

Guida pratica alla gestione dei progetti software

Valerio Teta
Ciro Di Leva



I diritti di traduzione, di riproduzione, di memorizzazione elettronica e di adattamento totale o parziale con qualsiasi mezzo (compresi i microfilm e le copie fotostatiche) sono riservati per tutti i paesi.

Nomi e marchi citati sono generalmente depositati o registrati dalle rispettive case produttrici.

Copyright © 2003 CUEN S.r.l.
In Area Industria della Cultura
Via Cordoglio, 156 – 80124 NAPOLI
Tel. 081 7352442 fax 081 2301044
e-mail cuen@zeus.idis.unina.it

Sommario

Sommario	3
Presentazione	9
Autori	11
Ringraziamenti	13
Premessa	15
Come nasce l'iniziativa.....	15
A chi è rivolta la guida	15
Articolazione della guida	16
Introduzione.....	17
Definizioni.....	18
Progetti software	21
Project Management.....	23
Ingegneria del software	24
Il Project Management ed il PMI.....	26
Le aree di conoscenza dell'ingegneria del software	33
Gli attori del software project management	35
Il Project Manager.....	40
Relazioni tra strutture organizzative e project management	42
Processi software	45
Le fasi del ciclo di vita di un prodotto software.....	49
Riepilogo.....	50
Definizione del progetto software	51
Generalità sulla definizione del progetto.....	52
Fattibilità e stima preliminare	53
Situazione di partenza e nuove esigenze.....	54
Specifiche di massima del nuovo sistema	56
Scelta del modello di ciclo di vita	60
Soluzione proposta.....	62

Dimensionamenti – generalità	62
Dimensionamento del prodotto	64
Dimensionamento del processo.....	66
Dimensionamento del controllo – Identificazione iniziale del rischio	68
Conclusioni.....	70
Dallo studio all'accordo contrattuale	70
Definizione dell'ambiente di pianificazione	72
Definizione dell'archivio di progetto	72
La struttura dell'archivio di progetto	73
I mezzi di archiviazione	74
Regolamentazione degli accessi	74
Riepilogo.....	75
Pianificazione del progetto software.....	77
Definizione e sequenziazione delle attività.....	79
Definizione di durate e date.....	81
Pianificazione delle risorse.....	82
Previsione dei costi.....	83
Sviluppo della schedulazione	84
Pianificazione della qualità.....	87
Sviluppo del piano dei controlli	87
Sviluppo del piano di monitoraggio.....	88
Sviluppo del piano degli acquisti	89
Piano di sviluppo dell'organizzazione di progetto	92
Pianificazione della configurazione.....	95
Sviluppo del piano di gestione dei rischi	96
Identificazione dei rischi	96
Analisi dei rischi.....	97
Pianificazione dei rischi	98
Sviluppo del piano di comunicazione.....	99
Oggetto	99
Destinatario	100
Frequenza	100
Formato del documento	100
Mezzi di comunicazione.....	100
Riepilogo.....	103
Esecuzione del progetto software	105
Introduzione.....	106
Kick-off Meeting.....	107

Assegnazione dei compiti individuali	108
Esecuzione dei processi software	110
Definizione del requisito utente	112
Raccolta dei requisiti	112
Documentazione dei requisiti	114
Accettazione dei requisiti	114
Gestione dei requisiti	115
Analisi dei requisiti software	115
Definizione del modello concettuale dei dati	116
Definizione del modello delle funzioni	116
Analisi dei requisiti software non funzionali	117
Prototyping	117
Produzione del documento di specifiche funzionali	117
Disegno Architetture	118
Progettazione dell'infrastruttura tecnologica	118
Progettazione della logica applicativa	118
Progettazione dei dati	119
Produzione del documento di progettazione	119
Disegno di dettaglio	120
Disegno di dettaglio dei componenti	120
Produzione del documento di disegno di dettaglio	120
Costruzione del componente e test unitario	121
costruzione del componente	121
costruzione della procedura di test	121
Esecuzione del test	121
Collaudo ed integrazione	122
Collaudo di integrazione	122
Collaudo di sistema	122
Collaudo utente	123
Conclusione del compito individuale	123
Distribuzione dell'informazione	124
Sviluppo della squadra (team)	126
Valutazioni iniziali	126
Condivisione/aggiornamento del piano di sviluppo	127
Assicurazione della qualità	128
Sorveglianza del test	129
Ispezioni	130
Individuazione ed eliminazione dei moduli difettosi	131

Il database dei difetti.....	131
Acquisti	131
Verifica dell'ambito del progetto.....	132
Riepilogo.....	134
Controllo del progetto software	135
Introduzione.....	136
Raccolta dei dati e comunicazione dei risultati sullo stato	136
Controllo dei rischi	139
Controllo all'ambito del progetto	142
Il processo di gestione delle modifiche ai requisiti	143
Il processo di gestione delle modifiche alla progettazione.....	146
Esempio di procedura	146
Il sistema di indicatori	150
Controllo dei costi	152
Controllo dei tempi.....	156
Controllo della qualità del prodotto.....	158
Distribuzione dei difetti per componente	159
Distribuzione dei difetti per origine.....	159
Distribuzione dei difetti per fase.....	160
Andamento dei difetti nel tempo	161
Controllo sul processo di gestione delle Risorse Umane.....	166
Risoluzione dei problemi	168
Riepilogo.....	169
Chiusura del progetto software	171
Introduzione.....	172
Chiusure tecniche	173
Installazione del prodotto software.....	173
Consolidamento della release software	174
Consegna dei documenti tecnici	174
Chiusura del dossier della qualità	175
Rilascio del personale	175
Verifica degli impegni individuali e definizione di un piano di uscita	176
Valutazione della prestazione	176
Valutazione delle competenze finali	176
Riunione di rilascio	177
Chiusure contrattuali	177
Chiusure amministrative	177
Riepilogo.....	178

Riepilogo.....	183
Appendici.....	185
Studio di fattibilità.....	186
Introduzione.....	190
Metodologia "EXTREME PROGRAMMING"	190
Metologia "SCRUM"	191
Metologia di Booch, Jacobson, Rumbaugh	191
Altre metodologia.....	191
Esempio di utilizzo di Rational Rose.....	192
Use Case View	192
Logical View	195
Component View.....	195
Deployment View	195
Esempio.....	195
I PATTERN NELL'INGEGNERIA DEL SOFTWARE	204
Alcune definizioni di pattern	204
Scheda descrittiva di un pattern.....	204
Elementi essenziali costituenti un pattern.....	205
Perché i pattern?	206
Che cosa sono i pattern	206
Che cosa non sono i pattern	206
Perché i pattern sono utili	207
I differenti tipi di pattern.....	207
Esempi di pattern.....	208
Riferimenti bibliografici	208
Libri sui pattern	209
In Internet	209
Scheda prodotto RATIONAL ROSE:	211
Scheda prodotto RATIONAL UNIFIED PROCESS (RUP):	211
Altri prodotti RATIONAL:	211
Scheda prodotto MICROSOFT VISIO	212
Scheda prodotto MICROSOFT PROJECT	213
Scheda prodotto COMPUTER ASSOCIATED ERWIN.....	213
Elenco dei principali tool e delle principali metodologie di Project Management	214
Introduzione ai Function Point	216
Calcolo degli ILF e EIF	217
Calcolo dei Function Point Transazionali (EI,EO,EQ).....	218

Calcolo dei Function Point non pesati	219
Calcolo del valore del fattore di aggiustamento	219
Calcolo finale dei Function Point	219
COCOMO Cenni.....	221
Metodo del conteggio dei blocchi funzione	222
Il modello a spirale.....	224
Bibliografia.....	226
Altri riferimenti bibliografici	227
Collegamenti ai principali siti web	232
Glossario	233

II

Presentazione

Il Project Management nel software è ancora materia conosciuta da pochi e l'industria software è zeppa d'esempi di fallimenti di progetti dovuti ad innumerevoli cause che spesso hanno un denominatore comune. Questo denominatore comune cui sono ascrivibili tanti fallimenti è costituito da un Project Management non sostenuto da metodi efficaci che porta a:

- *Progetti che costano molto più di quanto stimato e che si concludono molto più tardi del previsto,*
- *Progetti che non si concludono perché non se n'è saputa gestire, governare e ridurre la complessità*
- *Progetti che, pur se formalmente conclusi, alla prova del campo si rivelano clamorosamente incapaci di soddisfare le aspettative dei clienti.*

Il libro di Valerio Teta e Ciro Di Leva è un libro di cui chi lavora nel campo software semplicemente ne sentiva il bisogno: finalmente una guida pratica fondata su solide basi teoriche di una materia spinosa, complessa ed in continua evoluzione.

Sul Project Management nel software si è scritto molto ma spesso quanto scritto riflette troppo una o più delle seguenti caratteristiche:

- *Chi scrive non proviene dall'industria software e non vive dall'interno le problematiche relative alla conduzione di progetti software con i vincoli, i tempi, le risorse sempre limitatissime di chi opera nel settore globale ed estremamente competitivo del mercato software;*
- *Sono affrontati e sviscerati argomenti importanti ma parziali, magari perché dominio di ricerca di quel particolare autore o gruppo. In questi casi gli scritti sono a volte ottimi come approfondimento dei singoli aspetti ma poco utili a chi cerchi un manuale che presenti la materia in maniera organica e completa, a coloro che aspirano a diventare Project Manager o, essendolo già sul campo, vogliono consolidare le loro conoscenze sull'argomento;*

- *Il livello di presentazione è astratto, il linguaggio usato lontano da chi vuole apprendere o migliorare quello che in fondo è una professione con mille problemi pratici che vanno, ad esempio, dalla suddivisione dei compiti alla comunicazione interna al progetto, al reporting verso l'esterno, alla motivazione delle persone.*

Questo libro affronta la materia in maniera organica ma nello stesso tempo in modo semplice e snello, facendosi leggere con piacere presentando direttamente, facendone cogliere la potenza, molte tra le migliori Best Practices dell'industria software moderna. Dal libro traspare l'esperienza di chi queste problematiche le ha vissute su progetti reali e complessi. Ho trovato quindi questo libro specificamente adatto sia per chi vuole un buon riferimento nel suo percorso di autoapprendimento sia per chi vuole usarlo come materiale didattico di riferimento in un corso orientato alla formazione di figure professionali capaci di gestire progetti software.

Ettore Toscano
Segretario AICQ Meridionale

L'iniziativa di Valerio Teta e Ciro Di Leva è di notevole valore, non solo per i contenuti di sicuro spessore grazie alla loro maturata e consolidata esperienza, ma soprattutto apre una strada, traccia una guida. Mi auguro che il loro lavoro sia seguito da molti altri inaugurando un filone di pubblicazioni realizzate nell'area culturale dell'AICQ-Meridionale.

Diego Cerra
Presidente AICQ-Meridionale

Autori

Valerio Teta

Laureato in ingegneria elettronica. Dal 1975 progettista di software applicativo. Dal 1980 al 1982 in una multinazionale statunitense dell'informatica come sistemista in area software di base. Dal 1983 al 1986 responsabile dell'area sviluppo software in una softwarehouse napoletana. Dal 1986 al 1997 direttore dell'area sviluppo di impresa informatica di carattere nazionale. Dal 1994 responsabile del progetto del sistema qualità aziendale. Dal 1997 al 2000 quality manager. Dal 2000 responsabile di uno spin-off aziendale inizialmente dedicato alla gestione in outsourcing dei sistemi qualità del gruppo di origine ed orientato alla erogazione di servizi di consulenza e formazione.

Valutatore (1993) e Progettista (1995) di Sistemi Qualità nei Servizi secondo lo schema FITA-CONFIDUSTRIA.

Ciro Di Leva

Laureato in Scienze dell'Informazione.

Dal 1985 al 1993 come free-lance realizza applicazioni per piccole imprese.

Dal 1993 al 1995 progettista di software per una azienda di ICT.

Dal 1996 al 1998 lavora in una multinazionale statunitense del software come Analista nell'area delle Telecomunicazioni.

Dal 1996 è responsabile di progetti software nell'area sviluppo di una impresa informatica di carattere nazionale.

Dal 1998 ha una certificazione di Project Management Professional (PMP) del Project Management Institute.

Ciro Di Leva può essere contattato al seguente indirizzo di posta elettronica: **ciro.dileva@poste.it**.

Ringraziamenti

A Lorenzo

*La guida compendia risultati di anni di lavoro su progetti complessi.
Gli autori desiderano ringraziare :*

- *L'Alta Direzione del Gruppo Gepin per le opportunità e le risorse che nel tempo sono state rese disponibili in un ambiente fortemente dinamico;*
- *I collaboratori per la condivisione di problemi, soluzioni e conoscenze nei progetti vissuti insieme;*
- *Gli amici per l'incoraggiamento, il consiglio ed il sostegno nell'avventura della guida;*
- *I familiari per la pazienza dimostrata durante tutto il periodo di lavoro.*



Premessa

*“Uno spirito aperto e centrato è in grado di osservare le cose in modo distaccato.
Coltivate quindi la vostra saggezza e le vostre qualità interiori.
Sviluppate il senso di giustizia, imparando a distinguere il bene dal male
e studiando una per una le Vie delle arti.
Quando avrete imparato a non lasciarvi trarre in inganno, avrete realizzato la
saggezza della strategia. E' una saggezza del tutto particolare.”*
Il libro dei cinque anelli - Miyamoto Musashi

Come nasce l'iniziativa

La guida nasce come sussidio didattico al corso AICQ Meridionale sulla gestione dei progetti software ed intende offrire alle Persone interessate una chiave di orientamento nei processi di sviluppo nel rispetto di principi e vincoli espressi dalla Vision 2000.

La realizzazione di un'applicazione software richiede conoscenze, competenze, tecniche e strumenti che appartengono a discipline diverse: la pratica di gestione dei progetti software (il Software Project Management - SPM) deve essere capace di integrare i metodi dell'ingegneria del software, in particolare i criteri per la definizione dei processi di sviluppo, con i metodi del project management, in particolare i criteri per l'avvio, la definizione, la pianificazione, l'esecuzione, controllo e chiusura di un progetto di sviluppo software.

A chi è rivolta la guida

La guida si rivolge essenzialmente alle Persone che hanno la responsabilità della conduzione dei progetti software (Software Program Manager e Software Project Manager). La guida può essere utile a due ulteriori classi di Attori del Software Project Management:

Tecnici (Analisti e Progettisti software) che vogliono orientare il proprio percorso professionale verso la gestione di progetto;
Alta Direzione e Direzioni di funzione che vogliono migliorare la capacità di processo della propria organizzazione impostando la gestione per progetti.

Articolazione della guida

L'obiettivo di fornire al lettore gli strumenti per affrontare in modo strutturato le fasi di un progetto software si riflette nell'articolazione della guida in sei capitoli.

Nel primo capitolo, "**Introduzione alla gestione dei progetti software**", si danno le definizioni iniziali, si riassume il contesto di riferimento e si presenta l'approccio scelto ovvero una possibile integrazione, nel ciclo di vita del progetto, dei processi orientati alla direzione e dei processi orientati allo sviluppo.

Nel secondo capitolo, "**Definizione**", si presentano le relazioni tra l'opportunità offerta e le scelte progettuali che conducono alla definizione di un accordo contrattuale – identificazione del problema, definizione dello scopo dell'iniziativa progettuale, stima del prodotto software da consegnare e del processo da eseguire.

Nel terzo capitolo, "**Pianificazione**", presa la decisione di avviare il progetto si descrivono i compiti che, a partire dal contratto, consentono di sviluppare una pianificazione esecutiva del progetto caratterizzandone gli aspetti significativi (tempi, costi, qualità, Risorse Umane, ...).

Nel quarto capitolo, "**Esecuzione**", si descrive l'attuazione del piano – avviamento del lavoro con l'acquisizione delle risorse, la predisposizione dell'ambiente di lavoro e la costruzione della squadra; esecuzione delle attività pianificate e consegna dei prodotti previsti con le relative attività di comunicazione delle informazioni riguardanti lo stato del progetto; sorveglianza dell'andamento.

Nel quinto capitolo, "**Controllo**", si descrivono l'applicazione dei controlli previsti ai processi in esecuzione, la valutazione dei risultati disponibili e l'eventuale attuazione dei processi relativi alla gestione dei problemi e delle azioni correttive.

Nel sesto capitolo, "**Chiusura**", si descrivono la conclusione ordinata del progetto, le valutazioni dei risultati finali e l'immagazzinamento dell'esperienza compiuta – nuove conoscenze, nuove competenze e nuove tecnologie.

Introduzione

- 1.1** Definizioni
- 1.2** Progetti software
- 1.3** Project Management
- 1.4** Ingegneria del software
- 1.5** Il Project Management e il PMI
- 1.6** Le aree di conoscenza dell'Ingegneria del software
- 1.7** Gli attori del software Project Management
- 1.8** Il Project Manager
- 1.9** Relazione tra strutture organizzative e project management
- 1.10** I Processi Software
- 1.11** Le fasi ed il ciclo di vita di un progetto Software

*“Lo ufficio d’uno buono capitano è montare il primo a cavallo
e l’ultimo scenderne”*

La vita di Castruccio Castracani da Lucca – Niccolò Machiavelli

Dopo le definizioni iniziali di progetto e di software nel capitolo si esemplifica un possibile scenario dove s’innesca un progetto software sotto forma di appalto.

La Direzione dell’Impresa appaltatrice definisce un’organizzazione temporanea dedicata al progetto software ed ha la necessità di definirne il governo.

Il problema posto è stato scomposto in due parti: da una parte è stato individuato il processo gestionale (Project Management) e dall’altra il processo tecnico (Ingegneria del Software).

L’approccio scelto fa riferimento a SWEBOK (IEEE Computer Society) per l’Ingegneria del software ed a PMBOK (Project Management Institute) per il Project Management.

Individuati i temi da interpretare si è passati alla definizione degli Attori primari e delle Parti interessate con uno specifico approfondimento per il ruolo del Project Manager.

È stato brevemente descritto il contesto dell’azione di progetto.

L'organizzazione temporanea dedicata al progetto software vive la sua stagione all'interno dell'impresa appaltatrice, ne eredita la cultura, ne impegna la struttura ed alla fine ne influenza l'evoluzione.

Il Capitolo si conclude riprendendo il tema da interpretare e specificandolo attraverso i concetti di processo software e di modello di ciclo di vita.

La definizione dei processi software è vista come un sistema flessibile di produzione sviluppato in coerenza con la strategia d'impresa.

La scelta del modello di ciclo di vita, infine, può consentire un'adeguata personalizzazione della capacità di processo dell'impresa alle specifiche esigenze di un Cliente ed alle specifiche clausole di un contratto.

Caso di studio.

Per cominciare definiamo il contesto del caso di studio ed individuiamo gli scenari nei quali nasce il nostro progetto software.

XYZtelco è un'azienda a carattere regionale erogatrice di servizi telefonici e si rivolge sia all'utenza familiare sia all'utenza business.

Il volume di traffico di XYZ è cresciuto con una dinamica più vivace di quanto previsto: l'Alta Direzione ha deciso di rinnovare il proprio sistema di gestione delle informazioni per governare e sostenere i fenomeni di crescita.

La strategia di rinnovamento è stata strutturata in tre programmi:

1. amministrazione e controllo di gestione;
2. customer care;
3. infrastrutture di supporto all'erogazione del servizio.

Il programma più importante della strategia di rinnovamento è di customer care ed il progetto più importante dell'area customer care è quello di fatturazione.

Definizioni

Per arrivare alla definizione di progetto possiamo partire dal senso comune bene rappresentato da un popolare **[Zanichelli]** Vocabolario della Lingua Italiana

progetto

s. m.

1. *Piano di lavoro, ordinato e particolareggiato, per eseguire q.c.*
2. *Idea, proposito, anche vago, bizzarro o difficilmente attuabile: un progetto di matrimonio; fare troppi progetti; SIN. Intenzione, proponimento, proposito.*
3. *Insieme di calcoli, disegni, elaborati necessari a definire l'idea in base alla quale realizzare una costruzione o una macchina.*

piano (3)

s. m.

1. *Disegno industriale che rappresenta un oggetto, una macchina, una costruzione e sim. in proiezione su di una superficie piana.*
2. *Programma che determina i mezzi, i compiti e i tempi per conseguire alla scadenza un determinato risultato - Piano regolatore, atto amministrativo che determina la futura configurazione di una zona di insediamento e di un nucleo urbano preesistente - Piano di sviluppo economico, complesso di provvedimenti elaborati dallo Stato per garantire uno sviluppo equilibrato dell'economia.*
3. *Progetto, proposito: mandare a monte i piani di qc.*

programma

s. m. (pl. -i)

1. *Enunciazione verbale o scritta di ciò che è necessario o che ci si propone di fare: il programma della manifestazione - Proposito, progetto: ha fatto un programma per la serata - Trasmissione radiotelevisiva: i programmi della serata.*
2. *Piano di lavoro e di studi da realizzare entro un certo periodo scolastico: programma d'esame; svolgere il programma.*
3. *Opuscolo contenente le informazioni essenziali su uno spettacolo distribuito in teatri o in sale di concerto - Fuori programma, (fig.) cosa inaspettata, imprevista.*
4. *(elab.) Sequenza di istruzioni codificate che, inserite in un computer, lo abilitano a eseguire un ciclo completo di operazioni: programma di scrittura.*

La prima cosa che abbiamo immaginato quando abbiamo cercato di fissare nel contesto di nostro interesse il significato del vocabolo *progetto* è stata la realizzazione di grandi opere quali ad esempio un ponte, una nave, un palazzo o una metropolitana ed in ogni caso abbiamo pensato al coinvolgimento di un numero considerevole di risorse (umane e materiali) coordinate in modo appropriato per un tempo finito e finalizzate verso un unico scopo.

Ma quali sono le definizioni condivise di progetto? Di seguito ne riportiamo due, la prima di Kerzner e la seconda del Project Management Institute:

Un progetto può essere considerato come un insieme d'attività e compiti che hanno le seguenti proprietà:

- ***Hanno un obiettivo specifico da completare all'interno di certe specifiche***
- ***Hanno date d'inizio e fine definite***
- ***Hanno fondi limitati***
- ***Consumano risorse***

Harold Kerzner

Un progetto è uno sforzo temporaneo atto a creare un prodotto o un servizio unico.

Project Management Institute

L'ultima definizione pone l'accento su due punti; la temporaneità dello sforzo e la realizzazione di un prodotto o un servizio unico.

La temporaneità implica che il progetto avrà una durata finita, tale durata può essere di giorni o anni e può impiegare una o più risorse (umane e materiali) per svolgere il lavoro richiesto.

L'unicità del prodotto o servizio si riferisce al fatto che il progetto realizzerà qualcosa in esemplare unico che non era mai stato fatto o non esisteva.

Il software

I progetti cui siamo interessati sono quelli che hanno per fine la realizzazione o l'adeguamento di (prodotti) software.

Cerchiamo ora di giungere alla definizione di software attraverso approfondimenti successivi tratti dalla normativa di sistema e di settore.

Software come categoria di prodotto

Prodotto: risultato di un processo.

Processo: insieme d'attività correlate o interagenti che trasformano elementi in entrata in elementi in uscita.

Nota alla definizione di prodotto – Esistono quattro categorie generali di prodotti:

1. Servizi (esempio: trasporto);
2. Software (esempio: un programma per computer, il contenuto di un vocabolario);
3. Hardware (esempio: la parte meccanica di un motore);
4. Materiali da processo continuo (esempio: un lubrificante).

UNI EN ISO 9000:2000

Composizione del software

Il prodotto software è l'insieme di programmi per computer, procedure e possibilmente documentazione e dati associati (esempio: specifiche di prodotto, schema di data base, casi di test, manuale utente, istruzioni operative, codice sorgente, codice eseguibile, dati).

Nota – un prodotto software può essere sviluppato per essere consegnato come prodotto a se stante, come parte integrata in un altro prodotto o per essere usato nel processo di sviluppo.

ISO/IEC 12207:1995

Che cosa è ed a cosa serve il software

Il software è un prodotto intangibile costituito da stringhe di codici che rappresentano informazioni utili al governo di un processo.

TL 9000:2000

Caso di studio.

XYZ conduce un'indagine interna sulle esigenze software interne al processo aziendale di gestione delle vendite e del ciclo attivo: sulla base dei fatti emersi decide di avviare il processo d'acquisizione di un'applicazione di fatturazione. XYZ intraprende l'acquisizione attraverso i seguenti passi:

- 1- Definisce i requisiti dell'applicazione da acquisire;
- 2- Prepara una richiesta d'offerta e seleziona due Fornitori come candidati all'aggiudicazione del contratto.

Progetti software

In generale un progetto software può nascere quando una specifica esigenza software interna al processo aziendale di un possibile Cliente s'incontra con una specifica capacità di un Fornitore.

Il Cliente ed il Fornitore possono essere schematizzati come Organizzazioni specifiche per il software nel senso definito dalla norma **[ISO/IEC 12207]** e cioè Organizzazioni che comprano, vendono, producono, gestiscono in esercizio ed assistono il software come componente rilevante e significativa del proprio prodotto.

La norma citata suggerisce una suddivisione dei processi di un'Organizzazione specifica per il software in tre ampie classi:

- Processi primari
- Processi di supporto
- Processi organizzativi.

La norma citata presenta un secondo livello di strutturazione:

- Processi primari
 - Acquisizione
 - Fornitura
 - Sviluppo
 - Esercizio
 - Manutenzione
- Processi di supporto
 - Documentazione
 - Gestione della configurazione
 - Assicurazione della qualità
 - Verifica
 - Validazione
 - Riesame
 - Audit
 - Risoluzione dei problemi

- Processi organizzativi
 - Direzione
 - Infrastrutture
 - Miglioramento
 - Formazione.

I processi primari riguardano responsabilità e soggetti principali coinvolti nel ciclo di vita del software:

- **Acquisizione** definisce le attività del Cliente che acquisisce un software;
- **Fornitura** definisce le attività del Fornitore che fornisce un software;
- **Sviluppo** definisce le attività dell'Organizzazione che definisce e sviluppa un'applicazione software;
- **Esercizio** definisce le attività dell'Organizzazione che eroga il servizio di conduzione operativa di un impianto informatico consentendo agli utenti istituzionali di interagire con l'applicazione software secondo predeterminati livelli di servizio;
- **Manutenzione** definisce le attività dell'Organizzazione che eroga il servizio di manutenzione software ovvero gestisce gli interventi di carattere correttivo ed evolutivo per consentire all'applicazione di rimanere in esercizio secondo predefiniti livelli di servizio.

Caso di studio.

XYZ ha deciso di acquisire all'esterno anziché sviluppare all'interno un'applicazione software di fatturazione da integrare nel proprio ciclo attivo ed ha selezionato due Fornitori:

ABCsoftwaread hoc, azienda specializzata nello sviluppo di software su commessa;

QWEsoftwareofftheshelf, azienda specializzata nella commercializzazione di software e nell'erogazione di servizi di consulenza.

XYZ richiede ad entrambi un'offerta corredata da uno studio di fattibilità basato sui seguenti input: cosa fare, cosa non fare, vincoli e requisiti non funzionali.

Entrambi i Fornitori attivano il proprio processo di fornitura:

ABC presenta una soluzione caratterizzata dallo sviluppo di un'applicazione software tagliata sulla misura delle esigenze del Committente;

QWE presenta una soluzione caratterizzata dalla scelta/configurazione di componenti standard di una SUITEtutto pronto commercializzata già da un anno con un'eventuale personalizzazione per soddisfare i requisiti lasciati scoperti dalla SUITE.

XYZ valuta le due proposte e sceglie il Fornitore ABC anche per la facilità d'integrazione con l'applicazione di controllo di gestione in via di sviluppo.

I due processi (Acquisizione e Fornitura) che hanno reso possibile l'ingaggio iniziale tra Cliente e Fornitore continueranno ad interagire fino alla fine producendo il contratto e quella che si chiama vista contrattuale del progetto nel cui ambito avverranno gli scambi contrattuali.

Preso la decisione di avviare un progetto, la Direzione dell'Organizzazione Fornitrice deve innanzi tutto assegnare la responsabilità di condurre il progetto ad una Persona d'ora in poi denominata Project Manager (Capo-progetto).

Project Management

Riprendendo l'impostazione di Richard Thayer **[THAYERSPM]** cerchiamo di definire il concetto di project management (direzione/gestione di progetto) partendo dal concetto di management (direzione/gestione).

Un generico processo direzionale è composto d'attività messe in atto da un responsabile (manager/capo) con lo scopo di pianificare-coordinare-controllare le attività operative svolte da altre persone: l'azione del capo consente il raggiungimento di obiettivi che le singole persone operative **non** sarebbero in grado di raggiungere agendo da sole.

Il capo assolve l'incarico affidatogli (raggiungimento degli obiettivi del processo operativo) esercitando quelle che si chiamano funzioni manageriali.

Quando il processo operativo da governare è un progetto le funzioni del capo vanno a costituire ciò che s'intende per project management (direzione/gestione di progetto).

Secondo l'impostazione della norma **[ISO/IEC 12207]** per gestire correttamente i propri progetti, la Direzione dell'Organizzazione Fornitrice deve specializzare la struttura di un generico processo direzionale per la specifica pratica di direzione/gestione di progetto. Il risultato potrebbe essere rappresentato dalla seguente tabella di esempio:

Struttura di un generico processo direzionale	Processo di direzione di uno specifico progetto
<i>Avvio e definizione degli scopi</i>	Definizione del problema Definizione dei requisiti delle Parti interessate Definizione dell'ambito (scopi dell'iniziativa progettuale) Analisi di fattibilità e Stima del prodotto software da consegnare e del processo da eseguire
<i>Pianificazione</i>	Pianificazione del progetto in funzione degli obiettivi fissati in termini di tempi, costi, qualità, Risorse Umane, comunicazione, rischio ed approvvigionamenti
<i>Esecuzione</i>	Predisposizione del contesto di esecuzione: acquisizione delle risorse, attivazione degli impianti, costruzione della squadra ed avviamento del lavoro. Esecuzione delle attività pianificate. Comunicazione delle informazioni di progetto. Consegna dei prodotti intermedi e finali previsti.
<i>Controllo</i>	Esecuzione dei controlli pianificati per monitorare il grado di

	conseguimento degli obiettivi prefissati con eventuali azioni correttive (feedback) verso esecuzione e pianificazione
<i>Chiusura</i>	Rilascio al Committente del prodotto di progetto Produzione dei consuntivi e valutazione dei risultati raggiunti rispetto ai preventivi Rilevazione della soddisfazione delle Parti interessate

Tabella 1 – Struttura dei processi direzionali

Quest'approccio è pienamente compatibile con l'impostazione PMI illustrata successivamente.

Ingegneria del software

Una persona di buon senso, di fronte ad un problema di natura strategica, tende a sviluppare un approccio al problema ovvero un metodo di risoluzione.

Siamo andati a riprendere, sempre sul prezioso Zingarelli, i significati della parola "metodo":

- Via razionale da seguire nell'Operare, Insegnare, Discorrere, Ordinare ...
- Direzione da seguire nell'Operare, Insegnare, Discorrere ...
- Criterio da seguire nell'Operare, Insegnare ...
- Norma da seguire nell'Operare ...

Nella pratica la persona, per superare le proprie difficoltà, costruisce sulla base di evidenze (riproducibili ed accessibili) e di ragionamenti formalmente corretti un percorso (sistema di regole) da seguire ogni volta che si trova a risolvere quel tipo di problema.

Le dimensioni del problema (complessità, vastità, tempestività) possono favorire l'associazione di più persone in un gruppo che intraprende il percorso con maggiori probabilità di successo:

la condivisione di tutto o parte del sistema delle regole genera un metodo nel lavoro di gruppo che integra e valorizza il metodo nel compito individuale.

Da Royce [**ROYCE 1970**] a Boehm [**BOEHM**] fino ad oggi nella comunità dell'ICT si sono affermati numerosi metodi, detti anche impropriamente metodologie.

Grazie ad internet molte metodologie sono accessibili (ed acquistabili) via rete: nell'appendice **B** ne abbiamo sinteticamente descritte alcune tra quelle maggiormente in voga, nell'appendice **I**, invece, abbiamo riportato gli indirizzi di alcuni siti che abbiamo visitato.

Passando dal senso comune alla prestigiosa **IEEE Computer Society** troviamo la definizione dell'Ingegneria del Software come:

l'applicazione di un approccio sistematico, disciplinato, quantificabile ai processi di sviluppo, esercizio, manutenzione del software.

Il corpo delle conoscenze dell'ingegneria del software (SoftWare Engineering Body of Knowledge – SWEBOK) è l'insieme dei saperi e dei saper fare su cui si basa l'ingegnere del software per costruire applicazioni software **utili** ai clienti/utenti.

Nei Paesi industrialmente avanzati è all'ordine del giorno la questione del riconoscimento dell'ingegneria del software come disciplina e come professione: uno degli elementi cardine di questo processo è la piena condivisione del corpo delle conoscenze.

Lungo questa direttiva si muove il progetto SWEBOK della IEEE Computer Society; in particolare gli obiettivi del progetto rispetto al corpo delle conoscenze dell'ingegneria del software (www.swebok.org) sono:

- caratterizzare i contenuti;
- consentire l'accesso;
- promuovere una visione globale della disciplina;
- chiarire posizionamento e contorno dell'ingegneria del software in relazione alle discipline collegate;
- fornire i fondamenti per lo sviluppo di un curriculum e per la certificazione delle competenze individuali.

Il progetto SWEBOK ha prodotto (in versione provvisoria) un documento denominato "Guide to the SWEBOK" nel quale si caratterizza il corpo internamente strutturandolo in dieci aree di conoscenza:

- software requirements,
- software design,
- software construction,
- software testing,
- software maintenance,
- software configuration management,
- software engineering management,
- software engineering process,
- software engineering tools and methods,
- software quality

esternamente definendone le relazioni con sette discipline collegate:

- cognitive sciences and human factors,
- computer engineering,
- computer science,
- management and management science,
- project management,

- system engineering.

SWEBOK, ampiamente riferito nel testo, non rappresenta un'ulteriore metodologia ma descrive una struttura concettuale capace di ospitare le metodologie affermate come elementi del corpo delle conoscenze.

Come già dichiarato, il nostro scopo è di delineare una sintesi pratica tra metodi dell'ingegneria del software e metodi del project management. Per facilitare dopo la costruzione della sintesi è opportuno ora fare alcuni approfondimenti per delimitare le basi di partenza e cioè:

- definire il contesto (le aree di conoscenza delle due discipline),
- individuare gli attori del progetto,
- mettere a fuoco i processi da collegare,
- fissare i criteri per la pianificazione del ciclo di vita del progetto.

Il Project Management ed il PMI

Il Project Management Institute (PMI) è un'organizzazione statunitense che si occupa di standardizzare i processi di Project Management.

Il PMI ha identificato nella gestione del progetto 37 processi componenti. PMI suddivide tali processi in nove aree di conoscenza:

- **Integrazione** (integration)
- **Ambito** (scope)
- **Tempi** (time)
- **Costi** (cost)
- **Qualità** (quality)
- **Risorse Umane** (Human Resource)
- **Comunicazioni** (communication)
- **Rischi** (risk)
- **Approvvigionamenti** (procurement).

La seguente tabella riporta la suddivisione dei processi per area definita dal PMI (PMBOK Guide punto 1.3.2):

<p>Gestione dell'Integrazione</p> <ul style="list-style-type: none"> ● Sviluppo del piano di progetto ● Esecuzione del piano di progetto ● Controllo dei cambi 	<p>Gestione dell'ambito</p> <ul style="list-style-type: none"> ● Inizio ● Pianificazione dell'ambito ● Definizione dell'ambito ● Verifica dell'ambito ● Controllo ai cambi dell'ambito 	<p>Gestione dei tempi</p> <ul style="list-style-type: none"> ● Definizione dei tempi ● Ordinamento delle attività ● Stima della durata ● Sviluppo schedulazione ● Controllo schedulazione
<p>Gestione dei Costi</p> <ul style="list-style-type: none"> ● Pianificazione delle risorse ● Stima dei costi ● Budgeting dei costi ● Controllo dei costi 	<p>Gestione della qualità</p> <ul style="list-style-type: none"> ● Pianificazione della qualità ● Assicurazione della qualità ● Controllo della qualità 	<p>Gestione delle risorse umane</p> <ul style="list-style-type: none"> ● Pianificazione delle risorse umane ● Acquisizione dello staff ● Sviluppo dello staff
<p>Gestione della comunicazione</p> <ul style="list-style-type: none"> ● Pianificazione della comunicazione ● Distribuzione dell'informazione ● Rendiconti di performance ● Chiusura amministrativa 	<p>Gestione dell'approvvigionamento</p> <ul style="list-style-type: none"> ● Pianificazione degli approvvigionamenti ● Pianificazione delle sollecitazioni ● Sollecitazioni ● Selezione della sorgente ● Amministrazione del contratto ● Chiusura del contratto 	<p>Gestione dei rischi</p> <ul style="list-style-type: none"> ● Identificazione del rischio ● Quantificazione del rischio ● Sviluppo risposta al rischio ● Controllo risposta al rischio

Figura 1 Aree di conoscenza del Project Management [Fonte PMI]

Nella edizione 2000 della PMBOK Guide i processi passano da 37 a 39: la gestione dei rischi si arricchisce di due nuovi processi pianificazione del rischio e qualificazione del rischio.

Lo schema ad albero prosegue ed approfondisce il singolo processo attraverso i suoi elementi:

- Input
- Strumenti e tecniche
- Output.

Oltre alla decomposizione area-processo-elementi PMI propone un'ulteriore distribuzione dei processi in cinque gruppi:

- Avvio (Initiating)
- Pianificazione (Planning)
- Esecuzione (Executing)
- Controllo (Controlling)
- Chiusura (Closing)

I processi di **avvio** riconoscono che un progetto o una fase di progetto è iniziato e impegnano l'organizzazione a continuare.

I processi di **pianificazione** elaborano e mantengono aggiornato un programma di lavoro (piano) per indirizzare le azioni della squadra di progetto.

I processi di **esecuzione** coordinano persone e risorse al fine di eseguire le attività definite nel piano.

I processi di **controllo** assicurano che gli obiettivi di progetto siano rispettati attraverso il monitoraggio della esecuzione e la misura dei risultati disponibili. Quando necessario sono intraprese azioni correttive (feedback).

I processi di **chiusura** riconoscono che il progetto è concluso ed effettuano tutte le operazioni fino alla chiusura formale raggiungendo la fine del progetto in maniera ordinata.

PMI definisce (PMBOK Guide punto 3.5) una tabella di corrispondenza attraverso cui è possibile incrociare aree e gruppi.

I gruppi di processo si ripetono in ogni fase del ciclo di vita .

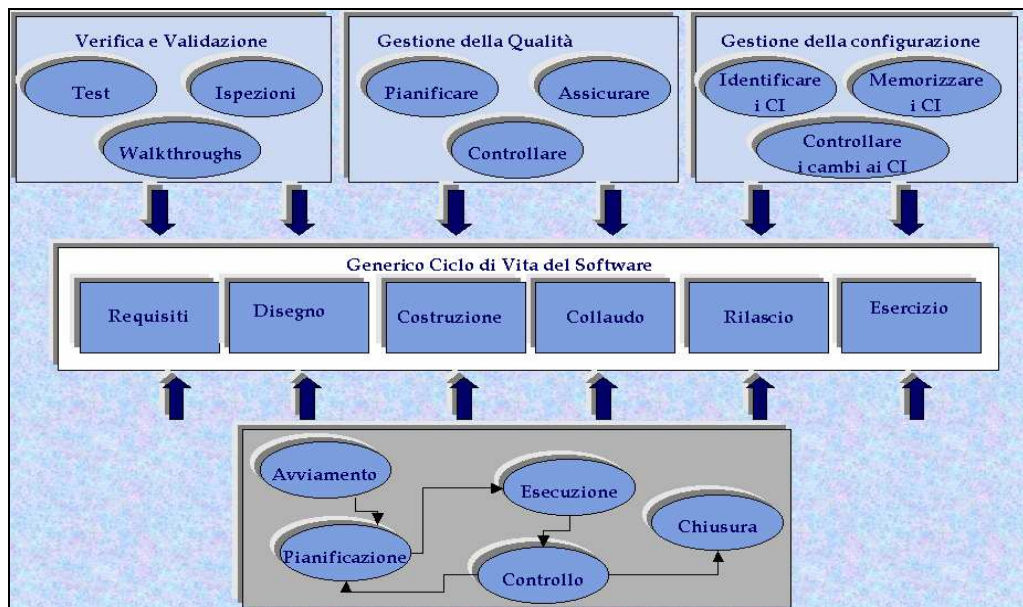


Figura 2 - Relazioni tra processi di Project Management e processi Software

Vi è inoltre una sovrapposizione dei gruppi nelle varie fasi del progetto. Nelle fasi iniziali la maggior parte del lavoro espresso è riconducibile ai processi di avvio e pianificazione mentre le attività di esecuzione, controllo e chiusura sono ridotte al minimo.

Nelle fasi finali la maggior parte del lavoro espresso è riconducibile ai processi di controllo e chiusura.

Tale distribuzione dei gruppi di processi lungo l'arco temporale del progetto è rilevabile anche nel ciclo di vita del software:

in definitiva il Software Project Manager all'inizio del progetto effettuerà attività intensive di avvio e pianificazione, mentre con il chiudersi delle fasi e soprattutto in costruzione effettuerà attività di esecuzione, controllo e chiusura.

Caso di studio

Ripercorriamo la fase d'ingaggio. La definizione concettuale inizia nell'impresa cliente. L'applicazione attuale di fatturazione va in crisi quando il numero delle fatture da stampare supera la soglia di centomila.

XYZ definisce la situazione attuale, i requisiti futuri e sollecita i Fornitori a formulare una soluzione.

I Fornitori definiscono un progetto di massima ed una stima a sostegno dell'offerta.

In particolare il fornitore ABC assegna lo studio a ROSSI che sviluppa lo studio attraverso i seguenti passi

Analisi dei requisiti - ROSSI riprende ed approfondisce il lavoro del Committente...

Scelta del modello di ciclo di vita - ROSSI sceglie un modello di ciclo di vita del prodotto basato su RUP.

Proposta di soluzione - ROSSI definisce le prestazioni da offrire...

Stima iniziale - La stima per determinare l'impegno delle Risorse Umane è stata ricavata utilizzando la stima statistica dei CSI relativa alla previsione di LOC, in funzione delle quali è stato ipotizzato il numero delle risorse e la loro distribuzione nel tempo.

La definizione si compie con la formulazione dell'offerta da parte dei Fornitori. Le Parti in gioco in fase di definizione hanno creato le premesse per un accordo contrattuale.

Overview del progetto dal punto di vista del fornitore

Ambito

In **avvio** sulla base dell'offerta-progetto (studio di fattibilità) il fornitore si aggiudica il contratto.

La direzione di ABC assegna a Rossi la gestione del progetto (Project Charter).

In fase di **pianificazione**, la descrizione del lavoro da svolgere è ricavata dal contratto, in particolare per quanto riguarda i requisiti ad alto livello del prodotto.

Cosa fare:

1. *Fatturazione convergente*
2. *Possibilità di fatturazione immediata...*

Cosa non fare:

Tutte le parti di interfaccia, cioè il software di formattazione delle diverse tipologie dei dati di traffico entranti.

requisiti non funzionali:

1. *Facilmente modificabile*
2. *Rendiconti telefonici flessibili*
3. *Integrazione web*

processi di sviluppo software contrattualmente previsti:

1. *Requisiti*
2. *Progettazione*
3. *Realizzazione*
4. *Collaudo*
5. *Installazione*

La pianificazione dell'ambito produce in conclusione una WBS basata sul modello di ciclo di vita scelto.

In fase di **esecuzione** il PM svolge tutte le verifiche interne prima di sottoporre le consegne al cliente per l'accettazione dei prodotti di fase.

Mentre in **controllo** il PM gestisce le richieste di modifiche all'ambito in termini di analisi, valutazione e decisione (se inglobare o no i cambi richiesti).

Integrazione

Rossi nelle **pianificazioni** ha deciso di integrare la pianificazione esecutiva in due passi:

1. Pianificazione esecutiva di base orientata al processo
2. Pianificazione esecutiva di dettaglio orientata al prodotto.

A fine integrazione tutti gli aspetti della pianificazione di progetto (Ambito, tempi, risorse, costi, rischi, qualità, ecc...) sono integrati nel piano di progetto.

Il PM in fase di **controllo** effettua delle valutazioni su tutti i controlli relativi ad ambito, tempi, costi, cambi, qualità, rischi, ecc.. utilizzando un cruscotto di indicatori.

Tempi

In **pianificazione** la definizione delle attività e lo sviluppo della schedulazione hanno recepito il vincolo della durata del progetto fissata per contratto in un anno solare. Considerando la distribuzione geografica della squadra si è utilizzata una tecnologia web based di supporto alla gestione dei tempi.

Il **controllo** dei tempi è stato sviluppato sulla base del contratto che prevedeva penalità economiche per ogni mese di ritardo. La tecnica dell'EARNED VALUE è stata utilizzata anche per il controllo dei costi.

Costi

In **pianificazione** il PM ha recepito la stima iniziale che ha consentito l'aggiudicazione del contratto. Su questa base è stato sviluppato un piano dei costi di dettaglio ed un piano delle risorse

In **esecuzione** il PM ha deciso di raccogliere le voci di spesa principali quali trasferte, risorse umane, facilities ecc.. con frequenza quindicinale anziché mensile, ciò per facilitare il controllo.

I dati raccolti poi sono stati acquisiti in un foglio di calcolo che mediante formule (tipiche dell'earned value) consentiva di effettuare controlli e previsioni.

In **controllo** il PM ha potuto scegliere liberamente la tecnica di controllo più adatta giacché ABC non aveva a disposizione un sistema avanzato per la gestione dei costi.

Così si è deciso di adottare la tecnica dell'Earned Value, che consente di controllare i costi ed i tempi in maniera efficace.

Qualità

In **pianificazione** la qualità è stata una delle aree che ha ricevuto maggiore attenzione. Sin dall'inizio si è capito che uno dei fattori di successo del progetto sarebbe stato consegnare un prodotto con un numero di difettosità molto basso.

Per gestire efficacemente i difetti si è deciso di utilizzare un sistema informatico. In conclusione sono stati emessi SPQP e SVVP.

In **esecuzione** il sistema informatico di gestione dei difetti ha consentito raccolta e catalogazione dei difetti riscontrati nelle varie fasi del ciclo di vita del software con accessi via WEB.

In **controllo** il PM e le Persone da lui delegate hanno monitorato il trend dei difetti.

Questa pratica ha consentito di intervenire in più di un'occasione per apportare i correttivi necessari alla sistemazione della qualità del prodotto.

