

TTR311: Managing Agile Software Development (4 days)



Managing Agile Software Development is a four-day, dynamic workshop designed to provide project stakeholders with an in-depth review of the Agile software development process, arming students with the skills required to execute these principles and practices in a practical manner, right after the course ends.

When new software has to be created and operational in a matter of days, rather than months, the standard linear waterfall development process can easily bottleneck these critical projects. The more current agile development approach is an excellent alternative to the linear process. Emphasizing user involvement and lightweight process, the agile approach provides software designers and developers with the means to create high-quality, low-defect applications that can be implemented in the minimum timeframes.

► Course Objectives: What You'll Learn

Geared for managers, programmers, developers, executives, and anyone interested in learning the benefits of and how to execute agile development practices, this course explores:

- The concept of development agility and the Agile Manifesto
- Each of the major agile development methods underscoring their strengths and weaknesses
- How to manage an agile environment even within a structured organizational approach
- How to introduce agility into a development organization

► Experiential Learning – Course Structure

Attending students will be led through a series of topics comprised of integrated lectures, group discussions and comprehensive demonstrations. The course provides a solid foundation in basic terminology and concepts, extended and built upon throughout the engagement.

► Audience & Pre-requisites: Who should Attend

This **introductory-level** course is geared for managers, programmers, developers, executives, and anyone interested in learning the benefits of agile development.

Attending students should have knowledge of current development processes, such as structured top-down development and the waterfall method.

► Related Courses – Suggested Options

Take Before: Students should have practical skills equal to or should have taken the following courses as a pre-requisite:

- For students requiring an introduction to Agile consider **TT1515 Agile Development Overview (1 day)**

COURSE SNAPSHOT

Duration: 4 days
Skill Level: Introductory +
Focus: Agile development
Audience: Technical managers, stakeholders, executives and developers, architects and analysts.
Hands-On: Expert lecture combined with open discussions and high-Level demonstrations; hands-on drawing
Delivery Format: Available for onsite private classroom presentation, or live online / virtual presentation
Customizable: Yes

Take Instead or After: We offer other courses that provide different levels of knowledge or focus in the Agile or TDD space

- For students who require more in-depth Agile training please consider **TTRS313 Implementing Agile Test-Driven Development (4 days)**

Take After: We offer a variety of introductory through advanced development, project management, engineering, architecture and design courses. Students may want to consider the following topics as follow-on training to this course.

- Additional agile or software design topics
- Test-driven development and
- Secure software development & secure design courses
- Architecture & Analysis courses
- Software Engineering, Design or Project Management tracks
- Service-Oriented Analysis and Design
- Web Services – Intro through Advanced
- Extensive developer courses for either .Net or Java/JavaEE platform application development

Please contact us for recommended next steps tailored to your longer term education, project or development objectives.

► **Student Materials: What You'll Receive**

Our robust course materials include much more than a simple slideshow presentation handout. Trivera Technologies Student materials include a comprehensive hard-copy course manual, complete with detailed course notes, code samples, diagrams and current reference materials, all directly related to the course at hand, indexed for ease of use. Step-by-step lab instructions and project descriptions are clearly illustrated and commented for maximum learning and ease of use.

In addition to everything students need for the course, students will receive our *LoadNGo Instant Classroom Kit* which includes any workshop labs and solutions; non-restricted workshop software (if applicable), APIs, documentation, technical education papers, and

specifications pertinent to the training course.

Our course kits are designed to serve as an excellent and useful reference set, long after we leave your classroom.

► **Optional Pre / Post-Testing & Skills Assessment**

We work with you to ensure that your resources are well spent. Through our basic pre-testing, we ensure your team is up to the challenges that this course offers.

We will work with you to come up with the best solution to ensure your needs are met, whether we customize the material, or devise a different educational path to prepare for this course.

