

***PERCORSI DI  
FORMAZIONE  
2010***



# PERCORSI DI FORMAZIONE 2010

*Il catalogo 2010 dei corsi Tecnet Dati è  
consultabile anche all'indirizzo Web:*

***<http://www.tecnetdati.it>***

**INFORMAZIONI** ..... pag. **2**

---

**L'ELENCO** ..... pag. **4**

---

**I CORSI** ..... pag. **6**

---

# INFORMAZIONI

## I PERCORSI DI FORMAZIONE 2010

---

Dal 1990 Tecnet Dati è sinonimo di formazione di alta qualità, lo dicono le centinaia di persone che hanno seguito i nostri corsi sia in House sia a Calendario. Nel corso degli anni i nostri clienti hanno apprezzato lo stile e le competenze dei nostri docenti, nonché i contenuti sempre orientati alle più recenti innovazioni ed al loro pratico utilizzo nei diversi contesti aziendali.

Oggi rinnoviamo il nostro impegno con un'offerta estesa ed aggiornata nei contenuti e nelle modalità di fruizione.

Le informazioni dettagliate su condizioni economiche, modalità operative e date di possibile effettuazione delle sessioni possono essere richieste a:

### **Tecnet Dati S.r.l.**

Corso Svizzera, 185 – 10149 Torino

Tel.011/77.18.090 – Fax 011/77.18.092

E-mail: [formazione@tecnetdati.it](mailto:formazione@tecnetdati.it)

## CORSI IN HOUSE

---

TecnetDati risponde alle esigenze di formazione aziendale con una proposta di corsi programmabili presso le sedi dei nostri clienti e adattabili alle loro specifiche esigenze. Le competenze e l'esperienza dei consulenti Tecnet Dati sono al servizio dell'azienda sin dal momento della progettazione del corso di formazione.

La formazione ad hoc produce valore e crea vantaggi significativi soprattutto se misurati in relazione ad un insieme di elementi fondamentali:

- la discussione e la condivisione della strategia e degli **obiettivi formativi aziendali**;
- l'analisi dei **fabbisogni specifici** delle risorse presenti all'interno della struttura organizzativa;
- la **creazione di materiali didattici**, business case e software di simulazione ad hoc, realizzati in funzione del contesto aziendale;
- la messa a fuoco di **modalità di verifica dell'apprendimento**, secondo logiche di breve e di medio termine.

## CORSI A CALENDARIO

---

I corsi contenuti in questo catalogo vengono promossi in veste pubblica (interaziendale) in partnership con un'azienda leader del settore della formazione aziendale, **Technology Transfer**.

Durante il corso Tecnet Dati fornisce il materiale di consultazione che è organizzato in modo da agevolare l'approfondimento anche dopo l'intervento formativo.

Tecnet Dati può offrire ai propri clienti, sotto forma di check periodici, di tutoring e di consulenza, il supporto per l'assistenza agli allievi durante il primo momento operativo, nel proprio contesto lavorativo, delle nozioni apprese durante il corso.

## QUICK TRAINING PROGRAM

---

Il Quick Training Program è il programma creato da Tecnet Dati per accompagnare le aziende in progetti di cambiamento che coinvolgono diversi livelli aziendali attraverso percorsi di **formazione ad hoc, workshop, consulenza e tutoring.**

Quick Training Program rappresenta il punto di riferimento per la progettazione e la realizzazione di programmi di formazione caratterizzati da:

- Elevata qualità in tutte le fasi di realizzazione del progetto
- Ampia portata dell'intervento nella collaborazione efficace a importanti progetti di cambiamento aziendali all'interno del settore IT
- Alta innovazione attraverso modalità quali la formazione esperienziale, one-to-one e una forte personalizzazione

## FORMAZIONE CERTIFICATA

---



Certificazione **ISO9001:2000** per lo sviluppo del software e l'erogazione dei servizi professionali.



Riconoscimento internazionale come **PMI Registered Education Provider** nell'ambito della formazione sul project management.

## DOCENTI

---

I docenti Tecnet Dati sono professionisti con anni di esperienza ed una specifica preparazione in campo didattico.

Il loro aggiornamento professionale nei rispettivi settori di competenza è garantito dal fatto che tutti i docenti affiancano all'attività di formazione un'intensa attività di consulenza nei confronti di realtà aziendali eterogenee.

L'attenzione verso gli standard condivisi è garantita dalle certificazioni individuali ottenute dai nostri docenti (per esempio PMI, IFPUG e ITIL). Un insieme qualificato di professionisti e aziende che operano nei settori della comunicazione e delle tecnologie informatiche, con cui Tecnet Dati ha stabilito rapporti di partnership orientati al lungo periodo, collabora alla realizzazione del nostro programma di formazione.

# L'ELENCO

## **ORGANIZZAZIONE E TECNICHE PER LO SVILUPPO**

---

Analisi dei dati	6
Analisi delle funzioni	7
Analisi e miglioramento dei processi aziendali	8
Gestione dei requisiti e casi d'uso	10
Guida all'utilizzo pratico dei Function Point	11
Function Point COSMIC 3.0	12
La qualità nel processo di sviluppo e manutenzione del software	14
Service Level Agreement	15
Stima di tempi e costi dello sviluppo software	16
Testing di applicazioni	17
Advanced testing	18

## **UML**

---

Pattern per la progettazione del software	19
Analisi Object Oriented con UML	20
Disegno Object Oriented con UML	22
Workshop di analisi e disegno UML	24
Model Driven Development	25

## **PROJECT MANAGEMENT**

---

Gestione di progetti (codice PMGGEP)	26
Gestione di progetti: avanzato (codice PMGADV)	27
Project Portfolio Management (codice PMGP3M)	28
Extreme Project Management (codice PMGXPM)	29

## **WEB, ARCHITETTURE E TECNOLOGIE**

---

Sviluppo di applicazioni con XML e Web Services	30
Sviluppo di applicazioni Web con AJAX	31
Web 2.0 e crossmedialità al servizio del business	32
Web 2.0: tecnologie e problematiche	34
La tecnologia Open Source in azienda	35
Programmazione JAVA	36

## **SOA**

---

SOA (Service Oriented Architectures): Best Practices	37
SOA per l'integrazione delle applicazioni aziendali	38
Implementare i processi di business con SOA e BPM	39

## **DATA WAREHOUSE**

---

Data Warehouse: architettura e principi	40
La progettazione del Data Warehouse	41

## **CAPACITY MANAGEMENT**

---

Misurare e mantenere le prestazioni delle applicazioni web	42
Capacity planning	43
Test di carico per il collaudo e per il tuning prestazionale	44
Fondamenti di ITIL® V3	45

## **FAMIGLIA DB2**

---

DB2 z/OS: architettura e funzioni	46
DB2 z/OS: sviluppo di applicazioni orientato alle performance	47

## **ORACLE**

---

Oracle 11g: architettura e funzioni	48
-------------------------------------	----

# ANALISI DEI DATI

## DESCRIZIONE ►

---

L'analisi dei dati è un'attività cruciale per la realizzazione dei sistemi informativi, in quanto ha un impatto diretto su tutte le fasi successive dello sviluppo. La definizione del modello dati è pertanto centrale per l'analisi sia in ottica strutturata che in ottica Object Oriented. Il corso presenta un insieme di tecniche (Entity-Relationship, normalizzazione) rivolte alla progettazione di un modello dei dati indipendente da ogni particolare ambiente di sviluppo tecnologico. Introduce inoltre i fondamenti per la derivazione di un diagramma delle classi di oggetti con la notazione UML. L'apprendimento, che avviene mediante un'alternanza di sessioni teoriche e di esercitazioni in aula, non richiede come prerequisito nessuna particolare conoscenza in campo informatico.

A conclusione del corso vengono svolte alcune considerazioni sull'implementazione del modello concettuale in basi di dati effettive, con particolare riferimento agli ambienti relazionali e alcuni cenni al data warehouse.

## DESTINATARI ▼

---

- Capi progetto
- Analisti
- Analisti programmatori
- Data administrator
- Data base administrator
- Utenti coinvolti nello sviluppo

## PREREQUISITI ■

---

Conoscenza generale delle problematiche relative allo sviluppo.

## CONTENUTI ●

---

- Introduzione - il ruolo dei dati nei sistemi informativi aziendali, la progettazione delle basi di dati
- Il modello Entity/Relationship - entità, relationship, attributi, domini, chiavi candidate e primaria, gerarchie di specializzazione
- Data type non tradizionali: immagini, suoni, oggetti complessi
- Il modello relazionale - struttura e manipolazione dei dati, corrispondenze con il modello Entity/Relationship
- Teoria della normalizzazione - il processo di normalizzazione, dipendenze funzionali, forme normali (dalla prima alla quinta forma normale)
- Definizione dei vincoli di integrità
- Costruzione dello schema concettuale-individuazione e raccolta dei dati, convenzioni di descrizione e denominazione, schemi concettuali analitici e di sintesi
- Il Reverse Engineering - tecniche e approcci, eliminazione di ridondanze e di altre caratteristiche "fisiche" degli archivi esistenti, determinazione di chiavi e relationship, integrazione delle entità
- Definizione e gestione di modelli complessi - integrazione dei modelli di progetto in modelli aziendali o di area, problematiche di proprietà e di condivisione delle informazioni, gestione dei dati storici
- Il diagramma delle classi di oggetti di UML - raffronto con l'Entity/Relationship
- Considerazioni sul disegno di data base - derivazione del modello logico e sua ottimizzazione, lo star schema del data warehouse
- Caso studio ed esercitazioni

## DURATA ◆

---

3 giorni

# ANALISI DELLE FUNZIONI

## DESCRIZIONE ►

---

Il corso presenta un insieme di tecniche e di modelli di analisi (Scomposizione funzionale, Dependency Diagram, Data Flow Diagram, State Transition Diagram, Action Diagram) per l'individuazione e la definizione delle esigenze funzionali di un sistema applicativo.

Sul piano teorico il corso si sofferma in particolare sulla tecnica di modellazione basata sui "Data Flow Diagram" (DFD), presentata in un quadro di riferimento ispirato ai principi dell'Analisi Strutturata e dell'Information Engineering.

Il corso prevede una serie di esercitazioni pratiche di difficoltà crescente, per permettere un'assimilazione effettiva dei concetti e delle tecniche proposti in sede teorica.

## DESTINATARI ▼

---

- Capi progetto
- Analisti
- Data administrator
- Data base administrator

## PREREQUISITI ■

---

Conoscenza generale delle problematiche relative allo sviluppo.

## CONTENUTI ●

---

- Introduzione - obiettivo dell'analisi, collocazione nel ciclo di vita dei sistemi, interazione con l'analisi dati
- Scomposizione funzionale
- Dependency Diagram
- Data Flow Diagram (DFD), concetti base
- Il diagramma di contesto
- Individuazione degli eventi
- Criteri per la scomposizione di un sistema
- Definizione della logica di un processo elementare
- Rappresentazione dei processi di regolazione e controllo di un sistema
- Diagrammi di transizione di stato
- Preparazione delle fasi successive di disegno tecnico e realizzazione
- Caso studio ed esercitazioni

## DURATA ◆

---

3 giorni

# ANALISI E MIGLIORAMENTO DEI PROCESSI AZIENDALI

## DESCRIZIONE ►

Il corso fornisce metodi e tecniche per analizzare, definire, ottimizzare e monitorare i processi aziendali, allo scopo di creare processi integrati e orientati al business aziendale.

Dopo aver fornito una panoramica sulla terminologia comune e sugli elementi costitutivi dei processi, il corso illustra l'approccio metodologico utilizzato ed i fattori chiave da considerare nell'analisi dei processi.

Per migliorare i processi aziendali è necessario infatti conoscerne i punti di forza e di debolezza, le eventuali inefficienze e colli di bottiglia, ed è quindi indispensabile misurarne le prestazioni.

Vengono presentati i principali standard di rappresentazione dei processi ed i criteri e le tecniche per la revisione, ottimizzazione ed implementazione del modello futuro con particolare attenzione al ruolo svolto dall'IT nel passaggio dai processi ai sistemi informatici.

## DESTINATARI ▼

- Capi progetto
- Analisti IT e di business
- Specialisti di metodi e standard

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- **Terminologia e aspetti generali dell'analisi dei processi**
  - Definizione di processo
  - L'azienda e i processi organizzativi:
    - classificazione dei processi;
    - organigrammi e funzionigrammi, mansionari;
    - modelli organizzativi e strutture snelle.
  - Approccio funzionale vs approccio per processi
  - Gestione dei processi
  - Elementi costitutivi dei processi:
    - attività, operazione, input/output, process owner, ...;
    - relazione tra processo e procedura.
  - Processi vs. progetti/commesse
  - Un po' di storia
- **Analisi situazione attuale (as-is)**
  - Definizione obiettivi e ambito (diagramma di contesto)
  - Rilevazione:
    - Fonti informative per la rilevazione (interviste, questionari, ...);
    - Attori (unità organizzative);
    - Procedure, processi e tempi (la value stream map);
    - Applicazioni informatiche di supporto;
    - Carichi di lavoro (misurazione, stima, parametrizzazione).
- **Rappresentazione dei processi**
  - Modellazione con BPMN
  - Cenni su altri linguaggi di modellazione (DFD, UML,..)
  - Strumenti di modellazione

- **Definizione modello futuro (to-be)**
  - Analisi criticità
  - Identificazione aree di miglioramento e innovazione (es. tecnologie,..)
  - Razionalizzazione delle attività
  - Ipotesi di soluzione e riprogettazione del processo
  - Definizione KPI
- **Implementazione modello futuro (approccio integrato)**
  - Job design e gestione del cambiamento (gestione motivazione, empowerment attori, comunicazione, metodi)
  - Dai processi ai sistemi informatici:
    - i passi per l'implementazione; il ruolo dell'IT nel cambiamento; cenni sulle opzioni tecnologiche.
- **Misurazione performance e monitoraggio processi**
- **Caso studio**

## DURATA ◆

---

3 giorni

# GESTIONE DEI REQUISITI E CASI D'USO

## DESCRIZIONE ►

Per mantenere il passo con l'aumentare della complessità dei sistemi software e delle richieste degli utenti, dobbiamo rendere sempre più mature ed efficienti le modalità in cui sviluppiamo, testiamo e gestiamo i nostri progetti. Il primo passo verso questa crescita consiste nel migliorare il processo di gestione dei requisiti. La gestione dei requisiti è un approccio sistematico per identificare, organizzare, comunicare, documentare i requisiti di un sistema software e gestirne i cambiamenti, con l'obiettivo di ridurre costi e rischi correlati allo sviluppo e all'evoluzione dei sistemi informativi.