Risorse Umane

In **pianificazione** il PM ha previsto una struttura organizzativa di progetto basata sulla divisione in gruppi di lavoro per area di prodotto: ciascun gruppo è stato assegnato ad un Team Manager direttamente alle dipendenze del PM. In funzione dell'allocazione prevista è stato formulato un piano di sviluppo personale.

Durante l'**esecuzione** del progetto le risorse umane assegnate sono state coinvolte sin dall'inizio con l'obiettivo di sviluppare il senso di appartenenza. La formazione prevista è stata effettivamente fruita sia in modalità tradizionale sia attraverso l'approccio training on the job.

In **controllo** il PM ha seguito lo sviluppo personale di ogni risorsa valutando i risultati della formazione e l'evoluzione di skill e prestazione.

Riunioni periodiche di team Building hanno contribuito a costruire un team di successo.

Comunicazioni

In fase di **pianificazione** è stata prevista una squadra di sviluppo distribuita geograficamente:

Un gruppo presso il cliente, un gruppo presso il laboratorio con sotto-gruppi dislocati nella stesso edificio ma su piani differenti, il fornitore ubicato in altra zona geografica.

Tale distribuzione ha imposto nel piano di comunicazione una valutazione oculata dei mezzi. Sono state scelte in prevalenza tecnologie web based: tool per teleconferenze quindicinali - Posta elettronica per le comunicazioni ordinarie - forum on line per le discussioni. Alla fine è stato emesso un piano di comunicazione.

Durante l'**esecuzione** l'informazione è stata distribuita sulla base del piano di comunicazione.

In fase di **controllo** è stata soddisfatta una richiesta del cliente che aveva esigenza di uno stato di avanzamento tecnico-economico mensilmente.

Rischi

In **pianificazione** sono stati identificati e previsti rischi di due categorie High level (alto livello - Gestiti direttamente dal Senior Management) sia i rischi a livello di Progetto (Function Level) di competenza del PM. Così come per i difetti è stato previsto un sistema informatico di gestione dei rischi.

In **esecuzione** i dati sui rischi (sintomi, nuovi rischi, ecc...) sono stati raccolti nell'archivio dei rischi.

Il **controllo** sui rischi è stato esercitato utilizzando l'archivio.

Approvvigionamenti

Si è ricorso ad una società esterna solo per lo sviluppo di componenti software in ambiente Mainframe.

La scelta di utilizzare una società esterna è stata motivata dalla necessità di ridurre i rischi derivanti dai tempi compressi imposti al progetto.

Per definire/pianificare l'acquisto di una parte software è stato emesso un documento di Statement Of Work (SOW).

Dei **controlli** periodici in casa del fornitore hanno contribuito a rilevare i progressi.

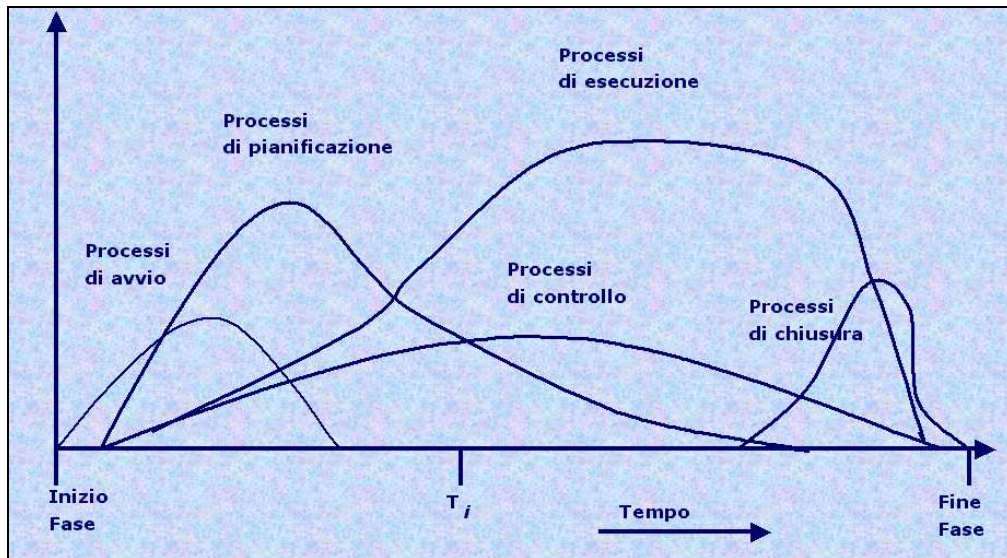


Figura 3 – Distribuzione dei gruppi di processo di PM per una generica fase [Fonte PMI]

Le aree di conoscenza dell'ingegneria del software

Facendo sempre riferimento al documento denominato "Guide to the SWEBOK" riportiamo la struttura interna del corpo di conoscenze dell'Ingegneria del software:

Area di conoscenza	Sub-aree
Requisiti software - software requirements	<ul style="list-style-type: none"> • Ingegneria dei requisiti • Raccolta dei ... • Analisi ... • Specificazione ... • Validazione ... • Gestione ...
Disegno del software - software design	<ul style="list-style-type: none"> • Basi del disegno del Software • Argomenti chiave del disegno del software • Struttura ed architettura del software • Analisi e valutazione della qualità del disegno • Notazioni per il disegno • Strategie e metodi per il disegno
Costruzione del software - software construction	<ul style="list-style-type: none"> • Riduzione della complessità • Anticipazione delle possibili evoluzioni • Strutturazione per (successive fasi di) test e validazione • Adozione di standard esterni (al progetto)
Test del software - software testing	<ul style="list-style-type: none"> • Definizione e concetti di base del Testing • Livelli di Test • Tecniche di Test • Misure relative al Test • Gestione del processo di test
Manutenzione del software - software	<ul style="list-style-type: none"> • Concetti di base • Il processo di manutenzione

maintenance	<ul style="list-style-type: none"> • Argomenti chiave nella manutenzione del software • Tecniche di manutenzione
Gestione della configurazione del software - software configuration management - SCM	<ul style="list-style-type: none"> • Gestione del processo SCM • Identificazione della configurazione Software • Controllo della configurazione Software • Stato di avanzamento della configurazione Software • Gestione dei rilasci e delle consegne del Software
Il processo di direzione nell'ingegneria del software - software engineering management	<ul style="list-style-type: none"> • Gestione dell'organizzazione -il ruolo dell'Alta Direzione • Gestione del Processo-progetto • Gestione delle misure per l'ingegneria del software
Il processo di produzione nell'ingegneria del software - software engineering process	<ul style="list-style-type: none"> • Ambito del processo di produzione nell'ingegneria del software • Infrastruttura del processo • Misura del processo • Definizione del processo • Analisi della qualità del processo • Implementazione e modifica del processo
Strumenti e metodi dell'ingegneria del software- software engineering tools and methods	<ul style="list-style-type: none"> • Strumenti software (per requisiti, disegno, costruzione, test, manutenzione...) • Metodi per lo sviluppo del software
Qualità del software - software quality	<ul style="list-style-type: none"> • Concetti relativi alla qualità del prodotto software (Software Quality Assurance - SQA e Verification and Validation - V&V) • Obiettivi e pianificazione dei processi di SQA e V&V • Attività e tecniche per i processi di SQA e V&V • Misure per i processi di SQA e V&V

Tabella 2 – Corpo di conoscenza dell'ingegneria del software [SWEBOK]

L'approfondimento delle singole aree di conoscenza esula dai nostri scopi: è opportuno, però, fare almeno due considerazioni.

La prima considerazione riguarda la relazione tra le aree di conoscenza di un ingegnere del software ed i processi di un'Organizzazione specifica per il software: le prime cinque aree di conoscenza ricadono nei processi primari, le altre nei processi organizzativi e di supporto.

La seconda riguarda le possibili sovrapposizioni tra scopi e contenuti dell'**area di conoscenza** "Il processo di direzione nell'ingegneria del software - software engineering management" ed il corpo delle conoscenze della disciplina Project Management: su questo punto il nostro riferimento sarà il PMI e la relativa PMBOK Guide.

Riteniamo interessante approfondire la prima considerazione e proponiamo al lettore la seguente tabella per individuare le correlazioni esistenti tra le aree di conoscenza di SWEBOK, i processi della norma ISO/IEC 12207:1995 e le prescrizioni della UNI EN ISO 9001:2000:

Are di conoscenza di SWEBOK	processi della norma ISO/IEC 12207:1995	prescrizioni della UNI EN ISO 9001:2000
Disegno del software	5.3.5 progettazione architettonica del sw 5.3.6 progettazione di dettaglio	7.3 progettazione e sviluppo 7.3.3 elementi in uscita dalla progettazione e sviluppo

	del sw 6.6 processo di riesame congiunto	7.3.4 riesame di progettazione e sviluppo
Costruzione del software	5.3.7 codifica e test unitario del sw	7.3 progettazione e sviluppo 7.3.3 elementi in uscita dalla progettazione e sviluppo
Test del software	5.3.8 integrazione del sw 5.3.9 test di qualificazione del sw	7.3.6 validazione di progettazione e sviluppo 8.2.4 monitoraggio e misurazione dei prodotti
Manutenzione del software	5.5 processo di manutenzione	7.5 produzione ed erogazione di servizi 7.3.7 tenuta sotto controllo delle modifiche della progettazione e sviluppo
Gestione della configurazione del software	6.1 processo di gestione della configurazione	7.5.3 identificazione e rintracciabilità
Il processo di direzione nell'ingegneria del software	7.1 processo direzionale 5.2.4 pianificazione	5 responsabilità della direzione 6.2 risorse umane 7.3.1 pianificazione della progettazione e dello sviluppo 7.4 approvvigionamento
Il processo di produzione nell'ingegneria del software	5.3.1 implementazione del processo 7.3 processo di miglioramento	4.1 requisiti generali 7.1 pianificazione della realizzazione del prodotto 8.5 miglioramento
Strumenti e metodi dell'ingegneria del software	7.2 processo di infrastrutture	6.3 infrastrutture 7.6 tenuta sotto controllo dei dispositivi di monitoraggio e misurazione
Qualità del software	6.3 processo di assicurazione qualità	8 misurazioni, analisi e miglioramento

Tabella 3 – Relazioni tra i processi di ingegneria del software e le norme ISO

Questa tabella può rappresentare anche una prima interpretazione della Vision 2000 per il settore del software: in attesa della nuova ISO9000-3 il lettore troverà un approfondimento del tema nel Quaderno n°11 pubblicato dal Sottocomitato per la Qualità del Software di AICQ-CI [AICQCI 2001].

Gli attori del software project management

Nei ruoli di soggetti gestori gli attori del Software Project Management sono due: la Direzione dell'Organizzazione ed il PM cui è stata assegnata la responsabilità del progetto.

Nella seguente tabella sono stati individuati, senza riferimenti ai tempi, i processi fondamentali entro cui Direzione e Capo-progetto giocano i rispettivi ruoli definendo il sistema relazionale di base:

Processo direzionale	Il ruolo della Direzione	Il ruolo del Capo-progetto
<i>Pianificazione</i>	Strategico:	Operativo:

<i>e controllo</i>	definisce gli obiettivi di mercato e rispetto ad essi pianifica e controlla la prestazione dell'Organizzazione	recepisce in avvio gli obiettivi del progetto e rispetto ad essi pianifica e controlla la prestazione del gruppo di lavoro
<i>Governo della struttura organizzativa</i>	<u>dell'azienda:</u> distribuisce i centri di responsabilità nella rete aziendale allineando il peso della struttura con gli obiettivi prefissati	<u>del gruppo di lavoro</u> assegna i compiti in funzione delle competenze allineando la formazione della squadra con gli obiettivi prefissati
<i>Governo delle competenze</i>	<u>dell'azienda</u> definisce quali e quante competenze debbano sostenere la capacità di processo dell'Organizzazione in relazione al mercato d'interesse	<u>del gruppo di lavoro</u> definisce quali e quante competenze debbano sostenere la capacità di processo della squadra in relazione al prodotto ed al cliente del progetto
<i>Pianificazione</i>	<u>della realizzazione del prodotto aziendale</u> definisce i criteri per configurare il processo di produzione in funzione del tipo di prodotto, del tipo di cliente e del livello di capacità espresso dall'Organizzazione	<u>dello sviluppo di uno specifico progetto</u> personalizza l'intervento progettuale pianificando il lavoro per produrre la qualità attesa dal cliente e per impiegare al meglio la capacità espressa dal gruppo di lavoro
<i>Valutazione della opportunità</i>	<u>assegnazione dell'incarico al Capo-progetto</u> Decide d'iniziare il progetto fissando gli obiettivi	<u>Analisi di fattibilità e stima</u> recepisce gli obiettivi del progetto, ne valuta la fattibilità e riporta i risultati della stima alla Direzione concludendo la fase d'avvio
<i>Valutazione della prestazione</i>	<u>Verifica ed approvazione del bilancio</u> Decide di terminare il progetto perfezionando la fase di chiusura	<u>Proposta di Bilancio</u> raccolge e sistema tutti i risultati del progetto ed esegue tutte le chiusure

Tabella 4 – Principali ruoli della direzione e del Project Manager

Il discorso fatto sugli attori del processo di SPM può essere utilmente esteso alle parti interessate al progetto:

- **Il Cliente/Utente**
- **Il Cliente/Committente**
- **I Fornitori**
- **La Direzione-Ente Qualità**
- **Il Personale**

Il **cliente/utente** (esterno ed interno) è la parte, designata dal cliente/committente, che riceve ed usa il prodotto del progetto. Nel caso di progetti d'investimento il cliente/utente è, almeno inizialmente, interno (es.: la funzione di marketing).

Il cliente/utente si aspetta che il prodotto realizzato sia rispondente all'uso che ne deve fare, rispondenza prefigurata dai requisiti espressi ed impliciti.

Il **cliente/committente** è l'ente che sottoscrive il contratto e che finanzia il progetto. Nel caso di progetti d'investimento il cliente/committente è l'Alta Direzione.

Il cliente/committente si aspetta che il prodotto sia realizzato rispettando il contratto in tutti i suoi aspetti (tecnici, economici e legali).

I **fornitori**, (interni ed esterni) sono coloro che forniscono prodotti (HW e SW) e servizi utilizzati nel processo di progettazione e sviluppo.

Il fornitore si aspetta d'avere, all'inizio, requisiti coerenti, stabili e completi ed alla fine un collaudo tempestivo ed efficace: il processo di fornitura si gioverà di una stretta collaborazione con l'Organizzazione sin dalle primissime fasi del progetto. I fornitori si aspettano che la collaborazione possa trasformarsi in una partnership da cui trarre stabilmente beneficio.

La **Direzione** dell'Organizzazione è stata già trattata come uno dei due attori primari del processo di SPM: si può aggiungere che la Direzione si aspetta dal progetto un margine adeguato all'impiego delle risorse. Espressa così l'affermazione sembra riduttiva e generica: un margine soddisfacente conseguito contro la soddisfazione delle altre Parti interessate al progetto è una mina vagante anziché un buon risultato.

È importante notare che questo è un concetto fondamentale ed è stato qui appena sfiorato.

La metodologia **BALANCED SCORECARD** di Kaplan e Norton [DeMarco 1999] considera per l'Organizzazione competitiva una prospettiva aziendale risultante da quattro componenti:

1. prospettiva economico/finanziaria basata sull'andamento di costi e ricavi,
2. prospettiva del cliente basata sulla soddisfazione dei clienti e sull'andamento della quota di mercato,
3. prospettiva interna basata sulla capacità di far evolvere il processo di produzione,
4. prospettiva di innovazione ed apprendimento basata sulle persone, i sistemi e le procedure organizzative.

Per evitare il rischio dello sbilanciamento della prospettiva aziendale, la Direzione sorveglia l'andamento del processo aziendale attraverso i sistemi di gestione di cui si è dotata: controllo di gestione, sistema di gestione della qualità, gestione delle Risorse Umane.

Il PMI traduce il problema del bilanciamento dall'Organizzazione al Progetto con l'efficace immagine del triangolo tempi - costi - qualità.

Per governare il bilanciamento, in particolare per quanto riguarda il sistema di gestione della qualità la Direzione generalmente delega all'Ente Qualità dell'Organizzazione le seguenti responsabilità:

- Assistenza alla Direzione per pianificazione e controllo strategico,
- Assistenza al PM per pianificazione e controllo operativo,
- processo di taratura delle metodologie in uso,
- processo di monitoraggio (misurazione, analisi e miglioramento).

Nell'assistenza alla Direzione l'ente qualità recepisce le indicazioni di carattere generale sull'aggiornamento (della definizione) del processo aziendale, analizza la compatibilità delle innovazioni (tecniche, standard, modelli) rispetto al sistema di gestione della qualità ed eventualmente ne cura l'aggiustamento.

Nell'assistenza al PM l'ente qualità all'inizio cerca di facilitare l'applicazione del sistema di gestione della qualità allo specifico progetto (scelta del modello, adattamento del modello al progetto) ed in corso d'opera, fino alla fine, recepisce gli esiti dell'applicazione (misurazioni, analisi dei dati e miglioramento).

Gli interessi del **Personale** dell'Organizzazione vanno analizzati e curati da due distinti punti di vista: dapprima l'Organizzazione deve dare attenzione al singolo Specialista software impegnato nel progetto, poi al gruppo di lavoro (squadra di progetto).

Lo Specialista

Il singolo Specialista software impegnato nel progetto è colui che esegue il compito individuale assegnato dal PM.

Lo Specialista si aspetta un'assegnazione chiara, obiettivi realistici, risorse adeguate, riconoscimento e valorizzazione della propria professionalità.

Lo Specialista deve avere competenze tecniche metodologiche ed operative per portare a termine il compito assegnato attraverso il processo di sviluppo personale (PSP) [Humphrey 1997].

Naturalmente le aree di conoscenza-competenza individuali sono relative alle tecnologie dell'informazione.

La base è costituita dalle prime cinque aree individuate da SWEBOK:

- competenze applicative per i requisiti software;
- competenze sul disegno architettonico;
- competenze sullo sviluppo di programmi;
- competenze sui processi di prove, controlli e collaudi;
- competenze sulla manutenzione.

È opportuno che Lo Specialista abbia conoscenze e competenze anche in altre aree che generalmente fanno capo alle funzioni di supporto dell'Organizzazione

- operation, tecniche di conduzione operativa di sistemi e reti;
- strumenti e metodi dell'ingegneria del SW;
- SCM, strumenti e metodi per l'identificazione e la rintracciabilità del prodotto SW;
- strumenti e metodi per misurazione analisi e miglioramento del software.

Il Progettista non nasce con tutte queste competenze, ma le acquisisce lungo l'arco di una carriera che dura anni; in molte Organizzazioni specifiche per il SW è prassi tracciare il percorso professionale attraverso la sequenza dei ruoli interpretati (figure specialistiche) secondo sviluppi di carriera che possono seguire linee diverse quali:

- Manageriale
- Tecnica (Technical o Professional).

La linea Manageriale può prevedere i seguenti passaggi:

1. Programmatore;
2. Progettista;
3. Analista;
4. Team Manager;
5. Project Manager;
6. Senior Manager (Dirigente di funzione e superiore, es. Direttore Tecnico).

I primi tre livelli sono tipicamente tecnico applicativi e consentono l'acquisizione nel tempo delle necessarie competenze nei processi di sviluppo per poi accedere (ultimi tre livelli) alle competenze e responsabilità di tipo organizzativo.

La linea Tecnica consente alla Persona una diversa evoluzione delle competenze verso un'alta specializzazione tecnica di ruolo (seniority):

- Programmatore;
- Progettista;
- Analista;
- Sistemista ed amministratore di reti;
- Amministratore di data base.

Il gruppo di progetto

Il gruppo di progetto è l'insieme delle Risorse Umane dal cui sforzo congiunto ed integrato deriverà il conseguimento degli obiettivi del progetto.

Il gruppo di progetto è un'entità unica costituita dai progettisti selezionati in base alle disponibilità dell'Organizzazione ed alle caratteristiche specifiche del progetto.

Il gruppo di progetto si aspetta di diventare una squadra (team) con:

- una meta unica e raggiungibile;
- una definizione chiara dei ruoli e delle relazioni tra ruoli;
- un'efficace circolazione di tecnologie;
- un'efficace comunicazione interna;
- un'efficace comunicazione esterna;
- un adeguato riconoscimento della prestazione comune.

La distinzione netta tra lo Specialista che sviluppa ed il PM che governa è una semplificazione necessaria per articolare il discorso.

La realtà aziendale è più complessa ed i confini tra le responsabilità sono più incerti: in qualche caso sarebbe giusto parlare di una squadra di sviluppo che risponde ad una squadra di conduzione nata dal processo di delega di parte dei compiti del PM.

Il Project Manager

Il Project Manager ha un ruolo determinante nella gestione del progetto software. Il PM può assumere vari titoli: Project Leader, Coordinatore di Progetto, Program Manager, Direttore Lavori, Capo commessa, Capo progetto, Capo area, Team Leader ecc...

La diversificazione dipende dal livello formale di autorità assegnata nell'Organizzazione che esegue il progetto e dal settore di appartenenza (Costruzioni, Farmaceutico, Software, ecc...).

Il PM è il responsabile dei risultati di un progetto software e questo è uno dei motivi per cui la scelta del PM da parte della Direzione è una decisione difficile.

Per interpretare un ruolo tanto importante il PM deve avere le seguenti caratteristiche professionali:

- Competenza tecnico applicativa – gestione dei requisiti del prodotto lungo tutto il ciclo di vita (dominio del problema e tecnologia);
- Competenza metodologico gestionale – padronanza delle aree d'interesse dell'ingegneria del software (per esempio: tecniche di disegno dei sistemi informativi) e delle aree d'interesse del project management (per esempio: conduzione dei progetti);

- Capacità organizzative per la gestione delle Risorse Umane assegnate al progetto – leadership riconosciuta per coordinare, motivare e guidare la squadra
- Capacità organizzative per la gestione di risorse materiali e servizi assegnati – senso pratico per acquisire, predisporre ed utilizzare al meglio le risorse;
- Capacità comunicative per definire e praticare i canali di scambio informazioni con le Parti interessate al progetto (direzione, funzioni aziendali, clienti ed utenti dei sistemi, fornitori);
- Capacità di prevenire e gestire i conflitti sia quelli interni alla squadra sia quelli esterni tra la squadra e le parti interessate;
- Capacità di prevenire e gestire i problemi – intuito nella individuazione, approccio analitico nella definizione, coraggio nel decidere sulla soluzione, senso della misura negli eventuali processi di escalation;
- Capacità di mediare su tempi, costi e qualità nell’ambito di progetto in maniera soddisfacente per le parti interessate;
- Buona cultura generale;
- Integrità morale, stabilità emotiva ed indipendenza di giudizio per interpretare adeguatamente, durante tutto il governo di progetto, il sistema di valori alla base della comunità di appartenenza;
- E per finire entusiasmo, energia e senso dell’umorismo.

Gli obiettivi invece che persegue un PM sono:

- Rispettare gli obiettivi di tempi/costi/qualità nell’ambito concordato del progetto;
- Conseguire la soddisfazione delle parti interessate;
- Motivare la squadra ed i fornitori diretti;
- Ottenere collaborazione e supporto degli altri attori (Linee funzionali, Cliente, ecc ..) coinvolti nel progetto;
- Incentivare e sviluppare la professionalità dei componenti della squadra;
- Dare visibilità del lavoro svolto e dei risultati ottenuti.

Le responsabilità di un PM si collocano all’interno dei macroprocessi tipici di progetto:

- Definizione
- Stabilire obiettivi di progetto e criteri di valutazione, identificare requisiti delle Parti interessate, identificare una possibile struttura del prodotto e del processo
- Pianificazione
- Programmare lo sviluppo in termini di Tempi, Risorse Umane e Materiali, Rischi, Costi, Qualità,
- Esecuzione

- Assegnare i compiti, monitorare l'esecuzione delle attività previste nel Piano di progetto, integrare i risultati, Distribuire dell'informazione in modo appropriato e tempestivo (Project Status, Memorandum, Agenda Riunioni, ecc...)
- Controllo
- Sorvegliare le caratteristiche del processo di sviluppo e quelle dei prodotti sviluppati in termini di Tempi, Risorse, Rischi, Costi, Qualità, Ambito; intraprendere eventuali azioni correttive
- Chiusura
- Raccogliere i dati di base, chiudere le attività in corso, formulare un rendiconto, esprimere le proprie valutazioni.

Le descrizioni date sono generiche e riconoscibili in tutti i settori industriali. Ciò che contraddistingue il Software Project Manager è la competenza specifica nell'Ingegneria del software.

Nel settore del software il successo del progetto dipende prevalentemente dalle prestazioni delle Risorse Umane impiegate, dunque le capacità più importanti che deve possedere un Software Project Manager sono di organizzare, comunicare, mediare: sopra tutte svetta la capacità di leadership per motivare e guidare una squadra ricca di forti individualità professionali e di creatività.

La considerazione svolta porta ad osservare che le Organizzazioni specifiche per il software trovano naturale orientarsi verso strutture organizzative piatte capaci di facilitare decentramento e partecipazione delle Persone.

Relazioni tra strutture organizzative e project management

La struttura organizzativa dell'azienda appaltatrice influenza l'impostazione dei progetti.

In una struttura funzionale l'azienda è organizzata per funzioni (fig.4), ogni funzione ha uno scopo da perseguire e possiede competenze omogenee.

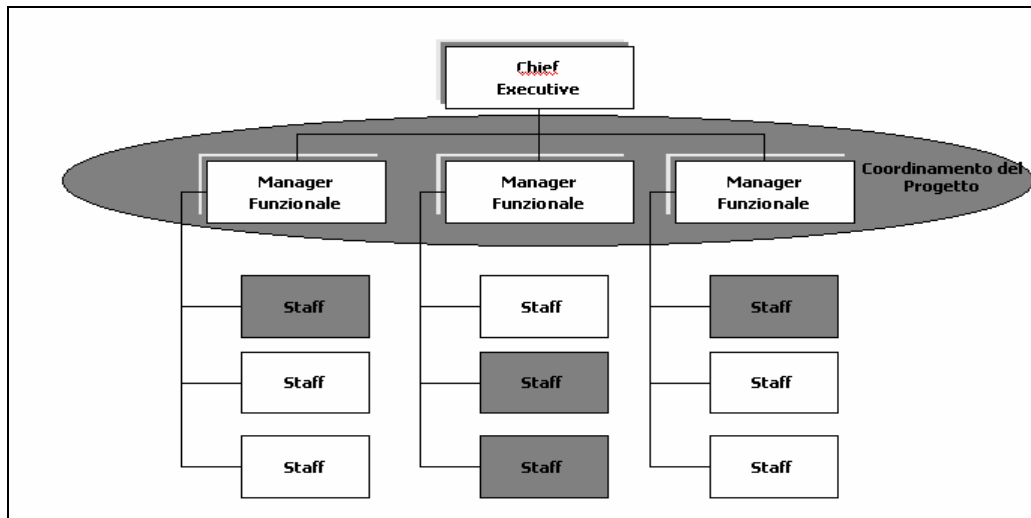


Figura 4 – Organizzazione di progetto basata su Funzioni [Fonte PMI]

All'interno della funzione vi è una gerarchia di comando.

I progetti attraversano una o più funzioni, ed ogni funzione esegue la parte di competenza svolgendo tutte le operazioni nel rispetto della linea gerarchica.

I vantaggi di un'organizzazione siffatta si traducono nei seguenti punti:

- Nessuna duplicazione di attività (le funzioni chiave sono svolte nell'ambito della funzione);
- Preciso percorso di carriera per le persone nell'ambito della linea gerarchica funzionale;
- Massima attenzione sulle funzioni chiave;
- Chiara definizione dei ruoli e delle responsabilità;
- Massima specializzazione delle risorse.

Mentre i principali svantaggi si riassumono nei seguenti punti:

- I diversi obiettivi delle funzioni portano a generare conflitti;
- Non è definita una responsabilità generale per il progetto;
- Il coordinamento è più difficile per l'attenzione posta dalle risorse sugli obiettivi funzionali invece che sul progetto.

In un'organizzazione progettata (Fig.5) al Project Manager sono assegnate tutte le risorse necessarie per il completamento del progetto.

Il PM ha l'intera responsabilità del progetto e delle risorse assegnate.

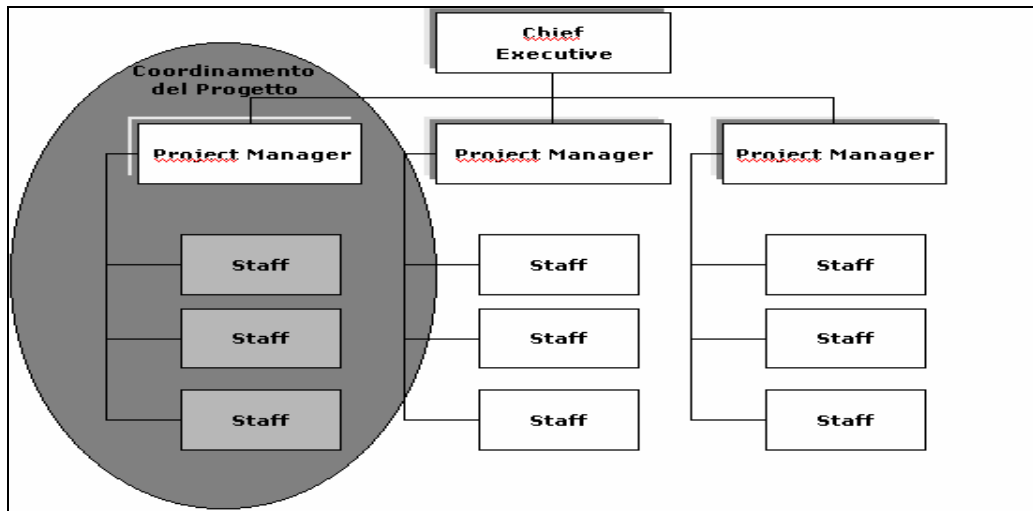


Figura 5 – Organizzazione basata su progetti [Fonte PMI]

Inoltre vi è una maggiore comunicazione (informale) fra il Project Manager e i componenti del gruppo.

Modello progettato e modello funzionale si scambiano punti di forza e punti di debolezza.

Esistono anche le organizzazioni a matrice le quali si suddividono ulteriormente in:

- Matrice debole;
- Matrice bilanciata;
- Matrice forte.

In queste forme di organizzazione il ruolo e l'autorità del PM cresce quando ci si muove da una forma debole ad una forte così come la percentuale di tempo assegnata allo staff.

In tabella 8 sono riepilogate le principali caratteristiche delle organizzazioni in termini di autorità e ruolo del PM, nonché la percentuale di utilizzo dello staff e l'impiego di un ufficio di progetto.

Caratteristiche del Progetto \ Tipo di Organizzazione	Funzionale	Matrice			Progettizzata
		Debole	Bilanciata	Forte	
Autorità del Project Manager	Poca o niente	Limitata	Da bassa a moderata	Da moderata a alta	Da alta a totale
% di personale dell'organizzazione assegnata a tempo pieno ad attività di progetto	Nessuna	0-25%	15-60%	50-95%	85-100%
Ruolo del Project Manager	Part-time	Part-time	Full-time	Full-time	Full-time
Titolo comune per il ruolo di PM	Coordinatore - Project Leader	Coordinatore - Project Leader	PM - Project Officer	PM - Program Manager	PM - Program Manager
Staff amministrativo di Project Management	Part-time	Part-time	Part-time	Full-time	Full-time

Figura 6 – Relazione tra le caratteristiche di progetto e il tipo di organizzazione [Fonte PMI]

Processi software

Prima di affrontare la questione relativa alla definizione dei processi SW riportiamo la definizione generica di *processo*:

Insieme di attività correlate o interagenti che trasformano elementi in entrata in elementi in uscita.

Note

Gli elementi in entrata in un processo provengono generalmente dagli elementi in uscita da altri processi;

I processi in un'Organizzazione sono di regola pianificati ed eseguiti in condizioni controllate al fine di aggiungere valore.

UNI EN ISO 9000:2000

Nel paragrafo dedicato al PMI abbiamo riportato i processi tipici della direzione/gestione del progetto e la loro strutturazione nei cinque gruppi (macroprocessi): Avvio, Pianificazione, Esecuzione, Controllo, Chiusura. I processi SW sono invece i processi operativi orientati alla realizzazione del prodotto.

Cerchiamo ora di approfondire l'importanza della definizione dei processi SW all'interno di un'Organizzazione.

Nel paragrafo dedicato all'Ingegneria del Software abbiamo accennato alla larga disponibilità di metodologie SW e ciascuna presenta una propria definizione dei processi SW.

Di fronte a tanta abbondanza d'informazione la questione della scelta dell'approccio progettuale sembra risolta e forse lo è per il singolo ingegnere del SW che lavora in autonomia.

Certamente non è risolta per un'Organizzazione:
 lo ricorda la norma (prescrizioni della **UNI EN ISO 9000:2000** punto 4.1)
 ma lo impone il mercato quando chiede una predefinita capacità di risposta
 a problemi sempre nuovi e sempre più incerti.
 Un'ipotesi operativa per giungere alla definizione dei processi SW in
 un'Organizzazione potrebbe essere sintetizzabile nella sequenza

- 1. Gestione strategica d'impresa,**
- 2. Governo della struttura organizzativa,**
- 3. Acquisizione della metodologia SW.**

In un'Organizzazione attenta all'innovazione della propria tecnologia la *metodologia* aziendale (strategia d'impresa) necessariamente precede ed ospita la *metodologia* SW (strategia di realizzazione del prodotto): siamo entrati nelle responsabilità dell'Alta Direzione.

Nella norma ISO/IEC 12207:1995 (7.1) troviamo definita la struttura generica di un processo direzionale: lo abbiamo già applicato per rappresentare la gestione di un progetto ed è ancora applicabile alla gestione strategica d'impresa.

Struttura di un generico processo direzionale	gestione strategica d'impresa
<i>Avvio e definizione degli scopi</i>	<ul style="list-style-type: none"> • Definizione del mercato di sbocco • Definizione dei requisiti delle Parti interessate • Definizione dello scopo dell'iniziativa imprenditoriale • Analisi di fattibilità e Stima: segmentazione del mercato, analisi della concorrenza, analisi delle fonti d'approvvigionamento
<i>Pianificazione</i>	<ul style="list-style-type: none"> • Business Plan • Caratterizzazione del marketing mix • Sviluppo degli obiettivi di vendita – piano commerciale • Sviluppo degli obiettivi di produzione – piano di produzione • Definizione della configurazione dei costi – piano d'acquisizione delle risorse • Definizione del processo – piano di sviluppo organizzativo
<i>Esecuzione</i>	<ul style="list-style-type: none"> • Avviamento del lavoro (processi primari e processi di supporto) • Acquisizione e distribuzione delle risorse • Assegnazione di obiettivi e responsabilità • Monitoraggio dell'andamento aziendale attraverso i sistemi software di gestione
<i>Controllo riesami e valutazioni</i>	<ul style="list-style-type: none"> • Verifiche periodiche del controllo di gestione e del controllo qualità • Sorveglianza e controllo degli indicatori di prodotto e di processo (cruscotto aziendale) • Analisi dei problemi ed eventuale gestione delle azioni correttive
<i>Chiusura</i>	<ul style="list-style-type: none"> • Bilancio annuale

Tabella 5 – Relazione tra la struttura di un generico processo direzionale e la gestione strategica di impresa.

Non è difficile incrociare le caratteristiche di tale processo con le prescrizioni della **UNI EN ISO 9001:2000** (punto 5)

Non basta fissare gli obiettivi e controllarne il conseguimento, è necessario governare la struttura organizzativa dedicata alla realizzazione del prodotto software (prescrizioni della **UNI EN ISO 9001:2000** punto 7.1): è un concetto che troviamo già indicato come *Definizione del processo* e che possiamo sviluppare riapplicando il solito schema.

Struttura di un generico processo direzionale	governo della struttura organizzativa
<i>Avvio e definizione degli scopi</i>	<ul style="list-style-type: none"> •Elaborazione degli obiettivi di business •Elaborazione degli obiettivi di produzione •Elaborazione dei risultati del riesame di sistema - valutazioni sull'attuale capacità di processo dell'Organizzazione •Definizione del livello tecnologico adeguato al mercato •Definizione dello scopo delle modifiche organizzative - aggiornamento della capacità •Analisi di fattibilità e Stima della ristrutturazione organizzativa
<i>Pianificazione</i>	<ul style="list-style-type: none"> •Definizione del processo - piano di sviluppo organizzativo •Definizione dei processi da inserire eliminare aggiornare •Definizione della mappa dei processi •Definizione delle responsabilità •Piano di ri-progettazione dei sistemi gestione aziendale (sistema qualità, sistema informativo) •Piano d'avviamento in esercizio dei sistemi di gestione ri-progettati
<i>Esecuzione</i>	<ul style="list-style-type: none"> •Acquisizione della tecnologia •Sviluppo dei sistemi di gestione aziendale •Assegnazione dei processi ai responsabili (process owner) •Pubblicazione dell'organigramma •Acquisizione delle risorse •Definizione degli obiettivi operativi •Monitoraggio dei processi attraverso il sistema informativo
<i>Controllo riesami e valutazioni</i>	<ul style="list-style-type: none"> •(Auto) valutazione della capacità di processo (livello di maturità) attivando se necessario la gestione dei problemi e delle azioni correttive
<i>Chiusura</i>	<ul style="list-style-type: none"> •Struttura di un'iniziativa di miglioramento del processo di realizzazione del prodotto software •Riconnessione al riesame di sistema

Tabella 6 – Analisi comparata tra capacità di processo e governo della struttura organizzativa

Il governo della struttura organizzativa ospita il processo di acquisizione della metodologia. Una direzione incisiva, capace di adeguare la struttura organizzativa alle condizioni di mercato deve dare buone risposte alle seguenti domande:

- Quali processi di sviluppo devono essere definiti?
- Quali e quante metodologie acquisire?
- Come collegare la definizione dei processi e la definizione delle metodologie?

Non abbiamo una risposta buona in generale: caso per caso riteniamo che ciascuna Organizzazione debba impegnarsi nell'individuare la migliore soluzione in relazione alla missione, al mercato ed alle risorse disponibili.

Intanto la norma ISO/IEC 12207:1995, oltre ad indicare i metodi da noi riportati, ci offre spunti interessanti per la definizione dei processi dell'Organizzazione.

Al punto 5.3 è definito il "Development process" (processo di sviluppo) in cui è possibile distinguere le seguenti tredici attività:

1. Process implementation – implementazione del processo;
2. System requirements analysis – analisi dei requisiti del sistema;
3. System architectural design – progettazione architettuale del sistema;
4. Software requirements analysis – analisi dei requisiti SW;
5. Software architectural design – progettazione architettuale del SW;
6. Software detailed design – progettazione di dettaglio del SW;
7. Software coding and testing – codifica e test del SW;
8. Software integration – integrazione del SW;
9. Software qualification testing – test di qualificazione del SW;
10. System integration – integrazione del sistema;
11. System qualification testing – test di qualificazione del sistema;
12. Software installation – installazione del SW;
13. Software acceptance support – servizi di assistenza alle attività di accettazione.

Al punto 6 della norma sono definiti i processi di supporto del ciclo di vita:

- Documentazione
- Gestione della configurazione
- Assicurazione qualità
- Verifica
- Validazione
- Riesame congiunto
- Audit
- Risoluzione dei problemi.

Abbiamo parlato di spunti non per sminuirne validità e significato ma per sottolineare ancora una volta l'opportunità che un'Organizzazione decida in proprio la definizione dei processi.

L'osservazione importante è un'altra.

Un'Organizzazione specifica per il SW che lavora per progetti non può formulare un'unica funzione di produzione proprio perché ciascun progetto è diverso da un altro in termini di prodotto e di cliente, né può definire ex novo il processo di sviluppo ogni volta che concorre ad acquisire un contratto.

D'altro canto l'esperienza consolidata di anni nella comunità degli ingegneri del software ha prodotto un buon numero di modelli di ciclo di vita, ciascuno con i suoi vantaggi e le sue controindicazioni.

Dunque il risultato del processo di governo della struttura organizzativa potrebbe essere, anziché uno o due modelli, un metodo messo a disposizione del PM per la "scelta e pianificazione del ciclo di vita".

Le fasi del ciclo di vita di un prodotto software

Nel paragrafo precedente abbiamo affermato un concetto che riteniamo importante: la capacità di personalizzare il processo sulle caratteristiche del contratto da acquisire non è un lusso ma una necessità per un'Organizzazione che lavora per progetti.

Da un lato le Organizzazioni ritengono opportuno darsi delle regole per disciplinare i propri processi.

Dall'altro l'inadeguatezza di una regola rispetto agli obiettivi di mercato, la mancanza di un'alternativa ad essa ed un malinteso significato della qualità portano le Persone a vivere una schizofrenica divergenza tra la qualità della carta e la qualità dei prodotti e dei processi reali.

Per evitare ciò un'Organizzazione che lavora per progetti deve possedere un sistema di gestione della qualità flessibile in grado cioè di adeguarsi ad ogni nuovo contratto.

È come se, una volta definiti i requisiti del prodotto da progettare e le esigenze delle Parti, l'Organizzazione dovesse attrezzare una specifica linea di produzione: ciò è possibile solo se l'Organizzazione possiede nell'ambito del proprio patrimonio tecnologico uno schema di linea di produzione adattabile-personalizzabile al contratto.

Tutto ciò si può fare attraverso lo sviluppo del concetto di modello di ciclo di vita come accennato nel paragrafo precedente.

Partiamo da una definizione.

Modello di Ciclo di vita

Uno schema di riferimento contenente i processi, le attività ed i compiti relativi a sviluppo, esercizio e manutenzione di un prodotto software, che abbraccia la vita del sistema dalla definizione dei requisiti sino al termine del suo utilizzo.

ISO/IEC 12207:1995

Uno dei primi proposti [**Royce 1970**] è il modello sequenziale lineare o "waterfall". Esso prevede che il lavoro proceda sequenzialmente attraverso

le fasi di

1. Analisi,
2. Disegno,
3. Codifica,
4. Collaudo,
5. Manutenzione.

Una fase comincia quando la precedente si è conclusa producendo l'effetto *cascata* (waterfall).

Una fase contiene una o più attività: con riferimento al paragrafo precedente la fase di disegno **potrebbe contenere** le attività di *Software architectural design* e di *Software detailed design*.

Potrebbe contenere e non contiene: questo perché la composizione di una fase non è un'ipotesi da accettare ma una decisione da prendere nel processo di definizione del modello.

Abbiamo già rilevato che ciascuno dei modelli disponibili nello stato dell'arte comporta vantaggi e controindicazioni, rischi ed opportunità: per esempio il modello a cascata espone il progetto a seri rischi nel caso d'incertezza sui requisiti iniziali, in compenso offre al gruppo di lavoro l'opportunità di strutturarsi al meglio e di trovare gradualmente gli equilibri più adatti al progetto.

Le varianti del ciclo di vita a cascata sono innumerevoli a testimonianza sia del successo del modello sia della vocazione delle Organizzazioni mature ad elaborare la proposta disponibile sulla base delle esigenze del mercato e della capacità interna.

Generalizzando il discorso un'Organizzazione che lavora per progetti dovrebbe dotarsi di due metodi:

il primo per l'inserimento di un modello dello stato dell'arte nel repertorio interno elaborandolo secondo le caratteristiche (culturali) dell'Organizzazione,

il secondo per la scelta dal repertorio interno del modello più adatto alle caratteristiche del contratto personalizzandolo allo specifico progetto.

Riepilogo

Il capitolo di introduzione è stato dedicato alla graduale costruzione dell'ipotesi di lavoro che sta alla base dell'intera guida. Una buona pratica di gestione dei progetti software capace, cioè, di rispondere a tutti i requisiti imposti da Vision2000 può essere facilmente sviluppata a partire dall'integrazione di due cicli: il ciclo relativo al modello generale di processo direzionale influenzato dalla struttura organizzativa dell'impresa ed il ciclo relativo al modello specifico di processo produttivo influenzato dalle tecnologie di prodotto e di processo adottate.

Definizione del progetto software

- 2.1 Generalità sulla definizione del progetto
- 2.2 Fattibilità e stima preliminare
- 2.3 Conclusioni
- 2.4 Riepilogo

“Un guerriero della luce studia con molta attenzione la posizione che intende conquistare. Per quanto il suo obiettivo sia difficile, esiste sempre una maniera di superare gli ostacoli. Egli verifica i cammini alternativi, affila la sua spada, e cerca di colmare il proprio cuore con la perseveranza necessaria per affrontare la sfida... ..”

Il Manuale del guerriero della luce - P. Coelho

Con la definizione del progetto inizia l'attraversamento delle vicende dello sviluppo di un'applicazione software dal punto di vista del Capo-progetto: è il primo passo della sequenza definizione-pianificazione-esecuzione-controllo-chiusura.

La Direzione valuta interessante per l'Organizzazione una opportunità di mercato ed ha la necessità di decidere sull'avvio del progetto: fissa gli obiettivi ed affida al Capo-progetto uno studio preliminare incentrato su fattibilità e stima.

Il processo di avvio può essere visto come un progetto il cui risultato definisce l'ambito e costituisce la base informativa delle decisioni della Direzione.

L'analisi di fattibilità è stata strutturata in passi: definizione iniziale dei requisiti, proposta di soluzione, identificazione delle tecnologie di prodotto e di processo, scelta del ciclo di vita, dimensionamenti (prodotto, processo, controllo), conclusioni dello studio.

Il capitolo, infine, affronta il tema dell'ambiente di pianificazione ovvero del sistema di gestione delle informazioni di progetto (Project Management Information System) utile per consolidare le informazioni prodotte, rendere tempestivi eventuali approfondimenti della Direzione, collegare i processi di avvio con i processi di pianificazione, rendere riusabile l'esperienza fatta per altre opportunità di mercato.

caso di studio.

XYZ (Parte committente) ha attivato il proprio processo di Acquisizione.

ABC è un'organizzazione competitiva e mantiene attivo un processo di Fornitura capace di:

catturare la sollecitazione dall'ambiente esterno,

analizzare gli elementi della richiesta d'offerta rispetto alle politiche aziendali,

caratterizzare l'opportunità,

portare la questione (decidere sulla partecipazione alla gara) all'attenzione della Direzione.

La Direzione di ABC ritiene interessante l'opportunità riconoscendo per essa una significativa collocazione nella propria strategia commerciale, decide di rispondere alla richiesta di offerta ed incarica Rossi assegnandogli la responsabilità di uno studio di fattibilità preliminare alla preparazione dell'offerta progetto.

