

spazio libri

Managing Software Requirements: A Unified Approach



Autore	Dean Leffingwell, Don Widrig
Editore	Addison Wesley
ISBN	0201615932
Lingua	Inglese
Anno	1999
Prezzo	£ 145.000
Pagine	544

In un progetto software, una fase cruciale e propedeutica alla scrittura del codice è la stesura, in collaborazione con il committente, di un documento dei requisiti, ovvero delle funzionalità che il sistema dovrà offrire. Fondamentale è poi la loro tracciabilità, ossia la corrispondenza fra un requisito e la parte di codice che lo realizza. Infine, i requisiti sono soggetti a cambiamenti, per necessità dello sviluppo o per mutate esigenze del cliente. Il testo tratta la gestione dei requisiti, utilizzando il formalismo UML degli use-case. Il processo viene diviso in sei fasi: analisi del problema; comprensione delle necessità dell'utente; definizione del sistema; gestione dei tempi, costi e requisiti del progetto; raffinamento della definizione del sistema; messa in esercizio. Per la prima fase vengono date cinque best practices: raggiungere un accordo sulla definizione del problema; comprendere le necessità che hanno portato alla richiesta del sistema; identificare gli utenti del sistema e gli stakeholder; definire i confini del sistema; definire i constraints del sistema. La parte dedicata alla comprensione delle necessità dell'utente raccoglie una serie di "esperienze sul campo" riguardanti le interviste, il brainstorming, lo storyboarding, il roleplaying, la prototipazione. Al requisito, "carpito" in maniera maieutica all'utente, occorre associare una serie di attributi che ne descrivono lo scopo, il grado di priorità, lo sforzo richiesto per la realizzazione. Durante la definizione del sistema, i requisiti vengono organizzati, in maniera formale e con l'ausilio di tool. Si redige il documento vero e proprio (Vision document, nel testo), e viene nominato un "campione" del progetto, che ne mantenga una visione globale. La gestione del progetto prevede il soddisfacimento delle specifiche, nei limiti dei tempi previsti e delle risorse assegnate. Ai tradizionali modelli waterfall ed a spirale viene affiancato l'originale modello RUP (Rational Unified Process). Durante lo sviluppo del sistema occorre definire i requisiti architetturali, e raffinare iterativamente lo use-case model proposto in prima stesura. Vengono introdotte la metodologia Modern SRS, tecniche di misura della qualità dei requisiti, e metodi formali di specifica dei requisiti architetturali: pseudocodice, macchine a stati finiti, alberi di decisione, diagramma delle attività, modelli entità-relazioni, analisi O-O, analisi strutturata. L'ultima parte è dedicata alla "messa in opera" del sistema: corrispondenza fra requisiti e prodotto finito, matrici di tracciabilità, validazione ed uso del ROI per determinarne l'effort, gestione dei cambiamenti. Un capitolo finale riassume, ad uso del lettore già esperto, le sei fasi precedentemente descritte.

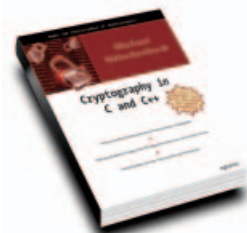
Pro

Un testo di ammirevole completezza, chiarezza, precisione, che riesce a formalizzare l'informalizzabile, ovvero il complesso lavoro di interazione con l'utente ed il team di sviluppo e di "visione" del progetto che è la caratteristica distintiva dell'analista programmatore, e di gestione della configurazione che è compito del responsabile della qualità.

Contro

È perlomeno sconcertante che assieme ad un testo di tale respiro non venga fornita una demo di un tool per la gestione dei requisiti, come Rational® RequisitePro, e la versione elettronica della preziosa documentazione citata nel libro.