Il corso fornisce ai partecipanti un quadro di riferimento per tradurre le necessità dei committenti in requisiti qualitativamente validi e verificabili, gestire con un processo strutturato il ciclo di vita dei requisiti e utilizzare la tecnica dei casi d'uso nella modellazione dei processi di business e dei sistemi informatici. Il corso prevede inoltre una serie di esercitazioni per permettere un apprendimento effettivo dei concetti e delle tecniche proposte nello svolgimento teorico degli argomenti.

## DESTINATARI ▼

- Capi progetto
- Analisti
- Esperti applicativi coinvolti nella gestione dei requisiti
- Addetti alle funzioni di assicurazione e controllo della qualità

## PREREQUISITI ■

Conoscenza generale delle problematiche relative allo sviluppo e all'evoluzione dei sistemi informativi.

## CONTENUTI ●

- Gestire i requisiti - Definizione di requisito, obiettivi, requisiti e sistema, ruolo dei requisiti nel processo di negoziazione con i committenti, ruolo dei requisiti nella progettazione e testing dei sistemi SW, il processo di gestione dei requisiti, classificazione dei requisiti, requisiti e cicli di vita del sistema, la gestione dei requisiti secondo il Capability Maturity Model del SEI e gli standard ISO 9000, la qualità del prodotto secondo ISO 9126
- Scoprire i requisiti - individuare gli obiettivi di business, identificare gli interlocutori, tecniche per l'individuazione dei requisiti: interviste, incontri, workshop, brainstorming, Joint Application Design, analisi documentazione esistente, simulazione di scenari - prototipi, i casi d'uso
- Specificare i requisiti - specifica efficace, analisi e controllo dei requisiti, attributi del requisito, assegnare una priorità ai requisiti, organizzare i requisiti, template e standard per la documentazione dei requisiti, tracciabilità dei requisiti, strumenti per la gestione dei requisiti
- Gestire l'evoluzione dei requisiti - un processo per gestire le modifiche
- Casi d'uso - definizioni di caso d'uso, attori e contesto del sistema, relazioni tra attori e casi d'uso, come identificare i casi d'uso, il modello dei casi d'uso e rappresentazione UML, scenari di un caso d'uso, casi d'uso e prototipi di interazione, casi d'uso (modelli di business e informatico), granularità dei casi d'uso, relazioni tra casi d'uso, template per la documentazione dei casi d'uso, partizionamento del sistema, ruolo dei casi d'uso nel ciclo di vita del sistema
- Caso studio ed esercizi

## DURATA ◆

2 giorni

# GUIDA ALL'UTILIZZO PRATICO DEI FUNCTION POINT

## DESCRIZIONE ►

---

Il calcolo dei function point è attualmente la tecnica più efficace per dimensionare i progetti software in modo comprensibile da parte dell'utente, in quanto permette di correlare lo sforzo necessario per produrre il software ai prodotti effettivamente rilasciati. La tecnica ha ormai raggiunto un alto grado di standardizzazione. E' sempre più utilizzata, inoltre, come metrica oggettiva per il dimensionamento del software, nella contrattazione tra aziende e fornitori.

L'applicazione dei Function Point, sebbene guidata da precise regole codificate (standard IFPUG 4.2), può tuttavia lasciare spazio, nella pratica, a dubbi e interpretazioni soggettive.

Il corso, articolato in forma di workshop, intende fornire un aiuto pratico per il conteggio dei Function Point, attraverso la presentazione e la soluzione congiunta delle situazioni applicative più ricorrenti nell'ambito dello sviluppo dei sistemi.

Vengono inoltre affrontati aspetti inerenti l'utilizzo dei Function Point nel governo dei progetti e nella stesura di contratti per la realizzazione del software.

## DESTINATARI ▼

---

- Capi progetto
- Analisti
- Specialisti nel campo delle metriche

## PREREQUISITI ■

---

Conoscenza delle problematiche di dimensionamento del software e dello sviluppo applicativo.

## CONTENUTI ●

---

- **Il metodo IFPUG 4.2** - criteri di utilizzo, ambiti applicativi, elementi e regole di conteggio, cambiamenti rispetto alla versione 4.1
- **Problematiche applicative** - stima dei Function Point (FP) nelle fasi "alte" di progetto, rilevazione del portafoglio applicativo esistente, dimensionamento dei progetti di manutenzione evolutiva in base all'effettivo sforzo richiesto, riutilizzo di componenti software già esistenti
- **Applicazione dei Function Point ai nuovi paradigmi di sviluppo** – Object Oriented con UML, applicazioni web, Data Warehouse
- **Cenni all'utilizzo dei Function Point nei contratti**
- **Indicatori per il governo di progetto tramite i FP**
- **Impiego dei Function Point per la determinazione del Service Level Agreement**
- **Caso studio 1: calcolo dei FP per un progetto di nuovo sviluppo**
- **Caso studio 2: calcolo dei FP per un progetto di manutenzione evolutiva**
- **Caso studio 3: calcolo dei FP nell'ambito di un'applicazione sviluppata ad oggetti**

## DURATA ◆

---

3 giorni

# FUNCTION POINT COSMIC 3.0

## DESCRIZIONE ►

Disporre di una metrica funzionale per il dimensionamento del software è ormai un'esigenza primaria in campo informatico. Nonostante la diffusione storica della FPA (Function Point Analysis) secondo le regole IFPUG, emergono alcuni limiti nella sua applicazione in contesti e paradigmi di sviluppo moderni.

Per questo dal 1998 si propone un nuovo metodo di misurazione funzionale chiamato 'COSMIC', dal nome dell'organismo internazionale che lo ha realizzato: the Common Software Measurement International Consortium.

Il metodo COSMIC si va costantemente diffondendo; l'ISO/IEC lo ha riconosciuto, al pari del metodo IFPUG Unadjusted, come metrica funzionale rispondente ai propri standard (rif. ISO/IEC 19761:2003). È chiaro che i Function Point COSMIC, pur appartenendo alla famiglia delle metriche funzionali, non sono identici ai Function Point IFPUG (pur essendo possibile una conversione dagli uni agli altri): i Function Point COSMIC permettono la misurazione di applicazioni "tradizionali", con risultati analoghi a quelli IFPUG, ma, in virtù di alcune caratteristiche particolarmente interessanti, si applicano con altrettanto successo anche a componenti software "particolari", per esempio di tipo GIS (cartografico), SOA (Service Oriented Architecture), web services, middleware evoluto, etc., e a processi software complessi, come nel caso dei batch.

Il corso fornisce gli elementi di base per l'apprendimento del metodo COSMIC Function Point 3.0 (2007) e per la sua applicazione pratica e le regole complete per il conteggio e la stima di applicazioni, progetti di sviluppo e progetti di manutenzione evolutiva del software. Il corso prevede inoltre l'analisi e la discussione di un Caso studio mutuato da esperienze pratiche, la discussione di casi proposti dai partecipanti, un confronto tra le metriche IFPUG e COSMIC e un'analisi delle modalità di conversione tra i due metodi.

## DESTINATARI ▼

- Capi progetto
- Analisti
- Specialisti nel campo delle metriche

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- **Introduzione**
- **Standard e riferimenti normativi**
- **Regole di misurazione Function Point COSMIC 3.0**
  - modello del software generale e del contesto
  - fase di strategia:
    - scopo della misurazione;
    - ambito della misurazione (strati, componenti);
    - utenti funzionali;
    - livelli di granularità (standard e superiori).
  - fase di mappatura:
    - processi funzionali;
    - oggetti di interesse e gruppi di dati.
  - fase di misurazione:
    - movimenti di dati (Entry, Exit, Read, Write);
    - funzione di misurazione;
    - aggregazione dei risultati e formule finali;
    - misurazione delle modifiche (manutenzioni evolutive).

- **Esempi** (applicazioni tradizionali, middleware, web services, etc.) & esercizi
- **Confronto & convertibilità** misure COSMIC-IFPUG e viceversa
- **Temi avanzati**
  - livelli di scomposizione (ambito della misurazione)
  - benchmarking
  - stima anticipata
  - modalità di reporting
- **Casi di studio**
  - proposti dal docente (porzioni)
  - suggeriti dai partecipanti (porzioni)

## **DURATA** ◆

---

3 giorni

# LA QUALITÀ NEL PROCESSO DI SVILUPPO E MANUTENZIONE DEL SOFTWARE

## DESCRIZIONE ►

Promuovere il miglioramento continuo del processo di sviluppo e manutenzione del software, mediante l'introduzione di un sistema qualità, è un obiettivo cui tutte le aziende stanno guardando con crescente interesse al fine di aumentare la propria competitività nel settore aziendale di riferimento. Prescindendo dall'introduzione del sistema qualità aziendale, tuttavia, vi sono elementi della qualità che possono essere adottati immediatamente al fine di migliorare l'efficienza dei progetti software.

Il corso illustra le linee guida per la progettazione, la realizzazione e la successiva (eventuale) certificazione di un sistema qualità aziendale conforme alle norme di riferimento del settore informatico (ISO9001; ISO/IEC 9126, 12207 e 15504; SEI CMM ).

Il corso intende inoltre fornire elementi teorici e pratici, mutuati da esperienze reali, che consentono un'immediata applicazione delle nozioni apprese ai progetti di realizzazione del software nell'ambito della propria realtà operativa.

## DESTINATARI ▼

- Responsabili dei sistemi informativi
- Personale coinvolto nella gestione della qualità (specialisti di metodi e standard, ...)
- Figure professionali coinvolte nel processo di sviluppo del software (capi progetto, analisti, ...)

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- Concetti base su qualità e certificazione
- Le norme applicabili al processo di sviluppo e manutenzione del software – interpretazione dei requisiti normativi
- Linee guida per l'impostazione del sistema qualità: analisi e re-engineering dei processi chiave relativi alla produzione del software - rilevazione dei requisiti, riesame del contratto, gestione dei progetti, gestione risorse umane, gestione infrastrutture, progettazione, produzione, approvvigionamenti, gestione della documentazione, verifiche ispettive, gestione delle non conformità e delle azioni correttive/preventive, piani di miglioramento, riesame della direzione, gestione del sistema
- Applicazione del sistema qualità alla gestione dei progetti: definizione e riesame dei requisiti, riesame del contratto, piano di qualità, ciclo di vita, analisi dei rischi, controllo della progettazione, gestione della configurazione, testing, controllo qualità, gestione anomalie, misurazioni [della qualità] del processo e del prodotto, monitoraggio e controllo del progetto, audit di progetto
- Valutazione dell'efficacia dei processi software e gestione del miglioramento
- Predisposizione alla certificazione: cenni

## DURATA ◆

2 giorni

# SERVICE LEVEL AGREEMENT

## DESCRIZIONE ►

La forte evoluzione delle tecnologie informatiche ed il volume sempre crescente del patrimonio software da gestire, costituiscono due delle principali criticità dei settori IT aziendali. L'impegno necessario per la manutenzione del software applicativo esistente infatti, condiziona e limita pesantemente lo sviluppo delle nuove "soluzioni", richieste dai reparti operativi aziendali ed in linea con i trend evolutivi IT. In questo contesto molte aziende si orientano verso la concessione in outsourcing di parte o tutto il servizio di Application Management, con il duplice obiettivo di definire in modo chiaro i costi relativi e liberare risorse per l'evoluzione dei Sistemi Informatici in modo coerente con il proprio core business.

La definizione dei livelli di servizio tra tutti i soggetti coinvolti nell'erogazione del servizio stesso, Cliente-Fornitore-Subfornitori, è una condizione fondamentale per il raggiungimento degli obiettivi di qualità ed efficienza richiesti.

Il corso costituisce una guida alla definizione di servizi di Application Management ed all'impostazione/redazione di un documento di Service Level Agreement. Il corso introduce all'utilizzo dei Function Point quale strumento per prima esprimere e poi monitorare in modo oggettivo i livelli di servizio.

## DESTINATARI ▼

- Account Manager
- Service Manager
- Sales Manager
- Application Manager

## PREREQUISITI ■

Conoscenza delle problematiche di manutenzione e assistenza di applicazioni software di grandi dimensioni.