Generalità sulla definizione del progetto

Nel capitolo precedente abbiamo individuato la prima fase del progetto come avvio e definizioni dello scopo del progetto software in seguito semplicemente "Definizione del progetto software".

Le azioni della definizione di progetto, tipicamente sono:

- Definizione del problema;
- Definizione dei requisiti;
- Analisi di fattibilità e stima.

Il risultato della definizione di progetto è lo studio di fattibilità che può anche può essere visto come il prodotto di un progetto limitato ad elaborare i dati disponibili sull'opportunità ed a produrre l'offerta tecnica.

Nell'avvio dello studio la Direzione incarica il responsabile e fissa gli obiettivi che sono raggruppabili per tipi diversi:

- *contrattuali* – qualità, tempi, esigenze degli Utenti;
- *economici* – ricavi, costi, margini;
- *commerciali* – opportunità d'entrare in un nuovo segmento di mercato, necessità di difendere la propria posizione da competitori emergenti;
- *tecnologici* – arricchimento/aggiornamento del patrimonio aziendale delle conoscenze e delle competenze.

La pianificazione dello studio (impegni, tempi, risorse) è naturalmente legata alla cronologia espressa nella richiesta di offerta ed alla capacità dell'Organizzazione.

L'esecuzione dello studio è incentrato su fattibilità e stima del lavoro richiesto rispetto agli obiettivi fissati dalla Direzione.

Il controllo dello studio è incentrato su impegni, tempi e competenze delle risorse impegnate.

I risultati dello studio sono le informazioni necessarie e sufficienti a formulare un'offerta progetto e cioè quali sono le **proposte** che l'Organizzazione è in grado di sostenere verso la Parte appaltante e quali sono le **condizioni** che rendono conveniente l'effettuazione del progetto.

Fattibilità e stima preliminare

caso di studio.

L'analisi di fattibilità e la stima preliminare affidata a Rossi è un piccolo progetto a se stante.

Rossi lavora da solo e decide di strutturare lo studio in quattro parti:

situazione di partenza, analisi degli input disponibili - identificazione del problema opportunità - esigenze - criticità - vincoli;

specifiche di massima del nuovo sistema, identificazione del prodotto;

scelta del modello di ciclo di vita, identificazione delle tecnologie di prodotto e di processo;

proposta di soluzione, dimensionamento del sistema (prodotto, processo, controllo).

Lo studio deve esprimere una argomentata valutazione sul grado d'effettiva conseguibilità degli obiettivi posti dalla Direzione.

Le attività di fattibilità e stima sono orientate a definire una ragionevole sintesi tra gli obiettivi posti e la capacità di processo della struttura organizzativa.

Alla fine lo studio deve produrre informazioni utili a confermare, annullare, aggiornare l'iniziativa presa.

Per i progetti di grande rilievo non basta che l'approccio allo studio si mantenga aderente alla struttura concettuale esemplificata, ma è generalmente necessario che lo studio sia anche **formalizzato** in un progetto preliminare propedeutico al progetto esecutivo.

Nell'Appendice A è stato riportato il modello che l'Autorità per l'Informatica nella Pubblica Amministrazione (AIPA) propone per il documento *studio di fattibilità*: nell'impostazione AIPA lo studio rappresenta il risultato finale dell'intera fase di definizione del progetto software.

Situazione di partenza e nuove esigenze

caso di studio.

Rossi inizia il lavoro analizzando gli input disponibili: il primo passo è il recupero e l'elaborazione del lavoro fatto da XYZ per formulare la richiesta di offerta - documentazione esistente sull'attuale sistema, caratteristiche dei nuovi processi, caratteristiche d'uso del nuovo sistema. Rossi non esclude eventuali interviste ed approfondimenti con il cliente/utente per individuare ulteriori esigenze e/o criticità.

Il secondo passo di Rossi è la descrizione del problema/opportunità: con la fine del monopolio delle telecomunicazione sono nati nuovi operatori di servizi di telefonia fissa.

I nuovi operatori, in particolare Regionali come XYZ, dovranno acquisire prodotti software da integrare nel sistema informativo di supporto alla gestione aziendale.

Tra i prodotti fondamentali c'è quello relativo alla fatturazione dei consumi. I prodotti software attualmente presenti sul mercato hanno costi proibitivi per operatori di questo tipo, i quali reagiscono ad una tale barriera all'acquisto utilizzando in alternativa strumenti fatti in casa poco flessibili e poco adeguati alla complessa problematica della fatturazione dei consumi telefonici.

L'obiettivo posto dalla Direzione di ABC consiste nel realizzare un prodotto software di fatturazione di servizi di telefonia fissa con le seguenti caratteristiche: basso costo, buona qualità e prestazioni ridotte secondo le esigenze di XYZ tipiche degli operatori regionali e locali.

Il progetto concluso con successo consentirà all'azienda di diventare leader di una nicchia di mercato piuttosto interessante.

La proposta di soluzione dovrà recepire/elaborare tempi, costi e prestazioni indicati nella richiesta di offerta.

I primi passi dello studio (analisi degli input disponibili, identificazione del problema/opportunità - situazione attuale, esigenze, criticità, vincoli) ricadono nel processo di gestione dei requisiti.

Cominciamo con una definizione:

Requisito: esigenza o aspettativa che può essere espressa, generalmente implicita o cogente.

- *Nota 1 "(di regola) implicita" significa che è uso o prassi comune, per l'organizzazione, per i suoi clienti e per le altre parti interessate, che l'esigenza o l'aspettativa in esame sia implicita.*
- *Nota 2 per individuare un particolare tipo di requisito, possono essere utilizzati elementi qualificativi, quali per esempio: requisito di prodotto, requisito di gestione per la qualità, requisito del cliente.*

- *Nota 3 un requisito specificato è un requisito che è precisato, per esempio, in un documento.*
- *Nota 4 i requisiti possono provenire da differenti parti interessate.*

UNI EN ISO 9000:2000

L'analisi e la gestione dei requisiti è un processo di raccolta, elaborazione e tracciamento delle esigenze espresse ed implicite del Cliente e più in generale di tutti gli elementi in ingresso alla progettazione e sviluppo.

La gestione dei requisiti è un processo tipico dello sviluppo e come tale è trattato nel capitolo dedicato all'esecuzione di progetto: in avvio è necessaria un'attivazione preliminare del processo limitata agli obiettivi di definizione del progetto.

La prima istanza del processo di gestione dei requisiti prosegue fin quando il Responsabile ritiene di poter consolidare una configurazione completa dei requisiti del progetto.

Qualunque decisione inerente alla definizione di progetto deve essere presa con riferimento alla configurazione al momento consolidata dei requisiti.

All'inverso una modifica alla configurazione consolidata dei requisiti di progetto deve essere attentamente analizzata per valutarne l'impatto sulle stime di prodotto e processo software:

la modifica, se accettata, deve condurre ad un nuovo consolidamento della configurazione dei requisiti di progetto.

Lo scopo fondamentale della gestione iniziale dei requisiti è definire cosa vuole il Cliente e cosa vuole fornirgli l'Organizzazione.

Si tratta di produrre un'ipotesi d'accordo tra le Parti: a tal fine il PM deve identificare ed approfondire chi sia il Cliente o, meglio ancora, quali siano tutte le Parti interessate al progetto (gli attori).

Da una gestione iniziale dei requisiti bene impostata e condotta è lecito attendersi:

- la definizione di un efficace canale di comunicazione tra Progetto e Parti interessate,
- la definizione di un accordo tra Cliente e Fornitore sui requisiti della fornitura,
- la definizione di un meccanismo di gestione delle modifiche ai requisiti della fornitura.

Specifiche di massima del nuovo sistema

caso di studio.

Una volta definito cosa vuole il committente XYZ Rossi passa ad identificare il nuovo sistema attraverso una suddivisione del sistema in blocchi applicativi (CSCI - Computer Software Configuration Items) cioè dei sottosistemi che raggruppano requisiti funzionali che operano su un sottoinsieme di dati.

Nella figura sottostante sono specificati i vari CSCI e i flussi di dati scambiati fra loro. La suddivisione è necessaria perché oltre ad essere utile per determinare la dimensione del prodotto serve anche alla pianificazione di rilasci incrementali del prodotto.

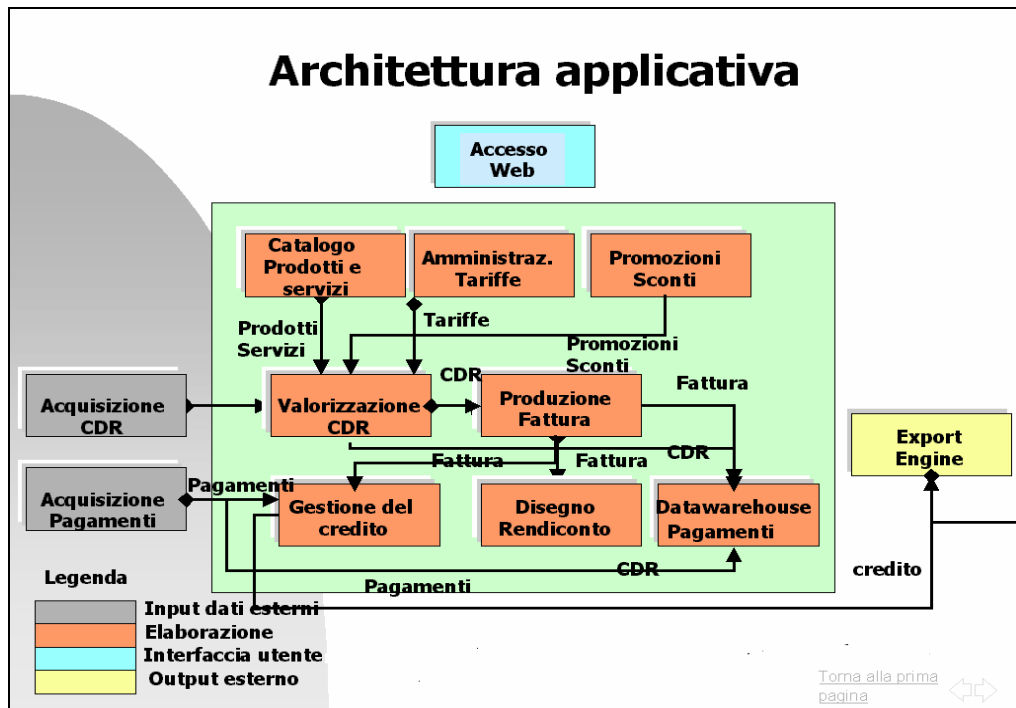


Tabella 7 - Architettura applicativa

Nell'ambito dell'architettura dell'intero sistema applicativo, Rossi approfondisce il sottosistema "disegno rendiconto":

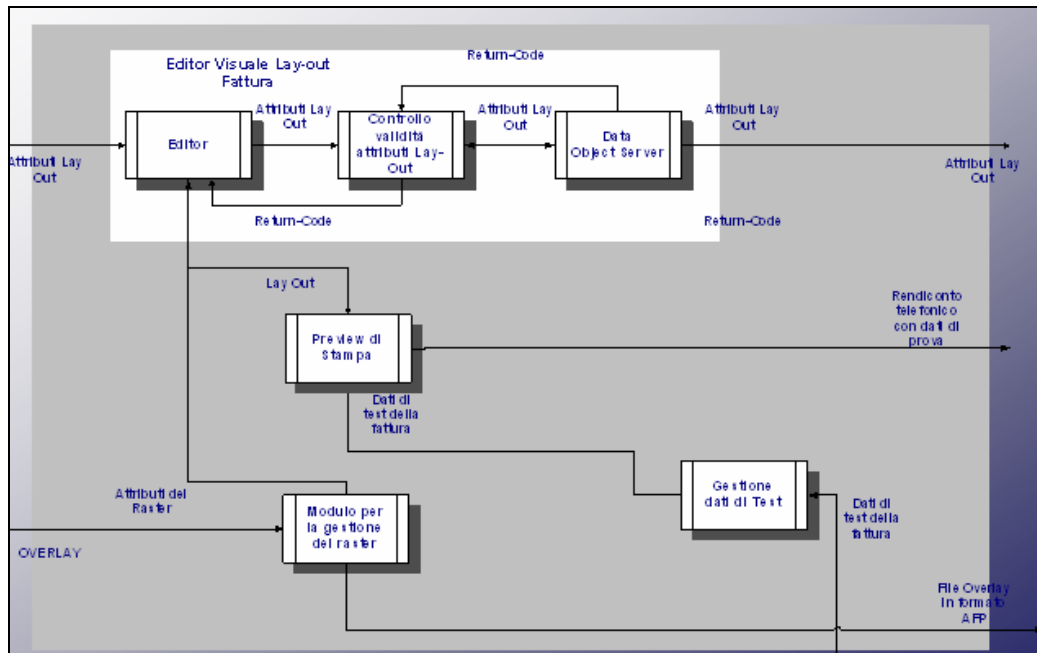


Figura 7 - Diagramma del sottosistema applicativo "disegno dei rendiconti"

Il sottosistema applicativo per il disegno dei rendiconti è composto di sei CSCI. Per ognuno di essi dovranno essere specificati i requisiti. Sulla base dei vincoli imposti dal cliente sul mantenimento dell'impianto pre-esistente d'elaborazione Rossi, tra le varie soluzioni, identifica le tecnologie di prodotto e di processo.

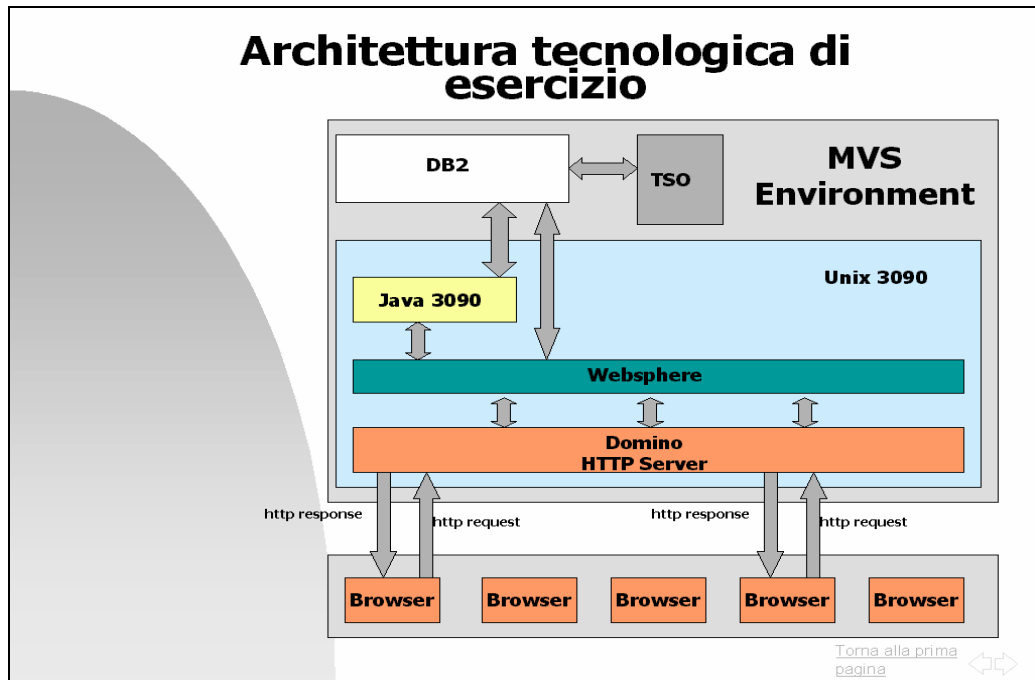


Figura 8 - Schema a blocchi dell'architettura di esercizio

La figura illustra i vari componenti software che compongono l'architettura tecnologica di esercizio. I vari componenti influenzeranno fortemente le scelte per gli ambienti di sviluppo, ad esempio il sistema operativo unix 3090 obbligherà il PM a scegliere il linguaggio C per lo sviluppo di programmi non interattivi. In coerenza con l'infrastruttura di esercizio scelta Rossi prosegue con la definizione dell'infrastruttura di sviluppo:

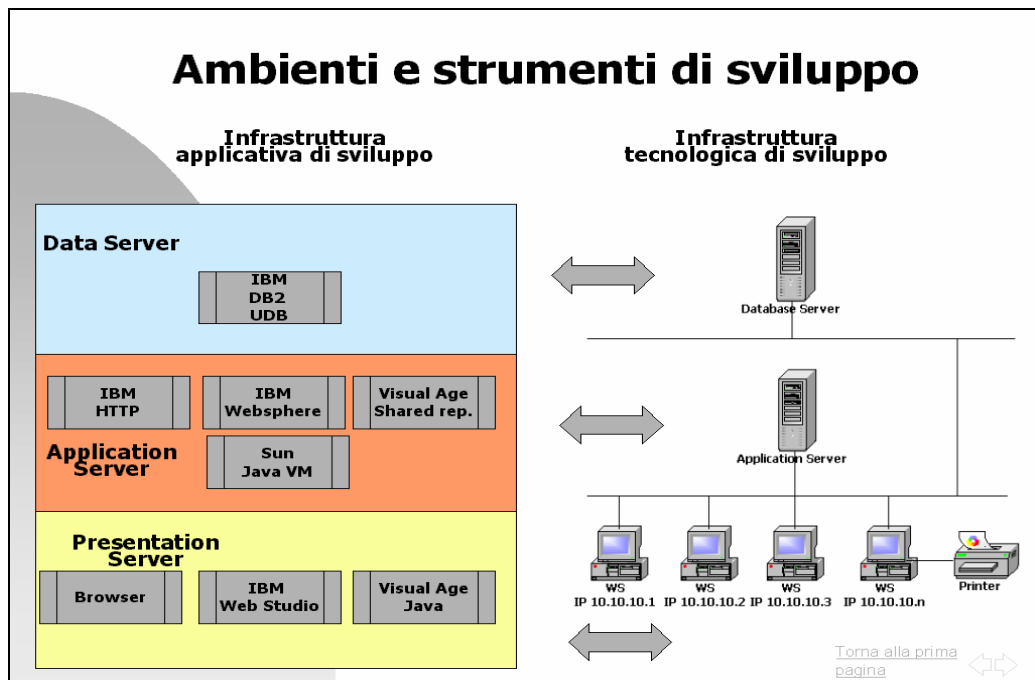


Figura 9 – Ambienti e strumenti di sviluppo

Coerentemente con l'infrastruttura tecnologica di esercizio il PM per lo sviluppo dei componenti software identifica i tools di sviluppo e gli ambienti per lo sviluppo, la scelta ricade su tools che consentono realizzare software che può essere sviluppato su piattaforme hw e software meno costose.

ABC non ha un mainframe per lo sviluppo di programmi C,DB2 ma il C e il DB2 possono essere utilizzati su piattaforma Microsoft Windows.

Le specifiche di massima del nuovo sistema non sono complete né dettagliate se confrontate con un disegno tipico di progetto: mentre un disegno attiva un processo esecutivo di sviluppo, la specifica di massima attiva dei processi di definizione/valutazione verso l'interno e l'esterno dell'Organizzazione.

Verso l'esterno la specifica di massima consente di stimare quale grado di copertura delle esigenze del Committente offre il nuovo sistema.

Verso l'interno la specifica di massima consente di stimare con quale bilanciamento costi-tempi-qualità il Fornitore è in grado di fare il lavoro richiesto.

Approfondiamo di seguito il percorso dello studio verso l'interno dell'Organizzazione.

Scelta del modello di ciclo di vita

caso di studio.

Rossi riprende ed approfondisce tutte le criticità emerse: quelle riconducibili ad esigenze e vincoli posti dal Committente (prestazioni richieste da consegnarsi in un anno solare), quelle conseguenti agli obiettivi della Direzione (rapporto qualità/costo compatibile sia con la strategia finanziaria sia con quella commerciale) e quelle generate dalle scelte tecnologiche appena operate (relativa novità delle tecnologie ad oggetti).

Egli valuta le criticità rispetto al patrimonio tecnologico dell'Organizzazione e di conseguenza sceglie il modello di ciclo di vita migliore per l'occasione.

Nell'Organizzazione sono presenti/praticati due modelli o, per essere più precisi, due categorie di modelli: modelli tradizionali basati sul ciclo di vita a cascata e modelli innovativi basati su UML.

In particolare le novità del prodotto, l'incertezza sui requisiti dell'utente e la disponibilità all'interno di eccellenti competenze di sviluppo su tecnologie ad oggetti convincono Rossi a scegliere il Processo di sviluppo unificato basato su UML.

Individuate le specifiche di massima lo studio prosegue spostando il focus sull'identificazione delle tecnologie di prodotto e di processo. Il discorso sulla tecnologia è particolarmente delicato perché spesso è incerto il confine tra i vincoli imposti dall'esterno e le scelte da effettuare all'interno del progetto. La distinzione tra tecnologie di prodotto e di processo è uno schema che ha valenza soprattutto pratica: si può partire dall'identificazione delle tecnologie di prodotto per arrivare all'identificazione delle tecnologie di processo ed in funzione di esse alla scelta del modello di ciclo di vita del software.

Allo stato attuale delle conoscenze non esiste un ciclo di vita migliore di tutti quanti gli altri.

La questione è venuta fuori quando abbiamo trattato quei processi strategici focalizzati sullo sviluppo della struttura organizzativa e della capacità di processo:

Abbiamo ipotizzato l'impegno della Direzione a rendere disponibile un repertorio dei modelli agibili dall'Organizzazione ed un metodo di scelta.

L'argomento è stato trattato in modo efficace da McConnell in **[RADMCCO]**.

Egli propone una tabella che presenta

- sulle righe le caratteristiche di capacità di un generico modello,
- sulle colonne alcuni modelli di cicli di vita,

Capitolo 2 - Definizione del progetto software

- all'incrocio tra riga e colonna una valutazione che esprime il livello di capacità del modello di fronteggiare la criticità di progetto associata alla caratteristica.

Esempio di tabella di McConnell limitatamente a due modelli: a cascata ed a spirale.

Lifecycle model capability Caratteristiche di capacità	Elementi (mal definiti e/o critici) del progetto	A cascata puro	A Spirale
Lavora con requisiti poco chiari	Incertezza su dati e requisiti di base	Mediocre	Eccellente
Lavora con un'architettura poco chiara	Incertezza su dati e requisiti di base	Mediocre	Eccellente
Produce sistemi molto affidabili	Affidabilità delle prestazioni del sistema	Eccellente	Eccellente
Produce sistemi soggetti ad evoluzioni	Dati e requisiti di base possono subire modifiche sensibili passando da una release all'altra	Eccellente	Eccellente
Gestisce i rischi	Evidenza di uno più rischi	Mediocre	Eccellente
È in grado di rientrare in una schedulazione predefinita	Date di consegna improcastinabili	Buono	Buono
Ha basso impatto sui costi	Monte ore disponibile risicato	Mediocre	Buono
Permette modifiche in corso d'opera	Dati e requisiti di base possono subire modifiche sensibili durante lo sviluppo	Mediocre	Buono
Fornisce al cliente visibilità progressiva	Stati d'avanzamento significativi per le prestazioni	Mediocre	Eccellente
Fornisce alla direzione visibilità progressiva	Stati d'avanzamento significativi per i costi	Buono	Eccellente
Richiede skill poco sofisticati per gestione e/o sviluppo	Modello facilmente apprendibile	Buono	Mediocre

Figura 10 – Tabella di comparazione tra ciclo di vita a cascata e quello a spirale.

Con la disponibilità di una simile tabella, il Capo-progetto responsabile della scelta analizza dati e requisiti di base della progettazione e poi valuta l'applicabilità delle caratteristiche di capacità al proprio progetto: la tabella diventa uno strumento di aiuto alla scelta.

Poniamo per ipotesi che l'Organizzazione abbia messo a disposizione di tutti i suoi Capi-progetto una tabella derivata da McConnell ed opportunamente ridotta-ampliata-adattata alla capacità di processo dell'Organizzazione.

A questo punto il PM si trova nelle condizioni di decidere/selezionare il modello più adatto al progetto e lo deve fare ora perché la scelta del modello influenza un processo iniziale fondamentale quale la gestione dei requisiti.

Sulla base dei risultati della prima istanza del processo di gestione dei requisiti, il PM valuta rischi ed opportunità di ciascun modello agibile stilando una graduatoria d'applicabilità [ZOLLO]

In cima alla graduatoria c'è il modello più adatto rispetto alle caratteristiche del contratto e della capacità dell'Organizzazione.

Il PM può scegliere la via da seguire:

- Selezionare il modello indicato dalla graduatoria come ottimale;
- Selezionare un modello diverso dal primo motivando la scelta;
- Ritenere insoddisfacente la graduatoria e riciclare reistanziando la gestione dei requisiti;
- Ritenere insoddisfacente il metodo ed adottare un modello fuori repertorio concordando la scelta con la Direzione-Ente Qualità.

Ci sono due casi particolari interessanti da analizzare:

- il Committente impone il modello al progetto,
- l'Organizzazione dispone di un unico modello agibile.

In questi casi particolari il processo di scelta può essere ancora utile per individuare i rischi connessi a ciò che oggettivamente si è costituito come vincolo: o si trova un'adeguata risposta ai rischi o si rinuncia alla gara/contratto.

Una trattazione più ampia sui cicli di vita è disponibile nel Quaderno AICQ-CI "Linee guida per la scelta e l'applicazione di cicli di vita del software".

La scelta del modello di ciclo di vita è solo la prima occasione, nello studio e nel progetto, in cui si prendono in considerazione le criticità o in altri termini i fattori di rischio del progetto: l'analisi dei fattori di rischio è un approccio cui il PM è chiamato ricorrentemente per prendere decisioni importanti nell'ambito delle sue responsabilità.

Soluzione proposta

Sulla base della prestazione offerta e della configurazione tecnologica definita approfondiamo di seguito impegni, costi e tempi: lo studio entra nella stima preliminare di progetto.

Dimensionamenti – generalità

I costi per la realizzazione o l'adeguamento del software sono dati principalmente dai costi dello sviluppo del prodotto.

I costi (variabili) di produzione (processo di sviluppo) dipendono dalle quantità prodotte (dimensione del prodotto software) e dai costi unitari (livello di produttività del gruppo di lavoro).

L'unità di misura inizialmente utilizzata per misurare la dimensione del prodotto software è stata la quantità di linee di codice sorgente SLOC (Source Line Of Code o semplicemente LOC).

Stimare la dimensione del prodotto software significava prevedere (con un certo margine di errore) il numero di linee di codice per ciascun programma dell'applicazione da sviluppare.

La LOC non è rimasta l'unica unità di misura utilizzata; i Function Point (punti funzione) sono oggi una valida alternativa per indicare la dimensione del software, inoltre sono stati sviluppati dei fattori di conversione da FP a LOC e viceversa per molti linguaggi in uso.

L'importanza di determinare la dimensione (in LOC o in Function Point) sta nel fatto che, se si conosce la produttività (cioè quante LOC o FP vengono prodotti mensilmente) per persona, è possibile determinare l'impegno totale e di conseguenza i costi totali.

Esempio.

Se abbiamo stimato che un'applicazione software avrà una dimensione pari a 20.000 LOC e conosciamo il numero potenziale di LOC mensili per persona (3.500 LOC al mese per persona) allora avremo un impegno stimato in

$$20.000 : 3.500 = 5,7 \text{ mesi di una persona.}$$

La stima dei costi software è una dei compiti più delicati dello studio.

Una stima *eccessiva* può far perdere l'opportunità di acquisire un nuovo contratto o, quanto meno, può causare una perdita di credibilità dell'Organizzazione nei confronti del cliente.

Una stima *ridotta* può causare perdite economiche e/o creare un sovraccarico di lavoro nel gruppo per rispettare i tempi imposti.

È una buona pratica rifare la stima più volte, come illustrato in figura, durante il ciclo di vita del software: così il Capo-progetto sarà in grado di formulare meglio lo stato di avanzamento del processo e del prodotto.

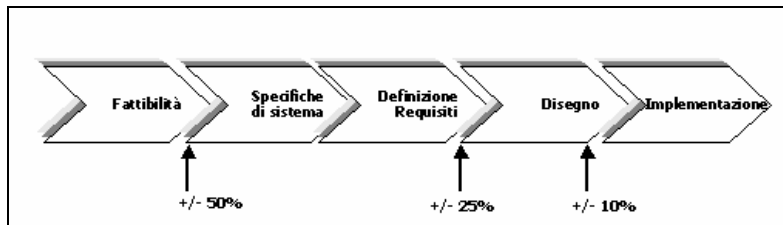


Figura 11 – Percentuale di accuratezza delle stime nelle varie fasi del ciclo di vita.

Più si va avanti lungo il percorso di progetto, più si riducono i margini d'errore sulle stime:

- al termine dell'analisi di fattibilità quando sappiamo in generale cosa il software deve fare (Requisiti ad alto livello) ma non conosciamo i dettagli dei requisiti le stime producono dei risultati con un margine di errore di +/- 50%;

- al termine della fase dei requisiti, quando conosciamo le funzionalità, le stime producono un margine di errore di +/- 25%;
- infine quando il processo di disegno di dettaglio è completo l'accuratezza delle stime è di +/- 10%.

Il 50% di margine di errore conferma quanto sia difficile effettuare una stima dei costi all'inizio, cioè quando si conoscono i requisiti ad alto livello **[Roetzheim 2000]**.

Ciò è vero soprattutto nel caso di realizzazione di nuovi prodotti, mentre la stima dei costi nel caso di adeguamento di un'applicazione in esercizio, le stime sono più accurate:

la maggior precisione è dovuta al fatto che si hanno maggiori informazioni (conoscenze sul problema, conoscenze sulle tecnologie di base, conoscenza dell'applicazione software, produttività individuale, etc..).

Per determinare in questa fase la dimensione del prodotto si possono utilizzare varie tecniche: Stima per analogia, Conteggio dei blocchi funzione, Stima statistica, function point, cocomo, ...

L'approfondimento delle tecniche di stima esula dagli scopi della guida: di seguito ci soffermeremo su una delle tecniche praticabili soprattutto per dare completezza al caso di studio. Il lettore interessato può trovare una panoramica completa in [SEDEHI].

Dimensionamento del prodotto

caso di studio.

Rossi ha segmentato, nell'architettura applicativa, il sistema di fatturazione per XYZ nei seguenti sottosistemi:

1. Amministrazione tariffe
2. Gestione del credito
3. Acquisizione dei pagamenti
4. Datawarehouse dei pagamenti
5. Acquisizione CDR
6. Valorizzazione CDR
7. Promozioni e sconti
8. Produzione fattura
9. Catalogo prodotti e servizi
10. Acquisizione dati da altri sistemi
11. Export engine.
12. Disegno dei rendiconti

Il sottosistema per il disegno dei rendiconti telefonici è costituito da:

1. Editor visuale
2. Gestione Overlay
3. Gestione dati di test

CSCI	CSC	Ottimistico	Pessim.	Probabile	Atteso	Deviazione Standard	LOC
Editor Visuale	Editor	6040	9060	15000	12517	1493	
	Controlli_Validità	5240	15720	7860	8733	437	
	Object Data Store	2520	5000	3000	3253	80	
					24503	735	26708
Gestione Overlay	Editor	5400	8100	16500	13250	1850	
	Controlli_Validità	4500	13500	7000	7667	417	
	Object Data Store	2200	22500	4500	7117	383	
					28033	837	30545
Gestione dati di test	Gestione strutture dati	300	900	750	700	75	
	Funzioni di gestione dati	180	300	2400	1680	370	
					2380	209	3006
Totale linee di codice atteso					54917	1781	60259

Figura 12 - Stima statistica del sottosistema software per il disegno dei rendiconti telefonici

Il metodo **[Gaffney e Britcher 1984]** presentato consiste nello stimare le funzionalità presenti all'interno di un sottosistema.

Il parametro di input di questo metodo sono le specifiche del sistema (le specifiche del sistema descrivono principalmente i sottosistemi software e le funzioni che ne fanno parte).

Il livello di decomposizione dipende principalmente dalle conoscenze del dominio applicativo possedute dagli analisti (e dal PM) in questa fase: generalmente 2 livelli possono essere considerati sufficienti a fornire una stima.

Nella terminologia utilizzata dagli autori del metodo s'indicano con CSCI un blocco di funzioni o un sottosistema software, mentre con CSC una macrofunzione o funzione.

I passi da effettuare sono i seguenti:

- Si contano o si stimano il numero dei blocchi ad un dato livello di dettaglio, CSCI o CSC
- Si moltiplicano il numero dei blocchi per il valore atteso (in LOC) della dimensione di quel tipo di blocco.

Il valore ottenuto è la stima della dimensione del software.

Il metodo non può essere di aiuto se ci sono meno di 3 blocchi funzione, inoltre può essere combinato con il metodo di stima statistica.

Il metodo utilizza l'esperienza posseduta dall'organizzazione su progetti simili per determinare statisticamente la dimensione del software da realizzare.

Tale metodo è stato descritto da [PUTNAM].

Dimensionamento del processo

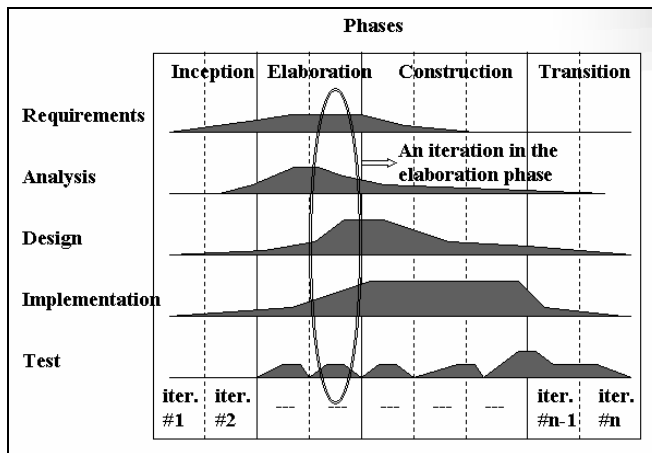


Figura 13 – Relazione tra fasi e iterazioni nel modello di sviluppo unificato

Nel caso di studio in esame le iterazioni #1,#2,... corrispondono rispettivamente a:

- #1 Fattibilità
- #2 Sviluppo prototipo
- #3 Architettura release 0.1
- #4 Architettura release 0.2
- #5 Architettura release 0.3
- #6 Iterazione Sviluppo prodotto
 1. Incremento 1 – Sviluppo del Disegno dei rendiconti
 2. Incremento 2 – Sviluppo parte web
 3. Incremento 3 – Processi batch di fatturazione (Valorizzazione CDR,Acquisizione CDR,Acquisizione Pagamenti,Produzione Fattura,Export Engine)
 4. Incremento 4 – Datawarehouse
- #7 Alfa Release
- #8 Beta Release 1.0
- #9 Beta Release 1.1
- #10 Release prodotto 2.0

D'ora in avanti il caso di studio in esame tratterà specificatamente il "Disegno dei rendiconti".

Caso di studio.

Rossi ha scelto un modello di ciclo di vita basato su RUP (Fig.13) ed il processo unificato di sviluppo.

In figura 14 è definito un piano di massima. Le fasi principali sono Concezione, Elaborazione, Costruzione e transizione.

Vengono fissate le milestone di fase e definite le varie iterazioni e gli incrementi per ogni iterazione.

Vengono definite 10 iterazioni e nell'iterazione 6 vi sono 4 rilasci incrementali.

Uno dei rilasci (il primo) riguarda Il Disegno dei rendiconti telefonici.

La dimensione stimata per questo sottosistema è di circa 60000 LOC, la produttività del laboratorio di sviluppo è di 1500 LOC/mese per cui l'effort complessivo è di 40 mesi uomo.

L'analisi condotta consentirà di stimare tempi e costi.

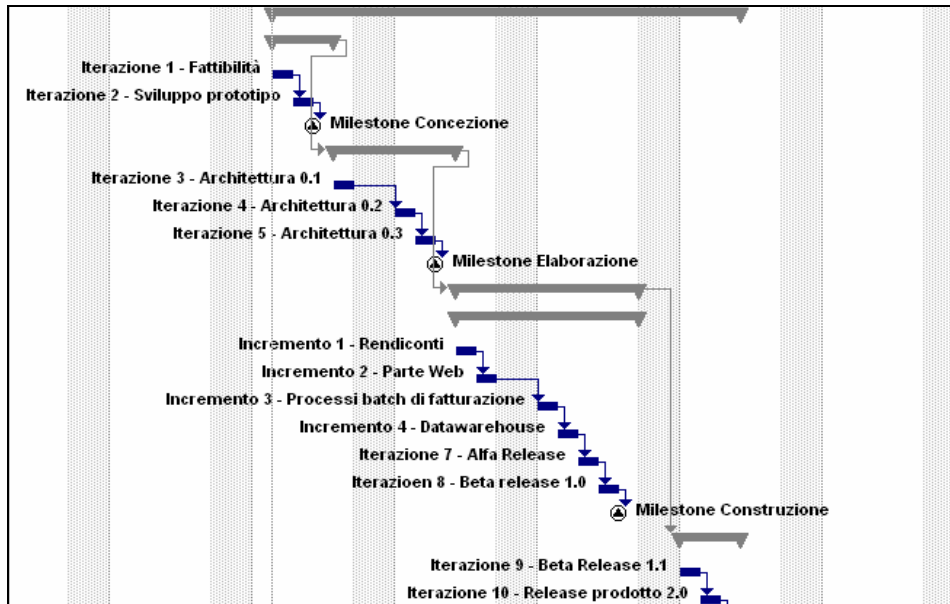


Figura 14 – Iterazioni ed incrementi nel ciclo di vita basato su processo di sviluppo unificato del caso di studio in esame

Dimensionamento del controllo – Identificazione iniziale del rischio

caso di studio.
 Rossi riprende e completa l'identificazione iniziale del rischio di progetto: in funzione dell'analisi condotta stabilisce dei punti di controllo sia a livello di piano di massima (major milestone) sia a livello di iterazione e a livello di incrementi. Per l'incremento relativo al Disegno dei rendiconti i punti di controllo sono settimanali, in tal modo Rossi può avere delle deviazioni dal piano di al massimo una settimana.

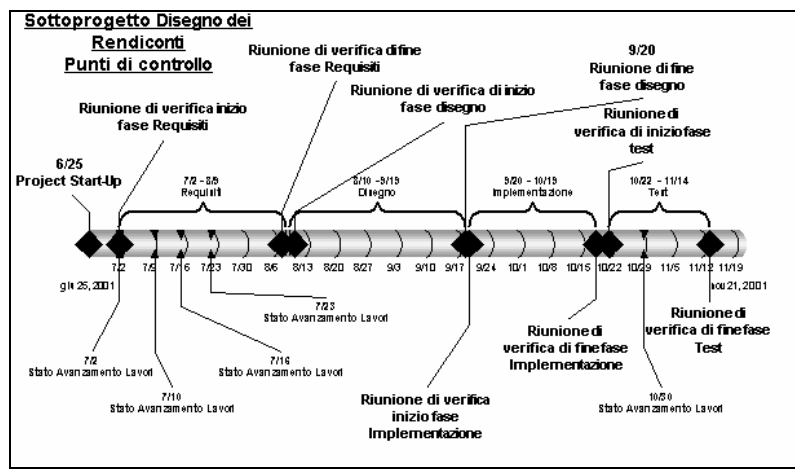


Figura 15 – Pianificazione dei punti di controllo del progetto caso di studio

Fin dall'inizio della guida abbiamo sostenuto l'affermazione secondo cui ogni progetto è diverso dagli altri e produce un prodotto unico. Sempre mantenendo il livello minimo prescritto dalla norma **ISO9001:2000**, un'organizzazione competitiva deve porsi il problema del dimensionamento del controllo. La caccia alle zanzare è cosa profondamente diversa dalla caccia al cinghiale in termini di strumenti, metodi e competenze. A lume di naso si può affermare che la dimensione del controllo deve crescere quando cresce il grado di complessità del progetto.

In termini organizzativi questa considerazione conduce a definire un metodo pratico per individuare la classe di incertezza iniziale (complessità e dimensione) di un progetto e poi dimensionare il controllo in funzione della classe individuata.

È opportuno evitare confusioni tra tale identificazione iniziale del rischio globale e la gestione dei rischi, area di conoscenza della disciplina di Project Management i cui processi sono trattati più avanti nelle fasi di pianificazione e di controllo del progetto.

La classificazione di progetto è un concetto che appartiene ad un argomento molto interessante, la gestione del portafoglio progetti, non trattato nella presente guida: il lettore interessato può riferirsi a **[Corti2002]**.

Ritornando al dimensionamento del controllo il metodo aziendale può avere un impianto analogo a quello per selezionare il modello del ciclo di vita.

Si tratta di mettere a disposizione del PM delle checklist a risposta chiusa derivabili dalle definizioni aziendali di complessità relativamente ai processi di sviluppo software.

È possibile, per esempio, definire la classe di un progetto come una funzione di due variabili:

classe del progetto = F(dimensione del processo, complessità del prodotto e del processo).

La funzione F può essere costruita empiricamente attraverso una tabella del tipo:

Grado	livello		impegno	
	Piccolo	Medio	Grande	Extra
44-55				
33-43				
22-32				
11-21				
0-10				

Figura 16 – Tabella per la classificazione della classe di progetto

Nella tabella di esempio si presuppone che:

- la stima della dimensione del processo conduca alla definizione di un livello d’impegno in una scala di quattro possibili,
- la valutazione della complessità di prodotto e di processo produca un punteggio compreso tra 0 e 55 riconducibile ad un grado di complessità in una scala di cinque possibili.

Dalla fase di dimensionamento del processo si dispone della stima dell'impegno (per esempio in ore uomo): con un criterio di classificazione si ottiene il livello d'impegno del progetto.

Sulla base delle informazioni consolidate attraverso il processo di gestione dei requisiti, il PM risponde ai quesiti della checklist ottenendo prima il punteggio e poi uno specifico grado di complessità del prodotto e del processo.

Con livello e grado si può entrare nella tabella esemplificata e ricavare la classe del progetto.

Un'organizzazione consapevole della propria capacità deve definire le classi ammissibili per se stessa e deve dare un significato operativo ad ognuna collegandole al dimensionamento del controllo.

Conclusioni

caso di studio.

Rossi ha portato a conclusione lo studio e cioè è riuscito a valutare con quali costi-tempi-qualità l'Organizzazione può fare il lavoro richiesto: arrivato a questo punto chiede un incontro con la Direzione per discutere/consolidare i risultati conseguiti.

La Direzione approva lo stato di avanzamento e riprende il possesso del processo di Fornitura: acquisisce i costi, fissa i ricavi ed assegna la responsabilità della trattativa con XYZ alla funzione commerciale.

Dallo studio all'accordo contrattuale

Per privilegiare l'esposizione, l'esecuzione dello studio è stata tratteggiata come una sequenza lineare di attività: in realtà l'intera fase di definizione avanza per approssimazioni successive ed è connotata da grande incertezza.

Il PM spende la sua opera per elaborare tale incertezza estraendone basi affidabili sui cui fondare gli accordi contrattuali e la pianificazione di progetto.

Esecuzione e controllo dello studio sono intimamente connessi: ogniqualvolta s'avverte la sensazione che l'incertezza aumenti anziché diminuire è opportuno fermarsi, fare il punto della situazione (magari attraverso lo strumento del riesame) e ricominciare dai requisiti per acquisire altri input, per risolvere contraddizioni emerse, per ricostituire le informazioni consolidate.

Dimensionato il controllo di progetto, il PM ha finalmente eseguito lo studio: sono disponibili risultati ed informazioni ma soprattutto c'è la prima definizione dell'ambito del progetto (definizione di cosa fare e cosa non fare, limitazioni nel tempo e nello spazio).

Sulle opzioni emerse si eserciterà la decisione della Direzione nella chiusura della fase di definizione: il **PM** propone una o più soluzioni ciascuna corredata da propri calcoli e stime e la **Direzione** sceglie la via da seguire magari rimodulando gli obiettivi iniziali.

Il processo di condivisione e scelta appartiene alla Direzione ma l'interpretazione del PM non è affatto trascurabile:

- ha un'idea globale del progetto, dei risultati attesi, delle dimensioni, della complessità, del rischio globale (visione);
- dispone degli elementi per ridiscutere con la Direzione la missione inizialmente definita;

sa cosa chiedere per continuare il cammino.

Il processo di condivisione e scelta consente la formalizzazione dei risultati dello studio e l'attivazione di un processo di negoziazione del contratto.

Ricapitoliamo i risultati dell'intera fase di definizione del progetto:

- configurazione delle esigenze del Cliente e del Committente
- specifiche di massima della soluzione
- modello del ciclo di vita;
- primo taglio del modello – nel dimensionamento di prodotto e processo sono state identificati rilasci, costi e fasi del modello applicabili all'iniziativa;
- secondo taglio del modello – nell'identificazione iniziale del rischio sono state definite le dimensioni del controllo (qualità);
 - identificazione degli input disponibili;
 - identificazione degli output attesi;
 - identificazione delle tecnologie necessarie nel ciclo di vita del prodotto.

Le informazioni disponibili in questo momento del progetto sono di importanza vitale per molti motivi:

- l'interazione tra PM e Direzione può richiedere approfondimenti che saranno tanto più tempestivi ed efficaci quanto più è buona la gestione delle informazioni,
- l'eventuale prosecuzione del lavoro richiede un collegamento tra definizione e pianificazione che deve funzionare indipendentemente dalle singole Persone coinvolte (predisposizione dell'ambiente di pianificazione),
- le conoscenze messe in gioco dall'Organizzazione sono di gran valore ed è sicuramente conveniente rendere l'esperienza fatta riusabile per altre opportunità di mercato (knowledge management).

Una parte delle informazioni e delle conoscenze prodotte trova un'adeguata rappresentazione nei documenti tipici previsti nelle pratiche aziendali coinvolte.

L'altra parte va opportunamente gestita nell'ambiente di pianificazione.

Definizione dell'ambiente di pianificazione

Una buona gestione delle informazioni può cominciare con una classificazione del tipo:

- gestione del prodotto – requisiti
- gestione del processo – Risorse Umane, pianificazioni
- gestione della qualità – controlli
- gestione della configurazione – struttura output
- gestione del Cliente – accordi

ecc.

La classificazione delle informazioni aiuta il PM a gestire relazioni fondamentali tra le informazioni, per esempio:

- la relazione tra fase del ciclo di vita e tecnologie di base da utilizzare aiuta nell'individuazione e selezione di strumenti, tecniche e metodi di lavorazione
- la relazione tra fase del ciclo di vita e skill delle Risorse Umane candidate all'esecuzione dei compiti individuali aiuta nella gestione delle competenze e nell'assegnazione dei ruoli

ecc.

Definizione dell'archivio di progetto

L'archivio di progetto è il contenitore di informazioni e documenti riguardanti il progetto.