Workshop Topics Covered

Session: Introduction to Agile

- Software Development Life Cycles
 - Mandatory elements
 - Transitions
 - Traditional Life Cycle Models
- Process-oriented development
 - CMM
 - Other process models
 - Downsides of process models
- Basis for Agile Development
 - Timeboxing
 - Iterative development
 - Incremental delivery
- Principles of Agile methods
 - Adaptation rather than prediction
 - People-oriented rather than process-oriented Working with the RAD

Session: The Agile Approach

- The rise of agile programming methods
- The agile manifesto
 - Individuals and interactions
 - Customer collaboration
 - Working software
 - Response to change
- Forms of communication
- The people factor
 - Involving the user
 - Involving the Business Analyst
 - The agile development team
 - Refactoring
- Testing

Session: The System Architecture

- Developing a system model
- Relationship of the model to agile development
- Documentation in the Agile environment
- Structure, method, documentation
- The “open-close” principle

Session: Agile Product Definition

- Customer interaction
- User stories
- The requirements backlog

Session: Incremental Delivery

- Classic approaches to delivery
- Principles of Incremental Delivery
- Benefits
- Approaches
- Agile Approach: Staged Evolutionary Delivery
 - Background
 - Principles of Tom Gilb
 - Structure
- Evolutionary delivery vs. exploratory development
- Methodology
- Structuring the evolutionary project

Session: Iterative Development

- Classic approaches to development
- What is iterative development
- Iterating development cycles
- Benefits of iterative development
- Web site iteration with wire frames
 - Roles of the UX and IA

- Static versus dynamic prototypes
- User involvement
- Agile Approach: Phased Iterative Development
- Team approach
 - Team member roles
- Prototyping sessions
 - Structure
 - Method
 - Documentation
- Timeboxing
- Phases
 - Data
 - Cosmetic
 - Function
- Change control

Session: Agile Analysis and Design

- Classic approaches to analysis and design
- Agile Modeling concepts
- Iterative design approaches
- Refactoring design
- Agile method: Feature-Driven Development (FDD)
- Background
- Processes
 - Develop an overall model
 - Build a features list
 - Plan by feature
 - Design by feature
 - Build by feature

Session: Agile Programming

- Classic programming approaches
- Principles of agile programming
 - Refactoring
 - Complexity
 - Mutual ownership of code
- Agile Method: Extreme Programming
 - Twelve Practices
- Iterative planning
- Pair programming
- Simple design
- Unit testing
 - Documentation
 - Process
- “Planning game” meeting
- Exploration
- Management
- Commitment

Session: Agile Software Development Management

- Classic software development management
- Agile Self-managing teams
- Business representation
 - Choosing the customer
 - Business analyst on the team
 - UX in XP
- Agile Method: Scrum
 - Background
 - The “wrapper” approach
 - The Scrum pattern
 - Timeboxing

- Principles
- The Sprint
- The Scrum meeting
- The backlog
 - Advantages
 - Scalability
 - Agile Methods: Adaptive Software Development and Crystal
 - Background
 - Predictive vs. Adaptive
 - Basic characteristics
- Mission focused
- Component based
- Iterative
- Timeboxed
- Risk driven
- Change tolerant
 - Speculate-collaborate-learn cycle
 - Principles
- Adaptive culture
- Adaptive framework
- Adaptive collaboration
- Adaptive scale
- Adaptive management
- Just in time methodology
- Humans and technology
- Team-orientation

Session: Adapting Agile Methods to Your Organization

- Decision points
 - Organizational culture
 - Systems organization culture
 - Post implementation processes
 - Overall focus
 - Applicability of standards
- Quality points
 - Ease of adoption
 - Availability of training
 - Thoroughness
 - Standards adherence
 - State of method evolution
- Establishing an agile development process

Session: The Bottom Line

- Ideas to Use
- Where to go for more information

***Need more details?** Please note that a more detailed outline of the course table of contents, lists of lab exercises and project descriptions is available. Please contact us at Training@triveratech.com for info.*

***Need courseware?** This course is fully customizable, and also available for license with complete support for qualified organizations. Please contact Courseware@triveratech.com for details.*