## CONTENUTI ●

- Il servizio di Application Management e le sue componenti - Analisi delle finalità e dei contenuti del servizio, mirata a facilitare l'individuazione del "perimetro" e la suddivisione nelle singole componenti, Assistenza, Manutenzione: la suddivisione tecnica e quella contrattuale, Definizione delle priorità di intervento.
- L'utilizzo dei Key Performance Indicator per la misurazione dei Livelli di Servizio - Utilizzo delle metriche di misurazione del software (Function Point) nei servizi di Application Management, I KPI più utilizzati.
- Il Sistema di Misurazione delle Performances ed il Reporting - La definizione di un "accordo" sui Livelli di Servizio implica un corrispondente adeguamento operativo, Il Sistema di Misurazione e la conseguente organizzazione operativa, La gestione del Reporting.
- I contenuti di un documento di Service Level Agreement - Analisi delle singole componenti di uno SLA.
- L'utilizzo del documento di SLA come strumento operativo per l'erogazione di un servizio qualitativamente valido.
- Il periodo di avviamento del Servizio - Gli obiettivi e gli aspetti organizzativi.
- Caso di studio.
- Rilettura di uno SLA, evidenziando gli aspetti salienti in relazione a quanto trattato nell'ambito del corso.

## DURATA ◆

2 giorni

# STIMA DI TEMPI E COSTI DELLO SVILUPPO SOFTWARE

## DESCRIZIONE ►

Stimare in modo efficace i tempi e i costi di un'attività di sviluppo software, che si tratti di un progetto articolato o di un semplice intervento di manutenzione, è una pratica con cui si cimentano costantemente i project manager ma anche, in generale, tutti gli addetti alla realizzazione del software. La stima basata sull'esperienza personale, che ad oggi rimane ancora la più consueta, pur potendo portare a buoni risultati presta, sempre e comunque, il fianco a critiche.

Nasce quindi l'esigenza di disporre di tecniche di stima più oggettive e formali, basate su criteri più scientifici, raffrontabili in ambienti eterogenei.

Il corso, dopo aver ripreso le problematiche più comuni nella realizzazione delle stime, illustra alcune tecniche di stima largamente utilizzate e le inquadra nell'ambito di un processo che ne guida l'utilizzo. Nel corso sono previsti ampi ed esaurienti riferimenti alla teoria dei Function Points, oggi strumento cardine per il dimensionamento del software, e al loro utilizzo nell'ambito delle stime.

### *Al termine del corso, i partecipanti:*

- hanno appreso i principi base dei Function Point e le problematiche legate alla loro applicazione (per poter iniziare ad applicare la tecnica a casi pratici);
- hanno chiara idea dei fattori di cui tenere conto nelle stime dei tempi e dei costi dello sviluppo software;
- conoscono le più diffuse tecniche di stima (applicabili in funzione delle caratteristiche e dimensioni del progetto) e le modalità con cui calare le stime in un processo di sviluppo del software
- hanno conoscenza dei principi fondamentali di un processo di gestione delle stime

## DESTINATARI ▼

- Capi progetto, team leader
- Analisti funzionali
- Specialisti nel campo delle metriche del software

## PREREQUISITI ■

Conoscenza delle problematiche legate al dimensionamento del software e alla stima di tempi e costi dello sviluppo del software.

## CONTENUTI ●

- **Le problematiche legate alle stime**
- **Il problema del dimensionamento del software** – le metriche del software, i Function Points: elementi teorici di base per conteggio e stima
- **Stima dell'effort** – la stima per analogia; la stima bottom-up; la stima top-down basata sui workproduct; la stima basata sui Function Point, la stima basata su modelli parametrici
- **Il processo di stima calato nel processo di sviluppo** – quando e come stimare (i momenti di stima e le tecniche più idonee); la gestione delle stime in ambito di progetto e in ambito aziendale
- **Gli strumenti per la stima** – templates, repository, ...
- **Caso studio 1** – stima dell'effort di un piccolo intervento di manutenzione evolutiva
- **Caso studio 2** – stima di tempi e costi di un progetto in "fase alta"
- **Caso studio 3** – stima finale di un progetto

## DURATA ◆

2 giorni

# TESTING DI APPLICAZIONI

## DESCRIZIONE ►

Il testing ha come obiettivo la verifica del funzionamento di un prodotto software rispetto ai requisiti del sistema.

Saper impostare ed eseguire una strategia di test è elemento chiave per garantire un elevato standard di qualità del software e richiede una corretta visione organizzativa e metodologica.

Il corso illustra in modo esauriente principi, strategie e tecniche su cui si fonda il testing: approccio di tipo sia funzionale ("black box" testing), sia strutturale ("white box" testing); strategia basata su più livelli e tipologie di test; esistenza di un ciclo di vita del test con conseguente definizione di attività, responsabilità, prodotti e metriche.

Partendo da questa base, si passa a specializzare le tecniche in funzione di diversi ambiti metodologici e tecnologici, quali ad esempio le applicazioni Web, la SOA e i metodi Agili.

Il corso fornisce inoltre gli elementi per inquadrare il testing nell'ambito del sistema qualità alla luce delle norme ISO/IEC 9126.

## DESTINATARI ▼

- Responsabili progetti Software
- Personale dell'assicurazione e del controllo di qualità
- Analisti e progettisti di applicazioni Software

## PREREQUISITI ■

Conoscenza delle problematiche di sviluppo applicativo.

## CONTENUTI ●

- Introduzione - obiettivo del testing, il testing nell'ambito del Sistema Qualità, fattori di qualità del software, le norme ISO/IEC 9126
- Livelli di test (Unit, Integration e System testing, Collaudo e Accettazione)
- Tipologie di test: funzionale e non funzionale
- Finalità di testing: verifica e validazione, "bug fixing", non regressione
- Tecniche di test: white, black e grey box
- Il processo di testing - Pianificazione (definizione della Strategia di test in funzione dei requisiti di qualità e degli obiettivi concordati con l'utente, livello di copertura funzionale e topologico, Piano dei test), Progettazione (Checklist, Casi di Test, Classi di equivalenza, Specifiche di Test, Unità di Lancio), Esecuzione (aspetti organizzativi, Scheda esito test, Rapporto anomalie), Gestione delle anomalie
- Adattare il testing in funzione delle caratteristiche del progetto - testing di applicazioni Web, testing di applicazioni SOA, il testing nei metodi iterativi e nell'Agile
- Cenni sulle metriche di prodotto e di processo - indice di difettosità del prodotto, efficienza nella rimozione dei difetti, valutazione dell'efficacia dei test, determinazione dell'impegno e valutazione oggettiva di parametri di qualità mediante la metrica dei Function Point
- Strumenti per la conduzione e l'automazione dei test
- Casi studio ed esercizi

## DURATA ◆

3 giorni

# ADVANCED TESTING

## DESCRIZIONE ►

Rispetto al corso base, che esamina le principali tecniche e le inquadra nell'ambito del ciclo di vita del software, il corso avanzato estende gli argomenti affrontati in precedenza ponendo l'accento su tre diversi aspetti che riguardano l'esercizio del test in un ambiente software "asset oriented" e precisamente Business, Metodologie e Governance.

L'aspetto Business comprende la gestione e il test dei Requisiti (funzionali e non). Le Metodologie riprendono ed affinano le tecniche specifiche e si soffermano sulla gestione del processo di Testing, mentre nella Governance rientrano la definizione delle strategie di qualità, il Risk Management, la raccolta e l'analisi risultati e i KPI per la costruzione del Cruscotto della qualità.

Il corso si sofferma inoltre sull'organizzazione e l'implementazione di una strategia di Test nel caso di Outsourcing di progetti in cui vi sono fornitori esterni coinvolti nei processi di Application Management.

Concludono il corso alcune considerazioni legate alla "vendita" del test ai clienti al fine di renderli consapevoli dei costi ma anche dei benefici.

## DESTINATARI ▼

- Analisti, Designer e Architetti
- Responsabili di programma/progetto
- Capi Progetto
- Personale dell'assicurazione e del controllo di qualità

## PREREQUISITI ■

Conoscenza delle problematiche di base di testing  
 Conoscenza e applicazione gestione requisiti  
 Conoscenza di base tecniche di software sizing

## CONTENUTI ●

- I tre aspetti della strategia di Testing: Business, Metodologie e Governance
- Business - Test dei requisiti funzionali e non, per esempio rispetto di esigenze di certificazione (PCI, Privacy ...), come la Risk Analysis può aiutare a definire una strategia di Testing
- Metodologie - Approfondimento delle tecniche di testing, ovvero Progettazione (Granularità dei Requisiti, Test di algoritmi e Rule Engine, Array ortogonali, ottimizzazione del numero dei Test Case) e Test di Non Regressione, Test Management, ovvero people (organizzazione e gestione del gruppo di Testing), process (Formalizzazione e gestione delle fasi di processo), metriche (stime di effort e difettosità residua e complessiva costruite sulla dimensione delle applicazioni in Function Point, metriche di prodotto e di processo)
- Governance - La strategia di qualità, la raccolta e l'analisi dei risultati, la definizione dei KPI di processo e lo sviluppo di un Data Warehouse (estimation vs actual) come base per l'implementazione di un sistema di qualità misurabile e tracciata
- Testing in regime di Outsourcing - Il Testing in presenza di terze parti nell'AM, dalla contrapposizione alla cooperazione, condivisione delle tecniche e degli strumenti di test e delle metriche
- Qualificare e saper comunicare la qualità al Cliente: perché il test costa caro ma vale ancor di più?

## DURATA ◆

2 giorni

# PATTERN PER LA PROGETTAZIONE DEL SOFTWARE

## DESCRIZIONE ►

I pattern di progettazione sono un “nocciolo duro” di best practices e soluzioni standard a problemi ricorrenti cui attingere abbondantemente nella fase di progettazione del software. Rappresentano un modo semplice ed efficace di riutilizzare le migliori tecniche ed esperienze maturate nella realizzazione di applicazioni robuste, modulari, scalabili: per questo oggi hanno assunto tanta importanza. Il corso, fatta una premessa sui concetti base relativi ai pattern e all’UML, illustra in modo approfondito i pattern più utili e diffusi.

I pattern vengono esposti prima in forma teorica e “platform independent”, avulsa da qualsiasi legame con specifiche tecnologie, quindi calandoli in casi tipici ed esempi pratici di utilizzo basati su specifiche piattaforme (J2EE in primis).

Nella esposizione ci si sofferma su come sfruttare i pattern per intervenire su alcuni, frequenti, problemi e situazioni progettuali non ideali, introducendo un miglioramento oggettivo.

Viene inoltre trattato il tema dell’utilizzo dei pattern in ambito di progetto: i criteri per scegliere i pattern da utilizzare nel proprio progetto, le modalità con cui diffonderli a progettisti e sviluppatori.

La notazione di riferimento utilizzata nel corso è UML: lo Unified Modeling Language, che nella nuova versione 2 è ancora più rispondente alle esigenze di modellazione di tutte le fasi della realizzazione.

## DESTINATARI ▼

- Architetti di sistema
- Analisti programmatori
- Specialisti di metodi e standard

## PREREQUISITI ■

Corso di “Analisi Object-Oriented con UML” o conoscenze equivalenti.

È molto utile l’esperienza di progettazione e programmazione con linguaggi Object Oriented

## CONTENUTI ●

- **UML e MDA: fondamentali**
  - UML: convergenza dei metodi, convergenza sul metodo
  - Principali artefatti di UML 2
  - concetti base dell’approccio Model Driven Architecture (PIM, PSM, ...)
- **Pattern**
  - Cos’è un “pattern”, pattern vs framework
  - Come si descrive un pattern e quali tipi di pattern: la descrizione e la classificazione secondo Eric Gamma
  - Pattern: MVC: il pattern architetturale di riferimento; Layer: altro pattern architetturale importante; Template (o algorithm); Visitor; Composite; Iterator; Proxy; Strategy; Adapter; Decorator; Singleton; State; Façade; Abstract Factory; Publish-Subscribe; Proxy; Data Access Object; Data Transfer Object; GRASP
  - Altri pattern, cenni
  - Usare i pattern nel progetto: scegliere, definire e modellare i propri pattern
  - Implementare i pattern con i framework: usare framework esistenti o creare i propri

## DURATA ◆

2 giorni

# ANALISI OBJECT ORIENTED CON UML

## DESCRIZIONE ►

Per sfruttare appieno le potenzialità delle nuove tecnologie e piattaforme di sviluppo del software (sviluppo per componenti, interoperabilità dei sistemi, riusabilità del codice) è necessario “pensare” in termini object oriented fin dalle prime fasi del ciclo di vita delle applicazioni.

Il corso presenta, con un’alternanza di lezioni teoriche ed esercitazioni, un procedimento generale di Analisi OO basato sulle più recenti esperienze pratiche e proposte metodologiche, e in particolare su UML: lo Unified Modeling Language, oggi giunto alla versione 2.0, di cui adotta notazioni e semantiche. Vengono esaminate le principali tecniche e best practices per progettare, fin dalla fase di analisi, applicazioni robuste ed estensibili.

Il corso fornisce inoltre un workflow di riferimento per le attività da svolgere e una serie di indicazioni pratiche per la produzione dei deliverables di ogni singola attività.

A conclusione vengono illustrate le principali opzioni tecnologiche oggi a disposizione e le linee guida per le successive fasi di disegno e implementazione.

## DESTINATARI ▼

- Data e database administrator
- Capi progetto
- Analisti
- Analisti programmatori
- Specialisti di metodi e standard

## PREREQUISITI ■

Conoscenza generale delle problematiche relative allo sviluppo e nozioni di modellazione E-R.