La quantità di documentazione prodotta per un progetto software può variare da una decina di pagine a milioni di pagine in funzione delle caratteristiche del prodotto software da realizzare.

Informazioni e documenti che vengono trattati generalmente sono così classificabili:

- Documentazione tecnica (Specifiche dei requisiti, specifiche delle operatività correnti, ecc.);
- Documentazione di Project Management (richiesta di offerta, Piano di rilascio, Contratto, Offerta, note, posta, verbali, ecc.);
- Manualistica tecnica;

- Procedure interne.

Per facilitare la gestione di informazioni e documenti da parte degli attori di progetto è importante che il Capo-progetto s'impegni a:

- definire la struttura dell'archivio di progetto
- individuare i mezzi di archiviazione da utilizzare
- definire le regole per gli accessi e individuare un responsabile per l'amministrazione dell'archivio di progetto.

In ogni caso la definizione e l'amministrazione dell'archivio di progetto devono tener conto delle politiche della qualità dell'azienda.

La struttura dell'archivio di progetto

La strutturazione dell'archivio di progetto dipende da progetto a progetto sebbene sia possibile definire uno schema generico e poi modificarlo a secondo delle caratteristiche del progetto.

Un archivio di progetto potrebbe essere organizzato nel seguente modo:

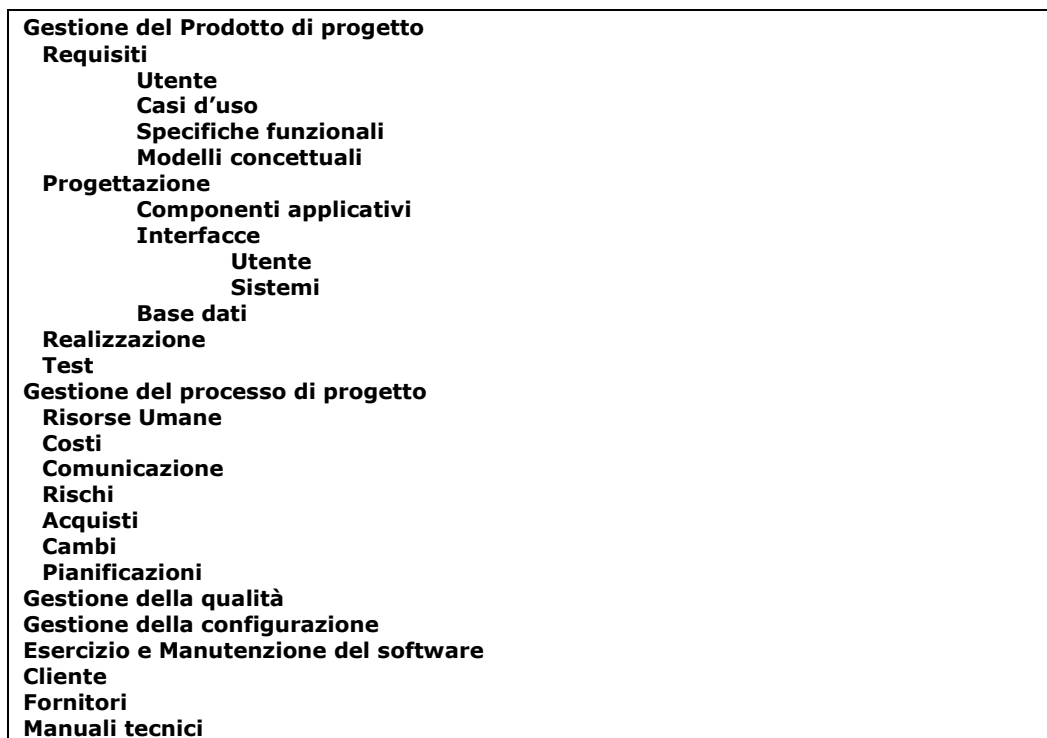


Figura 17 – Generica struttura di un archivio di progetto

Ogni sezione a sua volta può essere ulteriormente decomposta in funzione delle esigenze del progetto.

All'interno di ogni sezione poi verranno collocati i vari documenti.

I mezzi di archiviazione

Il tipo di mezzo da utilizzare dipende fortemente da:

- logistica del progetto (i componenti sono nella stessa stanza o sono dispersi geograficamente in più posti ?),
- disponibilità delle risorse hardware (sono in collegamento via LAN/WAN oppure sono in configurazione stand alone,
- costi previsti.

In definitiva si può scegliere tra i seguenti mezzi:

- Raccoglitori ad anello,
- Personal Computer dedicato all'archivio di progetto,
- Directory di progetto su rete LAN,
- Sito web.

È importante notare che la scelta di uno non implica l'esclusione dell'altro ma si possono utilizzare combinazioni di mezzi differenti.

Regolamentazione degli accessi

Sebbene la condivisione dell'informazione sia importante per la riuscita del progetto non tutti possono avere la stessa modalità di accesso all'archivio di progetto (es. le tariffe delle risorse non possono essere consultate da chiunque).

Per questo è necessario definire dei profili e definire la modalità di accesso alle sezioni e/o dei documenti dell'archivio di progetto.

Un documento di esempio che definisce i profili e le modalità di accesso è dato di seguito.

Sezioni	Amministratore	Project Manager	Componenti del team	SQA Manager	Direzione tecnica
Gestione del prodotto	CRUD	R	CRUD	R	
Gestione del progetto	CRUD	CRUD	R		R
Gestione della qualità	CRUD	R	R	CRUD	
Gestione della configurazione	CRUD	R	R	R	
Manuali tecnici	CRUD	R	R	R	
Esercizio manutenzione del sw	e CRUD	R	R	R	

C=Creazione,R=Lettura,U=Aggiornamento,D=Cancellazione

Tabella 8 – Matrice degli accessi per l'archivio di progetto

Riepilogo

Il capitolo di definizione è stato dedicato all'applicazione pratica dei principi di gestione della qualità alla gestione di progetto attraverso l'approccio definito nel capitolo di introduzione. In particolare evidenza appare il principio che invita a prendere decisioni sulla base di dati di fatto: l'approccio scelto dà conto di come la Direzione possa decidere sull'avvio e la prosecuzione del progetto sulla base dei dati di stima.

Pianificazione del progetto software

- 3.1 Definizione e sequenziazione delle attività
- 3.2 Definizione di durata e date
- 3.3 Pianificazione delle risorse
- 3.4 Previsione dei costi
- 3.5 Sviluppo della schedulazione
- 3.6 Pianificazione della qualità
- 3.7 Sviluppo del piano degli acquisti
- 3.8 Piano di sviluppo dell'organizzazione di progetto
- 3.9 Pianificazione della configurazione
- 3.10 Sviluppo del piano di gestione dei rischi
- 3.11 Sviluppo del piano di comunicazione
- 3.12 Riepilogo

“...Dantes si sarebbe precipitato sopra di lui nel momento in cui,ancora stordito per la caduta, non avrebbe potuto difendersi,lo avrebbe legato, gli avrebbe turato la bocca, ed allora tutti e due passando da una finestra di questa galleria,sarebbero discesi lungo la muraglia esterna coll'aiuto della scala di corde, e si sarebbero salvati. Dantes sbattè le mani, e i suoi occhi sfavillarono di gioia;questo piano era così semplice, che era impossibile non riuscisse.”

Il Conte di Montecristo - A.Dumas

Con la pianificazione del progetto il Capo-progetto giunge fino alla stesura del piano esecutivo dello sviluppo di un'applicazione software.

Il Capitolo è articolato in due parti.

Nella prima parte la pianificazione esecutiva declina scopi, tempi e costi consentendo al Capo-progetto un primo livello di integrazione del piano di progetto che è stato definito come pianificazione di base.

Il Capo-progetto parte dai risultati consolidati nella fase precedente, definisce le attività (WBS), sequenzia le attività, prevede le durate, definisce i compiti individuali, ipotizza la disponibilità delle risorse, prevede date di inizio e fine, identifica il fabbisogno di tecnologia, definisce i requisiti delle risorse, prevede i costi, sviluppa la schedulazione, formula il budget.

Nella seconda parte del Capitolo la pianificazione esecutiva declina qualità, gestione delle Risorse Umane, comunicazione, rischio e acquisti consentendo al Capo-progetto il secondo livello di integrazione del piano di progetto che è stato definito come pianificazione di dettaglio.

Il Capo-progetto approfondisce i temi della pianificazione di base e sviluppa i seguenti punti: il piano della qualità (controlli di verifica e validazione e monitoraggio), il piano degli acquisti, il piano di gestione della configurazione, il piano di gestione dei rischi, il piano di comunicazione, la definizione della struttura organizzativa di progetto.

Caso di studio

Sulla base dei risultati della fase di definizione (problema/opportunità, requisiti, ambito, pianificazione preliminare, ambiente di pianificazione...) la Direzione ha preso la decisione di partecipare alla gara: l'Organizzazione ha prodotto offerta e si è aggiudicata il contratto.

Ora si pone il problema di portare a termine il progetto con la soddisfazione di tutte le Parti interessate.

Nel nostro caso lo studio di fattibilità ha prodotto tutte le informazioni necessarie alla preparazione della risposta ed alla definizione del contratto. La pianificazione preliminare, però, non consente in alcun modo l'avvio dell'esecuzione: per gestire correttamente il processo ed il prodotto di progetto il PM deve scendere nel dettaglio di tempi, impegni, costi, dimensioni, prestazioni attraverso la fase di pianificazione esecutiva.

L'approfondimento dei temi della pianificazione scaturisce da una decisione che il PM prende tenendo congiuntamente presenti due riferimenti:

riferimento esterno -il Cliente si aspetta una risposta puntuale alle proprie esigenze secondo quanto definito nei documenti contrattuali,

riferimento interno - la Direzione si aspetta una pianificazione adeguata al rischio inizialmente identificato ed accettato.

La Direzione di ABC decide di assegnare il progetto a Rossi e glielo comunica attraverso una comunicazione contenente tra l'altro gli estremi del contratto (Project Charter)

Qualunque sia il progetto, la tecnica scelta per la pianificazione deve consentire la produzione di un piano agile/utile/aggiornabile lungo tutto il percorso dell'iniziativa.

Generalmente il PM prevede nel piano una o più attività di raffinamento e d'approfondimento della pianificazione: l'aggiornamento del piano va collegato alla disponibilità dei risultati dell'analisi dei requisiti e degli altri processi di sviluppo che nel tempo completano e stabilizzano il quadro di riferimento del progetto. In corso d'opera sono probabili anche aggiornamenti non pianificati necessari per recepire i ritorni dei processi di controllo.

Anche se la formulazione del piano non richiede necessariamente l'utilizzazione di uno strumento informatico, Rossi adotta uno dei tool di maggiore successo per la pianificazione e gestione dei progetti: Microsoft PROJECT (si veda App. E per maggiori dettagli).

Rossi decide di pianificare il progetto attraverso quattro documenti: uno (SPMP) orientato alla gestione del processo di sviluppo e gli altri tre (SCMP, SQMP, SVVP) orientati alla gestione del prodotto da sviluppare.

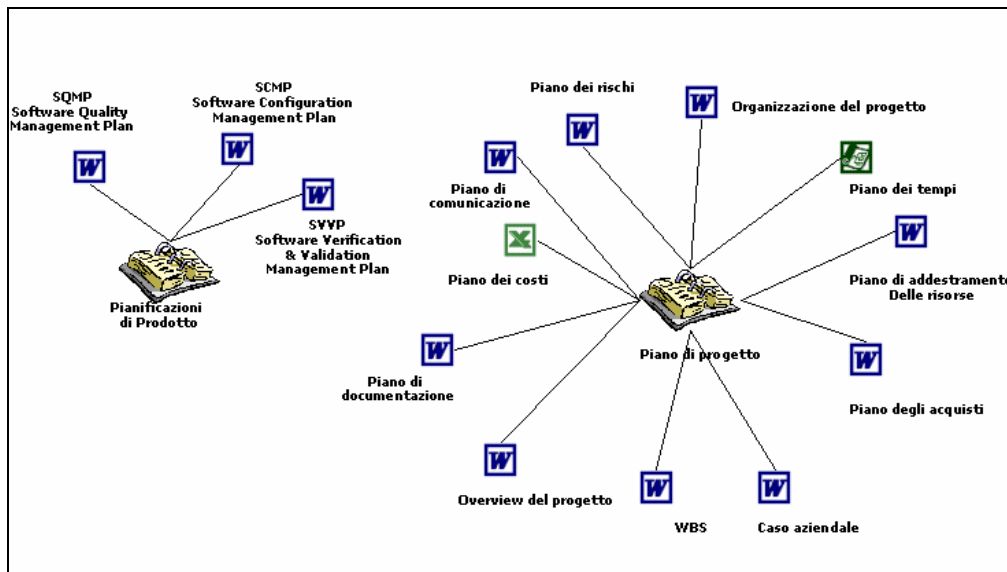


Figura 18 – Elementi costituenti la pianificazione

Definizione e sequenziazione delle attività

Per la definizione delle attività di progettazione è opportuno partire dai risultati dello studio di fattibilità e, più precisamente dalla definizione dell'ambito in esso contenuta: con le dovute approssimazioni potremmo dire che è una prima rappresentazione della struttura del prodotto software da realizzare.

È chiaro che la struttura del prodotto è un risultato disponibile solo nella successiva fase di esecuzione (disegno architettonico del software – ISO/IEC12207 punto 5.3.5).

Ora in pianificazione si tratta di formulare delle ipotesi o meglio di riprendere e sviluppare le ipotesi a suo tempo formulate in fase di studio. L'approccio iniziale alla strutturazione del prodotto consiste nei seguenti due passi:

- delineare una struttura che dia coerenza all'unione dell'insieme dei componenti da consegnare con quello dei componenti non oggetto di consegna,
- elaborare la struttura delineata per consentire i necessari ragionamenti sulla copertura dei requisiti.

Sulla bozza di struttura del prodotto appena definita il PM comincia a fare i suoi ragionamenti per impostare/rivedere le scelte di "make or buy": si tratta di capire (in base agli input di progettazione disponibili) quali parti del prodotto conviene che si facciano all'interno dell'organizzazione (make) e quali altre conviene che si acquistino già fatte da Fornitori qualificati (buy).

Quest'analisi produce implicazioni importanti sulla definizione delle attività perché si potrebbe decidere di affidare all'esterno una porzione del lavoro e quindi dividere il progetto in due o più sottoprogetti.

Ancora dalla fase di studio (definizione del progetto) abbiamo ereditato un modello di ciclo di vita tagliato sulle esigenze emerse, ma non è ancora uno schema operativo perché non è completo l'adattamento (istanziamento) del modello scelto sul caso specifico.

Nel modello le attività sono già identificate/strutturate ed il PM deve solo completarne la definizione assegnando input ed output a ciascuna:

da un lato dobbiamo rispettare lo spirito del modello (propedeuticità di fasi ed attività), dall'altro dobbiamo ricollocare dentro al ciclo di vita la struttura iniziale del prodotto (output) e la configurazione degli input di progettazione e sviluppo.

A questo punto possiamo affermare di aver individuato una struttura gerarchica (Breakdown Structure) del processo di sviluppo (Work): in altri termini abbiamo definito la Work Breakdown Structure (WBS) di progetto.

I legami (interdipendenze) tra le attività sono definiti a livello astratto/concettuale nel modello.

Caso di studio

In fase di studio Rossi ha scelto un modello di ciclo di vita basato sul processo unificato di sviluppo [ROYCESPM]. Ora in fase di pianificazione esecutiva Rossi personalizza la WBS generica del modello dettagliando le iterazioni e gli incrementi del progetto. In figura 20 è stato riportato il piano di massima del sottoprogetto relativo ai rendiconti telefonici.

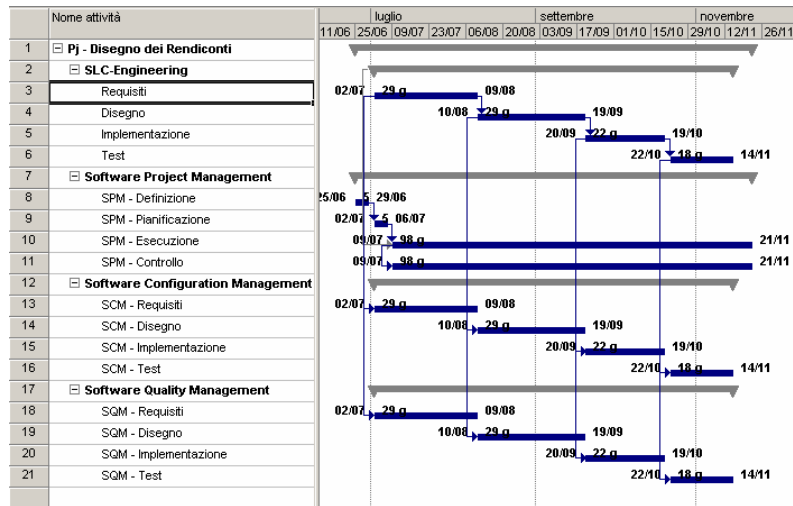


Figura 19 – Pianificazione di massima del sottoprogetto "Disegno dei rendiconti"

La graduale concretizzazione del modello (di ciclo di vita) rispetto alle condizioni specifiche del nostro progetto ha avuto un primo passaggio di raffinamento basato su input ed output, ne avrà un secondo basato su tempi ed impegni.

Definizione di durate e date

Dalla fase di studio (definizione del progetto) abbiamo ereditato il dimensionamento del processo e del controllo.

Dal passo precedente (definizione delle attività) abbiamo la disponibilità di una WBS e di una sequenza di attività che riescono a rappresentare lo sviluppo software come un processo capace di trasformare gli input disponibili negli output attesi.

Ora il PM può elaborare il dimensionamento del processo e del controllo trasportando l'ipotesi d'impegni sulle attività definite: l'abilità richiesta è quella di saper bilanciare obiettivi tra loro conflittuali come tempi/impegni/qualità mantenendo come riferimento l'ambito definito.

Così facendo il PM riesce, con i dovuti aggiustamenti e ripensamenti, a caratterizzare ciascun'attività della WBS con un impegno: per passare dall'impegno alla durata è necessario frammentare le attività fino ad arrivare al compito individuale.

Il compito individuale è un processo di sviluppo personale [Humphrey 1997] che il PM assegnerà ad uno Specialista del suo gruppo.

Il compito individuale è un'attività professionale, dunque lo Specialista incaricato va considerato come un professionista che agisce in pieno **autocontrollo** e che ha la delega del Capo-progetto per prove, controlli e collaudi della parte che realizza.

Il compito individuale può essere traguardato come un processo: lo scopo del processo di sviluppo personale è quello di trasformare un insieme di requisiti (specifiche di disegno) in una parte del prodotto di progetto che abbia caratteristiche tali da soddisfare le esigenze espresse (criteri ed evidenze di controllo).

L'aver definito la durata d'attività e compiti non significa aver definito la durata del progetto.

A questo punto il PM impone come vincolo una data d'inizio e, lavorando pazientemente sul numero delle risorse e sui parallelismi **tecnicamente** possibili, consegue, una per volta, le date definite nel contratto.

È necessario notare, infine, come questo passo di pianificazione da un lato possa produrre una retroazione sulla WBS e sulla struttura del prodotto, dall'altro prefiguri/anticipi la pianificazione delle risorse.

Pianificazione delle risorse

È opportuno rilevare che il risultato del passo precedente non è una schedulazione (programmazione operativa delle attività) ma solo una prima importante ipotesi di pianificazione del progetto: sulla base di tale ipotesi il PM elabora il fabbisogno di tecnologia (risorse materiali e Risorse Specialistiche) del progetto.

Nel passo precedente il PM ha scisso le attività fino ad arrivare al livello che lui stesso valuta atomico: ora fa il percorso inverso riaggregandole fino a ritrovare le fasi del modello di ciclo di vita selezionato.

Sfruttando tutta la visibilità che si è conquistato, egli deve definire le tecniche ed i vincoli del processo di sviluppo caratterizzando ciascuna fase dal punto di vista tecnico-produttivo.

L'elaborazione del fabbisogno di tecnologia non parte da zero ma riprende ed approfondisce i risultati dello studio: ci riferiamo, in particolare, alla scelta del ciclo di vita ed alla identificazione delle tecnologie di prodotto e di processo.

Esempio:

fase del modello selezionato	Tecniche e vincoli di sviluppo
Disegno	Tecniche di disegno dei dati Tecniche di disegno delle funzioni
Codifica e test	Tecniche di programmazione Linguaggi di programmazione Software di base Software d'ambiente Tecniche di test

Tabella 9 – Relazione tra fasi del ciclo di vita del prodotto e tecniche e vincoli di sviluppo

Da questo punto in poi l'analisi si sdoppia e prosegue su due linee:

1. individuazione delle risorse materiali e degli strumenti tecnici (tool) da allocare nell'ambiente di lavoro per ciascuna fase – per esempio: tool di disegno, tool di sviluppo, tool di test, stazioni di lavoro individuali, impianti di calcolo, impianti di telecomunicazione;
2. individuazione di numero e tipo delle Risorse Specialistiche cui affidare i compiti individuali di fase – per esempio: due analisti esperti di progettazione banche dati, tre programmatori esperti in ambiente "JAVA".

La pianificazione delle risorse deve indicare i requisiti di tutte le risorse da allocare sul progetto, sia quelle disponibili all'interno dell'organizzazione, sia quelle disponibili nell'offerta dei Fornitori qualificati.

Caso di studio

Rossi individua i profili delle Risorse Umane da coinvolgere nel progetto attraverso una tabella del tipo:

		Project Management		UML		Unix		DB2		Linguaggio C		Apache web		Tools di testing		Microsoft Project		Fatturazione		SCM CYS		Office	
		M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L
1	Project Manager	60	3	12	1	12	1	6	1	12	2				12	1	12	1	3	1	3	1	
2	Team Manager	24	1	12	1	12	1	6	1	12	2				12	1	12	1	3	1	3	1	
3	Analista			36	3	12	1	12	1	12	2						36	3	3	1	3	1	
4	Progettista			36	3	12	1	12	1	12	2						36	3	3	1	3	1	
5	Programmatore			12	1	12	1	12	1	12	1	12	1	12	1		12	1	3	1	3	1	
		84		108		60		48		60	12	12	24		108		15		15				

M=Mesi di esperienza

L=Livello 1= Utente, 2=Applicativo, 3=Esperto

Figura 20 - Profilo delle risorse necessarie con valutazione del grado di conoscenza/competenze.

Previsione dei costi

Il risultato della pianificazione delle risorse è una pietra miliare del cammino di pianificazione: è opportuno che il PM si fermi, raccolga le informazioni disponibili e li valuti insieme con la Direzione per capire il grado d'allineamento della pianificazione con gli obiettivi preassegnati.

Il riesame congiunto dei fabbisogni e la condivisione finale del piano (dei costi) d'acquisizione delle risorse è la maniera giusta di interagire con la Direzione, tanto per approfondire cosa fare (il prodotto di progetto – ambito), quanto per definire come farlo (il processo di progetto).

L'espressione "come fare il progetto", in questa interazione con la Direzione, ha un significato limitato alla definizione della configurazione di costi con cui conseguire la prestazione attesa dal Cliente e, di conseguenza, degli obiettivi di margine del progetto.

Per definire la configurazione dei costi del progetto, il PM deve elaborare gli obiettivi d'impiego delle risorse fino a definire i requisiti di ciascuna risorsa; per tale fine egli deve significativamente interagire con la politica dell'organizzazione in relazione a:

- Gestione della tecnologia
- Sviluppo delle Risorse Umane
- Approvvigionamenti
- Gestione delle infrastrutture.

Direzione e PM devono assumere decisioni che influenzano l'intero progetto.

In effetti, decidere su

- quali parti del prodotto costruire in casa e quali altre acquistare già fatte
- quali Fornitori coinvolgere sul progetto
- quali strumenti adoperare nelle lavorazioni
- quali Persone candidare per la composizione del gruppo di lavoro etc.

significa prefigurare quale grado di qualità di prodotto fornire al cliente, con quale produttività condurre il processo, quali rischi correre consapevolmente.

Sviluppo della schedulazione

Condivisa/approvata la pianificazione delle risorse con relativa previsione di costi, il PM è in grado di far evolvere il fabbisogno di tecnologia in una più articolata definizione del gruppo di lavoro (squadra di progetto).

Dall'ipotesi d'impiego e dai requisiti delle Risorse Specialistiche, il PM ipotizza assegnazioni di attività a ciascuna Persona, peraltro non ancora identificata dal punto di vista anagrafico.

Per formulare una prima schedulazione al PM basta ripercorrere i passi della pianificazione aggiornandola in funzione dell'ipotesi di squadra:

- Definizione delle attività e decomposizione del processo fino a definire i compiti individuali - caratterizzazione del compito attraverso input ed output;
- Previsione per ogni compito di un impegno, una durata, una data d'inizio ed una data di fine;

Capitolo 3 - Pianificazione del progetto software

- Definizione del fabbisogno di tecnologia – obiettivi d’impiego delle risorse;
- Previsione dei costi d’acquisizione delle risorse – requisiti delle risorse.

L’aggiornamento si basa essenzialmente sulla ri-assegnazione dei compiti rispettando i requisiti della Persona ipotizzati nella previsione dei costi.

La pianificazione comincia ad assumere una sua specifica fisionomia vicina all’aspetto del piano finale.

Tutte le decisioni importanti sono state prese.

Il PM ha trasformato un obiettivo di business iniziale (ricavi, costi e margini) in un’ipotesi operativa (ambito, tempi, costi, prestazioni) con un budget definito: disponiamo di un piano di gestione del processo di sviluppo e possiamo dare a questo fondamentale risultato il nome di **pianificazione di base**.

È necessario completare l’opera attraverso la definizione di un quadro di riferimento che consenta di strutturare l’impegno già previsto in termini di controllo di processo al fine di garantire al Cliente un livello predeterminato di prestazione (gestione del prodotto da sviluppare).

Un tale completamento può anche essere visto come un corollario tecnico della pianificazione di base e per definirne i confini abbiamo adottato come ulteriore riferimento la definizione dei processi chiave del livello 2 del CMM [CMMPAULK].

Key Process Area CMM Level 2	processi della norma ISO/IEC 12207:1995	prescrizioni della UNI EN ISO 9001:2000
Software Requirements	5.3.4 analisi dei requisiti software	7.2 processi relativi al cliente
Software Project Planning	5.2.4 pianificazione	7.3.1 pianificazione della progettazione e dello sviluppo
Software Project Tracking And Oversight	5.2.5 esecuzione della fornitura e controlli 6.4.2 verifica 6.6.2 riesame della gestione del progetto	7.3.4 riesame della progettazione e dello sviluppo 7.3.5 verifica della progettazione e dello sviluppo 7.3.7 tenuta sotto controllo delle modifiche della progettazione e dello sviluppo
Software Subcontract Management	5.1 processo di acquisizione	7.4 approvvigionamento
Software Quality Assurance	6.7 processo di audit 7.3.2 valutazione del processo di sviluppo F.2.10 valutazione del prodotto	8.2.2 verifiche ispettive interne 8.2.3 monitoraggio e valutazione dei processi 8.2.4 monitoraggio e

Capitolo 3 - Pianificazione del progetto software

	6.8 processo di risoluzione dei problemi 6.3 processo di assicurazione qualità	misurazione dei prodotti 8.3 tenuta sotto controllo dei prodotti non conformi 8.5.2 azioni correttive
Software Configuration Management	6.2 processo di gestione della configurazione	7.5.3 identificazione e rintracciabilità

Tabella 10 – Comparazione tra CMM e norme ISO

Il CMM definisce il livello 2 di un'organizzazione come il livello ripetibile:

- Politiche e procedure per la gestione di un progetto software sono definite
- La gestione di un nuovo progetto è basata sull'esperienza di progetti simili conclusi
- Le buone pratiche consolidate in un progetto sono identificate ed estese ai nuovi progetti rispettando le specificità dei casi diversi
- Pianificazione e tracciamento di un progetto sono processi stabili e applicabili a qualunque altro progetto.

La tabella propone al lettore una possibile trasformazione del riferimento CMM verso le indicazioni della normativa di settore e di sistema.

Rossi ha effettuato una pianificazione delle attività, con date, risorse e costi. Nelle tabelle seguenti sono dettagliate rispettivamente la distribuzione dell'effort, le risorse impiegate e la durata prevista per fase, la tipologia delle risorse e la relativa quantità inoltre la previsione dei costi per i 20 periodi di controllo.
L'effort totale (7040 ore) corrisponde a 40 mesi uomo con un mese uomo di 22 giorni lavorativi.

Fase	% Distribuzione effort	Giorni/ persona	Risorse Impiegate	Durata Prevista (gg)
Requisiti	20%	176	6	29
Disegno	20%	176	6	29
Implementazione	40%	352	16	22
Test	20%	176	10	18
				98

Capitolo 3 - Pianificazione del progetto software

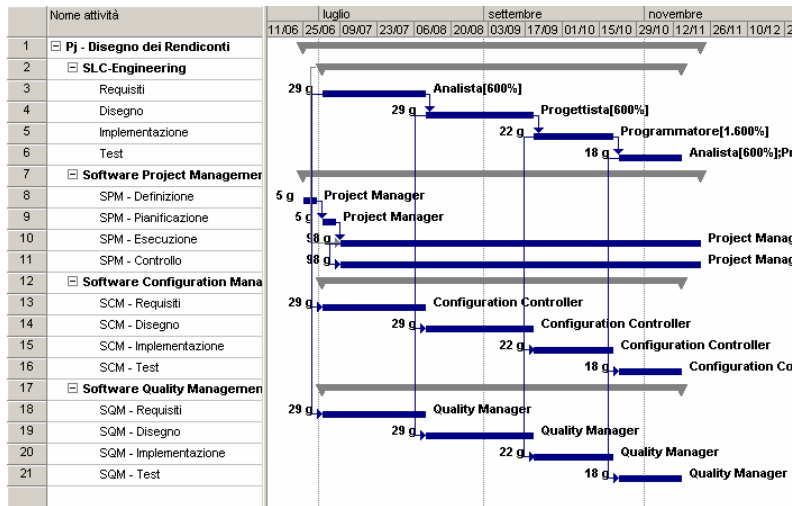


Figura 21 - Risorse impiegate per il progetto "Disegno dei rendiconti"

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2/7	9/7	16/7	23/7	30/7	6/8	13/8	20/8	27/8	3/9	10/9	17/9	24/9	1/10	8/10	15/10	22/10	29/10	5/11	12/11
40	1200	1250	2000	2100	2200	2300	2500	3500	4000	4100	4200	4250	4500	5000	5700	5800	6000	6500	7000

Figura 22 - Previsione di budget (effort in ore/uomo) per i 21 periodi di controllo del progetto "Disegno dei rendiconti", il 21° è pari al totale dell'effort previsto, cioè 7040

Pianificazione della qualità

La pianificazione della qualità ha lo scopo di assicurare che le prestazioni (consegne del progetto) siano adeguate alle attese del Cliente. A tale fine è necessario prevedere sia l'adozione di opportuni standard sia le attività necessarie a controllare la conformità di prodotti e processi di progetto agli standard adottati. Di seguito proponiamo due livelli di pianificazione della qualità.

Sviluppo del piano dei controlli

Abbiamo trovato una prima definizione dei controlli (verifiche e validazioni) nel modello di ciclo di vita, poi li abbiamo dimensionati in funzione della valutazione iniziale del rischio di progetto ed infine li abbiamo sistemati sull'asse dei tempi.

Lo sviluppo del piano dei controlli serve anche per mettere a fuoco i ruoli (PM e Progettisti) nel tracciamento e nella sorveglianza del progetto.

Il PM parte da una situazione esemplificata dalla seguente tabella:

Fase	Attività di fase	Prodotti di fase	Controlli di fase
Fase precedente			
Costruzione dei componenti	Codifica e test del componente 1 Codifica e test del componente 2	componente 1 testato componente 2 testato	Riunione di verifica
Fase successiva			
...			

Tabella 11 – Relazione tra fasi/attività/prodotti e controlli

Lo sviluppo del piano dei controlli di prodotto deve essere orientato all'efficacia delle relazioni tra il meccanismo di controllo scelto (per es. il processo di verifica), gli input al meccanismo (per es. i rapporti dei test), gli output dal meccanismo (per es. il verbale di verifica) e soprattutto i criteri che saranno adoperati per decidere sul consolidamento della fase (accettazione dei prodotti di fase).

Il discorso va completato risalendo dal prodotto al processo. Per analogia la situazione di partenza può essere esemplificata dalla seguente tabella:

Fase	Fattori chiave del processo	Controlli sui fattori
...		
Fase preceden.		
Costruzione dei componenti	Specialista 1 Specialista 2	Riunione di riesame
Fase success.		
...		

Tabella 12 – Relazione tra fasi/Fattori chiave del processo e controlli sui fattori

Anche lo sviluppo della pianificazione dei controlli di processo deve essere orientato alle relazioni tra il meccanismo di controllo scelto (per es. il processo di riesame), gli input al meccanismo (per es. lo stato di avanzamento del progetto ed una richiesta di modifica ad un requisito in lavorazione), gli output dal meccanismo (per es. il verbale di riesame) e soprattutto i criteri che saranno adoperati per decidere sulla prosecuzione della fase (ri-configurazione delle risorse, ri-assegnazione dei compiti e ri-lavorazioni).

Rossi definisce tutte le attività di controllo utilizzando un piano di verifiche e validazione basato sugli standard ESA (SVVP).

Sviluppo del piano di monitoraggio

L'ente qualità dell'organizzazione offre al PM un servizio di monitoraggio del progetto basato sul punto 8.2 (monitoraggi e misurazioni) della norma **UNI EN ISO 9001:2000**.

L'obiettivo del servizio di monitoraggio è facilitare le Parti interessate del progetto (Organizzazione, Cliente, Utenti e Fornitori) a condividere i risultati intermedi e finali del progetto rispettando gli accordi sottoscritti:

- La qualità richiesta – i requisiti – richiesta d'offerta e ordine
- La qualità offerta – le specifiche di prodotto - offerta
- La qualità sviluppata – le prestazioni del prodotto – collaudo
- La qualità percepita – l'uso del prodotto – validazione.

Lo sviluppo del piano di monitoraggio **[CIRAIPA]** è tutto incentrato sulla massimizzazione della qualità uscente dal progetto ad un dato livello di costo.

L'ente qualità e PM devono definire e condividere tipo e numero di misure su prodotto e processo di progetto.

L'elaborazione del monitoraggio della qualità uscente va fatta quanto prima a sostegno dell'intero processo di fornitura:

- Monitoraggio della preparazione dell'offerta
- Monitoraggio del contratto
- Monitoraggio della pianificazione
- Monitoraggio della esecuzione della fornitura
- Monitoraggio della consegna.

Lo sviluppo del piano deve indicare:

- l'oggetto della misura – il prodotto o il processo di progetto su cui si effettua la misura
- il metodo di misura
- i valori attesi per il tipo di misura
- il metodo di rappresentazione/comunicazione dei risultati elementari
- il metodo di aggregazione dei risultati elementari – verifiche ispettive sul progetto e rapporti di monitoraggio.

Il servizio di monitoraggio *può* (in alcuni casi *deve*) essere affidato in outsourcing ad un ente esterno all'organizzazione.

Rossi definisce tutte le attività di monitoraggio utilizzando un piano basato sugli standard ESA (SQAP).
--

Sviluppo del piano degli acquisti

Una buona parte del processo di acquisizione per il progetto è stata precedentemente consolidata; in particolare abbiamo definito:

- gli obiettivi d'acquisto – perché
- i requisiti d'acquisto – cosa
- la disponibilità a spendere – budget
- la sistemazione degli eventi d'ingresso (dei prodotti acquistati) sull'asse dei tempi – schedulazione.

Lo sviluppo del piano degli acquisti è tutto incentrato sulla massimizzazione della qualità entrante nel progetto ad un dato livello di costo.

Per ogni acquisto significativo per la qualità del progetto è necessario che siano definiti prove controlli e collaudi in ingresso:

- Per beni e servizi standard il PM si riferirà alle procedure aziendali;
- Per beni e servizi specifici (ad hoc) per il progetto il PM definirà i relativi meccanismi di controllo.

L'elaborazione dei controlli sulla qualità entrante va fatta quanto prima a sostegno dell'intero processo di acquisizione la cui struttura è tipicamente la seguente:

- preparazione della richiesta d'offerta
- ricezione delle offerte, selezione del Fornitore ed emissione dell'ordine
- monitoraggio del fornitore
- accettazione e completamento.

Lo sviluppo della pianificazione degli acquisti presenta prevedibili analogie con l'approccio ai controlli delineato precedentemente

Pianificazione acquisto	Sottosistema SW in subappalto
Scelta del meccanismo di controllo	Test di qualificazione
Input al meccanismo	Piano di test ed oggetto da testare
Output	Evidenze del test ed eventuali segnalazioni di errore
Criteri di uscita	Accettazione del prodotto in ingresso
Criteri di sorveglianza	Integrazione/utilizzazione nella realizzazione del prodotto

Tabella 13 – Relazione tra la Pianificazione di acquisto e le attività.

Caso di studio

Rossi integra la pianificazione delle risorse e la pianificazione dell'organizzazione di progetto, definisce gli acquisti ed elabora il seguente piano:

Capitolo 3 - Pianificazione del progetto software

FASE DEL CICLO DI VITA	TIPO DI ACQUISTO	DESCRIZIONE E ACQUISTO	CONTROLLO IN ENTRATA	ENTE RESPONSABILE	COLLAUDO DI VALIDAZIONE	Controlli sui fattori
Collegamento temporale alle attività schedate da assegnare come compito individuale	Servizio di consulenza specializzato per la costruzione di componenti software	300 ore persona skillata in ambiente java.....per la costruzione del componente 123	Meccanismo standard - colloquio per la valutazione delle competenze	Capo programma - responsabile Capoprogetto - collabora	Meccanismo standard - valutazione della prestazione	
Collegamento temporale alle attività schedate di integrazione e test di sistema	Subappalto per la costruzione di un Sottosistema di software applicativo da integrare nel progetto	Applicazione software per la gestione di secondo le specifiche dei requisiti sw DT/SRD/...	Meccanismo ad hoc - piano di test di sottosistema	Capo progetto	Meccanismo ad hoc - piano di test di sistema	
Fase di esecuzione	Prodotti informatici e telematici per il processo di sviluppo	Ambiente integrato di sviluppo visual age for java - 3 licenze	Meccanismo standard - procedura di installazione	Supporto sistemistico - responsabile Capoprogetto - collabora	Non applicabile	Riunione di riesame
Fase di esecuzione	Prodotti informatici e telematici per il processo di sviluppo	PC 400 MHZ, RAM 256 MB, HD 20 GB, Monitor 17" - 3 stazioni di lavoro	Meccanismo standard - procedura di installazione	Supporto sistemistico - responsabile Capoprogetto - collabora	Non applicabile	
Fase di esecuzione	Prodotti informatici e telematici per il processo di sviluppo	Collegamento veloce.....	Meccanismo standard - procedura di installazione	Supporto sistemistico - responsabile Capoprogetto - collabora	Non applicabile	

Tabella 14 – Generico piano di acquisti per il caso di studio in esame

Dall'esempio risulta particolarmente evidente come il PM debba saper interpretare il ruolo del "Cliente".

Piano di sviluppo dell'organizzazione di progetto

La definizione della struttura organizzativa di progetto riassume i risultati dello sforzo di pianificazione nell'area della gestione delle Risorse Umane. Il PM identifica le persone assegnate/candidate alla squadra, ne documenta le caratteristiche, definisce i ruoli di progetto, delimita le responsabilità, imposta i rapporti reciproci.

Caso di studio
 Rossi ha definito la struttura organizzativa del progetto attraverso la seguente tabella:

Figura professionale	Compiti	Responsabilità	Risponde a
Project Manager	Interagirà con l'utente, acquisirà i requisiti ad alto livello e ne valuterà congiuntamente con i team manager l'impatto sul progetto (tempi, costi, qualità, rischi, risorse)	Gestione dell'intero progetto in termini di: risorse risultati attesi dal cliente e dalla Direzione conformità agli standard	Capo area
Technical Leader - analisti e progettisti	Analizzerà i requisiti del progetto. Supporterà il Team Manager nell'analisi di impatto e nella stima dell'impegno per ciascun requisito e sue successive modifiche. Parteciperà alla raccolta dei requisiti utente, alle riunioni interne con il Project Manager sullo stato del progetto e le criticità Supporterà il PM nella gestione dei problemi di progetto	Analisi e disegno dei componenti software in conformità con i requisiti stabiliti e le esigenze d'uso. Pianificazione, progettazione, disegno dei collaudi a livello di Integrazione, Sistema e Utente. Gestione della conoscenza e della competenza di progetto a favore dei Team Member e del lavoro loro assegnato dai Team Manager	Project Manager
Team Manager	Coordinerà le risorse assegnategli. Fornirà le stime al Project Manager Controllerà l'avanzamento periodico del piano informerà il Project Manager sullo stato delle attività Controllerà che il software rilasciato venga acquisito dal cliente Gestirà le anomalie di competenze della propria area (valutazione, analisi delle priorità, assegnazione incarichi,	Gestione del processo di sviluppo dei componenti software appartenenti ad uno o più rami o fasi applicative analisi di impatto dei requisiti utente che insistono sui rami applicativi assegnatigli. Pianificazione di dettaglio delle attività di competenza. Sorveglianza sulla esecuzione delle attività di competenza. Controllo dei risultati delle	Project Manager

Capitolo 3 - Pianificazione del progetto software

	controllo interventi). Parteciperà alla gestione dei requisiti utente. Parteciperà alle riunioni interne con il Project Manager sullo stato del progetto e le criticità In caso di necessità farà opera di supplenza delle responsabilità assegnate originariamente a Technical Leader e Team Member	attività di competenza. Gestione della configurazione del software. Gestione delle relazioni interne al proprio gruppo di lavoro (ruoli e skill).	
Team Member - programmatori	Costruirà i componenti di dettaglio. Gestirà le tecnologie di costruzione (disegno, codifica e test). Gestirà la qualità del componente assegnatogli. Prenderà in carico dal Team Manager le anomalie sollevate durante tutte le fasi (Esercizio, Sviluppo, Collaudo Utente). Fornirà al Team Manager le soluzioni ai problemi assegnatigli. Supporterà il test di integrazione.	Disegno e codifica del componente. Pianificazione, progettazione, disegno dei collaudi a livello di componente elementare (test unitario).	Team Manager

Tabella 15 – Struttura organizzativa del progetto

Per quanto riguarda le competenze tecnologiche si fa riferimento alle definizioni della pianificazione delle risorse.

Dopo la definizione della struttura l'organizzazione del progetto prosegue verso la pianificazione della logistica dei gruppi di lavoro e le relative predisposizioni degli strumenti per lo svolgimento dei compiti: il PM collabora con la Funzione aziendale responsabile della gestione delle postazioni di lavoro e completa il piano di sviluppo dell'organizzazione attraverso il piano degli acquisti.

Per tutta la durata del progetto il PM deve seguire lo sviluppo professionale delle Risorse Umane coinvolte nella squadra.

Uno dei suoi compiti è ottenere la massima prestazione possibile dalle singole Persone e dalla squadra nel suo complesso.

Non è detto, a priori, che le Persone assegnate/candidate alla squadra abbiano caratteristiche (bagaglio di conoscenze, competenze, attitudine a stare in gruppo) tali da rispondere ai requisiti stabili e da consentire il compimento del cammino comune.

Per mettere in grado ciascun Candidato di partecipare pienamente al lavoro di progetto il PM deve impegnarsi nello sviluppo del team:

- stimare punti di forza e di debolezza
- Predisporre un piano a breve termine di miglioramento di conoscenze e competenze

Tale processo di progetto rimane distinto da altri processi aziendali di medio lungo periodo generalmente a carico di una specifica funzione aziendale (Gestione delle Risorse Umane, Personale).

Il processo di progetto si implementa sulla base dei rapporti professionali e personali che s'instaurano tra un Componente ed il Leader della squadra.

Per ogni area di conoscenza dell'Ingegneria del software significativa per il progetto il PM deve individuare i punti di forza e di debolezza.

La rilevazione dei punti di forza consente al PM di impiegare al meglio le competenze disponibili evitando così di trovarsi con Persone demotivate e scontente per sotto utilizzo.

La rilevazione dei punti di debolezza consente al PM di predisporre un piano di miglioramento individuale evitando fallimenti nell'esecuzione di un compito.

La rilevazione, in fase di pianificazione, avviene attraverso la raccolta delle informazioni sui candidati rese disponibili dal sistema informativo aziendale.

A partire dalla definizione della situazione iniziale di conoscenze e competenze il PM predisporre il piano di formazione.

Caso di studio

Rossi pianifica la formazione sia delle persone già assegnate sia dei candidati e struttura il piano di sviluppo personale di breve periodo attraverso la seguente tabella.:

Risorsa	Argomento del corso	Durata	Luogo	Data	Ente erogatore
Rossi...	UML	3g	Roma	1-1-2001	Esterno
Bianchi	UML	3g	Roma	1-1-2001	Esterno
Verdi	UML	3g	Roma	1-1-2001	Esterno
Risorsa 3	C	2g	Napoli	1-1-2001	Interno
Risorsa 4	C	2g	Napoli	1-1-2001	Interno
Risorsa 5	C	2g	Napoli	1-1-2001	Interno
Risorsa 6	C	2g	Napoli	1-1-2001	Interno
Risorsa 7	C	2g	Milano	1-1-2001	Interno

Tabella 16 – Piano di sviluppo personale per il caso di studio in esame

Il processo di formazione sul campo in affiancamento a persone esperte (training on the job) non è stato pianificato nelle precedente tabella perché sarà considerato successivamente nelle strategie di mitigazione del rischio.

Pianificazione della configurazione

Come ipotesi di lavoro prevediamo che l'organizzazione abbia definito e mantenga attivo un processo di gestione della configurazione che risponda alle prescrizioni del punto 7.5.3 (identificazione e rintracciabilità) della norma **UNI EN ISO 9001:2000**.

Come risultato delle precedenti attività di pianificazione il Capo-progetto dispone di:

- definizione della struttura del prodotto
- dimensionamento del controllo.

Lo sviluppo del piano di gestione della configurazione è tutto incentrato sulla struttura fisica del prodotto software per garantire nel tempo integrità, consistenza, aggiornamento, rintracciabilità del prodotto di progetto.

Il piano deve consentire uno scambio ordinato di oggetti software tra gli enti che cooperano nello sviluppo fino alla consegna.

Per il sistema di gestione della configurazione possiamo ragionevolmente ipotizzare tre livelli di librerie:

- la libreria dello Specialista con i risultati del compito individuale
- la libreria di base del Capo-progetto con i risultati dell'integrazione dei componenti nei (sotto) sistemi
- la libreria statica dell'organizzazione dove si conservano i prodotti consegnati al cliente.

