

Dr. Andrea Baruzzo

Consulente Indipendente

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

PI: 02567160300 CF: BRZNR73D15I403M

e-mail: abaruzzo@computer.org Mobile: (+39)3393955753

2010

Corso di Formazione: Testing e Testabilità

Descrizione del corso

Approcciare *professionalmente* al testing di sistemi software complessi è un'attività complessa che richiede un importante investimento culturale, la cui natura include almeno tre fattori distinti:

1. il *fattore tecnico* (per costruire del software di test è necessario padroneggiare strumenti, linguaggi, framework, ecc.),
2. il *fattore organizzazionale* (il testing va gestito, attraverso metodologie, processi, risorse, al pari delle altre attività della produzione di software)
3. il *fattore intellettuale* (per progettare e scrivere del software di test in modo efficace è necessario conoscere diversi elementi teorici quali, ad esempio, i modelli di fault e modelli di test più adatti a verificare lo specifico sistema da collaudare, ecc.)

Molto spesso, invece, nella pratica il test è relegato alla fine del ciclo di vita del prodotto, basandosi esclusivamente sulle capacità fornite da uno specifico tool di test. Indubbiamente uno strumento automatico di test semplifica e ottimizza la produzione e/o l'esecuzione di una suite di test. Nessuno strumento però può sostituire l'intervento umano, specialmente nel capire *come scrivere*, oppure *come organizzare* il codice di test. Nessuno strumento automatico può migliorare la qualità del sistema da collaudare: infatti per collaudare meglio è necessario prima *progettare meglio*, e il saper progettare è una prerogativa tipicamente umana. L'investimento richiesto da una progettazione orientata anche al test ha, tuttavia, 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,

rendendo il design più razionale e semplice da capire. Ecco allora che coniugare la progettazione con il test permette di ridurre in parte i costi globali di gestione del progetto software.

Una caratteristica essenziale del corso è quindi l'aggiornamento metodologico in chiave DFT (Design For Testability), 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). L'applicazione delle tecniche di DFT è spesso un passo preliminare alla costruzione di software object-oriented "orientato al test".

In conclusione, il corso ha come obiettivo affrontare la tematica del testing sin dalla fase di progetto, tenendo in considerazione i fattori tecnici (strumentali), i fattori organizzazionali (sociali), e quelli intellettuali (personali).

Pur essendo la tematica del testing di sistemi object-oriented ricca di aspetti teorici più o meno formali, il corso proposto mantiene un carattere *pratico*, rivolto principalmente agli sviluppatori, ai tester e agli architetti del software con una discreta esperienza di sviluppo.

A chi è rivolto il corso

Gruppi di test, architetti del software, capi progetto, programmatori.

Livello di difficoltà

Intermediate-Expert.

Prerequisiti

Minima familiarità con la modellazione del software in UML;

Buona conoscenza della programmazione orientata agli oggetti e dei principi di design orientato agli oggetti; Pregressa esperienza con le attività di testing del software è utile, anche se non vincolante.

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.

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.