## CONTENUTI ●

- **Modelli per l’Analisi**
  - l’analisi nel processo di sviluppo
  - l’approccio object oriented
- **Concetti base dell’Object-Oriented**
  - oggetti e classi, istanziazione, incapsulazione
  - messaggi e metodi
  - polimorfismo e ereditarietà
  - abstract data type
  - benefici dell’approccio OO
- **Unified Modeling Language**
  - UML come linguaggio universale di modellazione
  - Diagrammi UML e fasi del processo di sviluppo
- **Casi d’uso e analisi dei requisiti**
  - dai requisiti ai casi d’uso
  - attori e casi d’uso
  - delimitazione del contesto dell’analisi
  - scenari di un caso d’uso
  - relazioni tra casi d’uso
  - partizionamento del sistema in sottosistemi
- **Realizzazione dei casi d’uso (dai casi d’uso ai componenti)**
  - oggetti nel dominio dell’analisi
  - allocazione delle funzionalità del caso d’uso agli oggetti
  - ruoli, responsabilità e collaborazioni degli oggetti

- **Analisi Object Oriented: modello statico**
  - classi e tipi
  - identificatori, attributi e operazioni
  - relazioni fra classi (associazioni, gerarchie di specializzazione, gerarchie di aggregazione)
  - diagrammi delle classi e degli oggetti
  - vincoli di integrità
- **Analisi Object Oriented: modello dinamico**
  - interazioni tra gli oggetti: diagrammi di sequenza e collaborazione
  - ciclo di vita degli oggetti: diagrammi di transizione di stato
  - diagrammi di attività
  - confronto fra diagrammi di attività e diagrammi BPMN
- **Caso studio**

## **DURATA** ◆

---

3 giorni

# DISEGNO OBJECT ORIENTED CON UML

## DESCRIZIONE ►

Il disegno è un momento cruciale dello sviluppo di un'applicazione, in cui si porta a pieno compimento la definizione della sua architettura.

Nelle pratiche abituali, tuttavia, troppo spesso si passa dalle fasi "alte" dell'analisi direttamente all'implementazione, avendo definito in modo sommario l'architettura e non affrontando con la dovuta attenzione e disciplina la fase di progettazione. Si ha così un impatto fortemente negativo su qualità essenziali del software quali le prestazioni, la manutenibilità, l'efficienza, la portabilità. Inoltre non viene favorito il riutilizzo, non solo del codice ma in senso lato.

Il corso illustra inizialmente come si cala la fase di Disegno nei vari tipi di processo di sviluppo, definendone presupposti e risultati; vengono quindi esaminate le tecniche di Disegno ed i principi che guidano la realizzazione delle migliori architetture, attingendo abbondantemente alle "best practices" e ponendo particolare enfasi sui pattern di disegno.

In tale ottica va anche visto l'utilizzo della notazione UML (lo Unified Modeling Language: qui viene fatto riferimento alla versione 2.0) e il riferimento all'approccio MDA, Model Driven Architecture, di cui UML è un fondamento.

Il corso è completato da una serie di cenni all'implementazione e ai fondamenti di progettazione dell'interfaccia utente.

## DESTINATARI ▼

- Architetti di sistema
- Analisti applicativi
- Analisti programmatori
- Specialisti di metodi e standard

## PREREQUISITI ■

Corso di "Analisi Object-Oriented con UML" o conoscenze equivalenti. È utile l'esperienza di programmazione con linguaggi OO.

## CONTENUTI ●

- **UML e processi software**
  - UML: convergenza dei metodi, convergenza sul metodo
  - UML e processi software
  - I principali artefatti di UML
  - Un linguaggio senza processo?
  - Linee guida per uno sviluppo incrementale, iterativo, basato sull'architettura
- **Dall'analisi al disegno**
  - Ruolo dell'Analisi e del Disegno nel Processo software
  - Il Disegno in un processo di tipo iterativo/incrementale, basato sull'architettura e sui casi d'uso
  - Prodotti "ereditati" dall'Analisi: specifiche funzionali, requisiti non funzionali, Diagramma dei Casi d'Uso, Diagramma delle Classi, Diagrammi di Stato e di Attività, Diagrammi di Sequenza; caratteristiche richieste a questi prodotti
  - Estensione e affinamento in Disegno dei diagrammi UML "ereditati"
  - Documentare l'architettura applicativa: concetti base dell'approccio Model Driven Architecture (PIM, PSM, ...)
- **Principi per il disegno object oriented**
  - Coesione e coupling delle componenti
  - Ruoli, Responsabilità e Collaborazioni: contratti e deleghe, distribuzione delle responsabilità
  - Framework e Pattern: concetti fondamentali; come usarli per ottenere economie in fase di progettazione e per favorire il riuso non solo del codice

- Pattern fondamentali : MVC, Façade, Publish-Subscribe, Proxy, Abstract Factory, GRASP, ecc. Quali usare, quando e come
- **Disegno dell'applicazione**
  - Disegno dell'architettura applicativa: partizionamento in Sottosistemi, stratificazione dei Sottosistemi, tecniche di interfacciamento e disaccoppiamento
  - Disegno delle componenti: individuazione delle componenti, definizione delle loro responsabilità e collaborazioni, realizzazione dei diagrammi di disegno (diagrammi di sequenza, delle classi, delle azioni, delle componenti)
- **Linee guida alla implementazione**
  - Ergonomia: impostazione di uno standard di progettazione delle interfacce
  - Problemi applicativi: validazione dell'input, exception handling, accesso a fonti di dati eterogenee
  - Mapping a un DBMS relazionale: classi Vs tabelle, riferimenti all'uso di chiavi artificiali, trigger e stored procedure
  - Cenni all'Aspect Oriented Programming, nuovo paradigma di programmazione
- **Caso studio**

## DURATA ◆

---

3 giorni

# WORKSHOP DI ANALISI E DISEGNO CON UML

## DESCRIZIONE ►

---

UML si sta imponendo all'attenzione di quanti intendono estendere il rigore della notazione formale e dei meta modelli alle fasi "alte" del ciclo di sviluppo, senza sacrificare l'efficienza del processo di sviluppo.

Il Workshop rappresenta un'opportunità per favorire l'adozione e l'utilizzo di UML nelle aziende, in quanto permette di applicare i concetti e le tecniche esposte nei corsi teorici in un Caso studio di adeguata complessità, mutuato da un'esperienza reale.

In tal senso il workshop è il giusto complemento ai corsi di Analisi e Disegno.

I corsi teorici, infatti, sono pensati per trasferire ai partecipanti le conoscenze necessarie per comprendere le potenzialità e le modalità di utilizzo dei metodi UML; in tale ambito le esercitazioni proposte sono volte ad aumentare il livello di comprensione dei partecipanti. Il workshop, viceversa, ha l'obiettivo di permettere ai partecipanti di sperimentare le tecniche di analisi e disegno con UML apprese nei corsi nell'ambito di un progetto pilota.

Il corso si articola in un'alternanza di sessioni operative e momenti di consolidamento e verifica del lavoro svolto, che consentono di valutare assieme al mentore i risultati prodotti e fornire le linee guida per lo svolgimento delle attività.

Al fine di agevolare la definizione dei modelli UML, il workshop sarà svolto con l'ausilio di IBM Rational Rose.

## DESTINATARI ▼

---

- Analisti
- Analisti programmatori
- Capi progetto

## PREREQUISITI ■

---

La partecipazione ai corsi "Analisi Object-Oriented con UML" e "Disegno Object-Oriented con UML" o il possesso di conoscenze equivalenti.

## CONTENUTI ●

---

- Presentazione del Caso studio
- Elaborazione dei risultati dell'analisi (Individuazione e descrizione dei Casi d'Uso, Modello dei Casi d'Uso, Diagramma delle Classi, Diagrammi di Stato, Diagrammi di Sequenza)
- Estensione e affinamento dei modelli dell'analisi e definizione dei modelli del disegno con utilizzo dei pattern di progettazione (Definizione dell'architettura logico-fisica dell'applicazione, Diagramma delle Classi, Diagrammi di Stato, Diagrammi di Sequenza e di Collaborazione)
- Definizione dell'architettura fisica (Package, Diagrammi delle Componenti e di Allocazione delle Componenti (cenni))
- Aspetti implementativi (cenni)

## DURATA ◆

---

3 giorni

# MODEL DRIVEN DEVELOPMENT

## DESCRIZIONE ▶

L'esigenza delle aziende di essere sempre più produttive e di realizzare rapidamente applicazioni, anche per dispositivi e canali di fruizione eterogenei, ha posto il problema di come ottenere, a partire da un modello iniziale, soluzioni funzionanti su architetture e contesti differenti.

A tal fine, numerosi enti di standardizzazione, tra i quali l'OMG, hanno rilasciato un insieme di standard dove la realizzazione di un modello descrittivo dell'applicazione viene considerato come parte integrante e fondamentale del processo di realizzazione del software. I modelli, espressi principalmente in un formalismo basato su UML, vengono sottoposti ad un insieme di trasformazioni che consentono di guidare e automatizzare la realizzazione dell'architettura e della progettazione delle applicazioni, la generazione automatica del codice in base alla piattaforma target e alla generazione e gestione dei casi di test.

Il corso, partendo dalla descrizione di modelli UML, applica i concetti del Model Driven Engineering ad un Caso studio pratico che descrive tutti i passaggi da compiere per la realizzazione di un'applicazione, dalla descrizione dell'architettura, alla generazione automatica del codice fino alla fase di testing.

## DESTINATARI ▼

Tutti coloro che sono interessati agli strumenti di Model Driven Engineering e alla loro applicazione pratica in progetti reali.

## PREREQUISITI ■

Conoscenze base del linguaggio di modellazione UML

## CONTENUTI ●

- **Model Driven Engineering (MDE)** - Che cos'è il Model Driven Engineering, Il concetto di Modello, Definizione di modelli in UML, Trasformazioni
- **Model Driven Architecture (MDA)** - Definizione del modello architetturale, MDA Software Development life cycle, Modelli e matamodelli, Platform Independent Model, Platform Specific Model
- **Model Driven Development (MDD)** - Definizione dei modelli di sviluppo, Generazione automatica del codice, Trasformazioni per la generazione del codice, Agile Model Driven Development (AMDD)
- **Model Driven Testing (MDT)** - Definizione dei livelli di test (Unit, Integration, System,...) tramite Modelli, Data-Driven Testing, Real-Time and Performance Testing, UML Testing Profile
- **Framework e strumenti**

## DURATA ◆

2 giorni

# GESTIONE DI PROGETTI

## (codice PMGGEP)

### DESCRIZIONE ►

Il corso, frutto di un'esperienza maturata nella conduzione di progetti di ampio respiro, affronta il tema del Project Management considerando i principali aspetti implicati: validità di una strategia di "sviluppo per progetti"; individuazione dei momenti maggiormente critici di un progetto; definizione degli obiettivi di progetto; creazione di un piano realistico per il raggiungimento di tali obiettivi; assegnazione dei ruoli e delle responsabilità: attuazione di un'efficace azione di controllo e conduzione del progetto; gestione della qualità; gestione delle risorse umane (Peopleware).

Particolare peso viene dato all'esame degli aspetti riguardanti pianificazione, stima, controllo e convalida con una panoramica sulle nuove metodologie emergenti per la gestione dei progetti complessi.

Il Corso é conforme alle linee guida del PMBOK®.

### DESTINATARI ▼

- Responsabili dell'area sviluppo
- Capi progetto
- Analisti
- Responsabili di funzioni aziendali

### PREREQUISITI ■

Conoscenza delle problematiche di sviluppo applicativo.

### CONTENUTI ●

- **Introduzione:** che cos'è un progetto, la gestione degli obiettivi e dei risultati, bilanciamento tra costi, tempi, funzionalità e qualità
- **Il processo di Project Management:** avvio, pianificazione, esecuzione, monitoraggio e controllo e chiusura del progetto
- **Strutturazione del piano di progetto:** obiettivi generali, traguardi intermedi, ruoli e responsabilità
- **Tecniche per l'organizzazione, la pianificazione ed il controllo delle attività**
- **Controllo di un progetto:** che cosa significa controllare un progetto, supervisione e controllo dell'avanzamento (tempi e costi), azioni correttive
- **Rischi di progetto:** margine di rischio e controllo del progetto, fattori di rischio, modelli per la valutazione del margine di rischio
- **Gestione di un progetto complesso:** scomposizione in sottosistemi, coordinamento dell'avanzamento di sottoprogetti paralleli in fasi realizzative disomogenee; nuovi modelli di gestione (XPM, Metodi Agili,..)
- **Gestione dell'ambito (scope):** definizione, ordinamento in funzione delle priorità, verifica, definizione dell'ambito di progetto, governo delle modifiche
- **Stime:** problematiche e approcci metodologici
- **Creazione, sviluppo e mantenimento di un team efficace:** gestione dei rapporti interpersonali, fattori sociologici e loro impatto sulla produttività, leadership formale e informale, ruolo del capo progetto
- **La gestione della qualità all'interno del progetto**

### DURATA ◆

3 giorni

# GESTIONE DI PROGETTI: AVANZATO

## (codice PMGADV)

### DESCRIZIONE ►

Il corso, complementare rispetto al corso base, ha l'obiettivo di esaminare in modo più approfondito alcune aree di conoscenza che assumono oggi maggior rilevanza considerando che i progetti sono sempre più complessi, presentano maggiori rischi e coinvolgono in modo più evidente i fornitori esterni nella gestione di obiettivi e risultati.

