

## Agiles Testen und Test-Driven Development (TDD) von Embedded-Systemen - Live-Online-Training

Das agile Testen von Embedded-Systemen ist die konsequente Erweiterung der Agile-Methodik auf das gesamte Embedded-System. Analog zum agilen Entwickeln bedeutet der agile Testansatz einen Paradigmenwechsel. Unter Test-Driven Development (TDD) werden Tests erstellt und durchgeführt, bevor die eigentliche Komponente entwickelt wurde. Die Tests wachsen inkrementell mit dem System. Durch die kontinuierliche Testautomatisierung und Testwiederholung werden Fehler sehr früh im Entwicklungsprozess entdeckt und sind leichter lokalisierbar. In diesem Training werden Ihnen neben dem nötigen theoretischen Wissen auch praktische Umsetzungswege vermittelt, mit deren Hilfe Sie agiles Testen und Test-Driven Development in die Entstehung Ihrer Embedded-Systeme einführen.

### **Ziele - Ihr Nutzen**

Sie lernen wichtige Begriffe, Zusammenhänge, Methoden und Tools kennen, um maßgeblich zur Gestaltung, Umsetzung und Durchführung von agilem Testen und Test-Driven Development in der Entstehung Ihrer Embedded-Systeme (System-, Hardware- und Softwareentwicklung) beizutragen.

### **Teilnehmer**

Testingenieure, Test-Manager, Software-Entwickler, Software-Architekten, System-Architekten, Teamleiter, Projektleiter

### **Voraussetzungen**

Grundlegende Kenntnisse der Programmiersprache C oder C++; Projekt- und Prozess Erfahrung in der Entwicklung technischer Systeme sind von Vorteil.

## **Live Online Training**

\* Preis je Teilnehmer, in Euro zzgl. USt.

Anmeldecode: L-AGILTDD

### **Präsenz-Training - Deutsch**

#### **Dauer**

2 Tage

### **Live-Online - Englisch**

#### **Dauer**

2 Tage

### **Präsenz-Training - Englisch**

#### **Dauer**

2 Tage

## **Agiles Testen und Test-Driven Development (TDD) von Embedded-Systemen - Live-Online-Training**

### **Inhalt**

#### **Begriffe und Testansätze**

- Modul-, Integrations-, Systemtest
- Blackbox-, Whitebox-, Greybox-Test
- Regressionstest
- Kontinuierliche Integration und Test

#### **Test-Driven Development Ansatz**

- Gegensatz zu herkömmlichen Vorgehen
- Zusammenhang mit agilen Entwicklungsmethoden
- Kontinuierliche Integration und Test

#### **Embedded TDD-Strategie**

- Target Hardware
- Dual Targeting
- Embedded TDD Cycle

#### **Test-Erstellung**

- FIRST, die fünf Prinzipien zum Erstellen guter Tests
- Testumfeld und -abhängigkeiten
- Test Doubles
- Mocks und Stubs

#### **Test-Abdeckung**

- Testziele
- Kriterien für "done"
- Testwiederholung

#### **Der TDD-Zyklus**

- Red-Green Refactoring

#### **Design for Test**

- SOLID Design-Prinzipien
- Refactoring
- Testen von Legacy Code
- Testmuster und Zufallszahlen

#### **Praktische Übungen**

- Übungen zum besseren Verständnis der Methoden und Prinzipien
- Tools: ARM  $\mu$ Vision, Embedded-Unit