Lo sviluppo del piano deve indicare tutte le baseline previste per il progetto e per ciascuna baseline:

- nome
- descrizione
- momento di costituzione e di consolidamento
- struttura.

La gestione della configurazione deve continuare anche dopo la consegna del prodotto di sviluppo per consentire uno scambio ordinato di oggetti software tra gli enti che cooperano nella manutenzione e nell'esercizio fino alla dismissione del prodotto. Il cliente può riservarsi la responsabilità della gestione della configurazione fin dall'inizio del progetto: in questo caso il piano si riduce ad un raccordo operativo tra progetto ed un processo esterno all'Organizzazione.

Sviluppo del piano di gestione dei rischi

Nella fase di definizione del progetto il PM ha identificato in via preliminare il rischio globale di progetto al fine di dimensionare il controllo e di offrire una base informativa affidabile alla Direzione, che si assume l'onere del rischio decidendo di proseguire.

In pianificazione il PM poi suddivide il rischio globale in due grandi categorie:

- Rischi ad alto livello
- rischi funzionali.

Il PM trasferisce la gestione dei principali rischi individuati ad alto livello (es.: acquisizioni di Tecnologia, dimissioni di Risorse chiave, ...) alla Direzione e mantiene la responsabilità della gestione dei rischi funzionali, cioè rischi individuati nell'ambito delle funzioni/prestazioni da realizzare nel progetto identificati.

I rischi funzionali sono eventi possibili e non previsti che portano conseguenze negative sulla qualità del prodotto da consegnare e/o del relativo processo di produzione. La gestione del rischio è l'insieme delle azioni preventive e correttive atte a ridurre gli effetti degli eventi rischiosi al fine di centrare gli obiettivi di progetto.

Il piano di gestione dei rischi è il punto di arrivo di un processo strutturato nelle seguenti attività:

- Identificazione dei rischi
- Analisi dei rischi identificati
- Pianificazione dei rischi analizzati

Identificazione dei rischi

Quest'attività consiste nell'identificare i rischi relativi all'attività progettuale - altri rischi che non rientrano in tale categoria saranno sottoposti alla Direzione che provvederà a gestirli.

Il SEI (Software Engineering Institute) ha prodotto un questionario tassonomico per aiutare i PM ad identificare i rischi software.

Il questionario contiene un elenco di domande organizzate per:

- Classe di rischio - (Ingegneria del Prodotto, ...)
- Elemento di rischio - (Requisiti, ...)
- Attributo del rischio - (Stabilità,.....)

Se prendiamo ad esempio per Ingegneria del Prodotto la classe di rischio e i Requisiti come elemento del rischio e la Stabilità come attributo, le domande che aiutano ad identificare i rischi sono le seguenti :

Capitolo 3 - Pianificazione del progetto software

- Vi sono interfacce esterne che stanno cambiando ?
- Se i requisiti non sono stabili vi sono effetti sul test del sistema ?
- Se i requisiti non sono stabili vi sono effetti sul disegno del sistema ?
- Se i requisiti non sono stabili vi sono effetti sulla schedulazione del sistema ?
- I requisiti sono stabili ?
- Se i requisiti non sono stabili vi sono effetti sulla integrazione del sistema ?
- Se i requisiti non sono stabili vi sono effetti sulla qualità del sistema?

Ragionando ad esempio sulla prima domanda, supposto che il nostro software debba interagire con altri sistemi, è opportuno identificare il rischio relativo al fatto che una delle interfacce del nostro software deve interagire possa cambiare. Il questionario tassonomico del SEI contiene circa 180 domande.

Supponiamo di aver identificato i seguenti rischi:

- Cambio delle interfacce esterne
- Dimissioni di risorse chiave
- Requisiti tecnicamente difficili da implementare
- Personale sotto stress

Di seguito analizziamo in dettaglio i rischi identificati.

Analisi dei rischi

L'analisi dei rischi consiste essenzialmente nel valutare l'impatto che il rischio può avere nel caso questo si dovesse verificare e stabilire per ognuno di essi la priorità.

Prendiamo come esempio il rischio 3 e descriviamolo in dettaglio.

Id Rischio	3		
Descrizione	Personale sotto stress		
Classe	Ambiente di sviluppo		
Elemento	Ambiente di lavoro		
Attributo	Morale		
Sintomi	Calo della produttività, frequenti ritardi e assenze, commenti negativi		
Identificato da:	Project Manager	Data identificazione	Periodi di Pianificazione
Possibile impatto dal	Fine realizzazione	al	Poco prima delle consegne
Probabilità	Altissima=5		

Impatto	Qualità di prodotto scadente, Slittamento dei tempi di consegna, Richiesta di uscire dal progetto
Impatto tempi (gg)	22
Impatto costi (Valuta)	1000
Priorità	Altissima
Strategia da adottare	

Figura 23 – Descrizione di dettaglio di un rischio identificato per il caso di studio in esame.

Al termine di questo lavoro mediante l'utilizzo di un algoritmo (ad esempio per ogni rischio si effettua il calcolo tra la probabilità, Impatto tempi, Impatto costi e priorità) che stabilisce un ordine per ognuno si avrà una lista di rischi da gestire. McConnell suggerisce di gestire solo i 10 rischi più importanti [**RADMCCO**].

Per i rischi presenti nella lista sarà definita la contromisura (piano contingente di gestione del rischio), tutti gli altri non saranno presi in considerazione:

non c'è senso pratico nel proposito di gestire tutti gli eventi negativi che possono verificarsi.

Pianificazione dei rischi

Per ogni rischio analizzato bisogna scegliere la strategia. Sono possibili le seguenti strategie:

- Trasferire
- Mitigare
- Accettare

Trasferire il rischio significa individuare qualcun altro che si assuma l'onere della gestione del rischio.

Il trasferimento del rischio implica il trasferimento della responsabilità di gestione attraverso la stipula, quando possibile, di un contratto d'assicurazione.

La mitigazione del rischio implica la pianificazione e l'esecuzione di attività tendenti a ridurre la probabilità che il rischio si verifichi.

L'accettazione del rischio implica solo la definizione di un piano contingente di azione da intraprendere quando accade l'evento temuto.

Al termine di questa attività si produce un documento contenente tutti i rischi che si desidera gestire.

Il piano di gestione dei rischi può essere un documento a parte o essere integrato nel piano di progetto come sezione.

Caso di studio

Rossi identifica i rischi funzionali e li rappresenta attraverso un elenco per priorità. In accordo con McConnell [RADMCCO] si è deciso di pianificare la gestione dei primi 10 rischi.

Lista dei Rischi									
ID	Data creazione	Descrizione	Priorità	Probabilità	Impatto potenziale	Strategia utilizzata	Azioni da intraprendere	Piano di contingenza	Stato
1.		Mancanza di competenze nell'utilizzo delle primitive grafiche Java per la realizzazione sistema di gestione visuale del layout dei rendiconti telefonici	ALTA	0.5	Possibilità di annullare il contratto. Riduzione dell'ambito del prodotto	Mitigare	Realizzazione di un piccolo prototipo che utilizzi le primitive grafiche in Java	Appaltare l'attività a società specializzate	APERTO
2.		Requisiti sul Layout non chiari da parte dell'utente	ALTA	0.3	Software non corrispondente ai bisogni dell'utente	Mitigare	Realizzazione di un prototipo di interfaccia con sessioni JAD con l'utente		APERTO
3.									

Figura 24 – Lista parziale dei rischi per il caso di studio in esame

Sviluppo del piano di comunicazione

La capacità di comunicare è essenziale nella direzione di progetto. Lo scopo della pianificazione della comunicazione consiste nel prevedere/soddisfare le esigenze informative degli attori del progetto. Il piano includerà i seguenti elementi del processo di comunicazione:

- Oggetto
- Destinatario
- Frequenza
- Formato documento
- Mezzi di comunicazione.

Oggetto

L'Oggetto rappresenta il tipo di informazione da comunicare. Esempi:

- stato del progetto (stato degli avanzamenti, stato finanziario, ecc.),
- una riunione (convocazione, verbale di riunione),
- una nota aziendale (cambi nell'organizzazione aziendale, disposizioni amministrative, ecc.) ,

- una nota di progetto (ingresso o fuoriuscita di una risorsa dal progetto, disposizioni varie, cambio nelle priorità, ecc..),
- documenti di progetto (documenti tecnici, contratto, ecc..),
- informazioni sul Team (eventi particolari, riconoscimenti, anniversari, ecc..).

Destinatario

Il destinatario rappresenta l'ente e/o le persone che riceveranno l'informazione.

La scelta del destinatario viene effettuata studiando il contesto del progetto.

Il contesto del progetto è un diagramma che aiuta a capire quali sono le entità/organizzazioni che entrano in contatto con il progetto direttamente o indirettamente. vedi fig. 22-23.

Inviare l'informazione a persone non interessate oltre ad essere una perdita di tempo crea ambiguità e genera conflitti.

Frequenza

Per frequenza s'intende la frequenza del processo di distribuzione dell'informazione.

Esempio: una spedizione alla settimana, una al mese, una tantum, etc.

Una scelta adeguata della frequenza è importante:

- un'informazione inviata troppo frequentemente potrebbe richiedere un impegno eccessivo per essere esaminata,
- un'informazione inviata di rado potrebbe inficiare la tempestività delle azioni correttive da intraprendere.

Formato del documento

Il formato del documento indica il tipo di struttura fisica utilizzata per contenere l'informazione ad es. MS Word, MS Excel, MS Project, carta, MIME, etc..

Indicando il formato il PM mette in condizione il destinatario di predisporre alla lettura dell'informazione trasmessa.

Mezzi di comunicazione

L'informazione "mezzi di comunicazione" indica con quale mezzo si trasmetterà l'informazione; ad es. posta elettronica, fax, video, teleconferenza, ecc...

Tempi e costi di trasferimento dell'informazione guideranno la scelta del mezzo.

Con l'aiuto del diagramma di contesto, in cui sono individuate le funzioni con cui interagisce il Progetto e i flussi scambiati si sviluppa il piano di comunicazione.

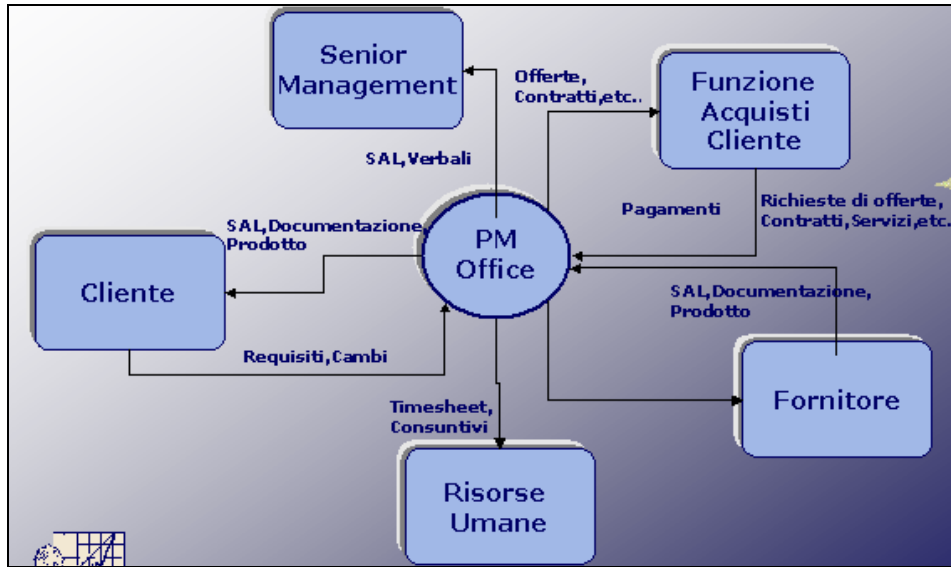


Figura 25 –Generico contesto di progetto

Caso di studio

Rossi affronta il problema della Comunicazione a partire dal contesto del progetto, ovvero analizzando i flussi di informazioni che vengono scambiati tra i vari attori:

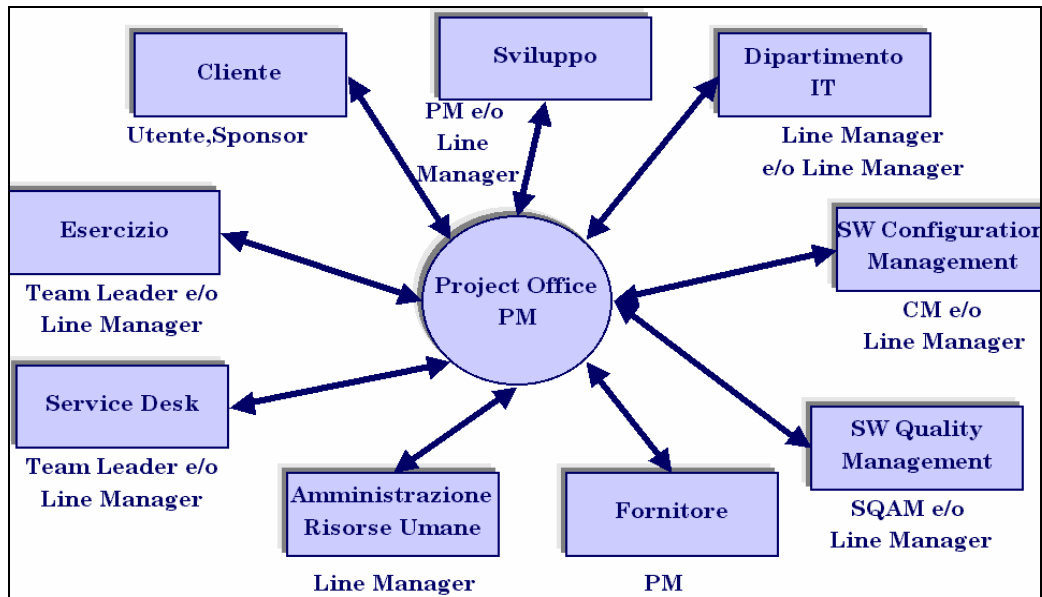


Figura 26 – Contesto di progetto dettagliato

Con riferimento al precedente contesto il PM sviluppa il seguente piano di comunicazione relativo al progetto:

OGGETTO	DESTINATARI	FREQUENZA	FORMATO	MEDIA
Piano di comunicazione	Tutte le parti coinvolte	Alla partenza della fase di esecuzione	MS Word	E-mail
Stato del progetto	Project Manager cliente, Manager Direzione	Ogni 2 settimane. Di venerdì	MS Word	E-mail
Agenda riunione periodica di progetto	Tutti i componenti del team	Venerdì di ogni settimana	MS Word	E-mail
Oggetti di consegna	Project Manager del cliente	Da piano o su richiesta	MS Office	E-mail
Documentazione contrattuale	Project Manager del cliente	Da piano	Cartaceo	Corriere o posta tradizionale
Note tecniche	Software Engineer	Una tantum	MS Office, MIME	E-mail

Tabella 17 – Esempio di piano di comunicazione

Il piano di comunicazione dovrà essere distribuito prima che inizi l'esecuzione del progetto a tutte le parti coinvolte.

La comunicazione riveste un ruolo **fondamentale** nella gestione di un progetto software.

Una comunicazione inefficace comporta ambiguità tra gli attori, conflitti tra le parti, impossibilità di gestire il gruppo: una tale escalation può condurre anche alla paralisi del progetto.

Durante l'esecuzione del progetto vedremo alcuni esempi di come si crea informazione utile e come la si distribuisce.

Riepilogo

La scelta di articolare la pianificazione esecutiva in un primo livello di base e nel successivo livello di dettaglio ha un senso eminentemente pratico. L'approccio scelto consente al Capo-progetto un'anticipazione delle interazioni con la Direzione e con le altre funzioni aziendali con il fine di consentire l'inizio dei lavori con una squadra ridotta per acquisire maggiore competenza prima di approfondire il livello di dettaglio ed impegnare il grosso delle risorse previste. Il meccanismo di aggiornamento del piano (da piano di base a piano di dettaglio) può essere generalizzato: i processi dei gruppi di esecuzione e controllo generano feedback preziosi per smorzare nel tempo l'incertezza inizialmente connessa alla pianificazione del progetto.

Esecuzione del progetto software

- 4.1 Introduzione
- 4.2 Kick-off Meeting
- 4.3 Assegnazione dei compiti individuali
- 4.4 Esecuzione dei processi software
- 4.5 Conclusione del compito individuale
- 4.6 Distribuzione dell'informazione
- 4.7 Sviluppo della squadra
- 4.8 Assicurazione della qualità
- 4.9 Acquisti
- 4.10 Riepilogo

“... a chi non basta l'animo di prender parte a questa battaglia, sarà data licenza di andarsene ,e gli sarà dato anche un salvacondotto e gli sarà messa nella borsa qualche corona per le spese di viaggio: noi non vorremmo morire in compagnia di un uomo che avesse paura di esserci compagno nella morte”
Enrico V - W. Shakespeare

Nella esecuzione del progetto il Capo-progetto avvia-governa-conclude le azioni pianificate per la realizzazione del prodotto di progetto.

Anche questo Capitolo è articolato in due parti.

Nella prima parte l'esecuzione declina il tema dell'integrazione consentendo al Capo-progetto l'ottenimento del prodotto (risultati del lavoro) di progetto.

Il Capo-progetto parte dal piano, riunisce la squadra, annuncia la formazione, assegna i compiti, sorveglia l'esecuzione dei processi di sviluppo software, conclude i compiti assegnati accettando i risultati.

Nella seconda parte l'esecuzione declina qualità, gestione delle Risorse Umane, comunicazione e acquisti consentendo al Capo-progetto il supporto al processo di sviluppo.

Il Capo-progetto facilita i processi principali in corso, distribuisce le informazioni, sviluppa le relazioni (quelle interne alla squadra e quelle tra progetto e parti interessate), assicura la qualità, verifica l'ambito del progetto, sorveglia l'entrata dei prodotti acquistati nel processo di sviluppo.

Introduzione

La pianificazione esecutiva è pronta, le risorse sono disponibili, i ruoli e le responsabilità sono definiti così come la logistica ed il flusso di comunicazione:

può iniziare la realizzazione del prodotto per conseguire gli obiettivi assegnati.

L'esecuzione è la fase di progetto in cui la squadra (organizzazione temporanea messa in piedi ad hoc) sviluppa gli oggetti previsti.

Il PM ha la responsabilità di garantire che il prodotto del progetto rispetti tempi, costi e qualità, con la soddisfazione delle Parti interessate.

Per conseguire gli obiettivi di fase il PM dovrà:

- Comunicare efficacemente alle risorse coinvolte la visione del progetto;
- Facilitare lo sviluppo professionale dei Componenti della squadra;
- Assicurare la qualità di prodotto;
- Comunicare efficacemente alle Parti interessate lo stato del progetto;
- Mantenere la rotta verso gli obiettivi (solo il lavoro richiesto sarà realizzato).

In fase di esecuzione emergono le doti di leader del PM, infatti, egli deve gestire l'ambiente turbolento di progetto ed in particolare:

- Il gruppo di lavoro con le sue dinamiche,
- Il cliente con le continue richieste di modifiche,
- La direzione con le sue pressioni per il rispetto dei tempi ed il contenimento dei costi
- I fornitori con le loro consegne e le relative esigenze contrattuali,
- I conflitti interni ed esterni.

È opportuno notare che nel caso delle *heavyweight methodologies* il mercato mette a disposizione del progetto infrastrutture tecnologiche integrate a supporto dei vari processi componenti (progettazione dei casi d'uso, controllo della produzione del SW, scrittura dei casi di test, produzione di documenti di progetto etc.) del processo di sviluppo software. Si vedano App. E ed App. C per maggiori informazioni

Kick-off Meeting

Nella fase di esecuzione del progetto le Risorse Umane previste per l'impiego sono disponibili e pronte ad iniziare e per iniziare bene è necessaria una riunione di inizio – kick-off meeting.

Il kick-off meeting ha lo scopo di:

- Condividere gli obiettivi del progetto
- Conoscere i Compagni di squadra ed instaurare le necessarie relazioni interpersonali
- Scambiare informazioni e dettagli sul progetto
- Discutere l'approccio alla realizzazione del prodotto di progetto (processi organizzativi e processi di sviluppo).

Il kick-off meeting è importante perché consente al PM di:

- Comunicare la visione
- Porre le basi per la costituzione di un gruppo di successo
- Diffondere un senso comune di appartenenza
- Indicare la strategia che s'intende seguire per arrivare alla meta
- Approfondire il progetto in termini di attori coinvolti, organizzazione, pianificazioni e consegne da effettuare.

Caso di studio

Rossi mediante una comunicazione convoca una riunione di inizio (Kick-off Meeting).

Kick-off				
		Tipo¹ intervento	Responsabile	Durata
1.	Apertura	N	PM Rossi	
2.	Il contratto	N	PM Rossi	
3.	Descrizione del prodotto	N	PM Rossi	
4.	Ambito del progetto	N	PM Rossi	
5.	Valore per l'azienda	N	PM Rossi	
6.	<u>Schedulazione di progetto</u>	N	PM Rossi	
7.	Introduzione dei componenti del Team, ruoli e responsabilità	N	PM Rossi	
8.	Presentazione del cliente	N	PM Rossi	
9.	Elenco dei contatti	N	PM Rossi	
10.	Procedure amministrative	N	PM Rossi	
11.	Diagramma organizzativo	N	PM Rossi	
12.	Piano di comunicazione	N	PM Rossi	
13.	Metodologie da utilizzare	N	PM Rossi	
14.	<u>Standards da utilizzare</u>	N	PM Rossi	
15.	Notebook di progetto	N	PM Rossi	
16.	Domande libere	N	Tutti	
			Totale durata	

¹ D=Decisione, N=Nota

Figura 27 – Agenda utilizzata da Rossi per la convocazione di riunione di kick-off

Rossi apre la riunione ed illustra il progetto in tutte le sue parti spaziando dall'esigenze per cui è nata l'iniziativa alla descrizione del prodotto, dalla organizzazione del lavoro alla comunicazione.

Rossi sfrutta l'occasione della presentazione per motivare le Persone al raggiungimento degli obiettivi facendo risaltare l'importanza della partecipazione di ciascuno al successo dell'impresa e dell'impatto che le caratteristiche del prodotto avranno sulle aspettative dell'utenza.

Assegnazione dei compiti individuali

Il Kick-off meeting è servito a diffondere maggiori informazioni riguardanti il progetto, a costruire/orientare il gruppo di lavoro: subito dopo è necessario assegnare le attività pianificate ai componenti del gruppo di lavoro.

A volte capita di assistere ad un progetto iniziato nel quale uno o più Progettisti non sanno esattamente quello che devono fare oppure sono completamente fermi nell'attesa di poter disporre dei necessari input: un tale progetto parte con il piede sbagliato.

Le cause di una falsa partenza possono essere:

- Pianificazione esecutiva insufficiente
- Requisiti insufficienti o ambigui
- Inesperienza del Project Manager.

Per partire bene il PM deve essere in grado di mettere in movimento **tutta** la squadra prima affidando a ciascuno il proprio compito e successivamente sorvegliando periodicamente i compiti assegnati.

Un Progettista per lavorare ha bisogno delle seguenti informazioni:

- Descrizione dell'attività
- Input per svolgere l'attività
- Output da produrre
- Criteri di partenza
- Data di inizio e fine prevista
- Criteri di terminazione

L'assegnazione dei compiti all'inizio dell'esecuzione, quando possibile, dovrebbe avvenire in una riunione: nel caso di team dispersi geograficamente il PM può assegnare i compiti mediante e-mail o fax o qualsiasi altro sistema in grado di distribuire l'informazione.

Capitolo 4 - Esecuzione del progetto software

Anche nel caso di gruppi di progetto costituiti da sottogruppi distribuiti geograficamente sul territorio, sarebbe auspicabile comunque la presenza nella riunione di assegnazione dei compiti di un rappresentante di ciascun sottogruppo residente fuori della sede di direzione del progetto: il rappresentante, rientrato nella sede distaccata di lavoro, completerà l'opera di assegnazione distribuendo gli incarichi agli altri Progettisti del sottogruppo.

Sebbene far incontrare le persone possa sembrare superfluo e costoso, il sistema delle riunioni è da preferire a tecnologie che non sono ancora in grado di ricreare il clima giusto per la comunicazione interpersonale.

caso di studio

Rossi ha deciso di assegnare il lavoro con comunicazione scritta ed ha registrato le informazioni di assegnazione in un documento strutturato nel seguente modo:

Progetto:	XYZ		
Fase:	Realizzazione componenti applicativi		
Nominativo:	Paolo Bianchi		
ID Attività/Package:	1020	Descrizione:	Aggiornamento Saldo
Data inizio prevista:	1/2/01	Data fine prevista:	15/2/2001
Data inizio effettiva:		Data fine effettiva:	
Input:	Specifiche funzionali, Modello fisico dei dati		
Attivazione	Terminazione attività 1019		
Elenco dei compiti e delle consegne			
	Compito	Consegne	Durata effettiva
1.	Requisiti	N.A.	
2.	Progettazione	N.A.	
3.	Disegno di dettaglio	Specifica di componente	
4.	Verifica e Validazione	Piano di test unitario, Disegno dei test, casi di test	
5.	Codifica componente	Codice sorgente	
6.	Esecuzione test	Rapporti di esecuzione, Rapporti di anomalie	
Stato di avanzamento	% SAL 1° - 3/2/01	% SAL 2° - 7/2/01	% SAL 3° - 12/2/01
Annotazioni varie			

Figura 28 - Descrizione dettagliata di un work-package

Un'alternativa se il PM adotta un tool come MS PROJECT alle comunicazioni di assegnazione potrebbe consistere nell'uso di un report personalizzato del tool.

Nella figura seguente si riporta un esempio di assegnazioni/verifica delle attività mediante l'opzione del tool "Gantt di verifica".

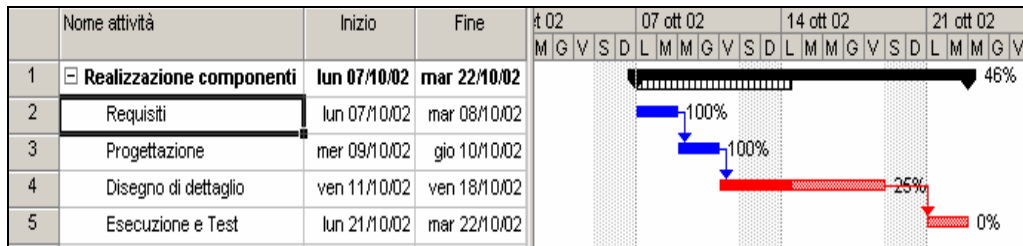


Figura 29 - Esempio di pianificazione Work-package e verifica mediante tools di PM

In mancanza di indicazioni, l'assegnatario è referenziato nel documento di pianificazione.

L'assegnazione non è un atto autoritario ma piuttosto una sorta di piccola trattativa: il Progettista accetta il compito, eventualmente rinegoziando i termini del preventivo, acquisisce gli input e predispone gli strumenti per lo svolgimento del compito e la misura della prestazione.

L'affidamento verbale degli incarichi rende più complicato il tracciamento degli eventi importanti del progetto

Esecuzione dei processi software

Il PM manda in esecuzione i processi software pianificati attraverso l'assegnazione dei compiti individuali.

Prima nella scelta e pianificazione del ciclo di vita e poi nella schedulazione il PM ha definito i processi necessari per realizzare il prodotto previsto nel contratto ad es. al progetto è affidata solo la sequenza disegno di dettaglio – costruzione del componente, oppure è affidata solo la sequenza disegno di architettura – disegno di dettaglio – costruzione del componente – test di integrazione.

Per completezza supporremo che il lavoro (ambito) del progetto richieda l'esecuzione di tutte le attività (ISO/IEC12207:1995) del processo di sviluppo software:

- Definizione del requisito utente
- Analisi dei requisiti software
- Disegno di architettura
- Disegno di dettaglio
- Costruzione del componente
- Collaudo ed integrazione.

Capitolo 4 - Esecuzione del progetto software

I processi componenti del processo di sviluppo (primari) devono essere necessariamente sostenuti dai processi di supporto; tra tutti quelli definiti dalla norma ISO/IEC12207:1995 nel seguito prenderemo in considerazione solo i seguenti:

- Verifica e Validazione
- Gestione della Configurazione Software
- Gestione della qualità

In figura è illustrato il legame tra i vari processi. Nella tabella seguente invece gli output delle varie fasi sia dei processi primari che quelli secondari.

	Documentazione Attività di fase di prodotto	Project Management di processo	Software Verification & Validation	Software Configur- ation Managem- ent	Software Quality Managem- ent
Requisiti	<ul style="list-style-type: none"> • URD • Use Case • SRD • Diagramma delle classi • Diagramma di collaborazione • Diagramma degli stati • Diagramma delle attività 	<ul style="list-style-type: none"> • SPMP/UR • Aggiornamento Gantt • Comunicazioni • Agenda riunione • Verbali di riunione • SPMP/SR 	<ul style="list-style-type: none"> • SVVP/UR • Piano di test • Progettazione test • Disegno casi di test • Procedure di esecuzione test • SVVP/SR 	<ul style="list-style-type: none"> • SCMP/UR • SCMP/SR 	<ul style="list-style-type: none"> • SQMP/UR • SQMP/SR
Disegno Architetturale	<ul style="list-style-type: none"> • ADD • Diagramma delle classi • Diagramma dei package • Diagramma di rete • Manuale utente 	<ul style="list-style-type: none"> • SPMP/AD • Aggiornamento Gantt • Project Status • Comunicazioni • Agenda riunione • Verbali di riunione 	<ul style="list-style-type: none"> • SVVP/AD • Piano di test • Progettazione test • Disegno casi di test • Procedure d'esecuzione test 	<ul style="list-style-type: none"> • SCMP/AD 	<ul style="list-style-type: none"> • SQMP/AD
Implementazione	<ul style="list-style-type: none"> • DDD • Diagramma delle classi dei componenti 	<ul style="list-style-type: none"> • SPMP/DD • Aggiornamento Gantt • Project Status • Comunicazioni • Agenda riunione • Verbali riunione 	<ul style="list-style-type: none"> • SVVP/SR • Piano di test • Progettazione test • Disegno casi di test • Procedure di esecuzione test • SPMP/UR 	<ul style="list-style-type: none"> • SCMP/DD 	<ul style="list-style-type: none"> • SQMP/DD
Test	<ul style="list-style-type: none"> • Piano di test unitario • Progettazione test unitario • Disegno casi di test unitario • Procedure di esecuzione test • Codice sorgente • Rapporti di esecuzione • Rapporti di anomalia • Riepilogo test 	<ul style="list-style-type: none"> • Project Status • Comunicazioni • Agenda riunione • Verbali di riunione 	<ul style="list-style-type: none"> • SVVP 		
Rilascio del software	<ul style="list-style-type: none"> • SRN (Note al rilascio del software) • Manuale di installazione • Manuale utente 	<ul style="list-style-type: none"> • Project History • Comunicazioni • Agenda riunione • Verbali riunione 			

Tabella 18 – Principali artefatti delle fasi di sviluppo software basato su UML

La trattazione successiva descriverà nell'ordine dato i processi primari includendo anche le attività dei processi di supporto.

Non si terrà conto delle strategie di sviluppo definite attraverso la scelta del modello (spirale, incrementale, a cascata, etc..).

Il presente paragrafo non ha minimamente la pretesa di trattare organicamente metodi/strumenti/tecniche del processo di sviluppo – il lettore interessato può riferirsi a SWEBOK; il nostro interesse è concentrato invece nella messa a fuoco delle relazioni tra PM e Progettisti nella fase di esecuzione per i fini della gestione di progetto.

Definizione del requisito utente

Il primo approccio ai requisiti utente c'è stato nella fase di definizione del progetto; ora in fase di esecuzione è possibile assegnare ulteriori compiti di definizione dei requisiti utente. Lo scopo del processo non cambia:

far emergere maieuticamente, raccogliere e tracciare le esigenze dell'utente lungo l'arco della vita del prodotto SW.

La definizione del requisito utente ha in input una descrizione del sistema/prodotto da realizzare in termini di requisiti ad alto livello e come output un documento del requisito utente ed eventualmente il documento dei casi d'uso.

Le pratiche di base sono:

- Raccolta dei requisiti
- Documentazione dei requisiti
- Validazione dei requisiti
- Gestione dei requisiti.

Raccolta dei requisiti

Nella raccolta dei requisiti sono coinvolti principalmente i rappresentanti degli utenti e gli analisti.

Il PM partecipa alle riunioni con gli utenti.

La tecnica dei casi d'uso [**JACOBSON**] porta alla raccolta dei requisiti.

I casi d'uso descrivono le interazioni tra gli attori, cioè gli utilizzatori del sistema/prodotto e le risposte del sistema.

Le informazioni da raccogliere per ogni requisito sono:

- Identificativo
- Classe del requisito (Funzionale, non funzionale, Vincolo)
- Tipo del requisito (Funzionale, Operazionale, Portabilità, Legislativo, Sicurezza del sistema, Interfaccia utente, Interfaccia con altri sistemi, sicurezza, etc..)

- Titolo
- Priorità
- Stabilità
- Caratteristiche (Fattibile, Chiaro, Verificabile, Completo)
- Origine
- Sorgente
- Condizione
- Stato (Proposto, Approvato, Incorporato, Verificato, Implementato)
- Data richiesta
- Richiedente
- Release

Caso di studio

Rossi per facilitare la gestione delle informazioni da raccogliere per ogni requisito ha deciso di utilizzare un'archivio informatico.

The screenshot shows a web-based form for managing requirements. The main section is titled 'Informazioni principali' and contains the following fields:

- ID***: 33
- Protocollo esterno**: (empty)
- Altro prot.**: M21
- Titolo**: Lista movimenti contabili
- Tipo**: Funzionale
- Classe**: No
- Macro ?**:
- ID Macro**: (empty)
- TBD**:
- Priorità**: TBD
- Stato**: APERTO
- Soddisfazione**: 1
- Insoddisfazione**: 1
- Sorgente**: Riunione del 25-05-2001
- Subsystem**: Invoice
- Baseline**: Novembre 2001

Below this section are four tabs: 'Requisito Funzionale', 'Requisito non Funzionale', 'Requisito di Vincolo', and 'Caratteristiche specifiche del requisit'. The 'Requisito Funzionale' tab is active, showing:

- Condizione**: In visualizzazione fattura l'evidenza della scrittura contabile relativa all'abbinamento pagamento giornaliero si deve riferire unicamente al pagamento di riferimento. La stessa logica deve essere prevista per le scritture contabili del bilancio.
- Capacità**: 1
- Velocità**: (empty)
- Accuratezza**: (empty)

At the bottom, there is a record navigation bar showing 'Record: 16 di 157' and a 'zazione Maschera' label.

Figura 30 - Form per la raccolta e la gestione dei requisiti

Una gestione dei requisiti basata su archivio consente la produzione di un report del tipo illustrato nell'esempio seguente.

<u>Descrizione</u>					
Titolo: Lista movimenti contabili:					
Tipo: Funzionale					
Sorgente: Riunione del 25-05-2001					
Data rilascio:					
Richiesto da: AFC/FT			Data richiesta:		30/11/2001
Approvato da: Cliente			Data approvazione:		07/10/2001
<u>Funzionalità</u>					
Condizione: In visualizzazione fattura l'evidenza della scrittura contabile relativa all'abbinamento pagamento giornaliero si deve riferire unicamente al pagamento di riferimento. La stessa logica deve essere prevista per le scritture contabili dell'arrotondamento.					
Capacità: 1					
Velocità:					
Accuratezza:					
<u>Vincoli</u>					
Descrizione: Non Applicabile					
<u>Caratteristiche</u>					
<input checked="" type="checkbox"/> Chiaro					
<input checked="" type="checkbox"/> Verificabile					
<input checked="" type="checkbox"/> Completo					
<input checked="" type="checkbox"/> Accurato					
<input checked="" type="checkbox"/> Fattibile					
<u>Stato del Requisito</u>					
Requisito	Stato	Note allo stato	Data allo stato	Responsabile	
33	Proposto	Proposta nella riunione del 7-10-2001	0/2001 1.28.48	Cliente	
33	Aperto		0/2001 1.30.42	PM Rossi	
33	Check	E' ancora in fase di negoziazione	1/2001 1.30.46	C.Di Leva	

Figura 31 - Esempio di dettaglio di un requisito software

Documentazione dei requisiti

Tutti i requisiti raccolti sono integrati in una forma coerente (es.: documento, tabella di database).

Un indice tipico di riferimento potrebbe essere quello riportato in [ESAGUIDE]

Accettazione dei requisiti

Il documento formalizzato è sottoposto all'utente per approvazione.

Il documento approvato servirà da guida nel processo di sviluppo secondo quanto prescritto nel modello di riferimento (es.: nel modello a cascata per tutta la durata del progetto).

Gestione dei requisiti

Il PM mantiene la responsabilità diretta della gestione dei requisiti e governerà il processo di sviluppo del prodotto in funzione delle priorità date ai requisiti.

In caso di necessità sarà suo compito sia rinegoziare gli accordi con l'utente sia gestire i rischi possibili.

La gestione dei requisiti è un'attività che attraversa l'intero ciclo di vita qualunque sia il modello di riferimento ed include due aspetti fondamentali quali la gestione delle varianti ed il tracciamento dei requisiti.

La gestione dei requisiti è orientata su un obiettivo irrinunciabile: la configurazione stabilita dei requisiti deve **sempre** rispecchiare il prodotto di progetto.

Processi di supporto	Attività specifiche per i requisiti utente
verifica e validazione	<ol style="list-style-type: none">1. Definizione del piano di collaudo (di accettazione) utente2. Progettazione dei casi di test (individuazione del requisito utente da collaudare)3. Disegno dei casi di test utente4. Disegno dei casi di test per i Casi d'Uso (Opzionale).
gestione della configurazione	<ol style="list-style-type: none">1. aggiornamento del piano di configurazione per le fasi successive2. aggiornamento della configurazione con i prodotti di fase3. esecuzione delle modifiche secondo procedure e tool aziendali (CVS, Visual Source Safe)
gestione della qualità	<ol style="list-style-type: none">1. Ispezione del documento del requisito utente2. Ispezione dei casi di test3. Ispezione dei casi d'uso

Figura 32 – Attività dei processi di supporto per la fase dei requisiti utente

Analisi dei requisiti software

Nel processo di analisi e specificazione dei requisiti software il Progettista elabora la proposta di soluzione formulata in fase di definizione di progetto con lo scopo di:

- Identificare **cosa** (requisiti funzionali) deve fare l'applicazione per rispondere alle esigenze dell'utente
- Precisare i confini dell'applicazione e le modalità di interazione con l'ambiente (requisiti di interfaccia)
- Trasformare il quadro dei vincoli d'uso nei requisiti non funzionali
- Definire l'elenco dei requisiti SW
- Elaborare le contraddizioni tra requisiti SW
- Identificare e mantenere il tracciamento tra requisiti utente e requisiti SW.

L'attività di analisi dei requisiti software ha in input il documento del requisito utente ed in output le specifiche funzionali cui possono essere allegati i modelli di analisi (Modello dati, Modelli funzionali nel caso si utilizzi una metodologia strutturata, Modelli ad oggetti nel caso si utilizzi una metodologia ad oggetti).

I Progettisti analisti sono gli attori principali, gli utenti possono essere interessati per dei chiarimenti, il PM controlla le attività ed è coinvolto solo in caso di necessità.

Le pratiche di base sono:

- Definizione del modello logico delle entità/oggetti
- Definizione del modello logico delle funzionalità
- Analisi dei requisiti software non funzionali
- Prototipazione (Prototyping)
- Produzione del documento di specifiche funzionali

Le pratiche di base evidenziano la forte correlazione tra la modellazione concettuale e le successive attività di disegno: l'analisi dei requisiti software anticipa temi tipici dell'architettura tecnologica ed applicativa.

Il Progettista impegnato in questo compito individuale può attingere ad una base di conoscenze condivisa:

La comunità dell'IT ha accettato diversi pattern utili per la risoluzione di alcune classi di problemi. Si veda App. D per maggiori informazioni.

Definizione del modello concettuale dei dati

Il modello concettuale dei dati descrive a livello logico le entità informative (dati) coinvolte nelle funzioni dell'applicazione.

Il mercato offre strumenti informatici di supporto alla definizione del modello concettuale dei dati. Si veda in App. E per maggiori informazioni.

La modellazione dei dati mediante oggetti può essere utilizzata in accordo con la configurazione tecnologica del progetto.

Ovviamente parliamo di oggetti persistenti, cioè di oggetti dati, oggetti memorizzati su memorie di massa.

Definizione del modello delle funzioni

La definizione del modello porta alla graduale e completa comprensione delle funzionalità da sviluppare.

Tra le tecniche disponibili per l'uso segnaliamo:

- Diagrammi di decomposizione funzionale
- Diagrammi di flusso dati

Il primo **[YOURDON]** consente di decomporre le funzionalità fino ad un livello oltre il quale non è più scomponibile (processo).

Il secondo **[YOURDON]** mette in relazione le funzioni con i dati.

Questi due modelli consentono di produrre le specifiche delle singole funzionalità software.

Analisi dei requisiti software non funzionali

Nell'elaborazione del quadro dei vincoli di progetto e nell'analisi dei requisiti software non funzionali il Progettista può riferirsi alla norma ISO/IEC 9126.

Prototyping

La realizzazione di un prototipo applicativo può essere di aiuto nel chiarire la contrapposizione tra le esigenze dell'utente e la soluzione proposta.

Decidere sulla pratica di prototipazione è una questione importante: se è vero che il costo di un prototipo è generalmente non trascurabile rispetto ai costi totali, è altresì vero che i benefici sul progetto sono veramente tanti.

Il prototipo può avere obiettivi di analisi diversi:

- il prototipo esterno è un prodotto orientato all'utente che a livello di interfaccia applicativa si avvicina molto al prodotto finale, con l'unica differenza che dietro l'interfaccia non esiste logica applicativa,
- il prototipo interno è un prodotto realizzato per simulare l'architettura tecnologica.

I prototipi, dunque, consentono di sperimentare e scegliere i percorsi più adatti per il progetto.

Produzione del documento di specifiche funzionali

Quest'attività consiste nell'integrare tutti i risultati conseguiti nell'analisi in una forma coerente, ad esempio un unico documento.

Processi di supporto	Attività specifiche per i requisiti software
verifica e validazione	<ol style="list-style-type: none">1. Definizione del piano di collaudo di sistema2. Progettazione dei casi di test (individuazione dei requisiti software da collaudare)3. Disegno dei casi di test di sistema
gestione della configurazione	<ol style="list-style-type: none">1. aggiornamento del piano di configurazione per le fasi successive2. aggiornamento della configurazione con i prodotti di fase3. esecuzione delle modifiche secondo procedure e tool aziendali

gestione della qualità	1. Ispezione del documento di specifiche funzionali 2. Ispezione dei casi di test di sistema
------------------------	---

Tabella 19 – Attività dei processi di supporto per la fase dei requisiti software

Disegno Architeturale

Nel processo di disegno d'architettura il Progettista definisce ad alto livello le strutture interne delle soluzioni formulate nel processo di analisi dei requisiti SW.

Il processo di Disegno Architeturale ha in input il documento di specifiche funzionali (dei requisiti software) ed in output il documento di specifiche dell'architettura software (in termini dei suoi componenti software).

I Progettisti Analisti sono gli attori principali, il PM controlla le attività ed è coinvolto solo in caso di necessità.

Le pratiche di base del Disegno Architeturale sono:

- Progettazione dell'infrastruttura tecnologica
- Progettazione della logica applicativa
- Progettazione dei dati
- Produzione del documento di progettazione

Progettazione dell'infrastruttura tecnologica

La pratica di progettazione dell'infrastruttura tecnologica definisce le caratteristiche tecniche degli ambienti nei quali saranno sviluppati, testati e rilasciati in esercizio i risultati intermedi e finali del lavoro di progetto.

La definizione scende nel dettaglio di strutture, legami e configurazioni dei componenti tecnologici che ospiteranno i componenti software dell'applicazione.

Tipicamente le caratteristiche da definire sono relative a:

- Sistemi Operativi.
- DBMS
- Sistemi Transazionali
- Reti di comunicazione

Progettazione della logica applicativa

La pratica di progettazione della logica applicativa è strutturata a sua volta nelle seguenti attività:

- Progettazione dell'Interfaccia utente
- Progettazione della Logica di presentazione
- Progettazione della logica aziendale
- Progettazione della Logica dei dati
- Progettazione interfacce con altri sistemi

La progettazione dell'interfaccia utente consiste nel progettare i componenti di interfaccia: nel caso di un'applicazione con interfaccia grafica (GUI) si tratterà di definire gli oggetti dell'interfaccia (Menù, Finestre di dialogo, Form di dati, etc..) e il loro comportamento.

Nella progettazione della logica di presentazione si progettano tutti quei componenti software che si occupano di gestire lo stato di presentazione del dato.

Nella progettazione della logica aziendale si progettano tutti quei componenti software che si occupano di implementare le regole del processo aziendale che ospiterà l'applicazione in sviluppo.

Nella progettazione della logica dei dati si progettano tutti quei componenti software che si occupano di colloquiare direttamente con il data base attraverso il DBMS.

Infine nella progettazione delle interfacce con altri sistemi si progettano tutti quei componenti software della nostra applicazione che consentono l'interazione con applicazioni diverse, ad es. importazione-esportazione di dati.

Progettazione dei dati

La pratica di progettazione dei dati recepisce il modello concettuale dei dati realizzato nell'analisi dei requisiti software per costruire il modello logico e fisico dei dati.

La progettazione a questo stadio dovrà necessariamente tenere conto del sistema di gestione dei dati (DBMS) che sarà utilizzato (File ad organizzazione random organizzati mediante indici, DBMS relazionali quali IBM DB2, MS SQL Server, Oracle, etc., DBMS reticolari, DBMS gerarchici o DBMS ad oggetti).

Le attività principali sono:

- Traduzione delle entità e delle relazioni in oggetti fisici del database (tabella, record, nodo, oggetto)
- Traduzione dei nomi concettuali in nomi fisici
- Creazione delle strutture fisiche
- Generazione delle DDL (Data Definition Language)
- Generazione dei componenti SW per il caricamento dei data base.

Produzione del documento di progettazione

Tutti i componenti software individuati sono descritti in un documento integrato.