Il corso, partendo dalle principali best practice, rielaborate anche alla luce delle esperienze maturate, affronta i temi riguardanti definizione e negoziazione dell'ambito di progetto (project scope), valutazione dei costi, analisi qualitativa e quantitativa dei rischi, gestione dell'outsourcing di progetto.

Il corso affianca agli aspetti teorici esercitazioni che hanno l'obiettivo di aumentare il livello di comprensione dei partecipanti e fornire indicazioni per l'utilizzo delle nozioni apprese.

Il Corso é conforme alle linee guida del PMBOK®.

### DESTINATARI ▼

- Responsabili dell'area sviluppo
- Capi progetto
- Analisti
- Responsabili di funzioni aziendali

### PREREQUISITI ■

Partecipazione al corso "Gestione di progetti" o il possesso di conoscenze equivalenti.

### CONTENUTI ●

- **Gestione del project scope:** Definizione dell'ambito di progetto (scope), product scope Vs project scope, negoziazione dei confini dell'intervento (tempi, costi e qualità), rischi connessi alla non corretta definizione del project scope. Il processo di gestione del project scope secondo PMBOK® : individuazione delle esigenze degli stakeholder, definizione dell'ambito, creazione della WBS (Work Breakdown Structure), accettazione dei deliverables, controllo dell'ambito
- **Gestione dei costi di progetto:** metodi e tecniche per la stima dei costi, determinazione del budget di progetto e creazione della cost baseline, controllo dei costi, metodo dell'earned value e tecniche correlate per l'analisi dei costi effettivi e la verifica dell'avanzamento fisico
- **Analisi dei rischi:** il processo di gestione dei rischi, analisi qualitativa (assessment, definizione matrice probabilità/impatto), analisi quantitativa (metodi statistici, distribuzione di probabilità, analisi della sensitività, tecniche di simulazione)
- **Gestione dell'outsourcing di progetto:** Il contratto (tipi di contratto, livelli di esternalizzazione, definizione ambito di intervento, regole per la gestione delle modifiche in corso d'opera, definizione KPI), i fornitori (criteri di selezione, interazioni dei fornitori con i progetti, gestione condivisa dei rischi), il governo dell'outsourcing (pianificazione delle forniture, monitoraggio su tempi e costi, accettazione dei deliverables, valutazione delle performance, gestione delle relazioni contrattuali con il fornitore, gestione dei conflitti)

### DURATA ◆

2 giorni

# PROJECT PORTFOLIO MANAGEMENT

## (codice PMGP3M)

### DESCRIZIONE ►

Se il Project Management può essere informalmente definito "...l'arte di fare bene i progetti", il Portfolio Management è "...l'arte di scegliere bene i progetti da fare".

Nell'ottica del Portfolio Management, i progetti rappresentano l'attuazione della strategia aziendale. Il governo del singolo progetto, tipicamente orientato al raggiungimento di obiettivi di ambito, tempi e qualità, deve essere inquadrato in un contesto più ampio, nel quale il suo avviamento, l'esecuzione, il controllo e la chiusura, devono essere eseguiti in un contesto di governo del portafoglio dei progetti dell'azienda.

Il governo del portafoglio dei progetti di impresa risponde all'esigenza di ogni Azienda di investire le proprie risorse sulle iniziative che sono in grado di restituire il massimo valore.

Il corso, conforme alle linee guida PMI's Standard for Portfolio Management, fornisce i concetti fondamentali per una corretta impostazione e gestione del Portfolio Aziendale in una visione integrata che unisce Project, Program e Portfolio Management (P3M).

### DESTINATARI ▼

- Capi progetto
- Program manager
- Responsabili di funzioni aziendali

### PREREQUISITI ■

Conoscenza delle problematiche legate alla gestione di progetti.

### CONTENUTI ●

- **Introduzione:**
  - La gestione per obiettivi in un'ottica integrata (Project, Program e Portfolio Management)
  - Portfolio Management:
    - Contesto
    - Obiettivi
    - Vantaggi
    - Stakeholder
- **Definizione dei criteri di valore**
  - Balanced Score Cards (Allineamento con strategia aziendale)
  - Pesatura dei criteri
- **Bilanciamento del Portfolio**
  - Valutazione progetti (Multidimension Decision Making)
  - Assegnazione priorità
  - Selezione nuovi progetti da realizzare
- **Assegnazione delle risorse ai progetti in base alla priorità**
- **Definizione masterplan dei progetti da realizzare**
- **Monitoraggio e controllo del Portfolio**
  - Valutazione del carico di lavoro
- **Caso studio**

### DURATA ◆

2 giorni

# EXTREME PROJECT MANAGEMENT

## (codice PMGXPM)

### DESCRIZIONE ►

L'elevata complessità e instabilità del contesto e la necessità di perseguire benefici con elevato ritorno di investimento, richiedono maggiore semplicità e flessibilità nella gestione dei progetti: non servono approcci più complessi per risolvere situazioni complesse. Con l'evoluzione dei nuovi modelli di sviluppo (Extreme Programming e altri metodi Agili) vi è la necessità di utilizzare tecniche di Project Management adeguate.

L'Extreme Project Management (XPM) ha come obiettivo quello di fornire una risposta concreta a questa necessità, focalizzando l'attenzione sul contesto e non solo sul contenuto interno, sulla comunicazione e condivisione di obiettivi e risultati con gli stakeholder e sulla gestione dei cambiamenti.

Il corso illustra i principi e le caratteristiche principali dell'extreme project management, che si distingue dal project management tradizionale (TPM) per il suo approccio maggiormente flessibile, leggero e non deterministico. Vengono inoltre analizzate le principali metodologie leggere (XP, SCRUM) su cui si basa l'XPM.

### DESTINATARI ▼

- Responsabili dell'area sviluppo
- Capi progetto
- Analisti
- Responsabili di funzioni aziendali

### PREREQUISITI ■

Partecipazione al corso "Gestione di progetti" o il possesso di conoscenze equivalenti.

### CONTENUTI ●

- **Introduzione:** l'Extreme Project Management e le metodologie leggere
- **La gestione di progetti estremi:** la gestione efficace del cambiamento (apertura al cambiamento), contesto vs contenuto, l'importanza dello sponsor, un project management a vita intera, il PM come facilitatore
- **Le metodologie agili:** concetti base ed approfondimenti su XP e SCRUM
- **La stima:** fattori umani che influenzano le stime dei progetti, Critical Chain Management basato su rimozione della "sicurezza nascosta", approccio di tipo "backward scheduling" per comprimere i tempi di progetto, gestione di buffer per assorbire ritardi, rimozione totale dei conflitti sulle risorse
- **Il Rapid Planning (RAP):** un processo di pianificazione frequente, cooperativo e rapido, con sessioni a cui partecipano tutti gli stakeholder chiave
- **La tracciabilità e il reporting:** verifica dell'avanzamento fisico vs impegno effettivo, analisi validità baseline
- **La gestione dell'ambiente sociologico del progetto:** gli stakeholder, un team motivato e responsabilizzato, la comunicazione e la gestione degli spazi del progetto
- **Caso studio**

### DURATA ◆

2 giorni

# SVILUPPO DI APPLICAZIONI CON XML E WEB SERVICES

## DESCRIZIONE ►

Lo sviluppo di applicazioni web enfatizza ulteriormente gli aspetti che riguardano l'indipendenza dell'applicazione dal dispositivo di visualizzazione, la comunicazione fra componenti realizzate con tecnologie eterogenee, spesso in concorrenza tra loro (Microsoft, Java, ...), e la creazione di componenti facilmente riusabili. Le tecnologie web standard non consentono di ottenere un buon livello di flessibilità nello sviluppo e una semplicità di manutenzione in grado di soddisfare le esigenze più complesse. Inoltre, con l'avvento del web 2.0 le tecnologie basate su XML si stanno imponendo come standard di fatto per lo sviluppo di applicazioni con interfacce web complesse.

Il corso, partendo dall'esame delle tecnologie basate su XML e dei nuovi paradigmi dello sviluppo (Web Services), intende fornire ai partecipanti un quadro teorico-pratico per la progettazione di applicazioni indipendenti dal formato iniziale dei dati, dai dispositivi di visualizzazione e dalla tecnologia utilizzata per lo sviluppo delle componenti software.

## DESTINATARI ▼

- Capi progetto
- Analisti
- Analisti programmatori
- Programmatori

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- **Introduzione allo sviluppo con XML:** Il linguaggio XML; DTD e XML Schema; Document Object Model; I fogli di stile XSL e l'indipendenza dal dispositivo di output; L'utilizzo di XML nei database: XQuery, XSQL, ...; La creazione di nuovi linguaggi di markup tramite XML; XHTML; Estensioni al linguaggio: XLINK, VXML, ...
- **Lo sviluppo di applicazioni XML:** L'architettura di un'applicazione XML Based; Le principali api per lo sviluppo con XML: DOM, SAX, ...; Lo sviluppo di applicazioni XML con tecnologia Microsoft; Lo sviluppo di applicazioni XML con tecnologia SUN
- **Interoperabilità con SOAP:** Introduzione a SOAP; SOAP e XML meccanismo di funzionamento; Architettura di un'applicazione basata su SOAP; Creazione di applicazioni SOAP con tecnologia Microsoft e SUN
- **Creazione di componenti riusabili tramite i Web Services:** Introduzione ai Web Services; Architettura di un sistema basato sui Web Services; La proposta Microsoft; La proposta Sun; Utilizzo congiunto di XML, SOAP e Web Services
- **Esempi e confronti fra le tecnologie**

## DURATA ◆

2 giorni

# SVILUPPO DI APPLICAZIONI WEB CON AJAX

## DESCRIZIONE ►

Una nuova generazione di applicazioni web si sta affacciando sul mercato; queste applicazioni hanno come principale caratteristica una elevata interattività, che si sposa con un aspetto gradevole e una elevata velocità di collegamento con il server, tanto da sembrare quasi applicazioni tradizionali. Queste applicazioni sono basate sulla tecnologia AJAX (Asynchronous JavaScript and XML), una tecnologia giovane che sta dando luogo a sviluppi interessanti, sia sul fronte delle applicazioni che sul fronte degli strumenti di sviluppo.

La tecnologia AJAX è anche un passo importante verso la realizzazione di applicazioni basate sul concetto di Service Oriented Architecture (SOA), ottenute integrando a livello client diversi servizi indipendenti basati sul concetto di Web Service. Gli sviluppatori di sistemi web saranno chiamati, nel futuro prossimo, a introdurre la tecnologia AJAX sempre più spesso nelle loro applicazioni.

Devono perciò avere il panorama della tecnologia, partendo dai concetti di base per realizzare applicazioni con AJAX fino a comprendere come usare i vari framework di sviluppo integrato. Il corso ha l'obiettivo di introdurre tutto questo. Si parte dai fondamentali, introducendo le caratteristiche essenziali di JavaScript, le nozioni fondamentali di XML, la rappresentazione DOM. Quindi, si passa a vedere il funzionamento del livello fondamentale di AJAX, la comunicazione asincrona con l'oggetto XMLHttpRequest.

Si procede usando il framework Yui (Yahoo User Interface) che consente di realizzare facilmente interfacce grafiche evolute con AJAX. Infine, si vedono i vari framework che consentono di sviluppare applicazioni client/server fortemente integrate, quali GWT (Google Web Toolkit), ASP.NET AJAX, Sun JSF (Java Server Faces). Durante il corso verranno svolti esercizi pratici di sviluppo di interfacce AJAX.

## DESTINATARI ▼

Tutti coloro che sono interessati all'utilizzo delle nuove tecnologie per l'implementazione di applicazioni Web complesse.

## PREREQUISITI ■

Conoscenze generali sulle piattaforme Web e conoscenze di base di programmazione. È necessario essere muniti di notebook, per svolgere gli esercizi di programmazione AJAX.

## CONTENUTI ●

- **La visione d'insieme:** Applicazioni Web Dinamiche; I Web Services; SOA: Service Oriented Architecture
- **I concetti fondamentali:** Il linguaggio JavaScript; XML: caratteristiche fondamentali; DOM (Document Object Model): la rappresentazione dei documenti XML; Alcuni esercizi pratici con il DOM in JavaScript
- **AJAX base:** L'oggetto XMLHttpRequest; Il lato server; Gestire la risposta; JSON: AJAX senza XML; Alcuni esercizi pratici
- **Le interfacce grafiche:** I framework grafici per la realizzazione di interfacce grafiche; Il framework YUI (Yahoo User Interface); AJAX e YUI; Alcuni esercizi pratici di realizzazione di interfacce grafiche con YUI
- **I framework integrati:** I framework integrati per la realizzazione di applicazioni client-server; GWT (Google Web Toolkit); ASP.NET AJAX; JSF (Java Server Faces); Piccoli esercizi pratici

## DURATA ◆

3 giorni

# WEB 2.0 E CROSSMEDIALITÀ AL SERVIZIO DEL BUSINESS

## DESCRIZIONE ►

La convergenza dei media e il fenomeno dell'“ubiquitous computing” rendono molto complesso lo scenario informativo, determinando un intreccio sempre più articolato fra mondo fisico e mondo digitale.

L'aumento della produzione e della disponibilità di informazioni porta l'utente a muoversi continuamente avanti e indietro fra diverse fonti di informazione, codificata e distribuita in un'ampia varietà di formati differenti.

Questo di conseguenza costringe ogni volta ad apprendere e utilizzare un modello di interazione e fruizione differente. Il rimedio a questo sovraccarico informativo è un mix alchemico che prevede la conoscenza dei paradigmi della comunicazione, delle euristiche per l'organizzazione dell'informazione e, infine, delle linee guida per progettare la crossmedialità.