Cryptography in C and C++



Autore	Michael Welschenbach
Editore	Apress
ISBN	189311595X
Lingua	Inglese
Anno	2001
Prezzo	£ 114.000
Pagine	432 + Cd Rom

Dirò subito che questo "Cryptography in C and C++", giunto alla seconda edizione, è un libro prezioso. La sua preziosità deriva dal fatto che la gran parte degli argomenti che tratta non sono facilmente reperibili nemmeno su Internet. Abbiamo di fronte quindi un libro che parla di crittografia moderna senza indugiare sulla storia o sulla teoria, ma partendo direttamente dagli aspetti matematici e computazionali dei principali sistemi crittografici asimmetrici oggi in uso: RSA e Rijndael (il nuovo American Encryption Standard). Si tratta di un viaggio di conoscenza a dir poco arduo, ma affascinante. I suddetti sistemi infatti si basano largamente sull'aritmetica dei grandi numeri, ragion per cui occorre "insegnare" al calcolatore a fare addizioni, sottrazioni e ogni altra operazione con questi mostri, che vengono rappresentati con un numero di bit ben superiore ai 32 che le nostre CPU sanno maneggiare. Perciò il libro comincia con una dettagliata analisi dei problemi derivanti dal calcolo con i grandi numeri, e illustra, passo dopo passo, una libreria in C che ha lo scopo di trattare con questi numeri (e con la loro rappresentazione, in qualsiasi base). Ogni capitolo può riservare sorprese: scopriamo per esempio che le divisioni tra grandi numeri interi vengono trattate con lo stesso algoritmo che ci è stato insegnato alle elementari, con tanto di diagramma. La libreria viene poi incapsulata, nel capitolo successivo, in una serie di classi C++. A questo punto il lettore ha gli strumenti per affrontare e capire le logiche che stanno dietro RSA e Rijndael, qui completamente descritti e commentati, con il codice sorgente disponibile sia sul libro, sia sull'utile CDROM a corredo. Tutto il codice è compilabile con il compilatore C/C++ di GNU (disponibile per Linux e Cygwin) o con Visual C++. In definitiva non posso che consigliare caldamente l'acquisto di questo libro: se state affrontando (per hobby o per lavoro) il tema della crittografia a chiave pubblica è addirittura necessario. Se invece siete solo dei cultori della matematica e dell'informatica nel suo senso più nobile, questo libro potrà essere fonte di divertimento per molti mesi. Alla larga, invece, chi cerca veloci tutorial sull'argomento o applicazioni già pronte per l'uso, o chi non abbia adeguate conoscenze di sviluppo in C o C++.

Pro

Un libro eccezionale, per profondità e accuratezza. Coglie l'obiettivo di fornire al lettore gli strumenti per comprendere la crittografia contemporanea.

Contro

Manca completamente un accenno alla teoria o alla storia della crittografia, che avrei magari voluto vedere accennate in una eventuale appendice.

Essential JAVA Style – Pattern for Implementation



Autore	Jeff Langr
Editore	Prentice Hall
ISBN	0130850861
Lingua	Inglese
Anno	1999
Prezzo	€ 125.000
Pagine	277

Dal momento che ormai anche la tecnologia ha un suo mercato e di conseguenza un suo marketing, per vendere un libro non è più una condizione necessaria che esso sia interessante, utile, sensato. Basta prendere qualche parola chiave del momento, metterla nel titolo e distribuire il prodotto in tutte le librerie. "Java" e "pattern" sono sicuramente due di queste buzzword e quello che ho pensato dopo aver letto *Essential Java Style* è che appartenga alla categoria "molto fumo e poco arrosto". In verità lo scopo del libro è ambizioso: fissare "a standard for building cost-effective, universally maintainable systems". Ma le lacune iniziano nel sottotitolo, "pattern for implementation". In realtà nel libro non si trovano affatto pattern per implementare qualcosa, bensì per come scrivere codice, cioè "coding". Il libro è suddiviso in 7 capitoli e 2 appendici; i pattern sono contenuti nei primi 6 capitoli, suddivisi per "Behaviour - Methods", "Behaviour - Messages", "State patterns", "Collections", "Classes", "Formatting". Il settimo capitolo "Development Example" fornisce un esempio che mette in pratica le tecniche proposte. Un'appendice riguarda l'ottimizzazione del codice e l'ultima sezione è una guida di riferimento che riassume i pattern descritti nei precedenti capitoli. Vengono presentati in totale 92 pattern, probabilmente troppi. Molti sono la messa in bella copia di cose ovvie; alcuni sono invece molto opinabili come il "query method" che presenta un modo di nominare i metodi di accesso alle proprietà di un oggetto differente dalle convenzioni per i JavaBeans. Particolarmente fastidiosa la sezione riguardante le Java Collection, dove vengono messi sotto forma di pattern come usare le classi della libreria (forse era necessario aumentare le pagine?). Opinabile anche una delle regole d'oro presentate nel testo "Make it run, Make it right, Make it fast". E il design dove lo mettiamo? C'è comunque uno spunto positivo nel porre un'accento particolare sulla leggibilità, chiarezza, manutenibilità del codice. Concludendo, visto anche il costo di copertina, considero il rapporto qualità/prezzo di *Essential Java Style* piuttosto sfavorevole.

