

# Test-Driven Development Training



## At a Glance

Rather than spending weeks or months on an extended “design phase,” agile teams use test-driven development (TDD) to evolve and optimize their design and implementation over time to meet the team’s cumulative understanding of the business requirements. In this three-day training course, you’ll learn the basic skills and gain the hands-on experience required to implement TDD in your agile development. You’ll be introduced to TDD-related techniques and concepts while working to turn example business requirements into working code.

### Key Outcomes

- Gain hands-on experience
- Learn TDD-related techniques and concepts
- Turn example business requirements into working code

### Course Essentials

#### Audience

Developers

#### Agile maturity

Intermediate

#### Maximum attendees

16

#### Length

Three days

#### Prerequisites

Participants should be familiar with the Java programming language and its commonly used libraries and APIs, and at least somewhat familiar with the Eclipse IDE. If interested in other languages or platforms, contact a CA Agile Training representative.

#### System requirements

Java® Development Kit version 1.5 or later, Eclipse IDE version 3.6 or later, JUnit version 4.X and JUnit plug-in for Eclipse, Mockito 1.9 or later. Note: System requirements can be made available on USB drives for each participant at an additional cost, as long as the attendees have administrative rights to install to their machine.

## Business Challenge

The modern company has to react fast and deliver faster. Your developers need to be able to deliver clean code—that works—and do it cost-effectively. That’s why so many companies have turned to agile methodologies. However, some agile practices are easier to adopt than others. TDD can help your software development teams consistently release high-quality code in short iterations. But with TDD, the learning curve is steep, and many experienced programmers bristle at the often evangelistic tone of some TDD adherents. The advantages of TDD make it worthwhile. By testing proactively, code changes become less costly, and adaptive change is encouraged. Developers can validate their work often and early. With immediate feedback, bugs are quickly detected. Which ultimately means your organization can continuously delight your customers.

## Learning Objectives

- Sketch an implementation model from user requirements, emphasizing the assignment of responsibilities to classes, components and sub-systems
- Apply the TDD Test-Code-Refactor cycle
- Code by intention and triangulate to arrive at an optimal solution
- Use unit tests as executable specifications that document the behavior of the code they test
- Detect common design deficiencies or “code smells,” and improve them through refactoring
- Isolate units to be tested using test doubles and mocking frameworks
- Describe the relationship between user stories, confirmations, acceptance tests and unit tests

## Intended Audience

This course is primarily intended for hands-on developers. Architects, technical leadership and anyone who must understand, review and communicate the use and utility of TDD by agile teams will also benefit. New agile teams that attend as a group will find the course helpful because it provides a common foundation of knowledge and experience for everyone on the team.

## Build Better Software

The practice of TDD allows an agile development team to efficiently convert user-defined requirements into a well-designed, fully tested, working implementation that not only meets the business need but also will be easy and safe to extend, maintain and enhance over time.

As a technical workshop, this course emphasizes hands-on coding and the use of widely adopted development tools and testing frameworks for the Java development language. The course can be modified to use real-life examples and situations from the students' own experiences.

## About CA Agile Training

Our training courses are built on sound principles of adult learning theory, general systems thinking and group process concepts. Seasoned practitioners, who use agile to teach agile, lead our highly interactive classes and put your organization at the center of every engagement—facilitating pragmatic discussions about how to adopt agile practices and foster an agile culture in your unique environment.

## Recommended CA Agile Coaching Services

Consider adding one or more coaching days to this class where activities could include:

- Performing test-driven development against the current backlog of user stories
- Refactoring portions of existing code
- Collaborating with Scrum Master, product owner and other team members
- Examining the code and test environment

CA Agile Coaching can hugely increase the value of your training by engaging expert consultants to help you apply learnings in your context and to your real environment and work. We can deliver coaching right after training or at another time—on-site or remotely. Speak with your CA sales rep for terms and conditions.

For more information, please contact us at [agilesales@ca.com](mailto:agilesales@ca.com)

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments. Learn more at [ca.com](http://ca.com).