organizing an Agile structure
- Supporting self-organizing teams
- Building an Agile-thinking company
- Applying Agile to your value stream
- Making your transformation plan

Prerequisites: Agile Foundations

Audience: Management, Business Management, Middle Management, Dev/QA Management, Scrum Master

Duration: 1 Day

Lab to Lecture Ratio: 70/30

Executable Requirements with Fitnessse

The traditional view of requirements communication results in a high degree of ambiguity and error. In this course we will examine the commonality between requirements and tests and how a requirement can be effectively communicated in the form of a test. Agile teams often struggle to fit testing into their iterations. By using FitNesse to create executable requirements, we will show students how to bring testing up front in their projects and therefore complete testing during the sprint without having a last minute large, stressful effort by the team.

Students in this class will learn how and when to use different fixture types in FitNesse. On the last day of class, students will bring user stories from their projects to class and will apply what they have learned to create acceptance criteria in the form of FitNesse Tables.

Topics Include:

- Gain hands on experience in new communication strategies to formalize requirements into executable tests
- How to use executable requirements to accelerates development and reduces defects found late in iteration cycles
- Integrating QA with Product Owners to define the acceptance criteria and hence the executable requirements
- How test first strategies support short iterations/sprints
- Selecting the appropriate table type to express executable requirements
- Strategies for organizing and automating entire test suites
- The use of various test styles for making tests clear and easy to read
- Knowledge of capabilities and limitations of the wiki based Fitnessse front end to Fit

Prerequisites: Agile Foundations

Audience: Product Owners, Product Managers, Business Analysts, Developers, QA

Duration: 3 Days

Lab to Lecture Ratio: 50/50

Agile Estimation and Planning

Estimating and planning is an important part of any agile project. This course will teach you how to efficiently and effectively drive the planning and estimating process resulting in improved accuracy. We will begin by examining the common pitfalls in planning and estimating processes so that new ideas have a baseline to be compared against.

We will discuss the motivations and challenges of story and task breakdown. Participants will learn an arsenal of techniques for breaking work into manageable chunks that fit neatly into iterations and days. We will discuss the trade offs and practicalities of estimating in story-points vs. ideal-days and how to make the appropriate decision for your project.

Product Owners will gain additional insight into Release planning and Backlog Management techniques. We will show simple but powerful ways to prioritize backlog items by financial impact, thematic organization and desirability. You will learn practical skills that will help you create useful plans that lead to reliable decision-making. You will leave with a solid understanding of how to conduct agile roadmap, release, and iteration planning.

Topics Include:

- Roadmap and Release planning
- Writing effective User Stories
- Planning Poker
- Iteration Planning
- Estimating work
- Plan Buffers and Uncertainties
- Estimating Velocity

Prerequisites: Agile Foundations or Certified Scrum Master Training

Audience: All Team Members

Duration: 3 Days

Lab to Lecture Ratio: 60/40

Agile Product Owner

This course is designed specifically to help Product Owners clearly understand their role on an agile team and how to work effectively and efficiently with the team to create, organize, and maintain a product backlog. During this two day course, students will apply what they learn by first creating a product vision and roadmap. They will then spend a significant amount of time learning how to write effective user stories that can then be estimated and prioritized to aid in release planning. Students will then learn how to prepare for an iteration planning meeting by creating acceptance criteria for prioritized user stories.

Near the end of the second day, students will learn how to track and communicate project progress using velocity metrics and release burn-down charts.

Topics Include:

- Proper care and feeding of a product backlog
- How to identify and write well-formed user stories
- How to assist in the estimation process

Continued...

For more information...us.training@valtech.com or 1 800 249 6595

www.valtech.us

(Agile Product Owner, Continued...)

- Techniques used to prioritize a backlog
- The importance of communicating acceptance criteria as tests
- Strategies for bridging the gap between business and technology
- How to allocate your time to keep fast iteration cycles running smoothly
- How to generate metrics that are meaningful and align with Agile fundamentals

Prerequisites: Agile Foundations

Audience: Product Owners, Product Managers, Business Analysts

Duration: 2 Days

Lab to Lecture Ratio: 50/50

Agile Database Design

The Agile Database Design course addresses issues previously relegated to the DBA group. Teams following iterative and incremental development cycles know how important refactoring is to emergent design. We now understand that development efficiency increases when the database design evolves with the software. An understanding of the existing barriers that inhibit agile database design is key to opening up this new approach. Agile database design requires new skills and tools and needs to be done in a fashion that lessens the operational risks on production versions of the system. This course teaches these skills in a mixture of classroom lectures and related lab exercises.