Le aziende, che vogliono uscire dalla nebbia informativa e rendersi visibili sul mercato, devono fare i conti con le nuove tendenze mediatiche: il Web 2.0 e la crossmedialità.

Il Web 2.0 sta cambiando radicalmente il modo in cui imprese, manager e professionisti impiegano Internet per il proprio business. Le regole, i Casi studio da analizzare e gli errori da evitare, sono gli elementi su cui porre l'attenzione. Il decalogo del Web 2.0 sintetizza chiaramente le 10 regole per competere con successo, mentre il decalogo dell'Advertisement 2.0 definisce le nuove regole per ottenere un ritorno degli investimenti. Ed infine, nell'era della crossmedialità e della convergenza digitale, l'offerta di contenuti informativi cambia e i canali web-tv-cellulare mutano la loro natura per diventare un prodotto mediatico globale, capace di garantire significative economie di scala. Progettare questi prodotti per dare alle aziende sempre più spazio e visibilità.

## DESTINATARI ▼

- Manager
- Responsabili dello sviluppo software
- Analisti programmatori
- Programmatori
- Responsabili delle applicazioni Web
- Capi progetto

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- **Le vie della comunicazione sono finite?**
- **New Media e comunicazione: paradigmi vecchi e nuovi**
  - Pratica della comunicazione
  - Il linguaggio dei new media
  - Quando la grafica diventa comunicazione visiva
  - Oltre i new media
- **L'architettura delle informazioni e della “partecipazione”, la nuova frontiera**
  - L'architettura delle informazioni centrata sull'utente: le tendenze orientate al Web 2.0
- **Web 2.0, istruzioni per l'uso**
  - Gli strumenti del Web 2.0
  - Fare business con il Web 2.0
  - Social network e modelli di business
  - Advertising nel Web 2.0: caratteristiche e logiche. Il potere ai consumatori

- **Le parole d'ordine: convergenza digitale e crossmedialità**
- **Organizzare e strutturare un progetto crossmediale**
  - Analisi dello scenario monomediale e multi-modale
  - Il contesto editoriale del progetto
  - Il concept
- **Progettazione del prodotto crossmediale**
- **Economie, sinergie e integrazioni nei processi di produzione del progetto crossmediale**
- **Gestione e scelta dell'utenza: la nuova domanda**
- **Internet: dal medium al progetto crossmediale**
  - Risorse integrate
  - Partnership di contenuto e tecnologiche
  - Marketing crossmediale
- **Caso studio**

## **DURATA** ◆

---

2 giorni

# WEB 2.0: TECNOLOGIE E PROBLEMATICHE

## DESCRIZIONE ►

Le applicazioni Web tradizionali sono poco interattive e offrono interfacce utente a volte poco ergonomiche e lente da utilizzare. Nel corso degli anni gli sviluppatori hanno adottato numerosi trucchi per ovviare a questi problemi ma le soluzioni trovate non sono mai state divulgate in modo corretto.

Con l'avvento del Web 2.0 e della crossmedialità gli utenti richiedono applicazioni con caratteristiche di usabilità e interattività che le tecnologie di sviluppo standard (ASP, JSP,...) non sono in grado di garantire. Le soluzioni tecnologiche esistono da molti anni ma solo oggi cominciano ad uscire i primi standard. Le tecnologie basate su XML e Javascript (AJAX) si stanno imponendo come soluzione principe per l'implementazione di applicazioni per il Web 2.0

Il corso fornisce un apparato teorico e pratico per la progettazione di applicazioni Web adeguate alle nuove caratteristiche del Web 2.0 e della crossmedialità. Il corso intende fornire inoltre indicazioni di base per la conoscenza degli strumenti di authoring, e delle diverse modalità di interazione possibili tra l'utente e il sistema e le linee guida per il publishing di pagine multimediali.

## DESTINATARI ▼

Tutti coloro che sono interessati all'utilizzo delle nuove tecnologie per l'implementazione di applicazioni Web complesse

## PREREQUISITI ■

La partecipazione al corso " Web 2.0 e crossmedialità al servizio del business " o il possesso di conoscenze equivalenti

## CONTENUTI ●

- **L'evoluzione del Web: dai primordi al futuro Web 2.0**  
Nuovi paradigmi della comunicazione; Cambiamenti nella fruizione dei servizi Web; Cambiamento del ruolo dell'utente: da passivo ad attivo; Confronto fra le applicazioni Web standard e le applicazioni Web 2.0; Gli strumenti del web 2.0; Enterprise 2.0 - L'azienda costruita sugli strumenti del web 2.0
- **Architettura delle applicazioni Web 2.0**  
Client/Server su Web; User interface dinamica; Back-end a servizi; Esempi di architetture; La comunicazione fra il client e il server
- **Tecnologie per l'implementazione delle applicazioni Web 2.0**  
Esecuzione lato client o lato server?; AJAX: un nome tante soluzioni; Microsoft Web Form ed estensioni 2.0; Microsoft Silverlight; SUN Java Server Face e relativi cloni; JavaFX; Tecnologie Open Source per l'implementazione di interfacce dinamiche; Confronto fra le tecnologie; Casi di studio ed esempi
- **Utilizzo di componenti multimediali**  
Integrazione di audio e video nelle applicazioni Web; Integrazione di contenuti proprietari
- **Portal server: evoluzione dei portali**  
Nuova generazione dei portali Web; Portlet e componenti riusabili
- **Criticità delle applicazioni Web 2.0**  
Ottimizzazione delle prestazioni; Sicurezza delle transazioni; Gestione dello stato del sistema

## DURATA ◆

3 giorni

# LA TECNOLOGIA OPEN SOURCE IN AZIENDA

## DESCRIZIONE ►

Negli ultimi anni il sistema operativo Linux e le applicazioni Open Source hanno cominciato a diffondersi all'interno delle realtà aziendali italiane. Ma cosa sono le applicazioni Open Source? Quali vantaggi possono offrire ad un'azienda? Sono realmente convenienti?

Il corso fornisce un quadro di riferimento per poter pianificare, in modo adeguato, un'introduzione delle tecnologie Open Source all'interno delle aziende valutando i vantaggi e gli svantaggi che ne derivano.

## DESTINATARI ▼

Dato il carattere introduttivo, il corso non richiede specifiche competenze e non è indirizzato a categorie particolari di pubblico

## PREREQUISITI ■

Nessuno.

## CONTENUTI ●

- **Che cos'è l'Open Source**
  - Definizione di Open Source
  - Tipologie di licenze del software
  - Chi sponsorizza le applicazioni Open Source (comunità, grandi vendor,...)
  - L'Open Source secondo i grandi nomi dell'informatica (IBM, ORACLE, HP,...)
- **L'utilizzo della tecnologia Open Source in azienda**
  - Costi e prestazioni
  - Stabilità e affidabilità delle tecnologie
  - Il problema del supporto tecnico
  - Maturità delle tecnologie
  - Compatibilità con i software e i formati proprietari
  - La produzione di software e di servizi basati sull'Open Source: quali sono le prospettive di guadagno?
  - Confronto fra le tecnologie Open Source e le tecnologie proprietarie
- **Esempi di applicazione della tecnologia Open Source**
  - I sistemi operativi Open Source (Linux, Free BSD e derivati)
  - Le tecnologie Open Source per la realizzazione di applicazioni web
  - Implementazione dell'infrastruttura dell'azienda tramite tecnologia Open Source (mail server, file e print server, sicurezza, ...)
  - Open Source per la progettazione e lo sviluppo del software (UML, Change Management,...)
  - Open Source nella Pubblica Amministrazione

## DURATA ◆

2 giorni

## DESCRIZIONE ►

---

Java si sta imponendo sempre più all'attenzione delle aziende. Lo testimoniano per tutti due fatti: da un lato l'interesse con cui grandi nomi dell'informatica come Ibm, Oracle e altri hanno esteso la propria offerta di prodotti e servizi per includere il supporto a Java. Dall'altro la coerenza con UML che fa di Java il linguaggio candidato per la costruzione di applicazioni OO, anche con riferimento a principi quali Tipo, Interfaccia, ecc.

Il corso, partendo da un quadro di riferimento introduttivo su metodologie e concetti OO, con riferimento a UML, illustra le caratteristiche di Java con l'obiettivo di introdurre i partecipanti all'uso di Java per l'implementazione di applicazioni e applet.

L'esposizione teorica è corredata da esercitazioni che permettono di tradurre in pratica i concetti esposti.

## DESTINATARI ▼

---

- Analisti
- Analisti programmatori

## PREREQUISITI ■

---

Per quanto la conoscenza dei concetti della programmazione OO renda più agevole l'assimilazione dei concetti esposti, il corso non richiede particolari requisiti.

## CONTENUTI ●

---

- Metodologie Object Oriented, con riferimento a UML
- Concetti fondamentali: classe, oggetto, tipo, interfaccia, package, incapsulamento, ereditarietà, polimorfismo
- Sintassi e semantica Java
- Librerie standard
- Gestione delle interfacce grafiche (Abstract Window Toolkit)
- Java Development Kit (JDK 1.2)
- Sviluppo di componenti software in Java – applicazioni, applet, componenti JavaBeans, servlet
- Gestione della memoria
- Gestione degli errori
- Thread e multithreading
- Integrazione di Java con altre applicazioni - Java native Interface
- Accesso ai dati – JDBC, JSQL, Java Blend
- Interconnessione tra componenti distribuite – RMI, CORBA
- Utilizzo di XML
- Java networking

## DURATA ◆

---

5 giorni

# SOA (Service Oriented Architectures): BEST PRACTICES

## DESCRIZIONE ►

Al fine di ottimizzare i costi e di migliorare la gestione dell'IT, le aziende guardano sempre con maggiore interesse verso le problematiche di governance dei processi aziendali. Fra i vari modelli di riferimento per l'implementazione e la gestione dei Business Process, SOA (Service Oriented Architecture) è quello più promettente e apprezzato. Il censimento dei processi e dei servizi aziendali e la loro implementazione all'interno di una infrastruttura SOA permette di sviluppare e mantenere le applicazioni con minor sforzo e costi minori, consentendo il riutilizzo del codice e ottimizzando le comunicazioni fra le varie aree di competenza presenti all'interno di un'azienda.

Le attuali tecnologie dei web services e i sistemi di BPM (Business Process Management) si pongono al vertice degli strumenti attualmente utilizzati per implementare una SOA ma le tecnologie distribuite meno recenti (CORBA, DCOM, MQ,...) stanno rivivendo una seconda giovinezza anche a causa della presenza di nuovi framework di sviluppo e alle prestazioni di livello superiore.

Di SOA si parla molto ma: quali sono i tempi e i costi necessari per l'implementazione di una SOA? Quali sono le piattaforme e le tecnologie più promettenti per l'implementazione di una SOA? Quali sono le architetture e i pattern più utilizzati per progettare una SOA? Quali sono gli impatti organizzativi che derivano dall'implementazione di una SOA? Quali vantaggi derivano e quali rischi si corrono nell'implementazione di una SOA?

Il seminario, partendo dal confronto delle architetture di sviluppo tradizionali, introduce i concetti fondamentali su cui si basa una SOA descrivendone le funzionalità e le peculiarità tramite l'utilizzo di Caso studio rappresentanti problemi reali.

Vengono poi affrontate le problematiche relative alla progettazione e all'implementazione di una SOA descrivendone gli aspetti metodologici, tecnologici e organizzativi.

## DESTINATARI ▼

- Capi progetto
- Analisti
- Professionisti IT
- Chiunque voglia approfondire gli aspetti legati alle SOA

## PREREQUISITI ■

Concetti base di progettazione e sviluppo applicazioni web

## CONTENUTI ●

- Che cos'è una SOA: descrizione, concetti fondamentali e confronto con le architetture tradizionali
- Requisiti, tempi e costi di una SOA
- Progettare oggetti di business orientati ai servizi
- Strumenti per il BPM (BPEL, BPEL4WS,...)
- Vantaggi e rischi di una SOA
- Architettura di una SOA e pattern di progettazione e sviluppo
- Tecnologie per l'implementazione di una SOA
- Impatti organizzativi, strategici, economici e tecnologici

## DURATA ◆

3 giorni

# SOA PER L'INTEGRAZIONE DELLE APPLICAZIONI AZIENDALI

## DESCRIZIONE ►

---

Lo sviluppo della nuova generazione di applicazioni web e la creazione di architetture orientate ai servizi riporta in primo piano il problema dell'integrazione delle applicazioni esistenti, siano esse web, client server o su piattaforme host con le nuove infrastrutture tecnologiche.

In tale ambito, infatti, non basta focalizzare la propria attenzione sull'interoperabilità dei protocolli per l'implementazione dei servizi (XML e SOAP), ma è necessario individuare meccanismi e tecniche sia per l'integrazione tra la rete e il sistema informativo aziendale che per il riuso di componenti applicative già esistenti. A tale scopo gli Enterprise Service Bus, gli integration server e le suite per il Business Process Management vengono in aiuto del progettista e dello sviluppatore.

Il corso, tramite l'utilizzo di Caso studio tratti da applicazioni reali, fornisce un quadro di riferimento per progettare in modo adeguato l'interconnessione tra le diverse componenti tecnologiche e applicative distribuite, utilizzando le principali soluzioni che le infrastrutture SOA rendono disponibili sul mercato.