► **Why Work With Trivera Technologies?**

- **Our courses are focused - no "fluff" included.** We offer more than a “laundry list” approach to teaching. All lessons have clear objectives, are fundamental to learning core agile practices, and are reinforced by demos, group labs and solid practical examples. Each lesson has performance driven objectives that ensure students will learn technologies and skills core to fundamental agile concepts – nothing more, nothing less. Students will learn the importance of developing well-designed applications.
- **Our materials are comprehensive, and current.** Each lesson has performance driven objectives that ensure students will learn technologies and hands-on skills core to agile development – nothing more, nothing less. Progressive labs are designed in such a way that students get a firm grasp on fundamental skills while they work toward building a complete solution. All lessons have clear objectives, are fundamental to learning core programming practices, and are reinforced by hands-on code labs and solid practical examples.
- **Our materials are robust.** Our comprehensive manuals include not only a hard copy of the course presentation, but also detailed reference notes, pertinent diagrams and charts, current lists of suggested online resources and articles, and often technical tutorials or white papers geared to the topics at hand. Hands-on courses also include our unique materials for each student, complete with course set up, software (as applicable) and a multitude of learning resources that complement the course.
- **We foster "Learning by Doing".** Progressive labs are designed in such a way that students get a firm grasp on fundamental skills while they work toward defending a complete application. All labs are take-home, and all solution code is presented in an easy to use self-study format for future use and review.
- **True content ownership gives us flexibility & quality above the rest.** These course materials are wholly-owned by our company and fully customizable, at little or no cost, to help you best meet your learning objectives. We have many dedicated experts available worldwide to

instruct your team, and can provide services around the globe, either locally or virtually. We work closely with you to produce the most effective events and materials for your team, within your desired timeframe and budget.

- **We have to adhere to higher standards.** As a courseware provider, our content and hands-on lab materials are licensed internationally by dozens of firms, and are therefore subject to very stringent quality requirements. Not only will your organization benefit from our own technical team's technical expertise, but also the feedback of hundreds of students and trainers using these materials, worldwide, on a regular basis. This unique fact guarantees that our materials are not only robust and interesting, but also technically correct, current and of the highest quality and usability.
- **We bring years of practical, current experience into the classroom and content.** Our instructors and course authors are also skilled mentors, Java, J2EE, .Net, SOA, Agile and web services developers, architects and security-oriented professionals. We believe that learning, using and maintaining solid software execution and delivery methods are as important as gaining sharp coding skills. Best Practices for software development and execution, beyond technical coding skills, are enforced throughout all of our courses and discussions. Our team brings this extensive experience into every classroom and engagement.
- **We're skills-centric.** Although our team has extensive experience using a variety of tools and solutions, our core content is "technology-centric". Our aim is to teach you the best skills and solutions out there – not to sell you software from any particular vendor.
- **We're authors.** Our team was selected to write the online *J2EE, EJB, EJB CMP-CMR and Web Services Tutorial Series for IBM developerWorks®* (www.ibm.com) These are the same instructors who train our classes and author the courseware. Most of our trainers/consultants have also authored additional articles on web services, EJB< Struts, J2EE and advanced Java topics, and are recognized speakers and presenters on the industry technical seminar circuit. Our team is comprised on several successful published authors. Members of our team have written or contributed to: *Eclipse Kick Start, Mastering Eclipse; Professional Jakarta Struts; Using Java Tools for Extreme Programming; Mastering Resin; Mastering TomCat and others.*
- **Our services are guaranteed.** Whether you're a stakeholder organizing your firm's educational services, a student in our live or virtual classroom or a trainer using our materials to educate your own client or team – **Our core mission is to make YOU a success in the classroom.**

► For Additional Information

Need dedicated training? All courses can be brought onsite for a **private presentation**, customized to suit your unique requirements or goals. Please contact Training@triveratech.com for course details, Public Schedule dates and locations, and Special Discount Offers.

Need courseware? Let us take the risk out of your classroom delivery! All materials are also available for corporate license with complete instructor support and free corporate branding. We guarantee our pricing and service. Samples of our course materials, as well as live client references for all of our services are available upon request. Please contact Courseware@triveratech.com for details.

For more information about our training, mentoring or courseware licensing options, or to see our complete list of course offerings and services, please visit us at www.triveratech.com, email Training@triveratech.com or call 609.953.1515.



Trivera Technologies is a 100%
 Female-Owned Small Business Concern
 GSA # GS35F-0188T
 Please contact us for details & pricing.