Students will learn the use of LiquiBase and migration and deployment strategies for schemas and data.

Topics Include:

- Thinking Agile
- Iterative and Incremental Database Design
- Table Migration Strategies
- Data Migration Strategies
- Deployment in an Agile environment

Prerequisites: Agile Foundations, Java programming background, use of Eclipse, JUnit and Ant. Some database theory and exposure to relational database design concepts.

Audience: Developers, DBAs, Architects

Duration: 3 Days

Lab to Lecture Ratio: 60/40

Agile Test Driven Development (for Java or any .Net CLR environment)

The Agile Test Driven Development course lays the foundation for modern agile development by teaching your team what unit testing is all about and how, if this is done properly, you can have the courage to design and write better code in a manner consonant with agile principles. We can only break the back of Big Design Up Front by having a code base that allows us to refactor our code relentlessly as the need arises. The only way to ensure that we can refactor relentlessly is to be able to ensure that each and every check-in to the repository happens in a way that no functionality is changed unintentionally as we rethink our solutions to the new realities of what the application is to do as we learn more about it.

This course is a mixture of practice and principle. We discuss what testing is, and why unit testing is not testing, as it is commonly perceived, but a coding practice. We will learn the basics of the unit testing framework, and how to use it effectively in your IDE. But test driven development is more than simply test first development. We learn how to avoid the pitfalls of an architecture-centric development model, and appreciate how test-driven development allows the architecture to organically rise up to meet project needs, rather than be imposed by a big design up front. We learn to understand that artificial up front thought on what design decisions we think we might need, especially given our lack of complete knowledge of what is actually needed at that stage, poses project and maintenance nightmares. We teach the necessity to be aware of the continuous integration nature of agile projects, and the process for insuring that our code never “breaks the build”. Appropriate XP coding practices are discussed and used. We also speak about what refactoring is and how you know the need for it when you smell it. Practical use of the refactoring tools built into your IDE are taught and practiced, but the emphasis is on learning how to accomplish thoughtful and more heavyweight refactoring. Finally, we discuss the importance of designing for testing and the role of the Inversion of Control pattern on the design of hard to test code in the form of Test Doubles and their kindred sorts.

The course is taught with a mixture of lecture and labs. Later labs benefit from the designation of a central lab machine to be the repository for the class, but this can be avoided in situations where no networks are allowed. Java students will use Java, Eclipse, JUnit and SubVersion during the course. .NET students will use one of the CLR managed languages (C#, VB, or C++), Visual Studio, NUnit and SubVersion during the course.

Topics Include:

- Why Test First?
- xUnit, the TDD workhorse
- The TDD Rhythm
- Testing and its Impact on Architecture
- Using Test Doubles
- Refactoring Patterns and Anti-patterns

Prerequisites: Agile Foundations, Java or any .NET CLR programming background and software design experience

Audience: Developers

Duration: 3 Days

Lab to Lecture Ratio: 60/40



Agile Automation

The speed at which agile projects run demands a high level of automation surrounding all repetitive activities. Nowhere is this more obvious than in a build/test/deploy process. Builds have traditionally been a source of pain; the natural reaction to this pain is for teams to put off the build and deployment activities, thereby incurring large amounts of 'project debt'. In an agile world we attack this problem by accelerating and automating the build and deployment processes. The goal of this class is to educate a team on how to apply automation to all stages of their work. We incrementally build a complete continuous integration strategy that is applicable and configurable for most any environment. Students will learn to use Ant and Cruise Control during the course.

Topics Include:

- Completely automating the build process
- Making builds portable across machines
- Integrating testing into the build
- Automating the deployment process
- Automating communication and messaging infrastructure for the entire build, to deployment, to monitoring process
- Unlocking the full potential of a source control through effective use of tags, branches, and merging

Prerequisites: Agile Foundations, Familiarity with application customizations, scripting facilities, Java, Eclipse and JUnit (.NET version uses CLR language, Visual Studio, and NUnit.)

Audience: Developers, QA, configuration managers

Duration: 3 Days

Lab to Lecture Ratio: 60/40

Agile Testing

When a software development team adopts agile practices, the testing team often struggles to understand what their role is and how they fit in. This two-day class provides participants with an overview of how testing changes when a team adopts agile methods. The class will begin with an overview of agile principles and practices and will be a combination of presentations and hands on exercises. Participants will leave this class with a clear understanding of how traditional testers can navigate through an agile adoption.