## DESTINATARI ▼

---

- Capi progetto
- Analisti
- Analisti programmatori
- Programmatori

## PREREQUISITI ■

---

La partecipazione al corso "SOA BEST PRACTICES" o il possesso di conoscenze equivalenti.

## CONTENUTI ●

---

- **L'evoluzione delle applicazioni in azienda: dal mainframe alle SOA**
- **Il problema dell'integrazione: obiettivi, scenari, implicazioni**
- **Integrare le applicazioni aziendali tramite SOA**  
Tipi di applicazioni; Gli strumenti per l'integrazione: Integration Server e Enterprise Service Bus, ...; I connettori di interfacciamento: tipologie e funzionamento; Architetture applicative per l'integrazione tramite SOA; Casi di studio ed esempi
- **Integrare SOA con i DBMS aziendali**  
Database e XML; Database federati; Architetture di integrazione; Casi di studio ed esempi
- **Integrazione dei processi aziendali**  
Gli strumenti per il Business Process Management; Orchestration e Choreography dei servizi e dei business process; Architetture per la gestione dei business process; Casi di studio ed esempi
- **Analisi comparata fra gli strumenti e le tecnologie di integrazione**  
Strumenti commerciali; Strumenti open source
- **Cenni sulle problematiche di sicurezza di una SOA**

## DURATA ◆

---

3 giorni

# IMPLEMENTARE I PROCESSI DI BUSINESS CON SOA E BPM

## DESCRIZIONE ►

SOA (Service Oriented Architecture) e BPM (Business Process Management) consentono alle aziende di creare nuovo valore di Business a partire dagli investimenti esistenti, di riutilizzare le componenti software e di Business realizzate nei vari progetti e ottenere nuovi livelli di agilità, grazie a una maggiore flessibilità e a strutture di minor costo.

Spesso SOA e BPM vengono confusi e descritti come intercambiabili, probabilmente perché molti dei vantaggi che li caratterizzano sono simili. Tuttavia si tratta di iniziative molto diverse... Il BPM è un complemento naturale della SOA, rappresenta un meccanismo attraverso il quale un'organizzazione può applicare la SOA a importanti e avanzate sfide aziendali. Il BPM rappresenta un nuovo modo per analizzare e misurare il Business in termini di processi aziendali che vanno oltre i tradizionali confini dell'organizzazione e dei sistemi.

Per l'IT, BPM e SOA rappresentano un nuovo modo per implementare le soluzioni aziendali, sottolineando l'importanza di ricorrere a meno programmazione, promuovere il riutilizzo dei componenti e ottenere un maggiore coinvolgimento dell'azienda. Sia SOA che BPM possono essere adottati singolarmente, ma è l'unione dei due modelli architetturali che esprime il massimo delle potenzialità: su scala aziendale, l'applicazione del BPM su una solida base SOA lo rende più agile e flessibile, lo diffonde più facilmente nelle varie realtà aziendali e strutture collaborative ed è più resistente alle modifiche dei sistemi IT sottostanti.

Il seminario, partendo da una breve introduzione a SOA, descrive i principali concetti relativi al BPM, analizza le problematiche di progettazione e implementazione dei Business Process tramite utilizzo di esempi e Casi di studio reali.

## DESTINATARI ▼

- Capi progetto
- Analisti
- Professionisti IT
- Chiunque voglia approfondire gli aspetti legati alle SOA

## PREREQUISITI ■

Concetti base di progettazione e sviluppo applicazioni Web, Il corso "SOA Best Practices" o conoscenze equivalenti.

## CONTENUTI ●

- **Breve introduzione a SOA:** cos'è SOA; vantaggi e svantaggi; perchè introdurre SOA in Azienda; i layer di una SOA
- **Introduzione al BPM:** cos'è e a cosa serve il BPM; I tre livelli di integrazione: Workflow Management, Business Process Management, Enterprise Management; Business Process e Business Model; pro e contro
- **Composizione di processi:** Workflow e BPM; Orchestration e Choreography; Pattern di Workflow; Business Process Design: Identificazione dei Business Process, Business Process Modelling Notation (BPML), Tools per la progettazione
- **Infrastruttura tecnologica:** Motori di Workflow; le suite per il BPM; esempi di architettura; i linguaggi per la descrizione dei processi (BPEL/WS-BPEL, WS-CDL)
- **Esempi e Casi di studio**

## DURATA ◆

3 giorni

# DATA WAREHOUSE: ARCHITETTURA E PRINCIPI

## DESCRIZIONE ►

Il Data Warehouse si trova oggi al centro dell'interesse di numerose aziende che intendono supportare in modo adeguato i processi decisionali.

I Data Warehouse poggiano le basi sui sistemi gestionali ma utilizzano anche dati di sistemi informativi esterni. L'organizzazione delle informazioni e lo stile di d'interazione tra utente e sistema sono evidentemente differenti rispetto ai sistemi informativi tradizionali.

Il corso non si limita ad illustrare i possibili benefici che si possono prospettare al management, ma esamina in toto il ciclo di vita del Data Warehouse, le tecnologie disponibili e le risposte che può dare a seconda della tipologia utente.

Obiettivo del corso è, infine, definire una chiara strategia di sviluppo e di utilizzo anche alla luce del nuovo paradigma di sviluppo applicativo Internet / Intranet.

## DESTINATARI ▼

- Responsabili dello sviluppo
- Analisti
- Analisti programmatori

## PREREQUISITI ■

Conoscenza di base su Qualità, Modello Dati, Teoria Relazionale.

## CONTENUTI ●

- Ragioni aziendali, tecnologiche ed economiche del Data Warehouse, e differenze fra i sistemi gestionali e i Data Warehouse
- Data Warehouse Framework – Esame delle componenti di base di un ambiente di Data Warehousing
- Problematiche e modalità di gestione/pianificazione di un progetto in ambiente di Data Warehousing
- Integrazione e Qualità: Problematiche e modalità di approccio
- Metadati: Ruolo all'interno dell'ambiente di Data Warehousing – Repository dei Metadati
- Architettura di un Data Warehouse – Confronto fra le modalità di approccio (Global Data Warehouse, Interconnected Data Mart). Confronto fra i modelli usabili : Relational, Star Schema e Snowflake Schema
- Area di ETL: Analisi delle componenti (Capture, Transform, Apply) – Problematiche e tecniche di approccio
- Le applicazioni tipiche che operano su un Data Warehouse - Query/Reporting; DSS / EIS; OLAP; Data Mining; distribuzione dei risultati
- EAI e Business Intelligence Portal (Closed Loop)
- Esempi e Caso studio

## DURATA ◆

3 giorni

# LA PROGETTAZIONE DEL DATA WAREHOUSE

## DESCRIZIONE ►

Il disegno logico e fisico di Data Warehouse e Data Mart è di cruciale importanza. Obiettivo del corso è fornire le conoscenze necessarie per utilizzare al meglio il modello dimensionale, sul quale si basa sia la progettazione sia la ricerca delle informazioni in un data warehouse.

Si analizzano in dettaglio le varie alternative per la progettazione Logica delle tabelle dei fatti e delle dimensioni, discutendo la validità delle scelte attraverso la presentazione di esempi applicativi concreti e numerose esercitazioni previste in aula.

Il corso affronta inoltre tutte le problematiche correlate: dal caricamento dei dati alla presentazione dei risultati, alla luce delle diverse tipologie di applicazioni che possono operare su un data warehouse.

I concetti e le tecniche esposte in sede teorica vengono applicati in un Caso studio di progettazione.

## DESTINATARI ▼

- Capi progetto
- Analisti e analisti programmatori
- Data base administrator
- Specialisti di Data Warehouse (Data Warehouse architect, Data Mart architect, gestore metadati, ...)

## PREREQUISITI ■

La partecipazione al corso "Data Warehouse: architettura e principi" o il possesso di conoscenze equivalenti; conoscenze di base sulla progettazione delle basi dati.

## CONTENUTI ●

- Introduzione al Data Warehouse - concetti di base e terminologia, architettura e componenti
- Definizione dei requisiti - tecniche per raccogliere e analizzare i requisiti funzionali e tecnologici; tipologie e locazione degli utenti coinvolti: profili di utilizzo, problematiche di sicurezza e di distribuzione delle informazioni
- Modelli dati per il data warehouse - pro e contro dei diversi approcci alla modellazione dei dati (Entità/Relazioni, Star Schema, Snow Flake)
- Il modello dei dati a stella (Star Schema) - fatti, dimensioni, gerarchie, la dimensione Tempo, dimensioni a variazione lenta, dimensioni degenerate, mini-dimensioni, fatti semi-additivi, assenza della tabella dei fatti
- Passi metodologici (Business driven + Data Driven) che permettono la trasformazione di un Modello Concettuale dei Dati in un modello Dimensionale Inziale (Star schema) e l'analisi dei requisiti per adeguare il modello alle esigenze utente
- Implementazione fisica del Modello Logico (Star Schema) utilizzando come Data Base target ORACLE
- Casi studio e esempi applicativi

## DURATA ◆

3 giorni

# MISURARE E MANTENERE LE PRESTAZIONI DELLE APPLICAZIONI WEB

## DESCRIZIONE ►

La qualità di un'applicazione Web dipende non solo dal contenuto e dalle funzionalità applicative, ma anche dal modo in cui le informazioni sono presentate all'utente e dalle prestazioni che egli percepisce nell'utilizzo dell'applicazione stessa. Le statistiche indicano che vi è una correlazione stretta tra prestazioni di un'applicazione Web e tasso di abbandono degli utenti.

È pertanto importante disporre, in ogni fase del processo che porta dalla realizzazione alla gestione delle applicazioni Web, di metodologie e strumenti che permettano di prevedere e misurare l'affidabilità, le prestazioni ed il livello di servizio erogato sulla base di diverse condizioni di funzionamento.

Obiettivo del corso è di fornire competenze e metodologie per affrontare le diverse fasi del processo di progettazione e gestione di applicazioni Web con un occhio di riguardo al livello di servizio delle applicazioni, sia in termini di prestazioni che in termini di presentazione delle informazioni.

Il corso, partendo da nozioni elementari di teoria delle code, affronta inoltre gli aspetti connessi con il capacity planning dei sistemi informatici.

## DESTINATARI ▼

- Responsabili delle applicazioni Web e dei sistemi informativi
- Addetti alla funzione di gestione e controllo della qualità
- Persone coinvolte nello sviluppo del software (capi progetto, analisti, ...)

## PREREQUISITI ■

Conoscenza generale delle problematiche relative alla progettazione, manutenzione o assistenza di applicazioni e sistemi.

## CONTENUTI ●

- La soddisfazione degli utenti Web: pazienza e frustrazione, i fattori di insoddisfazione, la percezione del tempo e le soglie di sopportazione, l'incremental loading, la frustrazione cumulativa ed il punto di non-ritorno, aspettative, motivazioni ed alternative degli utenti
- Prestazioni di un sistema Web: pagina versus transazione Web, tempo di risposta e throughput di un'applicazione, definizione del livello di servizio
- Come misurare le prestazioni end-to-end di applicazioni Web: tecniche attive e passive, packet level dump, analisi dei log, load testing, definizione del campione significativo, generazione di traffico artificiale, reti di misurazione
- Cosa è il capacity planning: quali gli approcci, costi-benefici, il grado di profondità dell'analisi, forecast e what-if analysis
- Dimensionamento dei sistemi e teoria delle reti di code: risorse e richieste, classificazione delle risorse (a coda, a ritardo, multiservente, passive), modello del carico e visite, calcolo dei parametri
- Tecniche risolutive: sistemi aperti e chiusi, tecniche approssimate e tecniche esatte, individuazione e rimozione dei bottleneck
- L'attività di capacity planning: definizione degli indicatori di business (business driver o business key indicator), modello del sistema e modello del carico, previsione del fabbisogno di risorse in funzione dell'evoluzione del business
- Casi studio di sistemi Web reali

## DURATA ◆

2 giorni

# CAPACITY PLANNING

## DESCRIZIONE ►

Se, da un lato, un sistema informatico deve soddisfare dei ben precisi requisiti di prestazioni nei confronti degli utenti del sistema, d'altro canto l'azienda deve contenere i costi dell'infrastruttura informatica.

Per questa ragione deve essere effettuata una attività periodica di Capacity Planning che garantisca un corretto dimensionamento dei sistemi al variare degli scenari di funzionamento dell'applicazione. L'obiettivo del corso è quello di fornire una metodologia efficace e di mostrare gli strumenti più idonei per effettuare periodicamente il Capacity Planning.

Il processo di Capacity Planning parte da una sistematica misurazione delle prestazioni e dei volumi di carico, passa per la parametrizzazione di un modello matematico di facile "intuizione" e si conclude con un'attività altrettanto sistematica di what-if-analysis, che principalmente consente di:

- valutare la capacità massima del sistema; ad esempio, stimare qual è il numero massimo di utenti che il sistema è in grado di gestire
- confrontare la capacità massima del sistema con le curve di trend del carico per prevedere se e sino a quando il sistema sarà in grado di funzionare entro i limiti operativi prefissati
- valutare l'impatto sulle prestazioni in seguito al fail-over
- identificare le risorse ed i sistemi che costituiscono un collo di bottiglia per le prestazioni in modo da pianificare eventuali upgrade
- valutare gli effetti sulle prestazioni di buffer applicativi quali, ad esempio, la dimensione del connection-pool verso un database server.

## DESTINATARI ▼

- Responsabili dei sistemi informativi, delle applicazioni e dei servizi
- Addetti alla funzione di gestione e controllo della qualità
- Persone coinvolte nello sviluppo del software (capi progetto, analisti, ...)