Processi di supporto	Attività specifiche per il disegno d'architettura
verifica e validazione	1. Definizione del piano di collaudo di Integrazione

	<ol style="list-style-type: none"> 2. Progettazione dei casi di test (individuazione dei componenti software da integrare) 3. Disegno dei casi di test di integrazione
gestione della configurazione	<ol style="list-style-type: none"> 1. aggiornamento del piano di configurazione per le fasi successive 2. aggiornamento della configurazione con i prodotti di fase 3. esecuzione delle modifiche secondo procedure e tool aziendali
gestione della qualità	<ol style="list-style-type: none"> 1. Ispezione del documento di disegno architettuale 2. Ispezione dei casi di test di integrazione

Tabella 20 - Attività dei processi di supporto per la fase di Disegno architettuale

Disegno di dettaglio

Nel processo di disegno di dettaglio il Progettista definisce a "basso livello" la struttura di ciascun componente SW dell'architettura formulata nel processo di disegno d'architettura.

Il processo di Disegno di Dettaglio ha in input il documento di Disegno Architettuale e in output produce il documento di disegno di dettaglio dei componenti software: è possibile che le conoscenze acquisite nel processo generino una linea di retroazione sull'intero progetto con eventuale aggiornamento delle specifiche di processo e delle specifiche di prodotto.

Gli attori principali sono i progettisti (analisti e programmatori), il PM controlla le attività ed è coinvolto solo in caso di necessità.

Le pratiche di base sono:

- Disegno di dettaglio dei componenti
- Produzione del documento di disegno di dettaglio

Disegno di dettaglio dei componenti

I componenti individuati nella fase di disegno architettuale sono dettagliati ulteriormente in termini di input, output ed elaborazione.

In fase di scelta e pianificazione del modello del ciclo di vita del progetto il PM potrebbe decidere di riassorbire il disegno di dettaglio nel processo a monte (disegno architettuale) oppure nel processo a valle (costruzione del componente): sono scelte orientate a bilanciare complessità dei processi e competenze disponibili.

Produzione del documento di disegno di dettaglio

Tutti i componenti software sono descritti in maniera dettagliata in un documento integrato.

Processi di supporto	Attività specifiche per il disegno di dettaglio
verifica e validazione	<ol style="list-style-type: none"> 1. Definizione del piano di test unitario 2. Progettazione dei casi di test (individuazione dei

	componenti software unitari da collaudare) 3. Disegno dei casi di test unitari
gestione della configurazione	1. aggiornamento del piano di configurazione per le fasi successive 2. aggiornamento della configurazione con i prodotti di fase 3. esecuzione delle modifiche secondo procedure e tool aziendali
gestione della qualità	1. Ispezione del documento di disegno di dettaglio 2. Ispezione dei casi di test

Tabella 21 – Attività dei processi di supporto per le attività di Disegno di Dettaglio

Costruzione del componente e test unitario

Nel processo di costruzione e test il Progettista costruisce il componente SW secondo le specifiche di disegno e prova il componente costruito secondo le specifiche di test.

Il processo di Costruzione del componente e test unitario ha in input il documento di Disegno di dettaglio e in output produce il codice e l'evidenza del test unitario: vale per la costruzione il discorso fatto per il disegno di dettaglio e cioè è possibile una retroazione sull'intero progetto. Gli attori principali sono i Progettisti programmatori, il PM controlla le attività ed è coinvolto solo in caso di necessità.

Le pratiche di base sono:

- costruzione del componente
- costruzione della procedura di test unitario
- esecuzione del test unitario

costruzione del componente

Il Progettista costruisce il componente software utilizzando strumenti, metodi ed infrastruttura tecnologica di sviluppo.

costruzione della procedura di test

Il Progettista analizza piano e specifiche di test e sviluppa la procedura operativa per provare il componente software.

Esecuzione del test

Il Progettista esegue la procedura di prova, registra i risultati e riduce la difettosità del componente: alla fine chiude il processo individuale di costruzione e test del componente.

Processi di supporto	Attività specifiche per la costruzione
verifica e validazione	Controllo di congruenza tra evidenza del test unitario e

	pianificazione del test unitario
gestione della configurazione	esecuzione delle modifiche secondo procedure e tool aziendali
gestione della qualità	Ispezione del codice

Tabella 22 - Attività dei processi di supporto per la fase di Test

Collaudo ed integrazione

Nel processo di collaudo ed integrazione Capo-progetto e Progettisti assemblano l'applicazione da consegnare secondo gli accordi contrattuali.

Il processo di collaudo ha in input i piani di test e le specifiche di prodotto e produce in output i risultati dei test.

Gli attori principali sono Progettisti e PM

- i Progettisti hanno la responsabilità dei livelli di collaudo previsti,
- Il PM ha la responsabilità delle consegne e sorveglia tutti i passaggi di responsabilità tipici del processo.

Le pratiche di base sono:

- Collaudo di Integrazione;
- Collaudo del Sistema;
- Collaudo utente.

Nel collaudo si possono individuare differenti tipi di attività:

- Predisposizione dell'ambiente di test;
- Individuazione dei dati di test;
- Progettazione e realizzazione di meccanismi software per il test:
 1. Scaffold - oggetti che simulano parti di software richiamate dal modulo sotto test;
 2. Driver - oggetti che simulano parti software richiamanti il modulo sotto test;

Collaudo di integrazione

Il collaudo di integrazione ha lo scopo di individuare difetti del prodotto realizzato rispetto ai meccanismi che consentono la comunicazione tra i rami dell'applicazione (es. passaggio di dati tra moduli).

Collaudo di sistema

Il collaudo di sistema ha lo scopo di individuare difetti del prodotto realizzato rispetto ai requisiti software (funzionali e non) specificati.

Collaudo utente

Il collaudo utente ha lo scopo di individuare difetti del prodotto realizzato rispetto alle esigenze dell'utente specificate.

Processi di supporto	Attività specifiche per collaudo ed integrazione
verifica e validazione	Controllo di congruenza tra evidenza del test e pianificazione del test
gestione della configurazione	esecuzione delle modifiche secondo procedure e tool aziendali
gestione della qualità	Vedi il controllo della qualità sviluppato successivamente nel controllo del progetto software.

Tabella 23

Conclusione del compito individuale

Il PM sorveglia il lavoro in corso sia con approcci formali (le riunioni di progetto) sia attraverso occasioni informali di incontro e scambio.

Alla fine del lavoro il Progettista rilascia il prodotto del compito individuale e lo mette a disposizione del PM.

Sulla base dei fatti accaduti il Progettista formula il consuntivo del compito e lo confronta con il preventivo concordato motivando eventuali scostamenti.

Il PM recepisce i risultati del compito sia in termini di prodotto sia in termini di processo integrandoli nello stato di avanzamento del progetto

Caso di studio

I componenti del team registrano le ore lavorate (time sheet) in un foglio excel, i dati servono all'amministrazione per la produzione della retribuzione (per i dipendenti) e della fattura per i consulenti, inoltre i dati raccolti servono al PM per controllare i costi. Le esecuzioni dei test unitario sono registrate su di un foglio di calcolo elettronico e daranno indicazioni al PM sulla conclusione del compito individuale del programmatore.

Capitolo 4 - Esecuzione del progetto software

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	
	1 gennaio 2001	2 gennaio 2001	3 gennaio 2001	4 gennaio 2001	5 gennaio 2001	6 gennaio 2001	7 gennaio 2001	8 gennaio 2001	9 gennaio 2001	10 gennaio 2001	11 gennaio 2001	12 gennaio 2001	13 gennaio 2001	14 gennaio 2001	15 gennaio 2001	16 gennaio 2001	17 gennaio 2001	18 gennaio 2001	19 gennaio 2001	20 gennaio 2001	21 gennaio 2001	22 gennaio 2001	23 gennaio 2001	24 gennaio 2001	25 gennaio 2001	26 gennaio 2001	27 gennaio 2001	28 gennaio 2001	29 gennaio 2001	30 gennaio 2001	31 gennaio 2001		
1																																	
2																																	
3	Rossi		8	8	8			8	8	8	8	9			8	8	9	9	8			9	8	9	8	8			8	9		166	
4	Bianchi	8	8	8	8			8	8	8	8	9			8	8	9	9	8			8	8	8	8	8			8	8		24	
5	Verdi			8	8					8	8	9			8	8	9	9	8			8	8	8	8	8	8		8	9		164	
6	Azzurra	8	8	8	8	8		8	8	8	8	9			8	8	9	9	8			8	8	8	8	8	8		8	9		189	
7	Visconti			8	8					8	8	9			8	8	9	9	8			8	8	8	8	8	8		8	8		155	
8	Diomedea			8	8					8	8	9			8	8	9	9	8			8	8	8	8	8	8		8	9		156	
9	De Franchis			8	8					8	8	9			8	8	9	9	8			8	8	8	8	8	8		8	8		138	
10																																0	
11																																0	
12																																0	
13																																0	
14																																0	
15																																0	
16																																0	
17																																16	
18																																0	
19																																0	
20																																0	
21																																8	
22																																0	
23																																0	
24																																0	
25																																8	
26																																8	
27																																0	
28	Ore	16	16	56	56	24	0	0	24	24	56	56	63	0	0	56	56	63	59	48	24	0	57	48	58	56	58	24	24	58	58	0	1024
29																																	
30	GG/uomo	2	2	7	7	3	0	0	3	3	7	7	8	0	0	7	7	8	7	6	3	0	7	6	7	7	7	3	3	7	7	0	128

Figura 33 - Consuntivo ore lavorate

RDE	ID Test Plan	Caso di test	Data e ora di esecuzione	Esito OK/NOK	RDA	Collaudatore	Data e ora verifica	Annotazioni
1	TP-0001	ABPA	19/03/2001 12.00	KO	1	Gruppo Supporto	12.05	
2	TP-0001	ABPA	19/03/2001 12.00	KO	1	Gruppo Supporto	12.15	
3	TP-0001	ABPA	19/03/2001 12.00	KO	1	Gruppo Supporto	13.00	
4	TP-0001	ABPA (1-37 C1=02)	19/03/2001 15.00	OK		Gruppo Supporto	16.05	
5	TP-0001	ABPA (1-37 C1=04,13,14)	20/03/2001 15.00	OK		Gruppo Supporto	8.30	
6	TP-0001	ABPA 14	20/03/2001 15.00	KO	2	Gruppo Supporto	11.00	
7	TP-0001	ABPA 19	20/03/2001 15.00	KO	3	Gruppo Supporto	11.00	
8	TP-0001	ABPA 36,	21/03/2001 11.00	KO	4	Gruppo Supporto	12.30	
9	TP-0001	TP Incassi 6	22/03/2001 0.00	KO	5	Gruppo Supporto	14.00	
10	TP-0001	11 TP	22/03/2001 0.00	KO	6	Gruppo Supporto	14.30	
11	TP-0001	11 TP	22/03/2001 0.00	KO	7	Gruppo Supporto	14.30	
12	TP-0001	12-13 TP	26/03/2001 0.00	OK		Gruppo Supporto		

Figura 34 - Controllo delle attività di test unitario

Nelle attività fin qui descritte il PM ha essenzialmente un ruolo di integratore sia dei risultati del lavoro sia degli input interni/esterni che insistono sui compiti individuali.

I processi di integrazione difficilmente sono delegabili dal PM così come la gestione delle comunicazioni e delle Risorse Umane trattate di seguito.

Distribuzione dell'informazione

Comunicare efficacemente durante lo svolgimento del progetto è uno dei fattori critici per la riuscita del progetto.

Un errore nella comunicazione (sbagliare l'individuazione dell'interlocutore o l'organizzazione dei contenuti o la scelta del momento) può creare ambiguità e generare conflitti.

Il piano di comunicazione sviluppato durante la fase di pianificazione guida il PM nel distribuire efficacemente le informazioni di progetto ai corretti destinatari, quando richiesto, con un buon livello di dettaglio.

L'esecuzione del progetto è il momento in cui si crea l'informazione dagli elementari disponibili.

La creazione dell'informazione è prevalentemente a carico del PM, è stimato [KERZNER] PM spenda il 90 per cento del suo tempo a comunicare.

Un PM che non ha doti di comunicazione sia verbale sia scritta dovrà ricorrere o all'aiuto di un'assistente oppure dovrà svolgere corsi di comunicazione.

Di seguito alcuni esempi pratici di comunicazione relativi al caso di studio:

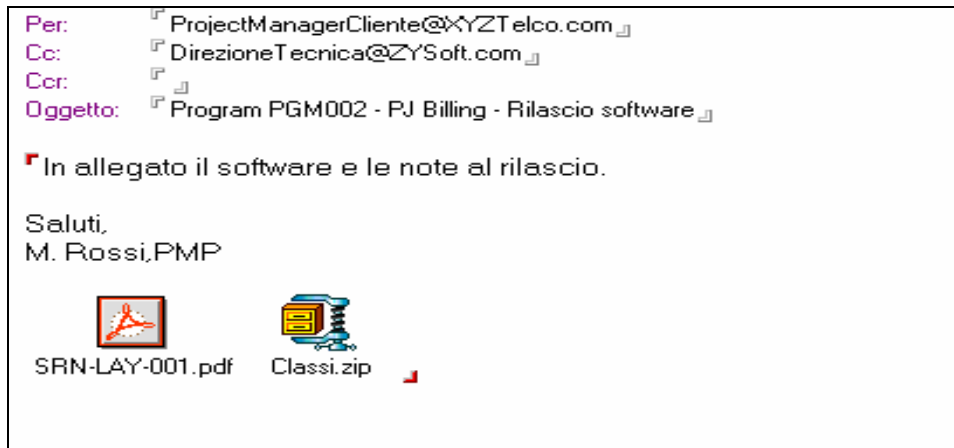


Figura 35 - Esempio di comunicazione di rilascio software



Figura 36 - Esempio di comunicazione relativo allo stato del progetto

Per: Start-up Team
Cc: Direzione.Tecnica@ZYSoft
Cc:
Oggetto: PGM002 - PJ Billing - Convocazione Riunione Studio di fattibilità

E' convocata riunione.
Di seguito l'agenda per la riunione in oggetto:

Oggetto : Studio di fattibilità - Rendiconti Visuali per la Fatturazione
Inizio ore : 15.00
Locazione : Sala riunioni - Centro Direz. Is E3 - 15° Piano
Prende nota : R. Cerva

Argomenti	Autore	Tempo previsto
1. Introduzione	M. Rossi	10m
2. Obiettivi	M. Rossi	10m
3. Rendiconti Visuali	M. Rossi	15m
4. Elementi dello studio di fattibilità	M. Rossi	30m
5. Feedback, Varie ed eventuali	Tutti	50m

Si prega di dare la propria disponibilità, onde consentire l'eventuale ripianificazione.

Saluti,
M. Rossi

Figura 37 - Esempio di convocazione riunione

Sviluppo della squadra (team)

In fase di esecuzione il PM deve **impegnarsi a:**

- Accogliere la Persona
- Valutare conoscenze e competenze reali
- Condividere/aggiornare le azioni formative previste
- Attivare/sorvegliare il processo formativo.

Valutazioni iniziali

La valutazione delle conoscenze e competenze realmente possedute dalla Persona che entra nella squadra di progetto può avvenire mediante:

- Colloqui informali tra PM e Progettista
- Osservazione dei comportamenti durante i processi individuali – elaborazione dei compiti assegnati
- Osservazione dei comportamenti durante le riunioni
- Rilevazione dati mediante questionari di autovalutazione.

Le informazioni raccolte dovrebbero essere schematizzate in modo da evidenziare efficacemente i punti di forza e di debolezza di ogni singolo individuo.

Caso di studio
 Rossi utilizza per la valutazione delle competenze in entrata lo stesso form utilizzato per la definizione delle competenze richieste:

			Project Management		UML		Unix		DB2		Linguaggio C Apache web		Tools di testing Microsoft Project		Fatturazione		SCM CVS		Office		
			M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L	M	L	M
1	PM	Rossi	60	3	0	12	1	6	1	12	2			12	1	12	1	3	1	3	1
2	TM	Bianchi	24	1	0	12	1	6	1	12	2			12	1	12	1	3	1	3	1
2	TM	Verdi	24	1	0	12	1	6	1	12	2			12	1	12	1	3	1	3	1
3	TL (analista)	Bruni			36	3	12	1	12	1	12	2			36	3	3	1	3	1	3
4	TL (progettista)	Manzoni			36	3	12	1	12	1	12	2			36	3	3	1	3	1	3
5	Team Member	Falco			12	1	12	1	12	1	0	12	1	12	1	12	1	3	1	3	1
	Team Member	Delfino			12	1	12	1	12	1	0	12	1	12	1	12	1	3	1	3	1
80																			

Figura 38 - Form per la valutazione delle competenze per il caso di studio in esame

Condivisione/aggiornamento del piano di sviluppo

Quando il PM ha chiari i punti di forza e di debolezza della Persona può e deve aggiornare/condividere il piano di sviluppo professionale.

La condivisione è importante perchè non avrebbe molto senso predisporre un corso per migliorare un punto di debolezza che la Persona non percepisce affatto oppure, pur riconoscendolo, non lo ritiene significativo per l'esecuzione degli incarichi previsti.

Una direzione assennata di progetto richiede **sempre** la condivisione di un obiettivo e della relativa strategia di conseguimento, **mai** l'imposizione: in caso di necessità sarà il PM, con il proprio carisma, a facilitare la confluenza di punti di vista differenziati in una sintesi operativa unitaria (gestione dei conflitti).

Caso di studio
 Sulla base delle valutazioni iniziali Rossi rivede il piano previsto in fase di pianificazione riducendo (da tre a due) il numero dei programmatori da addestrare nella programmazione in linguaggio C.

Capitolo 4 - Esecuzione del progetto software

risorsa	Argomento del corso	durata	luogo	data	Ente erogatore
Rossi...	UML	3g	Roma	1-1-2001	Esterno
Bianchi	UML	3g	Roma	1-1-2001	Esterno
Verdi	UML	3g	Roma	1-1-2001	Esterno
Falco	C	2g	Napoli	1-1-2001	Interno
Delfino	C	2g	Napoli	1-1-2001	Interno

Tabella 24 – Piano di addestramento per il caso di studio in esame

A completamento della pianificazione della formazione standard e del training on the job Rossi definisce per ciascuna Persona gli obiettivi di competenza funzionali al percorso di progetto:

Nominativo: Falco		Competenze	Modalità	Periodo previsto
		Tecniche di implementazione di algoritmi in C per la valorizzazione del traffico	Studio individuale Corso standard	Primo mese
		Competenza nella interpretazione delle norme di billing riportate nel TOM	Studio individuale	Secondo mese

Tabella 25 – Piano di sviluppo personale per il caso di studio in esame

Assicurazione della qualità

La gestione della qualità è una questione centrale nella gestione di progetto: i temi specifici sono stati anticipati attraverso l'indicazione delle attività applicabili ai vari processi primari. Di seguito ci proponiamo di recuperare una visione d'insieme.

Assicurare la qualità del progetto software (prevenzione, controllo e recupero della difettosità) è un compito del gruppo di lavoro e ricade nella responsabilità del PM: la sorveglianza sullo svolgimento del compito, generalmente, è a carico della funzione aziendale della qualità (monitoraggio del progetto).

L'assicurazione della qualità deve rispecchiare quanto definito nella pianificazione della qualità sviluppato durante la fase precedente.

Il PM deve fare in modo che il gruppo di lavoro sia attento alla qualità del prodotto e di processo:

- attenzione alla qualità del prodotto vuol dire **cura** della realizzazione del software e dei risultati intermedi relativi;
- attenzione alla qualità del processo vuol dire preventivazioni realistiche, disponibilità dei fattori di produzione, consuntivazioni veritiere.

L'attenzione alla qualità non può rimanere uno slogan: ricordiamo come il PM debba saper bilanciare obiettivi contrastanti espressi in termini di tempi, costi (impegno delle risorse) e qualità.

Spesso capita che si decida (a causa delle pressioni dei Clienti e della Direzione) di tagliare le attività di assicurazione della qualità per consegnare nei tempi previsti.

Tipico esempio si ha quando il progetto è in ritardo e non è possibile effettuare le consegne nelle date pianificate: l'errore frequentemente commesso è di saltare una o più fasi di test.

Il danno che se ne ricava è l'ulteriore lavorazione del software a causa delle anomalie che inevitabilmente saranno rilevate durante la fase di esercizio: in definitiva un rimedio furbesco e superficiale al ritardo aumenta il ritardo, il costo e l'insoddisfazione del cliente.

Tutto ciò è ampiamente dimostrato da Papaccio e Boehm - 1988 [**RADMCCO**] secondo cui la correzione di anomalie del software in esercizio costa dal 50% al 200% in più rispetto al livello di costo che si avrebbe se i difetti fossero stati rimossi durante la fase dei requisiti: in generale, facendo riferimento all'arco temporale del ciclo di vita, si può affermare che più tardi il difetto è riscontrato più aumenta il costo della sua rimozione.

L'abilità del PM sta proprio nel trarre il massimo risultato dall'opera di prevenzione e controllo magari con l'aiuto di strumenti dedicati alle attività di assicurazione della qualità; i tool possono dare concreto supporto ad operazioni del tipo:

- controllo di conformità della codifica rispetto agli standard - layout,
- controllo di dimensione del codice rispetto a valori di soglia,
- controllo di complessità ciclomatica,
- generazione di casi di test,
- gestione del processo di test,
- gestione di data-base per la registrazione e classificazione dei difetti, etc..).

Di seguito analizziamo nel dettaglio le principali responsabilità del PM durante le principali attività di assicurazione della qualità:

- sorveglianza del test
- Ispezioni
- Individuazione ed eliminazione di componenti ad alto tasso di difetti

Sorveglianza del test

Come già evidenziato, il processo di pianificazione-esecuzione-tracciamento dei test è una delle pratiche più conosciute di assicurazione della qualità. Il gruppo di lavoro esegue i test sul software con l'intento di individuare i difetti.

Il compito principale del PM è di sorvegliare l'andamento del test attraverso:

- Raccolta e classificazione dei difetti riscontrati
- RegISTRAZIONI dei tempi di esecuzione dei test.

La raccolta e la classificazione dei difetti saranno utili, come vedremo durante l'attività di controllo qualità per valutare la qualità del prodotto testato.

La registrazione dei tempi contribuirà a valutare la bontà del processo di test.

Il fatto che il numero ulteriore di difetti riscontrati nel tempo tenda a zero (ed il numero totale ad un valore di soglia prefissato) può avere significati alternativi:

- Un buon numero di difetti è stato riscontrato, la difettosità del SW è stata ridotta a sufficienza perciò è possibile interrompere il test in anticipo.
- I casi di test sono incompleti/parziali e non garantiscono un buon comportamento del SW dunque si dovranno progettare nuovi casi di test.

Se invece il numero totale dei difetti non si stabilizza al di sotto del valore di soglia prefissato, potremmo trovarci di fronte a problemi di costruzione, disegno (di architettura e/o di dettaglio) o addirittura di analisi dei requisiti.

È comunque responsabilità del PM dare la corretta interpretazione al fenomeno osservato, definire il problema e gestirlo al meglio attraverso un'azione correttiva e/o preventiva.

Ispezioni

Secondo Gilb e Graham **[RADMCCO]** le ispezioni consentono di rilevare in anticipo rispetto al test dal 60 al 90 per cento dei difetti in un programma con un'efficienza 20 volte maggiore del test e con un risparmio sull'impegno previsto dal 10 al 30 per cento.

Inoltre è stato dimostrato da Russel, 1991 **[RADMCCO]** che su software di grandi dimensioni **un'ora** spesa per ispezionare il software consente di far risparmiare circa **trentatre ore** di manutenzione.

La tecnica dell'ispezione è diversa dalla tecnica del test ma il fine è lo stesso: caratterizzare la (non) qualità del SW. Il PM può delegare l'esecuzione delle ispezioni assegnandole come compiti individuali ai Progettisti, deve acquisire i dati di registrazione/classificazione e come per il test mantiene la diretta responsabilità di individuare e gestire i problemi. Si può notare come l'intervento su un problema possa produrre una ripianificazione di progetto per considerare gli impegni aggiuntivi riconducibili alle necessarie attività di correzione dei difetti e rimozione delle cause.

Individuazione ed eliminazione dei moduli difettosi

Secondo Bohem,1978b [RADMCCO], il 20 per cento dei moduli in un'applicazione software sono responsabili dell'ottanta per cento dei difetti. Con questa indicazione di carattere pratico è possibile condurre un'analisi statistica sui moduli dell'applicazione rappresentando la distribuzione della difettosità ed individuando i moduli software con il più alto tasso di difetti: sulla base di questi dati di fatto il PM, in accordo con i progettisti, deve prendere una decisione.

Tale decisione può significare anche un riciclo completo del processo di sviluppo (analisi, disegno, codifica e collaudo) per uno o più componenti dell'applicazione.

Il database dei difetti

Come evidenziato tutte le attività di assicurazione della qualità richiedono la raccolta dei dati sui difetti e sull'impegno speso per trattarli.

I dati raccolti saranno poi analizzati consentendo così di intraprendere le opportune misure (azioni correttive ed azioni preventive) per il miglioramento del processo di sviluppo.

Per facilitare la gestione delle informazioni sulla difettosità il PM può predisporre e gestire un archivio con il quale controllare la (non)qualità di prodotto.

La struttura dell'archivio potrebbe essere la seguente:

- codice
- evento di segnalazione
- esperimento di prova
- classificazione
- descrizione
- origine nel ciclo di vita
- impatto sul prodotto

Acquisti

Generalmente il processo di acquisto appartiene ad una specifica funzione aziendale ed il nostro interesse consiste nel mettere a fuoco le relazioni tra il PM e la funzione aziendale responsabile.

L'attivazione del processo è basata su un piano di acquisto ed un budget predefiniti (cap.3) e parte attraverso una iniziativa del PM, "la richiesta di acquisto", verso la funzione responsabile.

Caso di studio.

Dal piano di configurazioni delle postazioni di lavoro è emerso la necessità di acquistare un Computer e 3 licenze del tools Visual Age Java per lo sviluppo del software. Di seguito si riporta un esempio di richiesta di acquisto.

ID	Oggetto	Data richiesta	Modalità di acquisizione
1	Ambiente integrato di sviluppo IBM visual age for java - 3 licenze	20-1-2001	
2	PC 400 MHZ, RAM 256 MB, HD 20 GB, Monitor 17"	20-1-2001	

Figura 39 - Generica richiesta di acquisto

Da questo punto in poi il processo d'acquisto prosegue all'interno della funzione responsabile. Tipicamente (ISO/IEC 12207 1995) il processo è articolato nelle seguenti attività:

- preparazione delle richieste d'offerte
- ricezione ed analisi offerte
- trattativa e definizione di ordine/contratto d'acquisto.

La funzione responsabile riattiva il PM passandogli le informazioni su ordine/contratto definito; il PM sulla base delle informazioni disponibili predispose il progetto per l'acquisizione attraverso le seguenti attività:

- setup del processo di progetto candidato a ricevere il bene ordinato
- monitoraggio del fornitore
- setup delle prove in entrata
- esecuzione delle prove e rapporto di collaudo
- valutazione del collaudo in entrata e benessere al pagamento.

Verifica dell'ambito del progetto

Il progetto è in esecuzione ed i vari compiti di lavoro assegnati sono in svolgimento; nel tempo i compiti giungono a conclusione e producono risultati intermedi non oggetto di consegna, documenti e componenti software da consegnare.

La verifica dell'ambito (il lavoro richiesto) è inclusa nell'accettazione formale dei risultati prodotti dai processi di lavorazione.

L'accettazione formale può essere fatta al termine d'ogni fase o prima della chiusura del progetto dalla direzione e/o dal cliente.

In generale, per assicurare il buon fine della consegna e della relativa accettazione, è opportuno programmare ed effettuare una o più verifiche interne di progettazione: il processo di verifica della progettazione e dello sviluppo (UNI EN ISO 9001:2000 punto 7.3.5) ricade nella responsabilità del PM, che condurrà il processo con il gruppo di lavoro interessato alle consegne.

Nella figura è illustrato un tipico GANTT di progetto software con fasi di lavoro e verifiche.

Alla fine d'ogni fase è programmata una verifica interna ed una con il cliente.

Durante la verifica interna, il PM verificherà che i prodotti in uscita dalla fase siano stati realizzati secondo i relativi requisiti in ingresso e gli standard previsti dal processo di produzione.

In funzione dei risultati della verifica il PM potrà:

- dichiarare chiusa la fase e contestualmente comunicare il rilascio del/dei prodotti al cliente, predisponendosi così al collaudo da parte del cliente se si sente sicuro del lavoro fatto,
- viceversa, se il/i prodotti sono incompleti oppure hanno caratteristiche inadeguate ai requisiti, il PM aggiungerà a piano le necessarie attività (azioni correttive) per il completamento e/o la rilavorazione.

Il processo di verifica del **lavoro fatto** da consegnare è necessario ma non è sufficiente: si pensi al caso in cui non ci siano le condizioni operative per recuperare incompletezza e/o inadeguatezza rilevata in verifica.

Per dare maggiore efficacia ed efficienza al progetto, il PM deve sorvegliare il **lavoro in corso** (compiti individuali in corso, compiti chiusi ma non integrati) attraverso ulteriori controlli formali e programmati (riesami di progettazione) e soprattutto attraverso iniziative informali.

il processo di riesame della progettazione e dello sviluppo (UNI EN ISO 9001:2000 punto 7.3.4) ricade nella responsabilità del PM, che condurrà il processo con le Persone chiave del gruppo di lavoro.

Durante il riesame il PM valuterà lo stato delle attività in corso, i risultati disponibili, l'allineamento del lavoro con gli obiettivi e le criticità.

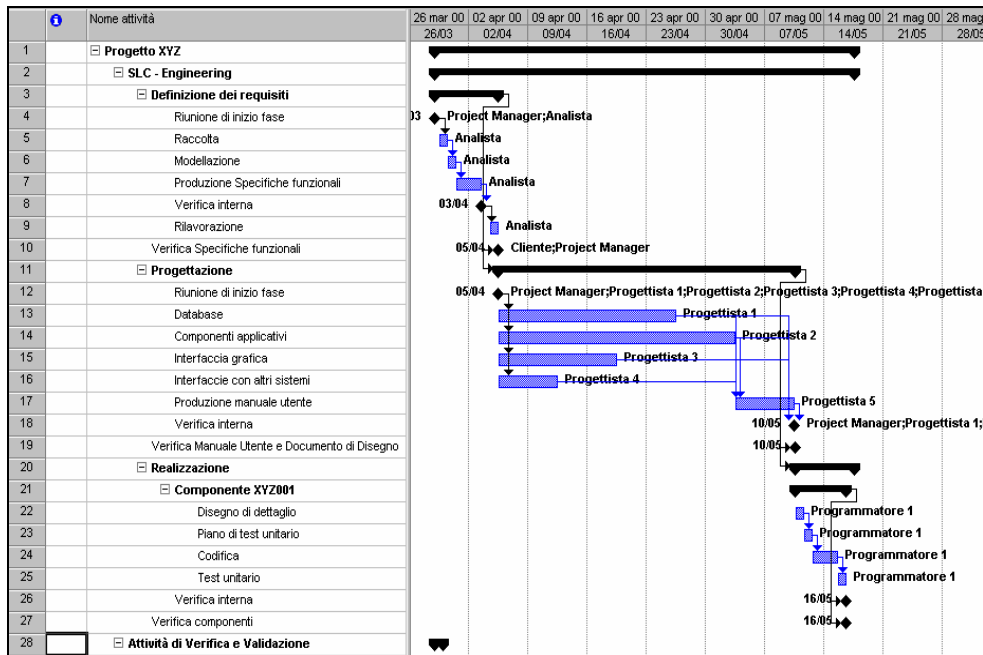


Figura 40 -Generico piano delle attività

Buon senso e controlli non pianificati eviteranno di arrivare ignari al termine della fase rilevando la sorpresa: la lavorazione non si è conclusa come previsto ed invece di consegnare il prodotto al Cliente possiamo solo comunicargli il problema.

Nel GANTT di esempio è riportata a piano un'attività di rilavorazione: Il dimensionamento di tale attività è fatta su base storica, cioè sulla conoscenza di progetti simili.

L'attività di rilavorazione è stimata al 10% dell'intera fase.

Riepilogo

Il Primo Attore sulla scena del progetto cambia con il susseguirsi dei tempi: in definizione è la Direzione, in pianificazione è il Capo-progetto, in esecuzione è il Progettista. Il Capo-progetto rientra nelle quinte e mantiene per se stesso il ruolo del regista (integratore e facilitatore). Convinzione condivisa è che il successo di progetto non sia un appannaggio predestinato di squadre tutte composte da stelle bensì la conseguenza di una paziente ed autorevole opera di mediazione e comunicazione tra professionisti motivati e collocati al posto giusto nel momento giusto.

Controllo del progetto software

- 5.1 Introduzione
- 5.2 Raccolta dei dati e comunicazione dei risultati sullo stato
- 5.3 Controllo dei rischi
- 5.4 Controllo dell'ambito del progetto
- 5.5 Controllo dei costi
- 5.6 Controllo dei tempi
- 5.7 Controllo della qualità del prodotto
- 5.8 Riepilogo

“Un buon capomastro deve conoscere pregi e limiti dei propri uomini e saper controllare l'esecuzione dei lavori, senza mai avanzare richieste irragionevoli. Deve conoscere le loro caratteristiche e le loro debolezze e, quando è necessario, deve saperli incoraggiare. Gli stessi criteri si applicano all'arte della guerra”

Il Libro dei cinque anelli - Miyamoto Musashi

Controllo ed esecuzione sono processi paralleli e concorrenti: il Capoprogetto utilizza il controllo per aggiustare il tiro nelle attività in esecuzione ed eventualmente estende il feedback fino alla ripianificazione del progetto. Controllare significa individuare-valutare scostamenti tra preventivo e consuntivo ed identificare eventuali azioni correttive: dunque per controllare è necessario misurare, per misurare è necessario avere la disponibilità dei dati.

Il punto di partenza del controllo sta nell'area della gestione delle informazioni e delle comunicazioni.

Il capitolo segue l'impostazione del PMI e sviluppa l'applicazione del controllo ad ambito, tempi, costi, rischi, qualità e Risorse Umane.

Introduzione

Controllare un progetto significa rilevare durante la fase d'esecuzione gli scostamenti tra la situazione reale ed il preventivo definito nel piano e tempestivamente intraprendere le azioni correttive capaci di riportare il processo di esecuzione nei limiti pianificati.

Il responsabile del controllo di progetto è il PM.

Per controllare è necessario avere a disposizione un sistema di misure con un numero significativo d'indicatori dello stato del progetto.

Le aree del progetto interessate al controllo devono necessariamente produrre i dati necessari alle misure; le aree identificate dal PMI sono:

- Rischi
- Ambito
- Costi
- Tempi
- Qualità
- Risorse Umane.

I dati prodotti in queste aree saranno raccolti, organizzati, elaborati per poi essere comunicati alle Parti interessate attraverso forme di rendiconti di prestazione.

Nelle sezioni successive, vedremo in dettaglio quale sono le misurazioni che vengono fatte per le aree sopra citate e quali sono le interpretazioni dei valori determinati.

Raccolta dei dati e comunicazione dei risultati sullo stato

La raccolta dei dati e la comunicazione dei risultati sullo stato del progetto hanno lo scopo di produrre informazioni utili alle Parti interessate su

- stato del progetto,
- progressi ottenuti,
- previsioni a finire.

L'attività consiste nell'acquisizione ed elaborazione dei dati elementari relativi alle aree fondamentali individuate.

Le fonti da cui acquisire i dati sono principalmente:

- sistema di supporto alla pianificazione per quanto riguarda informazioni su attività, tempi e costi previsti nel piano di progetto
- Sistema di gestione delle risorse umane o fogli elettronici contenenti le ore effettivamente lavorate
- Sistema di gestione di rischi/criticità o documenti contenenti l'elenco dei rischi/criticità da sorvegliare

Capitolo 5 - Controllo del progetto software

- Sistema di gestione delle anomalie software o l'archivio contenenti i difetti
- Sistema di gestione dei costi.

Un processo di progetto che genera grande quantità di dati elementari è la gestione del compito individuale sia come chiusura ed integrazione dei compiti conclusi (lavoro fatto) sia come sorveglianza dei compiti in svolgimento (lavoro in corso).

I componenti del gruppo possono comunicare lo stato delle proprie attività attraverso canali e metodi diversi:

- riunioni di progetto
- posta elettronica
- accesso diretto agli archivi di progetto organizzati con tecnologie web-based.

Per tutto il lavoro in corso il Progettista specificherà la percentuale di completamento e la previsione a finire.

Per il lavoro fatto il Progettista comunicherà la data effettiva di chiusura.

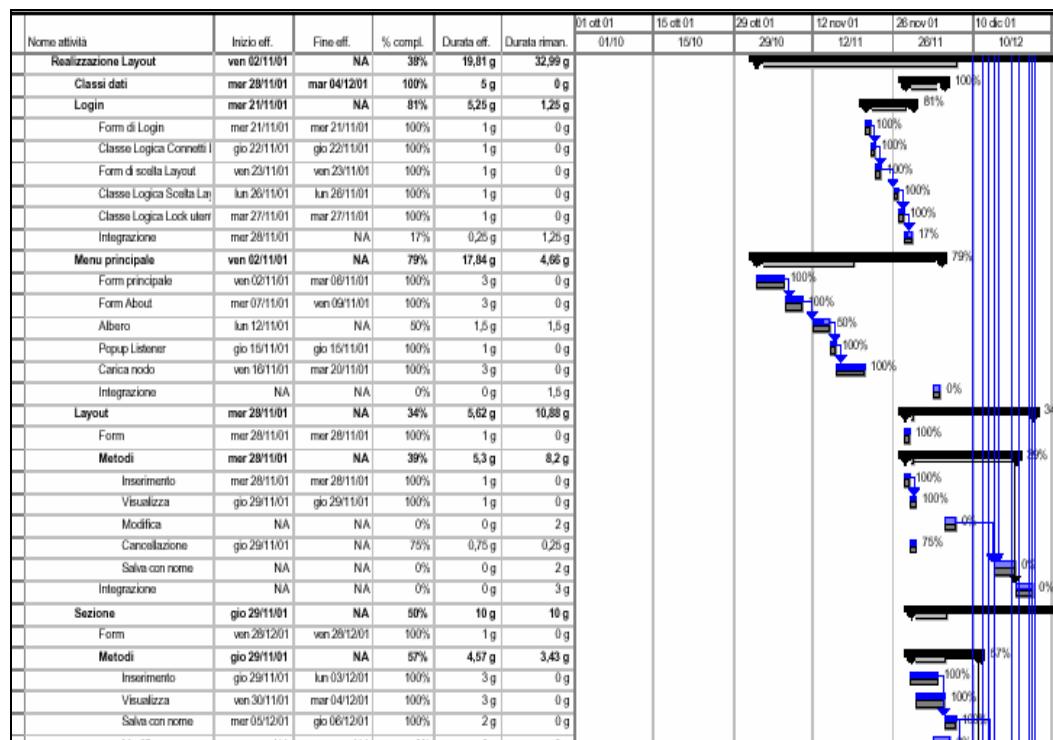


Figura 41 – Piano di dettaglio per l'implementazione dei componenti

In figura è illustrato un diagramma di Gantt relativo alla realizzazione dei componenti software facente parti del sottosistema dei rendiconti visuali. Ad esempio i componenti di Login da realizzare sono:

- La finestra di Login
- La classe logica che consente la connessione al DBMS
- Il form di scelta che consente la scelta del rendiconto
- La classe logica che implementa la parte di codice relativa alla scelta del rendiconto
- La classe logica che consente ad un utente di bloccare (Lock) quel rendiconto in modo esclusivo

Al termine vi è un'attività di integrazione dei componenti.

La colonna **%compl** riporta le percentuali di avanzamento.

I dati elementari sono raccolti, elaborati e poi utilizzati per la produzione delle informazioni di sintesi attraverso uno o più documenti (Stato del progetto, Rendiconto d'avanzamento, ...).

Questi ultimi vengono inviati ai destinatari interessati nei tempi e nei modi definiti nel piano di comunicazione.

E' importante in ogni caso che la raccolta dei dati sia congruente con la comunicazione dei risultati, ad es. se la distribuzione dei documenti è un evento periodico con periodo pari alla settimana allora i dati raccolti dovranno riferirsi a questo periodo.

I documenti possono essere separati oppure integrati in un unico documento anche in considerazione del grado di dettaglio e dei costi che caratterizzano la comunicazione.

A titolo esemplificativo di seguito si riporta una lista delle informazioni che tipicamente possono essere scambiate nel processo di comunicazione dei risultati del progetto:

- stato delle attività assegnate dettagliato per il periodo d'interesse (ID Attività, descrizione breve attività, stato, annotazioni, data fine prevista)
- rischi/Criticità
- Varianti ai requisiti
- costi - Scostamenti e Previsioni a finire
- tempi - Scostamenti e Previsioni a finire
- qualità e Prestazioni di prodotto
- clima del progetto e sviluppo delle risorse umane
- problemi e Decisioni prese.

Controllo dei rischi

Per ogni rischio presente nel piano il PM controlla se è scaduto (in altre parole se è trascorso l'intervallo temporale entro cui questo rischio era stato definito) oppure se è ancora possibile l'accadimento dell'evento temuto.

Nel caso in cui il rischio sia scaduto il PM lo chiude, lo elimina dalla lista ed eventualmente inserisce in lista un ulteriore rischio.

Nel caso in cui l'evento temuto possa ancora accadere si controlla la presenza di eventuali sintomi premonitori e si decide sulla risposta da dare (attivazione del piano contingente del rischio).

Nel caso in cui l'evento temuto sia accaduto il PM attiva il piano di risposta e ne controlla l'esecuzione.

E' importante che il PM tracci tutte le informazioni raccolte sui rischi, la storia servirà per ulteriori progetti.

Al solito, per progetti di grandi dimensioni, è difficile essere efficienti senza l'aiuto dell'informatica:

un database che classifica e conserva le informazioni è d'ausilio sia alla gestione dei rischi di progetto sia come archivio storico per progetti futuri.

Caso di studio

Il PM ha deciso di gestire il database dei rischi attraverso la tecnologia MICROSOFT ACCESS. L'applicazione implementa un questionario tassonomico (Figura 19) sulla base delle definizioni del SEI [SEITAXONO].

Il questionario suddivide rischi e relative sorgenti per aree di appartenenza. Tutte le sorgenti selezionate aiuteranno poi a descrivere ulteriormente i rischi, le cui informazioni vengono raccolte mediante (figura 18).

The screenshot shows a software application window titled "Rischio : Maschera". The window is divided into several sections:

- Identification Section:** Contains fields for "ID Rischio" (value: 3), "Titolo:" (empty), "Codice Commessa" (value: 31053-0001), "Sorgente" (value: 18), "Rango" (value: 24), "Identificato da" (value: M.Rossi), and "Data identificazione" (value: 10/10/01).
- Quantification Section:** Contains fields for "Probabilità:" (value: 3), "Impatto:" (value: 4), "Esposizione economica:" (value: L. 10.000), "Effort:" (value: 2), "Data prev. inizio impatto:" (value: 11/10/01), and "Data fine prev. impatto:" (value: 11/12/01).
- Consequences Section:** A text area containing the text: "Perdita di credibilità presso il cliente", "Pagamento di penali", and "Dimissioni".
- Navigation and Status:** At the bottom, there are navigation buttons and a status bar showing "Record: 1 di 2".

Figura 42 - Raccolta e gestione dei rischi

Nella figura precedente è illustrato un form di MS Access che consente la raccolta dei rischi di progetto.

Ogni rischio viene identificato con un Identificativo univoco (**ID Rischio**) ed un **Titolo** breve.

Il rischio viene descritto in **Descrizione del rischio**, in **Sorgente** vi è la classificazione del rischio che è basata sul questionario tassonomico del SEI.

In **Quantificazione** vi sono tutte le informazioni relative a **Probabilità, Impatto, Esposizione economica, Effort**, date in cui si prevede che possa accadere il rischio.

In controllo invece vi sono le annotazioni relative ai controlli effettuati per il rischio in esame.

In **risposta** vi sono tutte le informazioni (Attività, risorse coinvolte, durata prevista, responsabile, etc..) relative alle attività che vengono svolte in risposta al verificarsi di un rischio.

In **Piano di contingenza** vi sono informazioni dello stesso tipo di quelle presenti in risposta.

La differenza sta nel momento in cui si definiscono le informazioni relative.

In **stato del rischio** invece vengono tracciati gli stati in cui può trovarsi il rischio.

Questione	Check
Puoi misurare se il processo di sviluppo soddisfa gli scopi di qualità e produttività ?	<input type="checkbox"/>
Vi sono problemi di coordinamento quando si utilizzano più modelli di sviluppo ?	<input checked="" type="checkbox"/>
Vi sono piani di controllo formali per tutte le attività del ciclo di vita del software ?	<input type="checkbox"/>
I piani di controllo formali specificano bene il processo ?	<input checked="" type="checkbox"/>
Gli sviluppatori hanno familiarità con i piani ?	<input type="checkbox"/>
[79] Il processo di sviluppo è adeguato per questo prodotto ?	<input type="checkbox"/>
Il processo di sviluppo è supportato da un'insieme di procedure, metodi e tools ?	<input type="checkbox"/>
[77] Vi è più di un modello di sviluppo utilizzato ?	<input type="checkbox"/>
In che modo si garantisce che qualcuno segua il processo di sviluppo ?	<input type="checkbox"/>

Figura 43 - Form per l'identificazione dei rischi basata sul questionario tassonomico del SEI

Nella figura precedente invece il form consente di identificare i rischi di progetto. Per ogni processo componente dello sviluppo (Requisiti, Disegno, etc...) vi è un elenco di domande pertinenti all'area.

Selezionando una delle domande si identifica un rischio potenziale il quale verrà descritto ulteriormente.

Il PM sorveglia l'andamento del processo con l'aiuto di un report (Fig. 20) contenente l'elenco dei rischi.

In particolare MC Connell [**RADMCCO**] consiglia di tenere sotto controllo (gestire) solo i 10 rischi più importanti.