Topics Include:

- Comparison of Traditional QA practices vs. Agile Testing Practices
- Acceptance Test Driven Development
- Defining acceptance criteria
- Executable Requirements
- Automated Testing on Agile Teams
- Defect Management
- Continuous Integration
- Exploratory Testing

Prerequisites: Agile Foundations

Audience: QA, Testers

Duration: 2 Days

Lab to Lecture Ratio: 40/60

Agile Design

Being agile does not mean eschewing design. Good agile teams make design activities such a high priority that they are embedded in nearly every thought and activity. This course will take a team through the process of evolutionary design and show how difficult design constraints can be met more accurately than by traditional upfront design work.

Students will be given real world design problems and learn how to satisfy complicated requirements by incrementally ‘evolving’ a code base rather than pursuing an up-front design strategy.

We will also explore situations where large architectural changes are required late in a project life cycle due to changing requirements or faulty assumptions and how these changes can be safely undertaken.

Students will use Java, JUnit and Eclipse during the course. For .NET, students will use a CLR language, Visual Studio and NUnit.

Topics Include:

- Identifying and using design patterns
- Interface oriented design
- Designing for testability
- Creating “clean” code

Prerequisites: Agile Test Driven Development

Audience: Developers, Architects

Duration: 3 Days

Lab to Lecture Ratio: 60/40

Certified Scrum Master

This hands-on certification course is appropriate for all software team members, including Project Managers, Team Leads, Development Managers, Product Managers, Software Architects, Systems Architects, Testers, CIOs, and CTOs.

This course prepares all members of a Scrum team for the realities of working in a highly collaborative and effective software development environment. Scrum Master training provides an applied understanding of the Scrum process and helps participants begin to develop the tools, insights, and skills required to apply Scrum on their projects and across their organizations.

Course completion includes a Scrum Master Certificate and a one-year membership to the Scrum Alliance.

Topics Include:

- Learn the Scrum process and the important role of scrum master
- How to be an effective Scrum Master
- How to build and lead a Scrum team
- How to conduct effective planning, demo and retrospective meetings
- How to create and use information radiators
- How to build a self-organizing team

Continued...

For more information...us.training@valtech.com or 1 800 249 6595

www.valtech.us

(Certified Scrum Master continued...)

Prerequisites: None

Audience: All team members

Duration: 2 Days

Lab to Lecture Ratio: 60/40

Pragmatic Scrum Master

Valtech's Pragmatic Scrum Master training goes beyond the simple Certified Scrum Master (CSM) training to offer your team the real world techniques they will need in order to successfully participate in a Scrum based project.

This course assumes no previous working knowledge of scrum other than what can be acquired through our Agile Foundations course. During these intense two days participants will collaborate on a series of scrum based exercises that represent all key aspects of agile project challenges.

Specific areas that will be emphasized include: Backlog creation and maintenance, Sprint Planning, Release Planning, Task Estimation, Retrospectives, Daily Stand-ups, Information Radiators.

We will also discuss real world project 'smells' and give hard won advice on how they have been successfully addressed on other projects. Scaling scrum and interfacing with non-agile departments will round out our topic exploration. Each delivery of the class will, in true agile fashion, allow for the real-time course adjustment to meet student needs by emphasizing certain material relevant to your situation.

Topics Include:

- Deep-dive into the Scrum Master Role
- Templates for Successful Agile Planning
- More Tips for Effective Daily Standup Meetings
- Identifying and Repairing Agile Team Dysfunction
- Useful Metrics and Information Radiators
- Enlivening Drab Retrospectives
- Building Continuous Improvement into Your Team's Culture
- Pushing out Agile into the Enterprise
- Improving Your Agile Process Through Lean Principles
- Collaboration Strategies
- Tools to Support Agile

Prerequisites: Agile Foundations

Audience: Scrum Masters, Project Managers

Duration: 2 Days

Lab to Lecture Ratio: Approximately 30/70





19111 N Dallas Pkwy
Suite 300
Dallas, Texas 75287

**Contact us for more
information:
us.training@valtech.com
1 800 789 6595**

USA

Dallas :: Oklahoma City
Houston :: New York :: Raleigh

EMEA /ASIA

Denmark :: France :: Germany :: India
Korea :: Sweden :: United Kingdom

www.valtech.us