## PREREQUISITI ■

La partecipazione al corso "Misurare e mantenere le prestazioni delle applicazioni Web". Conoscenza generale delle problematiche relative alla gestione dei sistemi. Il corso non richiede particolari background matematici.

## CONTENUTI ●

- Richiami a concetti di base: prestazioni di un sistema informatico, tempo di risposta e throughput di un'applicazione, cosa è il capacity planning: modello del sistema e modello del carico
- Teoria delle reti di code: risorse e richieste, classificazione delle risorse (a coda, a ritardo, multiservente, passive), modello del carico e visite, calcolo dei parametri
- Tecniche risolutive: sistemi aperti e chiusi, tecniche approssimate e tecniche esatte, individuazione dei bottleneck
- Il processo di capacity planning: definizione degli obiettivi, analisi dell'architettura e dell'applicazione, costruzione del modello del sistema e del modello del carico, attivazione degli agenti di raccolta, calcolo dei parametri del modello, risoluzione del modello
- What-if analysis: capacità massima e residua di un sistema, bottleneck removal, analisi di un sistema in caso di failure o di upgrade
- Casi studio (applicazioni Web e intranet)

## DURATA ◆

2 giorni

# TEST DI CARICO PER IL COLLAUDO E PER IL TUNING PRESTAZIONALE

## DESCRIZIONE ►

---

Il processo di collaudo prestazionale è un'attività finalizzata a verificare, attraverso test di carico, se e in che misura, un'applicazione sia in grado di erogare un livello di servizio predeterminato su una specifica architettura hardware/software, prima del rilascio in produzione.

Lo scopo principale del test di carico per il tuning è quello di migliorare sistematicamente ogni livello di un sistema informatico complesso per garantire elevati standard qualitativi in termini di prestazioni. Questo obiettivo viene raggiunto mediante un processo ciclico che prevede di caricare il sistema, individuare le componenti che ne limitano le prestazioni ed eliminare i colli di bottiglia mediante interventi di tuning e upgrade.

Nella prima parte del corso viene descritta una metodologia che permette di sistematizzare il processo di collaudo prestazionale.

Inoltre vengono identificate e dettagliate le diverse fasi che costituiscono il processo di collaudo. Per ogni fase sono presi in considerazione gli aspetti organizzativi, i flussi informativi e i documenti alla base del processo.

La seconda parte del corso si focalizza su due tipi di test di carico: il load test, che permette di valutare le prestazioni del sistema nelle normali condizioni operative, e lo stress test, che permette di valutare come si comporta il sistema in condizioni di massimo carico.

Durante il corso vengono mostrati i casi reali, utilizzando come strumento di riferimento LoadRunner® di Mercury.

## DESTINATARI ▼

---

- Responsabili dei sistemi informativi
- Responsabili delle applicazioni
- Addetti alla funzione di gestione e controllo della qualità

## PREREQUISITI ■

---

Nessuno.

## CONTENUTI ●

---

- Definire i service level objectives, la pianificazione
- Il processo di test per il collaudo: la progettazione del test, la predisposizione degli ambienti, la progettazione esecutiva, l'esecuzione dei test
- La validazione dei risultati
- I limiti di una procedura di collaudo, errori comuni, un approccio safe
- Le tipologie del test di carico: load test e stress test
- Modalità di esecuzione dello stress test
- Principali problemi sotto carico: memory leak, resource locking, transactional limits, bandwidth limits, hardware bottlenecks, software bottlenecks
- Il processo di test per il tuning: la progettazione del test, la predisposizione degli ambienti, la progettazione esecutiva, l'esecuzione test e l'analisi risultati, il tuning hw/sw, la rendicontazione delle attività

## DURATA ◆

---

3 giorni

# FONDAMENTI DI ITIL® V3

## DESCRIZIONE ►

ITIL® è ad oggi il modello più diffuso ed apprezzato per la gestione dei Servizi IT. La versione 3 di ITIL si compone di cinque discipline fondamentali: Service Strategy, Service Design, Service Transition, Service Operation, Continual Service Improvement.

Le best practice di ITIL, basate sul ciclo di vita del servizio, hanno soddisfatto le aspettative di molteplici aziende permettendo una riduzione dei costi e un allineamento dei servizi IT agli obiettivi di business.

La riorganizzazione dei processi di gestione dei servizi IT richiede il coinvolgimento dell'intera azienda ed è quindi fondamentale che il personale disponga di competenze adeguate per essere in grado di offrire un concreto supporto.

Il corso fornisce ai partecipanti la visione dei concetti base necessaria per comprendere i processi, le relazioni tra di essi ed i benefici apportati dall'approccio orientato al ciclo di vita del servizio.

## DESTINATARI ▼

Tutti i professionisti IT che operano nella progettazione, implementazione, erogazione e supporto al Servizio

## PREREQUISITI ■

Esperienza delle problematiche IT.

## CONTENUTI ●

- **Framework ITIL** - Introduzione e obiettivi del Service Management, concetti chiave di IT Library, differenze con ITIL V2
- **Service Lifecycle** - I processi ITIL nel Ciclo di Vita del Servizio
- **Service Strategy** – Trasformare il Service Management in un asset strategico, Utility e Warranty, Value creation, I processi di riferimento, I ruoli e le responsabilità
- **Service Design** – Una guida per la progettazione e sviluppo dei servizi e dei processi di gestione dei servizi, Value to business, Il concetto di Architettura, I processi di riferimento, I ruoli e le responsabilità
- **Service Transition** – Lo sviluppo e il miglioramento delle capacità necessarie per la transizione dei servizi verso la produzione, I processi di governo dei cambiamenti, Il V-Model, Il Service Knowledge Management System, I ruoli e le responsabilità
- **Service Operation** – Efficacia ed efficienza nell'erogazione e nella gestione del servizio in linea, I processi e le funzioni di riferimento, I ruoli e le responsabilità
- **Continual Service Improvement** – Il miglioramento continuo alla ricerca del "cost effectiveness", Il CSI Model, Il concetto di misura, Il ciclo di Deming, I processi di miglioramento, I ruoli e le responsabilità

## DURATA ◆

2 giorni

# DB2 z/OS: ARCHITETTURA E FUNZIONI

## DESCRIZIONE ▶

---

Il corso presenta l'architettura, i servizi e le principali funzioni del sistema, con l'obiettivo di fornire una panoramica completa e dettagliata sull'ambiente DB2 z/OS.

Vengono esaminate in dettaglio le diverse componenti del sistema, e viene messo in evidenza l'inquadramento del DB2 nell'ambito dell'architettura z/OS.

Vengono inoltre esaminate le problematiche dell'ottimizzazione e le possibilità offerte dal DB2 distribuito.

L'esposizione teorica è corredata da esercitazioni che favoriscono l'assimilazione dei concetti esposti.

## DESTINATARI ▼

---

- Data base administrator
- Analisti
- Analisti programmatori
- Programmatori
- Sistemisti

## PREREQUISITI ■

---

Conoscenze di base sui sistemi relazionali.

## CONTENUTI ●

---

- Caratteristiche e configurazione del sistema
- Catalogo DB2 - - organizzazione ed utilizzo delle informazioni
- Prestazioni e problematiche di ottimizzazione
- Integrità fisica dei dati - - problematiche di backup/recovery
- Componenti e funzionalità del DB2 distribuito
- Organizzazione dei dati - caratteristiche ed utilizzo degli oggetti DB2
- Integrità logica dei dati
- Funzioni di utilità
- Modifiche al disegno del data base
- Gestione della concorrenza
- Riservatezza dei dati
- Elementi relativi alle features delle nuove versioni

## DURATA ◆

---

3 giorni

# DB2 z/OS: SVILUPPO DI APPLICAZIONI ORIENTATO ALLE PERFORMANCE

## DESCRIZIONE ►

Il corso ha come obiettivo quello di fornire ai partecipanti le conoscenze necessarie per sviluppare applicazioni DB2 ottimizzate in termini di prestazioni.

Vengono esaminati in dettaglio gli aspetti che influenzano le prestazioni delle applicazioni e gli interventi da effettuare per ottenere i risultati desiderati.

Vengono inoltre forniti criteri e norme per l'ottimizzazione delle query SQL, ed una chiave di lettura per valutare le scelte effettuate dall'ottimizzatore.

L'esposizione dei relatori è corredata da esercizi che affrontano problematiche di graduale complessità.

## DESTINATARI ▼

- Data base administrator
- Analisti
- Analisti programmatori
- Programmatori

## PREREQUISITI ■

Conoscenza dettagliata dell'SQL; partecipazione al corso "DB2 z/OS: architettura e funzioni" o conoscenze equivalenti.

## CONTENUTI ●

- Analisi delle problematiche e individuazione delle aree di intervento
- Il processo di ottimizzazione
  - componenti e fasi fondamentali,
  - tecniche di accesso ai dati,
  - analisi dei predicati
- Analisi delle prestazioni
  - analisi delle informazioni che compongono il piano di accessi,
  - correlazione tra informazioni statistiche e piano di accessi,
  - impatto delle informazioni statistiche contenute nel catalogo DB2 sull'ottimizzazione
- Disegno dei dati
  - modifiche da apportare al modello dei dati per ottimizzare le prestazioni
- Utilizzo degli indici per migliorare le performance
- Integrità logica dei dati
  - implicazioni applicative e studio delle alternative
- Linee guida per il disegno delle applicazioni DB2 distribuite e non distribuite
- Utilizzo efficiente delle funzioni SQL
  - criteri e norme per individuare l'alternativa più conveniente
- Elementi relativi alle features delle nuove versioni

## DURATA ◆

3 giorni

# ORACLE 11i: ARCHITETTURA E FUNZIONI

## DESCRIZIONE ►

---

Lo scopo del corso è quello di fornire una visione il più completa possibile dell'architettura di un dbms Oracle e delle funzioni che mette a disposizione.

Viene fatta una panoramica sulla gestione delle reti e i parametri di configurazione, mentre vengono illustrate più in dettaglio le diverse componenti del sistema. Sono inoltre esaminate le varie tecniche di Backup e Recovery e l'utilizzo di RMAN.

## DESTINATARI ▼

---

- Data base administrator
- Analisti
- Analisti programmatori
- Programmatori
- Sistemisti

## PREREQUISITI ■

---

Conoscenze di base sui sistemi relazionali e sul linguaggio SQL

## CONTENUTI ●

---

- **RDBMS:** Introduzione (architettura e processi); Spfile; OMF; Sql e iSQL\*PLUS; Cenni su Oracle Enterprise Manager Grid Control
- **Panoramica e Configurazione di Oracle Net**
- **Dizionario Dati: Organizzazione e utilizzo delle informazioni**
- **Definizione di uno schema:** Redo Log File; Datafile; Tablespace; Tablespace Sysaux; Tablespace Undo; Extend; Segmenti e tipi; Rollback Segment; Undo Data; Tabelle; Indici; Vista; Materialized view; Sequence
- **Partizionamento:** Tipi di partizionamento: Range, Hash, List e Composite; Indici partizionati; Gestione delle partizioni
- **Componenti PL/SQL:** Function; Procedure; Package; Trigger
- **Le estensioni Object Relational**
- **Integrità referenziale:** Entity Integrity; Referential Integrity; Check Constraints; Deferred Constraints
- **Gestione della concorrenza:** Modalità, durata e tipologia di lock
- **Gestione della sicurezza:** Utenti; Profili; Ruoli
- **Utility:** Data Pump Export; Data Pump Import; Sql \*loader; External Table
- **Backup e recovery:** Gestioni dall'utente; Panoramica e Configurazione di RMAN

## DURATA ◆

---

3 giorni

## **ALCUNE AZIENDE CHE HANNO SEGUITO I NOSTRI CORSI**

---

3i  
Accenture  
ACI Informatica  
Air Liquide  
Alitalia  
Almaviva  
Atos  
Banca Caboto  
Banca d'Italia  
Banca Popolare di Lodi  
Banca Toscana  
Banksiel  
Bisiel  
Blu Team  
Banca Nazionale del Lavoro  
Caridata  
Cedacri  
Cedati  
Centrale Bilanci  
Comune di Modena  
Consip  
CPI Progetti  
CSI Piemonte  
Datsiel  
EDS  
Electrolux  
Elsag  
ENEL  
Engineering  
Enidata  
Ericsson  
Ferrovie dello Stato  
Fiat Auto  
Fiat Iveco  
Finsiel  
Fondiaria SAI  
Galileo  
Gepin  
Global Value  
GS Supermercati

INA Assitalia  
INAIL  
Infogroup  
Informatica Trentina  
Insiel  
Intesa Sanpaolo  
ISA  
Italgas  
ITS  
KeyG Consulting  
Marconi  
Mediaset  
Monte dei Paschi  
Nuova Tirrena  
Pirelli  
Provincia di Bologna  
Raffinerie di Milazzo  
RAI  
Reale Mutua Assicurazioni  
Sara Assicurazioni  
SAS Institute  
SEP  
Setup  
Siemens Informatica  
Sipra  
Sistemi Informativi  
Sodalìa  
Sofiter  
Softlab 2  
Sogei  
Teksid  
Teleap  
Telecom  
Telesoft  
TIM  
TNT  
Unicredit  
Value Team  
Wind  
Zucchetti



**Tecnet Dati S.r.l**  
C.so Svizzera, 185 – 10149 Torino  
tel. +39 011 771 80 90 – fax +39 011 771 80 92  
<http://www.tecnetdati.it>