ID rischio	Probabilità	Titolo	Data apertura	Data scadenza	Assegnato a
R001	.7	Tecnologia nuova	10-10-2001	10-12-2001	Rossi
R002	.6	Cambio delle interfacce	8-9-2001	10-12-2001	Bianchi
R003	.5	Dimissioni di persone chiave	8-9-2001	10-12-2001	Senior Management
R004	.5	Cambio nei requisiti	8-6-2001	10-9-2001	Rossi

Tabella 26 – Parte dei principali rischi del caso di studio in esame

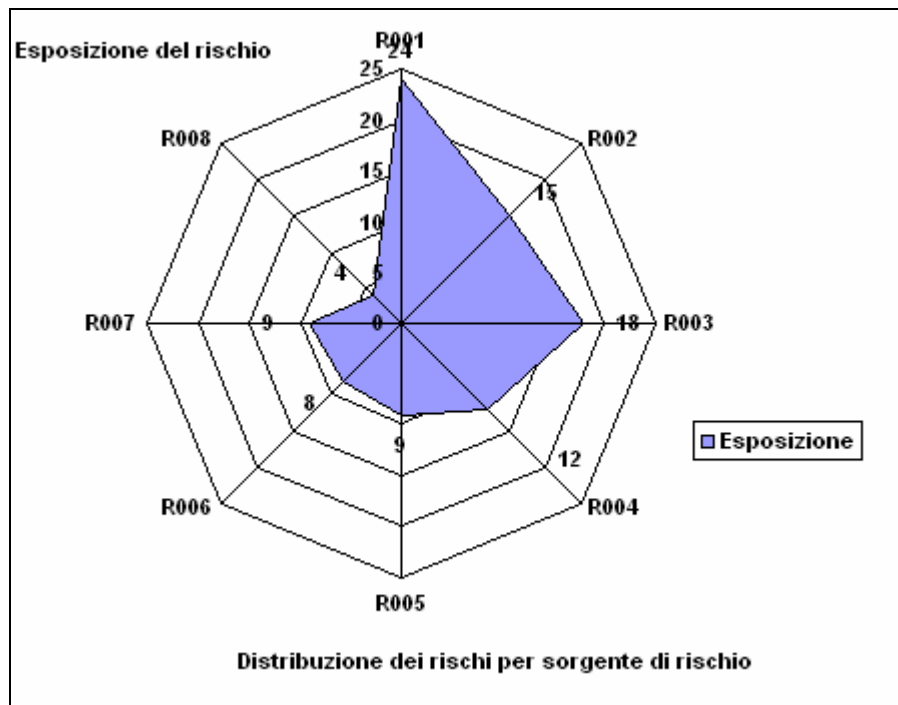


Figura 44 – Grafico Radar per i rischi del caso di studio in esame (i valori sono dati dal risultato del prodotto probabilità per impatto).

Il grafico (Radar) evidenzia i rischi con maggiore probabilità di verificarsi, esso fa riferimento ai dati della tabella precedente.

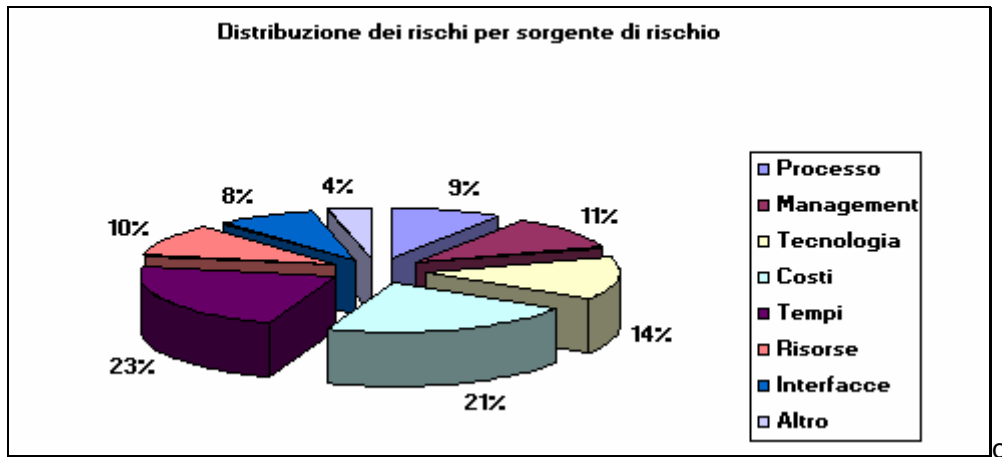


Figura 45 – Distribuzione dei rischi per “sorgente”

Il grafico a torta illustrato in figura invece evidenzia come si distribuiscono i rischi di progetto identificati per sorgente [SEITAXONO].

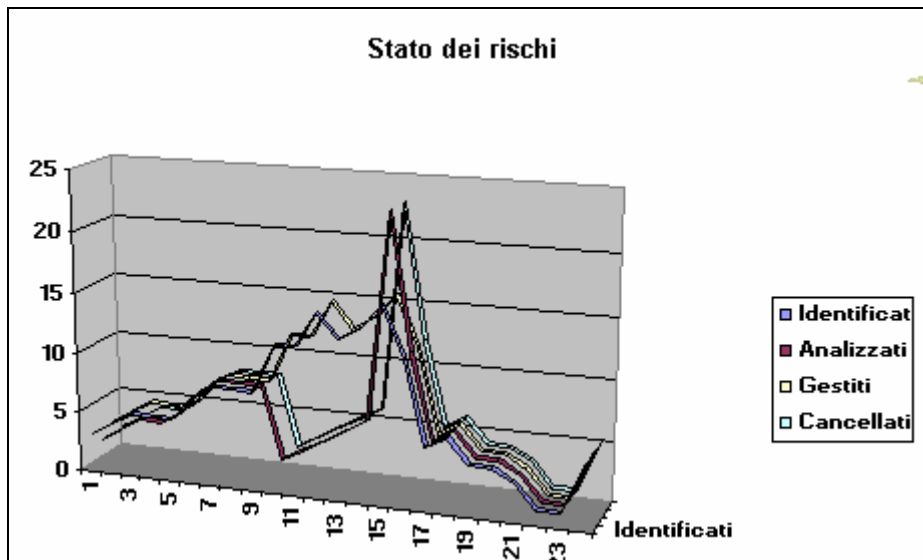


Figura 46 – Stato dei rischi nei periodi di controllo

Il grafico in figura invece illustra lo stato dei rischi identificati, nel tempo. Esso dà un'idea immediata di quanti rischi sono monitorati, quanti sono in fase di analisi, quanti sono scaduti.

Controllo all'ambito del progetto

Il controllo dell'ambito del progetto agisce su:

- Lavoro necessario per la realizzazione del prodotto – cosa si sta facendo
- Requisiti del prodotto software – cosa si aspetta il cliente.

Per certi versi attività che controllano l'evoluzione dell'ambito le abbiamo già incontrate nella fase d'esecuzione (verifica dell'ambito):

- Processo di accettazione ed integrazione dei risultati di ciascun compito individuale
- Processo di verifica del lavoro fatto
- Processo di riesame del lavoro in corso da completare e del lavoro in programma da iniziare.

Il PMI colloca questi processi nel gruppo di esecuzione e la presente guida segue tale impostazione ma si ritiene opportuno sottolinearne la peculiarità classificandole come autocontrollo, intendendo con ciò una serie di azioni e di **micro modifiche** atte a mantenere l'evoluzione del progetto nell'alveo definito nel piano e nel contratto.

Progettista e PM agiscono nella sfera di autonomia professionale connessa ai rispettivi ruoli conseguendo nel tempo i vari risultati previsti: le difficoltà incontrate sono state superate applicando ed interpretando il piano definito e l'azione di governo è tracciata nei verbali dei riesami e delle verifiche di progettazione.

Se poi la squadra non è più capace di garantire che il flusso dello sviluppo rimanga all'interno dei parametri stabiliti in pianificazione allora è necessario innescare attraverso una Richiesta di Azione Correttiva un processo formale di gestione delle varianti, in altre parole un processo che consente di:

- Richiedere una variante
- Analizzare la variante richiesta
- Implementare la variante
- Tracciare la variante

Il controllo dell'ambito risponde a più di una prescrizione della norma ISO9001:2000 relative alla realizzazione del prodotto (punto 7).

Il processo di gestione delle modifiche ai requisiti

La norma citata prende in esame prima (punto 7.2) i processi relativi al cliente ed in particolare la gestione delle modifiche ai requisiti, il relativo riesame e la necessaria comunicazione.

Il processo "definizione dei requisiti utente" componente del processo di sviluppo software produce risultati che costituiscono le basi della preparazione del contratto ed eventuali successivi aggiornamenti:

- un accordo iniziale tra le parti interessate per la definizione dei requisiti di fornitura – contratto
- la definizione di un canale di comunicazione tra le parti
- un meccanismo di gestione delle modifiche contrattuali.

Per rendere operativo il processo di gestione delle modifiche contrattuali è necessario definire input, output, risorse materiali, ruoli, competenze, responsabilità ed autorità: il PM prende in carico la richiesta di modifica, ne analizza la fattibilità, attiva il processo di gestione delle modifiche progettuali, si sospende in attesa del risultato, ottenuto il quale si riattiva per chiudere un nuovo accordo sui requisiti.

Caso di studio.

Rossi ha deciso di gestire le modifiche ai requisiti attraverso un data-base e relative statistiche capaci di rappresentare le informazioni fondamentali: quanti requisiti sono stati trattati, quanti requisiti sono stati (ri)proposti etc.

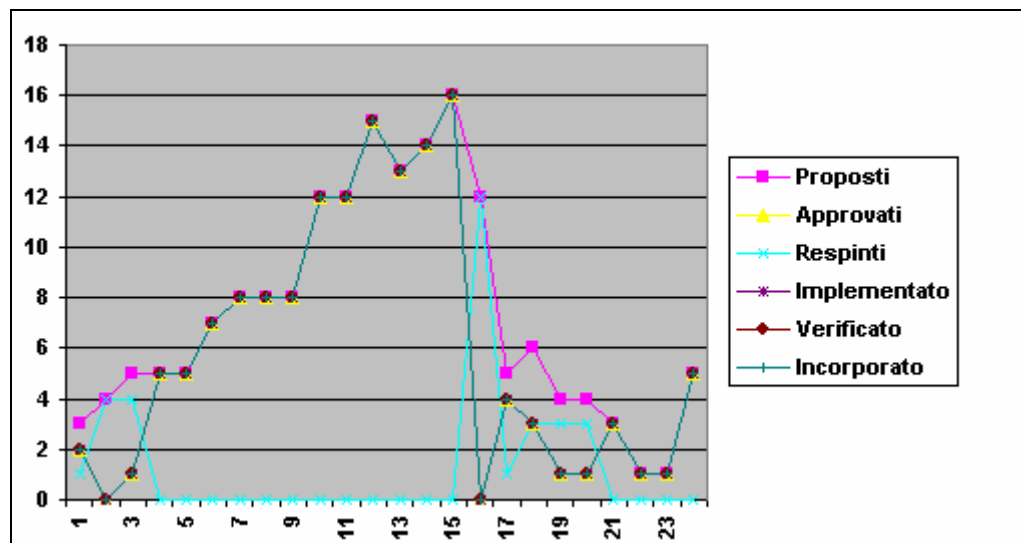


Figura 47 – Stato dei nuovi requisiti per giorno

In figura è illustrato lo stato dei requisiti nel tempo. Esso permette di controllare il processo relativo ai requisiti e in futuro di dimensionare al meglio la fase, consente di mettere in relazione ad esempio il numero dei requisiti proposti con quanti ne vengono incorporati nel software.

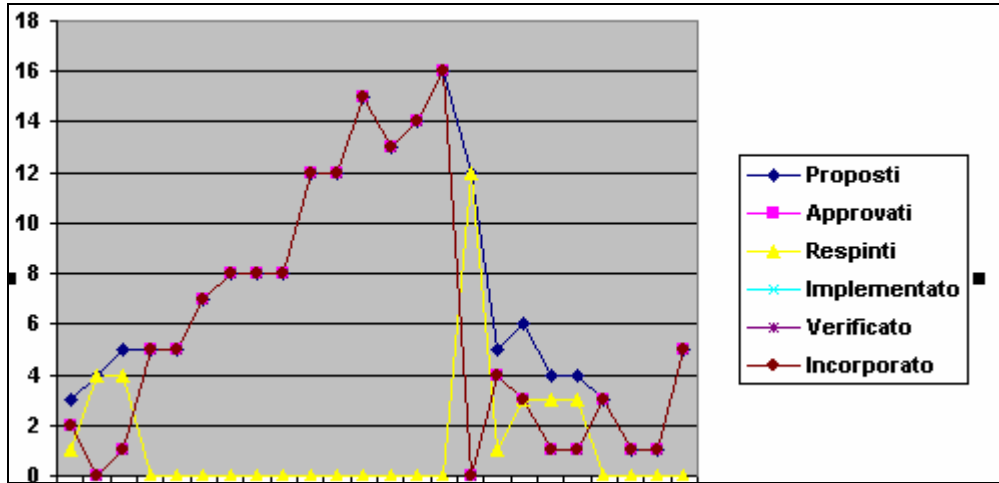


Figura 48 – Stato dei cambi dei requisiti incorporati per i periodi di controllo

La figura precedente invece illustra lo stato dei cambi ai requisiti nel tempo. Il grafico dà utili indicazioni circa la stabilità dei requisiti.

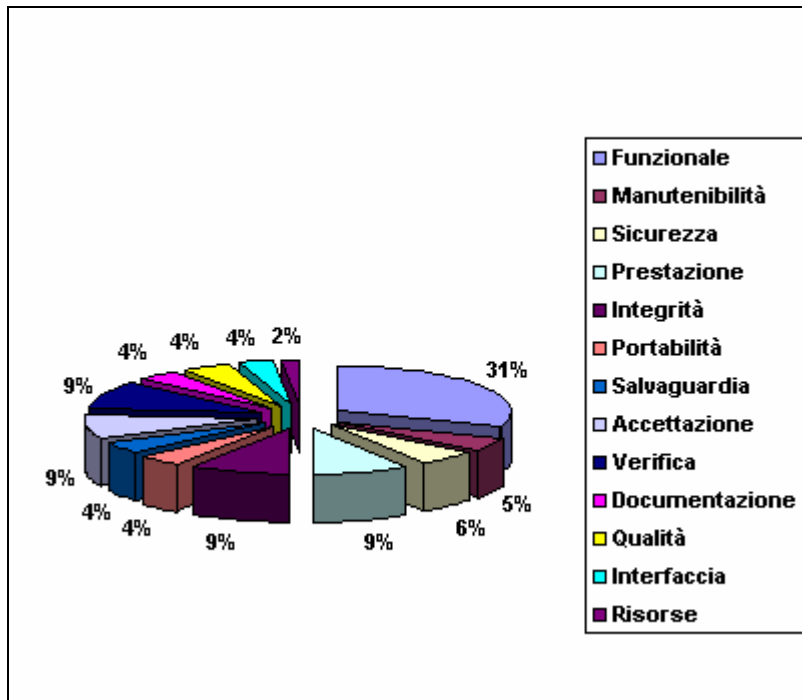


Figura 49 - Distribuzione dei requisiti per tipo di requisito

La figura precedente illustra la distribuzione dei requisiti per tipo di requisito. Per il prodotto in esame il 31% è costituito da requisiti funzionali.

Il processo di gestione delle modifiche alla progettazione

Successivamente (punto 7.3) la norma citata prende in esame i processi di progettazione e sviluppo ed in particolare (punto 7.3.7) la tenuta sotto controllo delle modifiche alla progettazione ed allo sviluppo.

Un controllo efficace richiede la definizione formale e l'attivazione di una procedura relativa al processo di gestione delle modifiche di progettazione (cambi): se una tale procedura non fa già parte del sistema di gestione dell'organizzazione allora deve essere definita all'avvio del progetto dal PM.

La procedura deve consentire di:

- Prendere decisioni appropriate su tempi, costi e valore (qualità) percepiti dal Cliente,
- Tracciare lo stato di tutti i cambi proposti assicurando così che nessuno di essi sia trascurato,
- Governare l'evoluzione della struttura fisica del prodotto attraverso una efficace relazione delle modifiche alla progettazione con il processo di gestione della configurazione software.

E' bene sottolineare, che un processo formale di cambi non deve essere visto come un ostacolo alle attività del progetto, ma un mezzo per governare l'evoluzione assicurando che solo i cambi appropriati siano adottati e gli impatti sull'intero progetto siano minimizzati.

Esempio di procedura

Tutti gli esempi riportati di seguito sono liberamente tratti dal capitolo 17 "Managing Change Request" in [SRWIEGER]

Cominciamo con la figura seguente che propone un indice tipico per una procedura di gestione dei cambi.

Introduzione
Ruoli e responsabilità
Stato delle richieste
Attività
Proporre una variante
Analizzare la variante
Implementare la variante
Testare la variante
Comunicare la variazione
Verifiche
Moduli

Figura 50 - Indice tipico di un piano di gestione dei cambi

La sezione "Introduzione" descriverà

- lo scopo del documento,

- l'obiettivo della procedura,
- i destinatari;

è necessario riportare anche altre informazioni di supporto alla lettura quali glossario, acronimi, abbreviazioni etc...

La sezione "Ruoli e responsabilità" elencherà per ogni ruolo le responsabilità per le attività definite. In particolare sarà necessario definire un comitato per la gestione dei cambi, comprendente il PM e i rappresentanti delle Parti interessate che avranno l'autorità per decidere sulle richieste di cambio.

Nella sezione "Stato della richiesta" verranno descritti gli stati che una richiesta di cambiamento può assumere. Nella figura si riporta un diagramma di transizione di stati per un processo di richiesta di cambio.

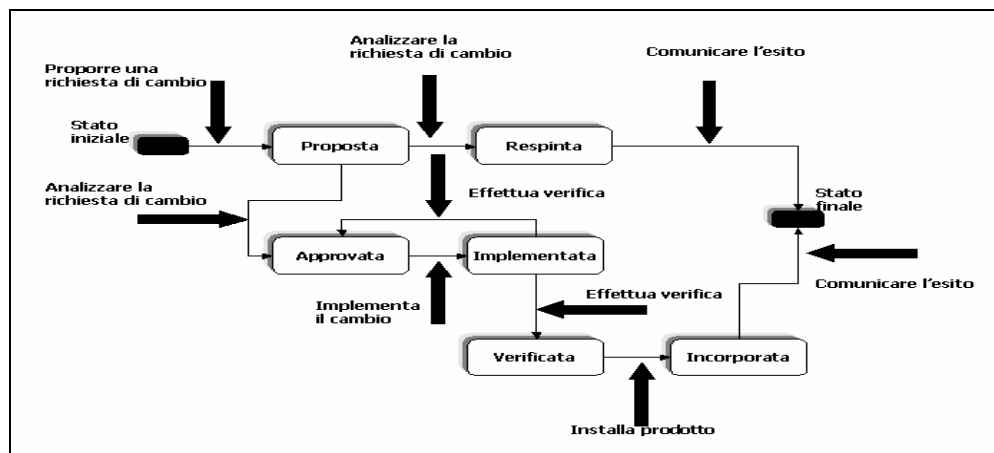


Figura 51 – Stati che una richiesta di cambio può assumere

La sezione "attività" descriverà per ogni compito del processo gli input, la sequenza d'operazioni, gli output e le regole (moduli e istruzioni) per l'operatività.

La sezione "verifiche" descriverà le attività di controllo che il PM effettuerà sugli oggetti (requisiti, contratto, documenti tecnici, moduli software, etc..) che sono coinvolti nella richiesta di cambio.

Nella sezione "moduli" infine sono elencati le strutture dei documenti di registrazione della qualità (Richiesta di cambio, Analisi d'impatto, Stato della richiesta, Comunicazione del cambio, statistiche, etc..).

Caso di studio

Richiesta variante				
ID Richiesta:	1			
Tipo richiesta:	Evoluzione			
Titolo richiesta:	Toolbar			
Priorità richiesta:	Alta			
Richiedente:	Data richiesta:	10-10-2001		
Descrizione richiesta:	Si deve consentire nella creazione dei rendiconti telefonici la possibilità di inserire gli oggetti da una barra degli strumenti inserendoli nel piano di lavoro			
Benefici:	Riduzione dei tempi nella realizzazione dei rendiconti telefonici			
Data prevista rilascio:	10-12-2001			
Stato Corrente:	Approvato			
Giornale di stato				
Stato	Data allo stato	Note allo stato	Responsabile	Data richiesta
Proposto	10-8-2001		Marketing	
Approvato	10-20-2001		PM	

Analisi di impatto			
ID Richiesta:			
Stima delle priorità	Alta	Benefici:	Soddisfazione del cliente,
		Penalità:	Nessuna
		Rischi:	Bassi
		Costi:	Normale
Costi previsti		Impegno:	6 mesi/p
		Tempi:	2 mesi
		Costi:	10000 euro
Impatti:		Qualità:	N.A.
		Altro:	
Dettaglio impegno			
Id.	Attività	Impegno previsto	
1.	Analisi	1 mese persona	
2.	Progettazione	1 mese persona	
3.	Realizzazione	2 mesi persona	
4.	Test	1 mese persona	
5.	Rilascio	15 gg persona	
6.	Adeguamenti documentazione	15 gg persona	

Figura 52 - Modulo per la richiesta di cambio/variante e analisi di impatto

Comunicazione	
Da:	Project Manager
A:	Richiedente richiesta di cambio
CC:	Parti interessate
Oggetto:	Richiesta di cambio n° 1
	La richiesta in oggetto è stata
	<u>approvata/respinta/verificata/incorporata/implementata</u>

Figura 53 - Esempio di comunicazione relativo allo stato di un cambio/variante

Quando i volumi delle informazioni da gestire sono cospicui un sistema informatico potrebbe essere un utile supporto al processo di gestione dei cambi.

Le richieste potrebbero essere formalizzate attraverso un sistema web-based.

Gli aggiornamenti e le valutazioni possono anch'esse essere effettuate via web, inoltre gli stati oltre ad essere disponibili via web possono essere comunicati mediante un sistema di notifica basato su e-mail.

In tal modo l'utente viene avvisato in tempo reale quando si ha un aggiornamento della richiesta di variante.

Una gestione basata su web offre più di un vantaggio:

- si migliora la **comunicazione**,
- si riducono i **tempi**,
- i dati raccolti nel database saranno utilizzati per calcolare velocemente gli indicatori necessari a controllare **l'ambito**.

Il sistema di indicatori

Il processo di gestione dei cambiamenti di progetto può essere rappresentato dai seguenti indicatori:

- Distribuzione delle varianti per stato (proposti, approvati, respinti, implementati, verificati, incorporati)
- Trend delle varianti nel tempo per stato
- Distribuzione delle varianti per origine (Marketing, Sviluppo, Direzione, etc..)
- Distribuzione delle varianti per area (requisiti, tempi, costi, qualità, rischi, acquisti, età ..)

La lista riportata non esaurisce il repertorio degli indicatori e la scelta dipende essenzialmente dalle politiche aziendali e dalle considerazioni che fa il PM all'inizio del progetto (costi per la raccolta dei dati e caratteristiche critiche del progetto).

Per un progetto software è importante che il PM si concentri su indicatori relativi al prodotto software in particolare per quanto riguarda la stabilità dei requisiti.

Essi sono:

- Numero totale dei requisiti nel tempo
- Numero delle varianti cumulate nel tempo,
- Numero di requisiti da definire nel tempo (TBD - To Be Defined)
- Numero di varianti per stato nel tempo
- Numero di varianti ai requisiti per origine

Caso di studio

Di seguito vengono dati degli esempi e le interpretazioni dei valori.

Nel grafico in figura è illustrato il legame tra le varianti ai requisiti nel tempo, il numero totale dei requisiti nel tempo e il numero di requisiti da definire nel tempo.

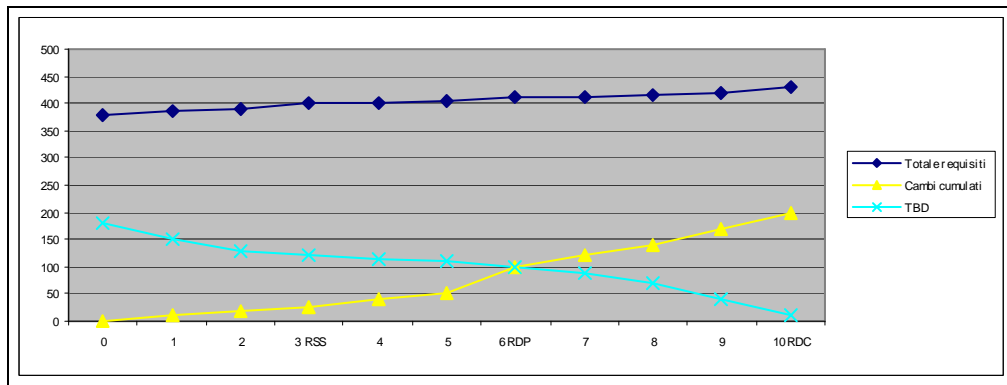


Figura 54 - Relazione tra le varianti ai requisiti nel tempo, il numero totale dei requisiti nel tempo e il numero di TBD.

L'analisi di questi valori porta a concludere che i requisiti TBD con l'avanzare del tempo tendono a zero, invece i cambi cumulati crescono nel tempo e ciò dà un'indicazione di instabilità del progetto.

Nel grafico nella figura seguente è illustrato l'andamento nel tempo dello stato dei requisiti.

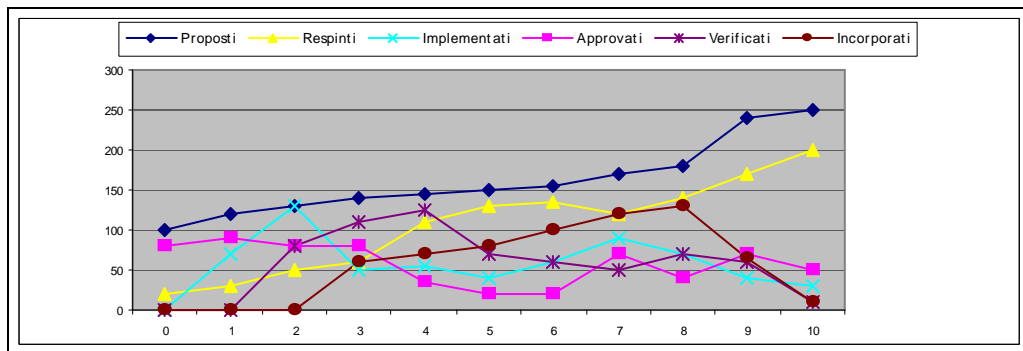


Figura 55 - Andamento dello stato dei requisiti nel tempo

Il grafico consente di effettuare una valutazione sul processo dei requisiti.

Nel grafico in figura è illustrato il collegamento tra le proposte di requisiti e la tipologia di utente/proponente.

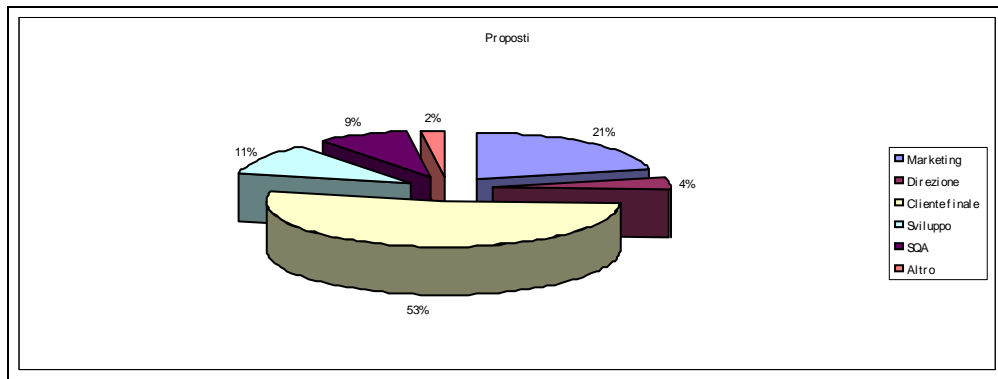


Figura 56 - Percentuale di distribuzione delle richieste ai requisiti per area funzionale aziendale

Quest'indicazione è utile perché consente di individuare quali sono i principali utenti del prodotto e conseguentemente intraprendere delle corrette strategie di CRM.

Controllo dei costi

Il calcolo degli scostamenti tra i costi preventivati e i costi effettivi consente di controllare il costo solo alla fine del progetto e non consente di fare, in corso d'opera, stati di avanzamento/previsioni a finire dei costi del progetto.

La tecnica dell'EARNED VALUE (valore assorbito), con l'introduzione della grandezza intermedia BCWP, supera questi limiti.

Tale tecnica [**FLEMING**] si basa su tre grandezze:

- Costo effettivo del lavoro eseguito (ACWP) – costi sostenuti,
- Costo previsto del lavoro eseguito (BCWP) – valore (dei costi) del lavoro effettivamente svolto calcolato ai costi previsti,
- Costo previsto del lavoro programmato (BCWS) – costi preventivati.

Queste tre grandezze possono essere espresse sia in euro, quando siano disponibili le informazioni sui costi, oppure in unità di impegno (ore/persona).

Attraverso opportune elaborazioni dei dati di base la tecnica dell'EARNED VALUE produce ulteriori informazioni quali:

- EAC (Estimate At Completion)
- BAC (Budget At Completion)
- CV (Cost Variance)
- VAC (Variance At Completion)
- CV% (Cost Variance Percent)
- SV (Schedule Variance)

- SV% (Schedule Variance Percent)
- CPI (Cost Performance Index)
- SPI (Schedule Performance Index)

Nella tabella seguente sono riepilogate le grandezze sopra citate, le rispettive formule e il loro significato.

I valori d'esempio si riferiscono ad un progetto nel quale

- il prodotto di progetto da realizzare è stato stimato pari a 100 moduli software ciascuno con un costo unitario di 22 giorni/uomo
- il processo di progetto è stato stimato pari a 2200 giorni/uomo (BAC).
- In pianificazione il PM ha previsto un numero di Progettisti costante dall'inizio alla fine del progetto e pari a 10.

Alla data corrente (130° giorno su una durata complessiva di 220 giorni – un anno solare) il piano prevede un costo del lavoro pari a 1300 ed una quantità di lavoro pari a 59 moduli = $1300/22$.

BCWS=1300

In esecuzione il gruppo di lavoro ha avuto un numero di Progettisti costante dall'inizio al momento corrente pari a 9 ed il costo effettivamente sostenuto è pari a $1170 = 9*130$.

ACWP=1170

La *differenza contabile* tra preventivo e consuntivo dei costi è $BCWS-ACWP=1300-1170=+ 130$: ho speso, ovviamente, meno del previsto.

Tale differenza contabile può trarre in inganno perché se è vero che abbiamo speso meno del previsto non siamo, però, in grado di dire se stiamo spendendo di più del dovuto, di meno oppure siamo in linea con le previsioni.

Con la tecnica dell'EARNED VALUE invece s'introduce la grandezza BCWP la quale indica il valore (dei costi) del lavoro effettivamente svolto calcolato ai costi previsti.

A quella data facciamo una verifica dello stato di avanzamento e rileviamo che abbiamo realizzato 50 programmi (Work Performed): se valorizziamo il lavoro fatto (BCWP) ai costi previsti nel budget otteniamo un valore $1100=50*22$.

La differenza $CV=BCWP-ACWP=1100-1170=-70<0$ mi dice che sto spendendo più del dovuto: l'informazione sulla perdita è data dalla differenza tra costo preventivato e costo sostenuto relativamente al lavoro effettivamente svolto.

L'indicatore $CV\% = [(BCWP - ACWP)/BCWP]*100 = [(1100-1170)/1100]*100=-6\%$ mi dice che la perdita accumulata ad oggi è circa il 6% del costo previsto.

L'indicatore $CPI = BCWP/ACWP = 1100/1170 = 0,94$ mi dice che la stima del costo effettivo dell'intero progetto (EAC) è pari a $1/CPI = 1,06$ volte la previsione (BAC): invece di costare 2200 (BAC) il progetto costerà $2340 = (1170/1100) * 2200$ - la stima della perdita finale è data da $VAC = BAC - EAC = 2340 - 2200 = 140$ (circa il 6% della previsione iniziale).

Il progetto non è finito: abbiamo realizzato il 50% del prodotto previsto (restano ancora 50 programmi da realizzare), abbiamo consumato circa il 53% del budget ed il trend indica che il progetto si concluderà in perdita.

Indicatore	Calcolo	Descrizione	Valore	Interpretazione
BCWS	Somma di tutti i BCWS dei package schedulati alla data	Quanto lavoro avrei dovuto fare? Quanto costa il lavoro che avrei dovuto fare?	1300	Alla data corrente il costo del lavoro preventivato è di 1400 e la quantità di lavoro è 63.6 moduli Prodotto e processo previsto alla data corrente
ACWP	Somma di tutti gli ACWP dei package schedulati alla data.	Quanto è costato il lavoro realizzato?	1170	Al tempo corrente sono stati spesi 1300. Stato di avanzamento del processo: il processo ha assorbito costi per una quantità pari a 1300
BCWP ₁	Somma di tutti i BCWP dei package schedulati alla data	Quanto lavoro è stato effettivamente realizzato? Del lavoro che ho effettivamente realizzato quanto budget ha speso?	1100	Sono stati realizzati 50 moduli. La realizzazione dei 50 moduli ai costi previsti vale 1100. Stato di avanzamento del prodotto: il prodotto ha assorbito valore per una quantità pari a 1100
BAC	Somma di tutti i BCWS	Qual è il costo totale previsto del progetto?	2200	Il costo previsto dell'intero lavoro è di 2200
CV	$BCWP - ACWP$	Qual è lo scostamento nei costi?	-70	Lo scostamento negativo indica la perdita accumulata fino all'istante di rilevazione
CV %	$[(BCWP - ACWP) / BCWP] * 100$	Scostamento percentuale nei costi	-6%	La percentuale negativa indica la stima della perdita di progetto rispetto alle previsioni iniziali
SV	$BCWP - BCWS$	Valorizzazione dello scostamento nei tempi	-200	Lo scostamento negativo indica la quantità di lavoro che manca rispetto alle previsioni
SV %	$[(BCWP - BCWS) / BCWP] * 100$	Scostamento percentuale nei tempi	-15%	La percentuale negativa indica la stima del ritardo rispetto alla durata inizialmente prevista
CPI	$BCWP / ACWP$	Indice di efficienza nei costi	0,94	Il rapporto tra costo previsto e costo effettivo rappresenta il fattore di riduzione della produttività inizialmente prevista - per ogni euro speso nel processo il prodotto assorbe valore per 84 centesimi
SPI	$BCWP / BCWS$	Indice di efficienza nei tempi	0,85	Il rapporto tra durata prevista e durata effettiva rappresenta il fattore di riduzione della tempestività - Per ogni mese trascorso il progetto accumula un ritardo di tre giorni
EAC	BAC/CPI	Quale sarà il costo finale stimato del progetto?	2340	Il valore indica il costo totale stimato a fine progetto
VAC	BAC - EAC	E' la differenza tra il costo totale preventivato e il costo totale stimato alla data per completare il progetto	-140	Lo scostamento negativo indica la perdita totale stimata al termine del progetto

Tabella 27 - Riepilogo dei principali indicatori utilizzati nella tecnica dell'earned value

Controllo dei tempi

La tipica forma dell'andamento dei costi rispetto al tempo consente di applicare la tecnica dell'EARNED VALUE anche al controllo dei tempi (schedulazione).

Riprendiamo l'esempio utilizzando le grandezze $SV/SV\%/SPI$.

La differenza $SV=BCWP-BCWS=1100-1300=-200<0$ mi dice che sto impiegando più tempo del dovuto: l'informazione sul ritardo è data dal valore della quantità di lavoro (200) che manca rispetto alle previsioni.

L'indicatore $SV\% = [(BCWP - BCWS)/BCWS]*100 = [(1100-1300)/1300]*100 = -15\%$ mi dice che il ritardo accumulato ad oggi è il 15% del tempo trascorso.

L'indicatore $SPI=BCWP/BCWS=1100/1300=0,85$ mi dice che la stima della durata effettiva dell'intero progetto (TE) è pari a $1/SPI=1,18$ volte la previsione (TS): invece di concludere nel giorno n°220 (TS) concluderò nel giorno n°260= $(1300/1100)*220$ - la stima del ritardo finale (TE-TS=40) è circa il $[(1300/1100)-1]*100=18\%$ della previsione iniziale (due mesi rispetto all'anno solare).

Il progetto non è finito: abbiamo realizzato il 50% del prodotto previsto ed il trend indica che il progetto si concluderà con una perdita pari a 140 e con un ritardo di circa due mesi.

commenti - il problema sulla produttività impedisce di fatto l'azione correttiva più semplice ed immediata e cioè aumentare la dimensione del gruppo di lavoro

Periodo	Data	BCWS	ACWP	PC	EAC	BCWP	CV	VAC	CV%	PS	CPI
1	02/07/2001	40	40	1%	4000	70,4	30,4	3.040	43%	1%	1,76
2	09/07/2001	1200	800	15%	5333	1056	256	1.707	24%	15%	1,32
3	16/07/2001	1250	1250	16%	7813	1126,4	-123,6	-773	-11%	16%	0,90
4	23/07/2001	2000	2000	20%	10000	1408	-592	-2.960	-42%	20%	0,70
5	30/07/2001	2100	2050	25%	8200	1760	-290	-1.160	-16%	25%	0,86
6	06/08/2001	2200	2100	30%	7000	2112	12	40	1%	30%	1,01
7	13/08/2001	2300	2150	32%	6719	2252,8	102,8	321	5%	32%	1,05
8	20/08/2001	2300	2300	34%	6765	2393,6	93,6	275	4%	34%	1,04
9	27/08/2001	3500	2400	35%	6857	2464	64	183	3%	35%	1,03
10	03/09/2001	4000	2500	36%	6944	2534,4	34,4	96	1%	36%	1,01
11	10/09/2001	4100	3000	40%	7300	2816	-184	-460	-7%	40%	0,94
12	17/09/2001	4200	3500	45%	7778	3168	-332	-738	-10%	45%	0,91
13	24/09/2001	4250	4000	50%	8000	3520	-480	-960	-14%	50%	0,88
14	01/10/2001	4900	4100	55%	7455	3872	-228	-415	-6%	55%	0,94
15	08/10/2001	5000	4500	65%	6923	4576	76	117	2%	65%	1,02
16	15/10/2001	5700	5000	67%	7463	4716,8	-283,2	-423	-6%	67%	0,94
17	22/10/2001	5800	5100	70%	7286	4928	-172	-246	-3%	70%	0,97
18	29/10/2001	6000	5400	85%	6353	5984	584	687	10%	85%	1,11
19	05/11/2001	6500	5500	90%	6111	6336	836	929	13%	90%	1,15

Figura 57 - Foglio elettronico contenente i principali indicatori dell'earned value per il caso di studio in esame.

La figura precedente illustra un foglio excel che consente di tenere il progetto sotto controllo mediante la tecnica dell'earned value.

Essa riporta per ogni periodo tutte le indicazioni relative a CV (Varianza dei costi, VAC varianza a completamento, etc... in funzione di tre sole grandezze ACWP,BCWP e BCWS.

I colore giallo rappresenta una situazione di stabilità per il progetto o comunque i valori ottenuti sono in un intervallo di "normalità", il verde indica buone performance e il rosso indica criticità.

caso di studio

Rossi utilizza un foglio excel (Figura 57) per il controllo dei tempi e dei costi .Ogni settimana per un totale di 21 periodi pianificati rileva le ore consumate (ACWP) e la l'earned Value (BCWP).

Le figure seguenti (Figura 58 e 59) consentono di tenere sotto controllo il progetto.

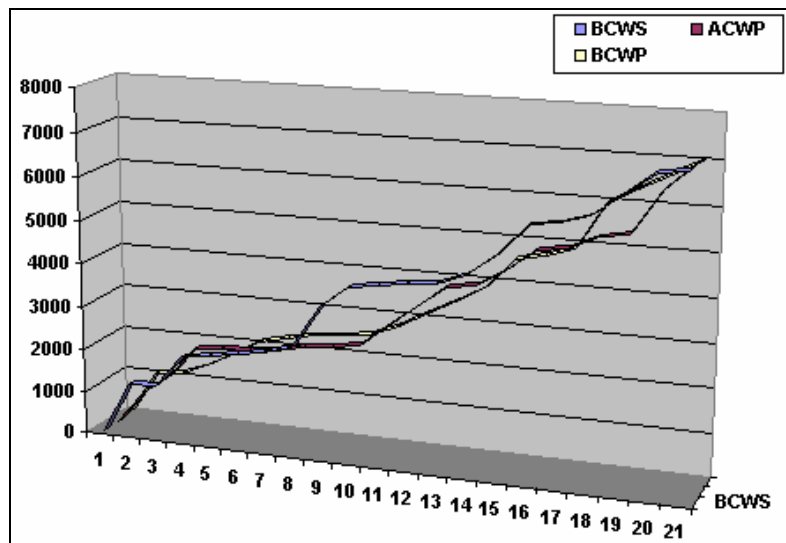


Figura 58 - Grafico indicante i valori di ACWP,BCWP e BCWS relativo al foglio di calcolo del caso di studio in esame.

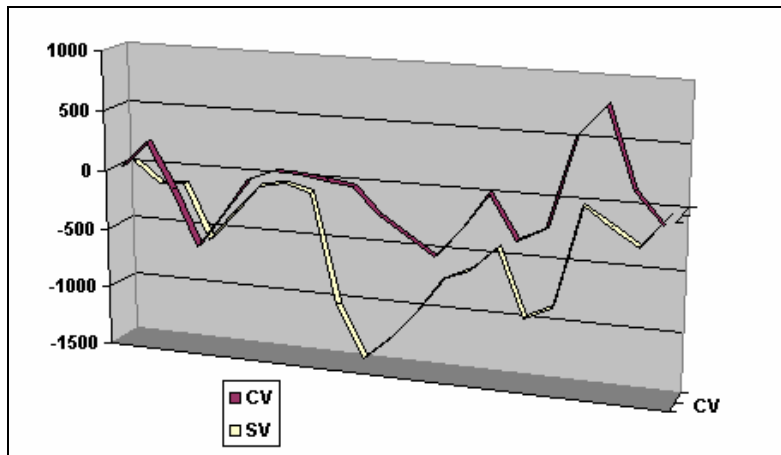


Figura 59 - Grafico indicante i valori di CV ed SV relativo al foglio di calcolo del caso di studio in esame.

Il grafico in figura illustra l'andamento nel tempo della varianza dei costi (CV) e della varianza della schedulazione (SV).

Controllo della qualità del prodotto

Il PM controlla la (non)qualità del prodotto di progetto mediante l'analisi dei dati raccolti sui difetti rilevati grazie soprattutto alle attività d'assicurazione della qualità.

Di seguito si riportano le informazioni sul difetto che rivestono un fondamentale interesse per il controllo della qualità:

- Quando (in quale fase di lavoro) è stato riscontrato (Ispezioni, test unitario, test integrazione, test di sistema, test utente, esercizio)
- Quando è stato iniettato (requisiti, disegno, costruzione, Ispezioni, test unitario, test integrazione, test di sistema, test utente, esercizio)
- Dove (in quale componente software) è stato riscontrato
- Qual'è la causa che ha originato il difetto
- Qual è la gravità del problema - classificazione del difetto.

Tali informazioni consentono di definire un insieme d'indicatori che permettono di controllare la qualità del prodotto software che si sta realizzando.

La registrazione di queste informazioni in un database consente un'analisi più efficiente.

Per controllare la qualità è possibile utilizzare i seguenti indicatori:

- Distribuzione dei difetti per componente (subsystem o modulo sw)

- Distribuzione dei difetti per origine
- Numero dei difetti riscontrati per fase
- Numero dei difetti prodotti per fase

Distribuzione dei difetti per componente

Nella figura sottostante il grafico rappresenta la distribuzione dei difetti per componente o sottosistema.

Tale indicatore ha lo scopo di individuare i componenti software o i sottosistemi software che hanno il maggior numero di difetti.

Le informazioni fornite dalla distribuzione consentono di:

- Distribuire meglio i Progettisti nella correzione dei difetti
- Migliorare la qualità complessiva del prodotto
- Decidere sulla rilavorazione parziale/totale del componente.

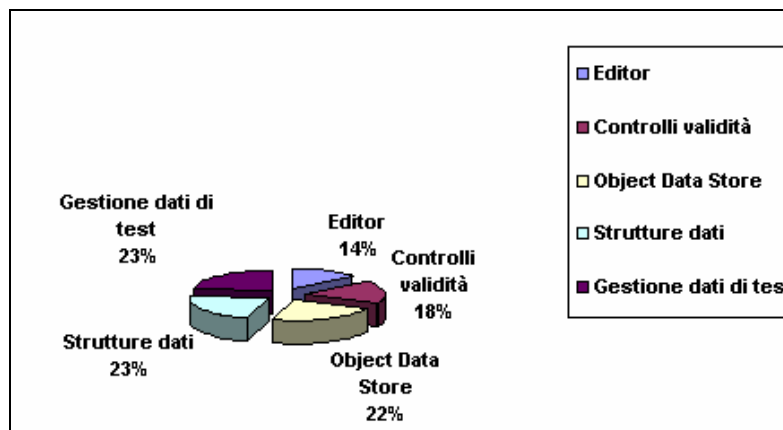


Figura 60 – Distribuzione dei difetti per CSCI del caso di studio in esame

Distribuzione dei difetti per origine

Nella figura sottostante il grafico rappresenta la distribuzione delle cause dei difetti

Tale indicatore ha lo scopo di individuare le cause principali dei difetti.

Le informazioni fornite dalla distribuzione consentono di:

- Migliorare il processo di sviluppo (ad es. se i difetti sono causati dai requisiti, ciò può significare che è necessario migliorare il processo sui requisiti)

- Migliorare le prestazioni dei Progettisti (es. se vi sono molti difetti dovuti a causa d'errori di programmazione, questo può significare che gli sviluppatori hanno necessità di approfondire le tecniche di programmazione)

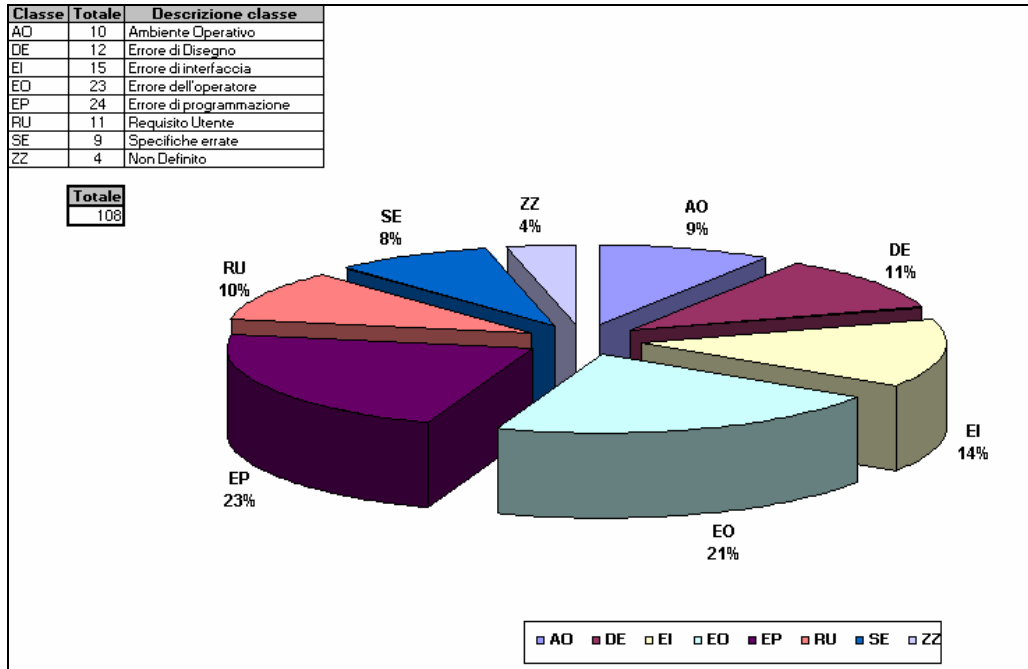


Figura 61 – Distribuzione delle cause dei difetti del caso di studio in esame

Distribuzione dei difetti per fase

Nella figura precedente il grafico rappresenta la distribuzione dei difetti per fase di collaudo.