Pro

Si presenta bene, con una grafica sobria, gradevole e leggibile. Ottimo l'accento sull'importanza della manutenibilità del codice.

Contro

Delude le aspettative. Alcuni pattern presentano cose ovvie, la validità di altri è discutibile. Non tutti sono realmente utili. Il prezzo di € 125.000 è sicuramente elevato.

Stefano Fornari

Reti di computer



Autore	Andrew S. Tanenbaum
Editore	Utet Libreria
ISBN	8877504536
Lingua	Italiano
Anno	1998
Prezzo	€ 90.000
Pagine	782

Gli argomenti trattati in questa terza edizione si allineano perfettamente al nuovo scenario tecnologico che vede Internet come protagonista indiscusso delle applicazioni basate sulle reti di calcolatori. Quest'opera, non tralasciando l'approccio classico definito dal modello ISO/OSI, affronta le nuove tecnologie di comunicazione cellulare, satellitare e ATM nonché i nuovi mezzi di comunicazione come la fibra ottica. Maggior enfasi è data al TCP/IP che è oggi il protocollo dominante. Non manca, inoltre, una panoramica sulle caratteristiche di IPv6, IP mobile e IP multicasting. Gli argomenti sono suddivisi non secondo i sette livelli del modello ISO-OSI, bensì secondo un modello ibrido a cinque livelli: livello fisico, data-link, rete, trasporto e applicazione. Questo modello si avvicina molto a quello TCP/IP. Per ogni livello, prima vengono esaminate le problematiche e gli approcci generali utilizzati per risolverle e poi vengono esaminate le varie tecnologie che interessano il livello stesso, il tutto descritto in modo assolutamente lontano dall'essere noioso. Nella parte dedicata al livello fisico troviamo argomenti quali ISDN a banda larga, radio cellulare e satelliti a orbita bassa. In quella dedicata al livello data-link troviamo argomenti quali la correzione degli errori, lo standard IEEE 802 per LAN e WAN, i bridge e un esempio di un semplice protocollo data-link con tanto di listato in linguaggio C. Nella parte dedicata al livello rete, prima troviamo un'esauriente discussione sugli algoritmi di routing e di controllo della congestione e poi la trattazione del livello rete sia in Internet, in particolare del protocollo IP, e sia nelle reti ATM. Nella parte dedicata al livello di trasporto vengono trattati i protocolli TCP e UDP e i protocolli di livello AAL di ATM e, come nel livello data-link, viene presentato l'esempio di un semplice protocollo di trasporto con annesso listato in linguaggio C. In ultimo, nella parte dedicata al livello applicazioni vengono presentati aspetti di sicurezza delle reti, il DNS, il modello SNMP, il World Wide Web con tanto di introduzione a Java e al suo utilizzo per scrivere applicazioni per il web e in conclusione una panoramica delle tecnologie multimediali che interessano applicazioni basate su reti di computer, come, ad esempio, il video su richiesta. Questo libro, scritto dall'autorevole professor Andy Tanenbaum, fa parte di una quadrilogia conosciuta in tutto il mondo e molto usata in ambito universitario per scopi didattici.

Pro

Argomenti descritti in modo semplice e mai noioso. L'autore presenta in modo organico e ben organizzato una grande varietà di tecnologie riguardanti le reti di computer. I numerosi esempi facilitano la comprensione dei vari argomenti facendo di questo testo un utile strumento didattico.

Contro

Poiché questo libro ha come obiettivo quello di spiegare le caratteristiche generali dei vari aspetti che interessano le reti di computer, spesso può sembrare troppo superficiale su alcuni argomenti. Inoltre la traduzione di alcune tabelle fa rimpiangere la versione in inglese.

Diego Bovenzi