

## Dr. Andrea Baruzzo

Consulente Indipendente

Via Circonvallazione 26, 33055 Muzzana del Turgnano, (UD) – Italy

PI: 02567160300 CF: BRZNR73D15I403M

e-mail: [abaruzzo@computer.org](mailto:abaruzzo@computer.org)

Mobile: (+39)3393955753

Formazione

Corso di Formazione Professionale: Design for Testability (DFT) nei sistemi software orientati agli oggetti

### Descrizione del corso

Progettare sistemi critici, ossia sistemi software di grandi dimensioni, altamente affidabili e manutenibili nel lungo periodo è inerentemente complesso. Molte delle tecniche che si imparano nei corsi di programmazione non “scalano” sufficientemente bene in questi sistemi. Ciò è vero particolarmente per il test. Un qualsiasi sistema critico non può essere reso affidabile sulla base delle sole tecniche di test, poiché nel momento in cui esso viene collaudato la sua qualità intrinseca è già stata determinata (indipendentemente dal tipo di metodologia di sviluppo utilizzata). Affinché tali sistemi siano collaudabili efficacemente, è necessario *progettarli* fin dall’inizio con questo obiettivo in mente. L’investimento richiesto da una progettazione orientata anche al test ha delle ricadute importanti anche sulla manutenzione a medio e lungo termine: una gestione attenta delle dipendenze semplifica le operazioni di estensione e correzione dei bug, rende il design più razionale e semplice da capire, riduce i costi di gestione del progetto.

Il corso proposto è di carattere *pratico*, rivolto principalmente agli sviluppatori e agli architetti del software con una discreta esperienza di sviluppo. Le tecniche proposte spaziano dalla progettazione dei singoli componenti alla definizione dell’architettura globale, affrontando aspetti sia di design logico, sia di design fisico. Molti esempi vengono proposti e discussi, valutando l’impatto di ciascuna alternativa di design rispetto a manutenzione e test.

Una caratteristica essenziale del corso è l’aggiornamento metodologico in chiave DFT, applicando i concetti tipici di questa disciplina alle moderne tecniche di progettazione del software

basate sui modelli (UML, design pattern, sistemi software orientati agli oggetti). Il corso è indipendente da uno specifico linguaggio di programmazione, anche se per i particolari domini applicativi che contraddistinguono i sistemi critici, una certa enfasi sarà posta sul linguaggio C++, storicamente il più diffuso linguaggio orientato agli oggetti in questi ambiti.

Il corso, infine, può essere visto come parte di un percorso formativo più completo e specificamente indirizzato verso i sistemi critici (quei sistemi che tipicamente ricadono in una delle seguenti denominazioni: dependable, safety, real-time, embedded, fault-tolerant systems). Alternativamente, può essere anche pensato sia come prosecuzione di un primo corso sulla progettazione del software orientato agli oggetti, sia come introduzione preliminare ad un corso orientato esclusivamente al testing.

### **A chi è rivolto il corso**

Architetti del software, programmatori, team di test.

### **Livello di difficoltà**

Parte: Intermediate-Expert.

### **Prerequisiti**

Minima familiarità con la modellazione del software in UML;

Buona conoscenza della programmazione orientata agli oggetti e delle tecniche di design orientato agli oggetti;

Conoscenza di base delle tecniche di test applicate ai sistemi orientati agli oggetti (in particolare test di unità e di integrazione, test gerarchico e incrementale).

### **Personalizzazione del corso**

I contenuti del corso possono essere personalizzati su richiesta in base alle specifiche necessità dell'azienda, con riferimento particolare alla composizione del gruppo di lavoro e/o del tipo di progetti. Per maggiori informazioni sui contenuti e per concordare durata e programmazione, contattatemi.

### **Esempio di programmazione del corso**

40-48 ore, normalmente suddivise in 10-12 lezioni da 4 ore ciascuna presso il cliente.

## **Il docente**

Il Dr. Andrea Baruzzo si è laureato nel 2001 in Scienze dell'Informazione presso l'Università degli Studi di Udine dove, nel 2008, ha conseguito il dottorato di ricerca in informatica. Si occupa di ricerca e formazione sia in ambito accademico, sia in ambito industriale. Ha svolto attività di consulenza e trasferimento tecnologico in svariati domini applicativi: in avionica (sistemi UAV velivoli teleguidati senza pilota), in progetti di tipo gestionale orientati sia ai dati, sia ai task/servizi (sistemi informativi enterprise, sistemi supply-chain, sistemi di simulazione finanziaria), in sistemi GIS (Geographic Information System) e grafica 3D.

Le sue principali aree di interesse sono: l'analisi, la progettazione e lo sviluppo di sistemi software ad oggetti (OOA/OOD/OOP), le tecniche di test e la modellazione del software. È inoltre membro di IEEE Computer Society.