Tale indicatore ha lo scopo di fornire indicazioni sull'efficacia delle attività d'assicurazione della qualità.

Se analizziamo attentamente il grafico, notiamo che gli errori (NOK) diminuiscono con il procedere delle fasi di test. Questo può significare che il Test è stato efficace, viceversa se invece nel passare da una fase all'altra gli errori aumentano, vuol dire che i test in fase di UNIT non sono stati efficaci oppure ne sono stati disegnati pochi.

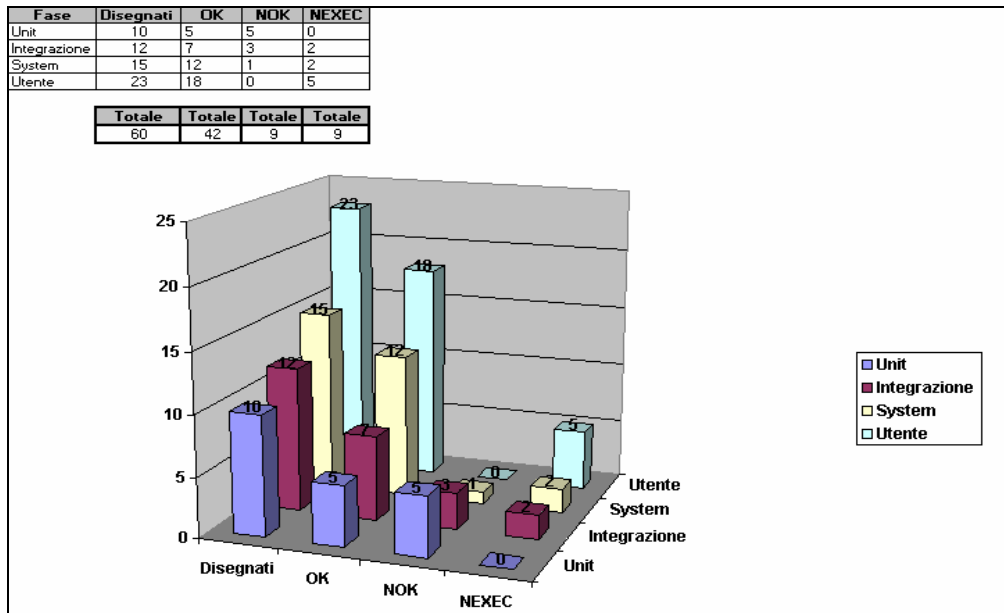


Figura 62 – Grafico di esempio relativo all’andamento dei difetti aperti e chiusi per fase del caso di studio in esame

Andamento dei difetti nel tempo

In figura il grafico riporta l’andamento dei difetti aperti e chiuso nel tempo. Esso da due utili informazioni, una riguardante il prodotto e l’altra riguardante le performance del gruppo di lavoro nel risolvere le anomalie. Nel primo caso è evidente che se il numero di anomalie aperte nel tempo aumenta ciò può significare che le anomalie o non sono corrette efficacemente oppure che il prodotto non è stabile.

Caso di studio

Rossi utilizza per il controllo l’elenco delle anomalie aperte ordinate per gravità decrescente

Elenco delle anomalie aperte Black List								
ID	Area	Gravità anomalia	Ranking	Oggetto	Affidata a	Durata prevista (hh)	Data prevista consegna	Ritardo
309	INCASSI	Molto Alta	125	(214*) - Incasso pendente				
281		Alta	80	Err. SIACR (N°103) - errata distribuzione automatica (atop) prot. 538				
284		Normale	75	Err. SIACR (LEGATO ALLA N°20) - c-ma ab prot. 541				
314	SCRITTURE CONTABILI	Normale	45	Err. SIACR (N°118) - scritture contabili "ANOMALE" prot. 598				
300	INCASSI	Normale	9	Enhancement SIACR (n°211) - CMA 05 anziché CMA 30 per faccodezza derivante dal rigo TP				
221		Normale	9	Err. SIACR (N°78) - scritture contabili relative all'arrotondamento prot. 459				
312	SCRITTURE CONTABILI	Bassa	2	(204*) - No mi DS regola bilica contabile				
319	ALTRA AREA	Normale	0	Dibattimento dati db. Anomalia interna. Rivolta da Domenico Amato.				
490	INCASSI	Normale	0	I. Enhancement SIACR (n° 203) - mancato recupero TCG scaricata a Sage in presenza di CI 29 PROT. 872				
419		Normale	0	anomalia (mia n° 180) - nell'fiche con importo a recupero tog a V - sistemazione pregresso prot. 181				
379		Normale	0	Enhancement SIACR (N°144) - annullamento contabile ed annullamento parziale dei relativi pagamenti - da parte di Luzzo prot. 887				

Figura 63 – giornale delle anomalie critiche (Black List)

in figura l'elenco consente al PM di ottimizzare il rapporto costo/prestazione: da un lato le risorse (limitate) di progetto vanno concentrate sulle anomalie di gravità elevata, dall'altro è necessario conoscere esattamente le anomalie (di gravità bassa) che si intendono rilasciare al cliente.

Caso di studio

Rossi gestisce tutte le anomalie del prodotto di progetto attraverso il giornale delle anomalie aperte

Giornale anomalie aperte						
ID Anomalia	Classe Errore	Descrizione origine	Descrizione anomalia		Elapsed	Durata
379	ZZ	Da Definire	Aperta	10/11/00 11.41.57		0
419	ZZ	Da Definire	Check Aperta	02/01/01 11.38.54 23/11/00 8.37.42	L'anomalia non è legata all'ABPS legata all'abps	0
483	ZZ	Da Definire	Aperta	19/12/00 9.57.39		0
231	EP	Dati incorretti o definiti in maniera inconsistente	Enh. SIA/CR(N°76) - scritture contabili relative all arrotondamento prot.469			0
		Sospesa	15/12/00 15.40.27	Perché Landi è su un nuovo requisito ABPS.		
		Chiusa	07/11/00 10.59.18			
		Aperta	10/10/00 9.27.56			
		Lavorazione	16/10/00 11.51.32			
		Assegnazione iniziale	12/10/00 12.49.44			
		Aperta	21/11/00 17.09.35			
		Diagnosi	10/10/00 14.52.26			
503	ZZ	Da Definire	Aperta	03/01/01 17.27.48		0
314	EP	Dati incorretti o definiti in maniera inconsistente	Enh. SIA/CR (N°118) - scritture contabili "ANOMALE" prot.586			0
		Lavorazione	06/11/00 9.52.22			
		Aperta	19/12/00 11.47.13			
		Aperta	23/10/00 14.06.56			
		Assegnazione iniziale	26/10/00 14.46.29			
		Chiusa	07/12/00 15.31.37			
294	EP	Dati incorretti o definiti in maniera inconsistente	Enh. SIA/CR(LEGATO ALLA N°20) - oma ab prot.541			0
		Chiusa	31/10/00 17.41.18	Chiusa dall'ABPS del 27/10/2000		

Figura 64 – Giornale delle anomalie aperte

In figura il giornale riporta per ogni anomalia identificativo, origine, stato corrente, descrizione e l'assegnatario. Il gruppo di sviluppo immette i dati elementari nel database dei difetti durante l'esecuzione. Questo flusso informativo serve a tenere sotto controllo il progresso delle anomalie senza dover aspettare la successiva riunione.

Caso di studio

Rossi analizza l'andamento delle anomalie nel tempo attraverso il grafico che rappresenta il numero delle anomalie aperte e di quelle chiuse per ciascun giorno di un determinato periodo di progetto.

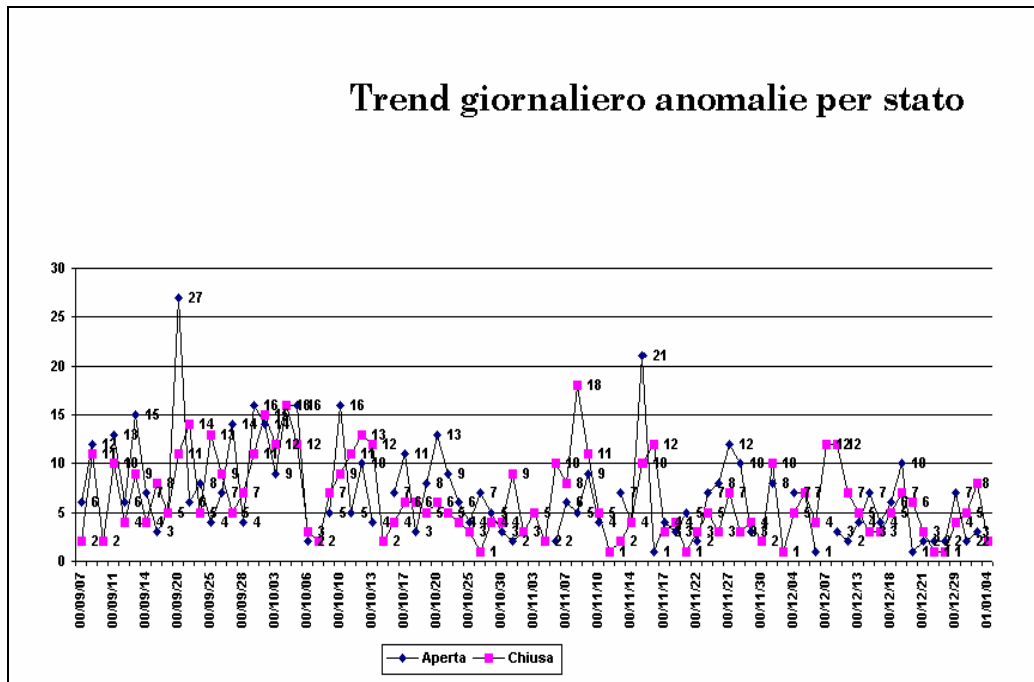


Figura 65 – Trend delle anomalie Aperte e chiuse nel tempo

il grafico in figura consente al PM di interpretare il fenomeno, risalire alle cause ed intraprendere eventuali azioni correttive; ad esempio: se vengono aperte molte anomalie, questo è indice di un problema nella progettazione di qualche componente, se le anomalie vengono chiuse celermente invece è indicazione che le anomalie sono facilmente risolvibili.

Caso di studio

Rossi analizza come varia la durata della risoluzione di ciascuna anomalia attraverso il grafico che riporta sull'asse delle ascisse i giornie sull'asse delle ordinate il numero di anomalie

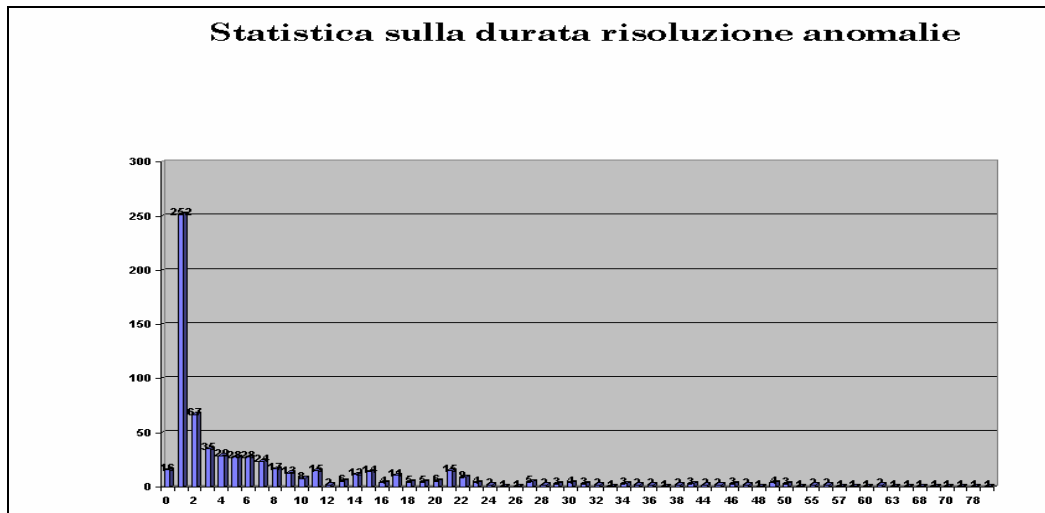


Figura 66 – Statistica sulla durata delle risoluzioni delle anomalie

il grafico in Figura consente al PM di valutare la capacità del gruppo di lavoro nell’elaborazione delle anomalie.

Caso di studio

Rossi rappresenta lo stato delle anomalie alla data corrente attraverso il grafico che riporta sull’asse delle ascisse la durata della risoluzione e sull’asse delle ordinate il numero di anomalie risolte

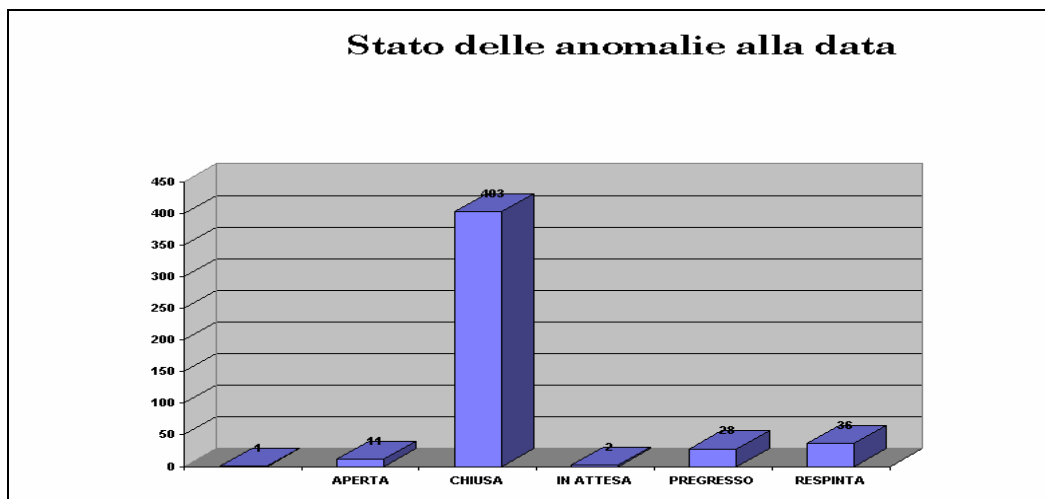


Figura 67 – Stato delle anomalie ad una certa data

il grafico consente al PM di capire istante per istante quanti difetti sono in un determinato stato.

Questo consente di capire anche quanto effort bisognerà ancora spendere per correggere i difetti.

Spesso tali valori vengono richiesti dall'utente.

Controllo sul processo di gestione delle Risorse Umane

Per esercitare un controllo adeguato sul processo di gestione delle Risorse Umane il PM deve impegnarsi a:

- Controllare periodicamente l'avanzamento di conoscenze, competenze e motivazioni
- Riconoscere i risultati raggiunti
- Attivare eventuali azioni correttive.

Caso di studio.

Rossi ha deciso di controllare la gestione delle Risorse Umane attraverso quattro statistiche:

- 1 risorse pianificate ed attuali alla data,
- 2 distribuzione delle risorse per fase,
- 3 staff turnover,
- 4 presenze/assenze.

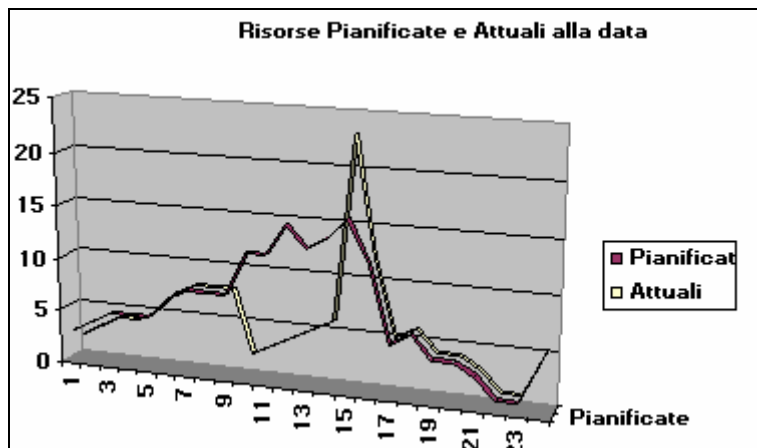


Figura 68 - Risorse pianificate e Attuali alla data

risorse pianificate ed attuali alla data - andamento della dimensione della squadra.

il processo di allocazione/deallocazione delle Risorse Umane sul progetto si sviluppa attraverso forme di negoziazioni tra PM e Direzione.

Da un lato c'è la capacità della Direzione di mettere a disposizione del progetto Risorse adeguate ai requisiti specificati.

Dall'altro lato c'è la capacità del PM di integrare nella squadra le Risorse disponibili secondo i tempi del piano di progetto.

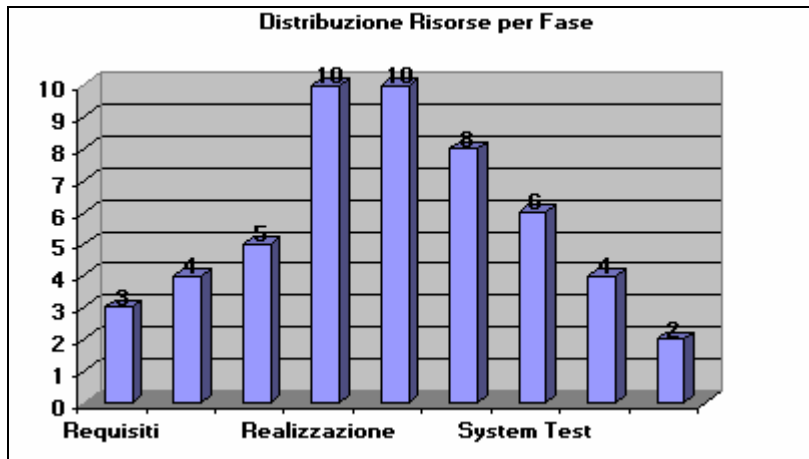


Figura 69 – Distribuzione delle risorse per fase

distribuzione delle Risorse per fase – andamento dell’impegno sui processi di sviluppo del modello di ciclo di vita.

La distribuzione dell’impegno effettivo sui processi di sviluppo tipici del modello tecnologico adottato rappresenta il modo con cui il PM ha utilizzato le competenze disponibili.

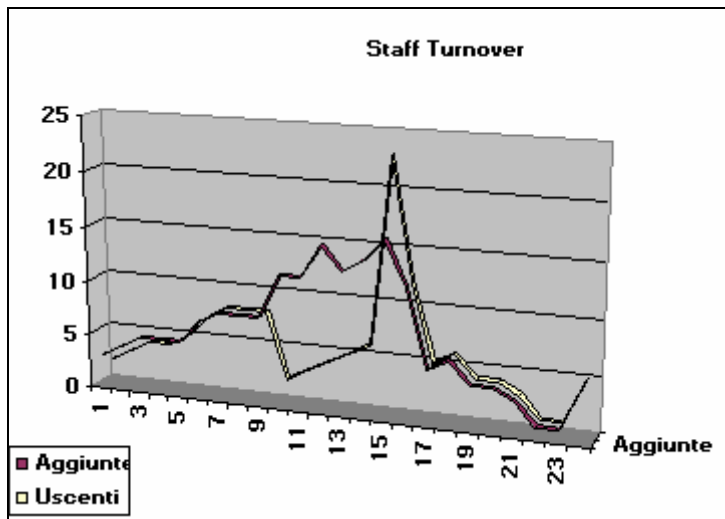


Figura 70 – staff Turn-Over

staff turnover – andamento delle variazioni (entrate ed uscite) della squadra.

Il turn-over di progetto fornisce indicazioni sulla stabilità

- una squadra compatta capace di assorbire nel tempo i cambiamenti (previsti/imprevisti) nella formazione è una garanzia di successo,

- un alto turn-over impedirà la creazione dei legami e delle relazioni interne alla squadra creando problemi di competenze e di capacità.

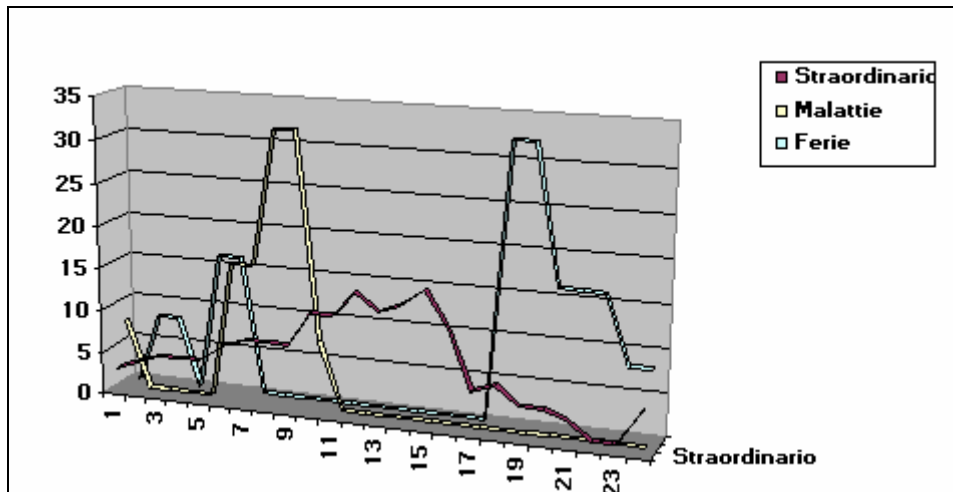


Figura 71 - Andamento dello straordinario, ferie e malattie

presenze/assenze.

Il confronto tra l'andamento di ferie, malattie e straordinario del progetto e quello dell'azienda nel suo complesso consente al PM di individuare tempestivamente sintomi di malessere e/o disaffezione delle Persone (clima di progetto).

Oltre alle statistiche presentate è opportuno che il PM controlli con l'aiuto della funzione aziendale preposta l'andamento della formazione specifica di progetto:

la soddisfazione del personale impegnato nella formazione fornisce indicazioni sulla motivazione al lavoro e sulla condivisione degli obiettivi.

Risoluzione dei problemi

In esecuzione abbiamo notato come i processi componenti del processo di sviluppo (primari) siano sostenuti dai processi di supporto ed in particolare sono stati presi in considerazione Verifica e Validazione, Gestione della Configurazione Software, Gestione della qualità.

Per esercitare un controllo efficace il PM si avvale costantemente di un ulteriore processo di supporto: il processo di risoluzione dei problemi.

La risoluzione dei problemi è un processo dedicato a rilevare, analizzare e risolvere/gestire i problemi che il progetto eredita, crea e trasmette nel corso della sua vita.

Nelle aree della gestione di progetto tipiche del controllo (rischio, ambito, costi, tempi, qualità, Risorse Umane) abbiamo cercato di collegare l'analisi dei dati con le azioni correttive e preventive attraverso l'integrazione del processo di risoluzione dei problemi nell'azione di controllo. Sebbene ciascun tipo di controllo possieda proprie specificità, è possibile individuare i caratteri generali del processo di risoluzione dei problemi (ISO/IEC 12207 1995):

- definizione del meccanismo di cattura del problema e di inserimento nel processo di gestione con
 1. criteri di attivazione
 2. criteri di comunicazione
 3. attenzione alle cause ed impegno alla rimozione di esse
 4. criteri di tracciamento dello stato del problema
 5. criteri di registrazione dei dati di base;
- definizione dello schema di classificazione e di prioritizzazione del problema;
- orientamento dell'analisi verso l'obiettivo di individuare le linee di tendenza sottostanti alle manifestazioni dei problemi;
- definizione dei meccanismi di sorveglianza del problema e delle risoluzioni adottate e collegamento con i processi di miglioramento.

Riconosciuta l'appartenenza del problema al progetto, il PM tenterà di gestirlo secondo i metodi predefiniti: se le risoluzioni si dimostrassero inadeguate alle esigenze il PM attiverà le procedure di escalation, che scalano, cioè, la struttura organizzativa verso l'alto passando il problema al Senior Management.

Riepilogo

Un'Organizzazione che non ha problemi né difetti o è fortunata o totalmente immatura e comunque non ha bisogno di controllo. Un'Organizzazione matura conosce i propri difetti ed è capace di mantenere la compatibilità tra efficacia (livello di difettosità) ed efficienza (struttura dei costi di prevenzione, controllo ed insuccesso). L'osservazione rimane assolutamente valida anche quando l'applichiamo al progetto: il Capo-progetto è il responsabile del controllo ovvero del complesso equilibrio che si deve stabilire tra ambito, tempi, costi e qualità.

Chiusura del progetto software

- 2.1 Chiusure tecniche**
 - 2.2 Rilascio del personale**
 - 2.3 Chiusure contrattuali**
 - 2.4 Chiusure amministrative**
 - 2.5 Riepilogo**
-

“O capitano, mio capitano, il nostro viaggio tremendo è terminato. La nave ha superato ogni ostacolo, l’ambito premio è conquistato, vicino è il porto...” (Walt Whitman)

La chiusura s’innesca quando almeno una delle Parti interessate al progetto vede ormai prossima la fine del percorso.

L’Organizzazione cattura l’evento e la Direzione può decidere di chiudere il progetto: definisce gli obiettivi di chiusura ed incarica il Capo-progetto.

Il Capo-progetto incaricato pianifica le operazioni di chiusura in termini di tempi ed impegni e si attiva per i vari passaggi: il prodotto software di progetto passa dallo sviluppo all’esercizio, le informazioni su prodotto e processo passano dai vecchi ai nuovi responsabili, le risorse passano dal progetto ad altre attività aziendali.

I controlli sono incentrati sulla definizione dei rapporti tra progetto e funzioni aziendali: il Capo-progetto assiste la Funzione Commerciale nella chiusura di ogni residuo adempimento del contratto di fornitura, la Funzione Acquisti nella chiusura di ogni residuo adempimento dei contratti di approvvigionamento ed in generale facilita l’Organizzazione nell’acquisire le eredità del progetto.

Controlli e passaggi vanno avanti fino all’esaurimento di tutte le attività: in conclusione il Capo-progetto formula i consuntivi, li confronta con i preventivi, elabora gli scostamenti, offre la propria interpretazione dei risultati finali e chiede l’approvazione della Direzione.

Introduzione

La Direzione dell'Organizzazione può decidere di chiudere il progetto in corso quando:

Il Cliente accetta il prodotto software realizzato,

La Funzione aziendale responsabile propone la dismissione del prodotto software in sviluppo/manutenzione perché ne ritiene esaurito il ciclo di vita sul mercato,

I fondi disponibili per il progetto si sono esauriti.

In analogia con la definizione in apertura, la chiusura di un progetto software può essere considerata a tutti gli effetti come un nuovo progetto limitato ad elaborare i risultati finali del processo di sviluppo.

L'approccio dell'organizzazione alla chiusura varia in funzione delle dimensioni in gioco:

in grandi organizzazioni/grandi progetti la direzione di solito formalizza la chiusura in un nuovo progetto e lo affida ad un nuovo PM,

in piccole organizzazioni/piccoli progetti il PM che ha condotto il progetto compie le operazioni di chiusura e chiede la approvazione alla direzione,

nelle situazioni intermedie si può ipotizzare una sequenza del tipo - il PM propone la chiusura, l'ente qualità interno la verifica, la direzione l'approva.

I motivi dell'attenzione della Direzione al processo di chiusura possono essere diversi. Ad esempio:

in caso di successo la direzione è interessata ad estendere le buone pratiche di gestione sperimentate nel progetto in chiusura a progetti simili,

in caso di insuccesso la direzione è interessata a risalire alle cause dei problemi per prevenire il ripetersi di inconvenienti simili.

La chiusura può incontrare nell'Organizzazione attriti e resistenze; per chiarire il discorso estremizziamo oltre i limiti del buon senso l'esempio:

se siamo di fronte ad un successo, è possibile che i componenti del gruppo di progetto, per paura delle future allocazioni, coprano l'esaurimento del lavoro creando ulteriori impegni nella speranza di procrastinare la fine naturale delle attività;

se siamo di fronte ad un insuccesso, è possibile che i componenti del progetto occultino i problemi per evitare che la direzione possa risalire a specifiche responsabilità.

Comunque sia una chiusura di progetto insufficiente impedisce all'Organizzazione di acquisire conoscenze fondamentali per migliorare e competere.

Gli obiettivi di chiusura sono, dunque, fundamentalmente obiettivi di conoscenza: la Direzione ha bisogno di sapere che fine hanno fatto i propositi di partenza e quale equilibrio è stato trovato tra aspetti contrattuali, economici, commerciali e tecnologici.

Il PM pianifica la chiusura definendo tempi ed impegni delle principali attività che ricadono in linea di massima nell'area della gestione dell'informazione e della comunicazione del progetto:

- Chiusure tecniche
- Rilascio del Personale
- Chiusure contrattuali
- Chiusure amministrative
- Riepilogo.

Chiusure tecniche

Il prodotto software di progetto passa dallo sviluppo all'esercizio attraverso il processo di collaudo.

Al termine del collaudo d'accettazione il Cliente decide se le caratteristiche del prodotto software fornito soddisfano i requisiti utente concordati.

In tal caso si registra formalmente la comunicazione dell'accettazione del prodotto e tale formalizzazione consente di iniziare la chiusura di progetto.

Se le ipotesi di chiusura sono diverse dall'accettazione del cliente sarà la direzione dell'organizzazione a **formalizzare** l'inizio della chiusura.

Il primo passo della chiusura di progetto è la chiusura tecnica strutturata nelle seguenti attività:

- Installazione del prodotto software
- Consolidamento in configurazione della release software installata
- Consegna dei documenti tecnici
- Chiusura del dossier della qualità

Installazione del prodotto software

L'installazione del prodotto software può essere svolta da uno dei seguenti enti

dal gruppo di progetto,

da un diverso gruppo di lavoro dell'organizzazione specializzato nell'assistenza al cliente,

dal cliente stesso.

In tutti i casi, il PM dovrà consegnare al responsabile operativo dell'installazione tutti i componenti software del prodotto da installare corredati da un manuale d'installazione ed una nota di rilascio del software.

La nota di rilascio conterrà le seguenti informazioni:

- definizione della release del prodotto software

- Principali caratteristiche della release
- Elenco dei componenti software e rispettive versioni
- modalità di archiviazione/conservazione.

Il livello di dettaglio delle informazioni sarà adeguato alle caratteristiche professionali dell'operatore incaricato dell'installazione.

Consolidamento della release software

È buona norma che il PM consegni gli oggetti necessari all'ente responsabile dell'installazione prelevandoli dalla libreria (statica) dell'organizzazione: ciò implica che prima della consegna deve essere consolidata sotto controllo di configurazione una baseline del prodotto (attività prevista nel piano di gestione della configurazione).

L'uso di strumenti informatici di supporto al software configuration management faciliterà e semplificherà le operazioni.

Oltre alle tecniche abilitanti è di fondamentale importanza che l'organizzazione adotti una procedura che disciplini l'intero processo assegnando le responsabilità delle operazioni.

Ad esempio la responsabilità del prelievo per consegna potrebbe essere assegnata al software configuration manager, mentre la responsabilità del consolidamento della release software potrebbe essere a carico del software project manager.

Consegna dei documenti tecnici

La documentazione tecnica è parte integrante del prodotto software. Generalmente gli accordi contrattuali specificano i documenti da consegnare insieme con il software applicativo.

Un elenco tipico dei documenti da consegnare potrebbe essere il seguente:

- Documento dei Requisiti Utente
- Documento di Specifiche Funzionali
- Documento di Disegno Architettuale
- Documento di Disegno di Dettaglio
- Documentazione di Verifica e Validazione
- Manuale di installazione
- Manuale Utente.

In analogia con la configurazione anche la documentazione evolverà attraverso un processo aziendale ordinato e definito basato sia su strumenti informatici (sistemi di gestione dei documenti in formato elettronico) sia su procedure organizzative che diano la certezza che il documento giusto sia distribuito alla persona giusta nel tempo e nel luogo previsto dall'operatività dei processi clienti del processo di sviluppo.

Chiusura del dossier della qualità

Il dossier della qualità raccoglie registrazioni, documenti ed evidenze del processo di gestione della qualità eseguito secondo il piano (controlli e monitoraggi - assicurazione della qualità).

La struttura del dossier riflette dimensione e complessità del progetto; in generale può contenere le seguenti informazioni:

- Piano
- Esiti dei controlli
- esiti delle verifiche ispettive condotte sul progetto,
- evidenze della gestione di non conformità ed azioni correttive
- misurazioni di prodotto
- misurazioni di processo
- analisi statistiche sulla difettosità.

La chiusura del dossier della qualità consiste nel verificare che il profilo di qualità del prodotto sviluppato sia adeguato alle fasi del ciclo di vita (esercizio, manutenzione) successive alla fase di sviluppo.

Solo dopo aver accertato che tutto sia in ordine è possibile chiudere il monitoraggio del progetto passando il dossier della qualità all'ente responsabile del monitoraggio dell'esercizio (gestione della difettosità in garanzia, validazione di progettazione, gestione dei livelli di servizio della manutenzione e dell'assistenza).

La chiusura del dossier della qualità dovrebbe avvenire in modo predefinito o secondo il citato piano, o secondo una procedura organizzativa: comunque è necessario disciplinare i passaggi di responsabilità all'interno dell'organizzazione.

Rilascio del personale

Con l'esaurimento delle attività di sviluppo gradualmente tutte le Persone che compongono il gruppo di progetto abbandonano la squadra.

Il processo di rilascio del personale è strutturato nelle seguenti attività:

- Verifica delle attività correnti per ogni Progettista e definizione di un piano di uscita,
- valutazione della prestazione
- valutazione delle competenze finali
- Riunione di rilascio

Verifica degli impegni individuali e definizione di un piano di uscita

Per ogni Componente del gruppo il PM dovrà verificare su quali attività è coinvolto e determinare il momento in cui potrà uscire dal progetto senza conseguenze negative per il cliente.

Il PM dovrà formulare un piano di uscita delle persone dal progetto e chiederne l'approvazione alla Direzione: in risposta, oltre alla condivisione/aggiornamento del piano, otterrà informazioni sulle future allocazioni.

Un esempio di piano di uscita è dato di seguito:

ID	Nominativo	Attività corrente	Data fine prevista attività
1.	Falco	Redazione del manuale utente	08/01/2001
2.	Delfino	Costruzione del kit d'installazione	15/01/2001

Tabella 28 – Piano di uscita per il caso di esempio in esame

Valutazione della prestazione

La valutazione della prestazione complessiva di ciascuna Persona si mantiene in linea con quanto affermato in pianificazione ed esecuzione: i processi di valutazione della prestazione di progetto sono processi di breve periodo distinti anche se collegati ad analoghi processi aziendali di medio lungo periodo.

Sarà poi compito della funzione aziendale preposta mettere in relazione la prestazione individuale con il concetto di retribuzione di risultato e la serie storica delle prestazioni individuali con il concetto di avanzamento di carriera.

Nei riesami di progettazione e negli stati di avanzamento il PM rileva la prestazione individuale in corso d'opera: in chiusura raccoglie e sintetizza le informazioni elementari esprimendo una valutazione complessiva.

Valutazione delle competenze finali

Il PM registra nel sistema informativo di progetto le competenze finali e dispone per ciascun individuo della differenza tra le competenze in entrata e le competenze in uscita.

Le informazioni elementari raccolte saranno trasmesse alla funzione aziendale preposta per consentire l'aggiornamento dello skill.

Oltre ai dati elementari il PM esprime un commento ed una valutazione sul grado di competenza acquisito da ciascuna Persona per consentire nel futuro la migliore utilizzazione del patrimonio tecnologico aziendale.

Riunione di rilascio

L'obiettivo principale della riunione di rilascio è di sancire l'uscita dal progetto e di comunicare le prossime assegnazioni dei Componenti del gruppo di lavoro. È auspicabile che questa riunione sia anche occasione, non formale, per ringraziare le persone dell'impegno profuso e per riconoscere il valore dimostrato sul campo.

Chiusure contrattuali

Le chiusure contrattuali riguardano le definizioni dei rapporti tra progetto e Parti esterne all'Organizzazione.

Nella chiusura del contratto di fornitura il PM assiste la Funzione Commerciale nella verifica che tutti gli obblighi contrattuali siano stati rispettati e nella chiusura di ogni residuo adempimento verso il cliente/committente/utente.

Nella chiusura dei contratti di approvvigionamenti il PM assiste la Funzione Acquisti nella chiusura di ogni residuo adempimento dei contratti di approvvigionamento: è fondamentale fornire alla direzione anche un'attenta valutazione sul lavoro e sulla professionalità del fornitore perché ulteriori collaborazioni possano essere decise sulla base di dati di fatto.

Chiusure amministrative

Le chiusure amministrative riguardano le definizioni dei rapporti tra progetto e Funzioni dell'Organizzazione.

Le attività da svolgere riguardano principalmente:

- *Rilascio delle dotazioni* – è buona pratica rendere disponibili ad altri gruppi di lavoro Postazioni di lavoro, software, manuali, ...
- *Invio della documentazione relativa agli acquisti* – è necessario fornire tutte le informazioni che possano facilitare i processi relativi al ciclo passivo (pagamenti, dati di fornitura, valutazione fornitore)
- *Invio della documentazione relativa alla fatturazione* - è necessario fornire tutte le informazioni che possano facilitare i processi relativi al ciclo attivo (incassi per la fornitura del prodotto, per i servizi a corredo, per le spese sostenute)
- *Invio della documentazione relativa a costi e ricavi* - è necessario fornire tutte le informazioni che possano facilitare i processi relativi al controllo di gestione
- *Invio alla funzione personale dei dati riguardanti i progettisti* - è necessario fornire tutte le informazioni che possano facilitare i processi relativi alla gestione delle Risorse Umane (competenze varie acquisite, note di merito, formazione effettuata etc..)

Riepilogo

A termine delle attività il PM produce un riepilogo: formula i consuntivi, li confronta con i preventivi, elabora gli scostamenti, offre la propria interpretazione dei risultati finali e chiede l'approvazione della Direzione. Lo scopo del riepilogo è lasciare traccia dell'esperienza compiuta qualunque sia l'esito del progetto terminato.

È importante focalizzare l'attenzione su cosa tracciare:

- **Eventi** - cosa è accaduto,
- **Scelte** - cosa si è deciso di fare,
- **Azioni** - cosa è stato fatto,
- **Motivazioni** - perché è stato fatto.

Si ritiene opportuno che il responsabile della chiusura interpreti la storia del progetto e lo faccia mediante un documento contenente le seguenti informazioni:

- Descrizione dei principali risultati
- Organizzazione del progetto
- Tecniche utilizzate per ottenere i risultati
- Punti di forza e di debolezza
- Esperienze maturate
- Prestazioni del prodotto
- Prestazioni del processo (costi e tempi).

L'eventuale assenza e/o lacunosità di questa pratica all'interno delle organizzazioni che sviluppano software sono ascrivibili alla scarsa lungimiranza della direzione: il riepilogo è considerato troppo costoso rispetto ai benefici che potrebbero scaturirne.

In realtà ciò è vero nel breve periodo (il riepilogo serve poco al progetto in chiusura) ma non nel lungo (il riepilogo è molto utile ai prossimi impegni dell'organizzazione):

un documento del genere ha un valore inestimabile per il *knowledge management* dell'azienda perché consente di imparare dall'esperienza migliorando le prestazioni sui progetti futuri.

Di seguito l'indice di un documento di chiusura di progetto.

<p>Generalità sul progetto Descrizione dei risultati principali Caratteristiche reali del prodotto Caratteristiche reali dell'utente Generalità sull'organizzazione utente – analisi del rapporto costo/benefici Ambito – collocazione del prodotto nell'ambiente dell'utente assistenza in corso – generalità sul processo di garanzia</p> <p>Organizzazione Struttura organizzativa del progetto Scelte iniziali Cambiamenti in corso d'opera Relazioni con le funzioni aziendali Vendite Acquisti Qualità Controllo di gestione Relazioni con il cliente – enti ed ambienti interessati al prodotto Relazioni con i fornitori – valutazione degli acquisti</p> <p>Tecnologia Vincoli e scelte tecnologiche Modello di ciclo di vita Tecniche e strumenti utilizzati Risultati delle scelte tecnologiche Impatto su prodotto – tecnologie incorporate nel prodotto Impatto su processo Infrastrutture tecnologiche di sviluppo ed esercizio Competenze dei Progettisti, dei Clienti e dei Fornitori</p> <p>Punti forza e punti di debolezza Forza – nuove opportunità di vendita generate dalle soluzioni di progetto Debolezza – minacce alla posizione occupata dall'Organizzazione presso il Cliente generate dai problemi di progetto</p> <p>Esperienze maturate Esperienze principali – Criticità e Rischi In positivo In negativo Lezione appresa – cosa abbiamo imparato dalle esperienze fatte</p> <p>Prestazioni del prodotto Analisi statistica dei dati di collaudo e garanzia Rilevazione della soddisfazione del cliente Validazione di progettazione – verifica dell'adeguatezza del prodotto all'uso</p> <p>Prestazioni del processo Tempi Costi Analisi dei dati e valutazione della capacità di processo Analisi delle prestazioni individuali e relative valutazioni di impatto sul processo Rilevazione della soddisfazione del Personale</p> <p>Conclusioni del PM Analisi degli obiettivi di progetto Valutazione della copertura di ciascuno degli obiettivi assegnati</p> <p>Allegati</p>

Figura 72 – Indice tipico di un documento di chiusura di progetto

Caso di studio

Rossi come ultima attività, rilascia un documento che contiene informazioni relative alla storia del progetto. Di seguito la storia del progetto software per il caso di studio in esame.

Storia del progetto software (Chiusura progetto)

Codice progetto: SW.BIL.100
Nome progetto Fatturazione dei servizi e prodotti telefonici
Descrizione: Sviluppare un sistema informativo su piattaforma tecnologica evoluta a basso costo per il processo di fatturazione integrata
Iterazione Disegno dei rendiconti telefonici

Ambito

	Previsti	Attuali
Requisiti:	200	175
Cambi:	20	25

Tutti gli obiettivi prefissati sono stati raggiunti.

L'istituzione di un sistema informativo ad hoc per la gestione delle varianti ha consentito di ottimizzare i tempi relativi al processo di software change management, in tal modo è stato possibile controllare efficacemente la stabilità dei requisiti.

Dimensione del prodotto software

	Previsti	Attuali
Function Point:	650	700
LOC:	60000	65000

Le stime iniziali ottenute mediante l'utilizzo della tecnica basata su stima statistica dei CSI ha fornito un valore in LOC con un errore di -12%, valore che in fase iniziale era da considerarsi accettabile, il conteggio dei FP invece è stato valutato in fase iniziale pari a 100 LOC, mentre gli attuali sono stati contati in maniera puntuale.

Schedulazione

	Previsti	Attuali
Durata solare	98 g/p	98/gp
Effort	7040 ore	7100
Overtime	0	70 ore

I tempi sono stati rispettati con un effort aggiuntivo di 70 ore sul pianificato, ciò è stato determinato dal tempo necessario per l'acquisizione delle competenze relative alla tecnologia, ai linguaggi e ai processi di supporto quali: Project Management, Software Change Management e gestione delle anomalie.

Risorse

	Previsti	Attuali
Analisti	6	6
Progettisti	6	6
Programmatori	16	16
Altro	3	3

Il coinvolgimento delle risorse sin dalle prime fasi del progetto nonché l'utilizzo di nuove tecnologie ha portato entusiasmo nel gruppo di lavoro, questo ha consentito di raggiungere tutti gli obiettivi, l'overtime non è stato un problema in quanto è stato utilizzato fondamentalmente per l'acquisizione di nuovi processi e nuovi tools.

Costi

BCWS: 7040
CV: -60
CPI: 0,99

La tecnica dell'earned value è stata utilizzata sia per la gestione dei costi che dei tempi.

Questo ha consentito di tenere efficacemente il progetto sotto controllo, per cui ogni scostamento negativo aveva come controparte una serie di azioni correttive.

I dati iniziali hanno evidenziato la necessità di ricorrere all'overtime per recuperare il tempo perso nell'apprendimento dei processi e dei tools.

Rischi

	Previsti	Attuali
High level	3	0
Function level	20	5

Non ci sono stati molti problemi relativamente ai rischi, in particolare l'utilizzo per la prima volta in azienda del processo di gestione del rischio e dei tools a supporto permetterà di introdurlo come pratica di base per progetti futuri.

Comunicazione

La difficoltà di gestire Team dispersi geograficamente è stata risolta mediante utilizzo di tools basati su tecnologia Internet. In questo modo si è ottenuto un duplice risultato:

- Riduzione dei costi derivanti da possibile trasferte
- Comunicazione efficace ed effettiva

Qualità

	Previsti	Attuali
Analisi	100	150
Progettazione	100	50
Codifica	100	85
Test	500	430

Il controllo della qualità mediante l'utilizzo di un archivio dei difetti individuati ha consentito di intervenire rapidamente apportando gli opportuni correttivi.

Lo sforzo compiuto in fase di progettazione di casi di test ha permesso di identificare difetti immediatamente sin dalle prime fasi dello sviluppo, riducendo in tal modo la difettosità del prodotto.

Criticità

L'introduzione di sistemi al supporto delle attività progettuali ha causato un effort aggiuntivo per l'apprendimento.

I team manager non avevano gli skill richiesti di Project Management

Lezioni apprese

- Un nuovo tools o un nuovo sistema di gestione per quanto facile da utilizzare richiede sempre del tempo.
- E' necessario, in caso di team dispersi geograficamente, una maggiore interazione.
- Le tecniche e i tools di Project Management devono far parte di un corso preliminare per i Team Manager

Raccomandazioni per ulteriori miglioramenti

- Includere nelle pianificazioni e quindi nei costi previsti l'effort relativo all'apprendimento dei tools a supporto delle attività di sviluppo software.
- Sviluppare una procedura di project management

Allegati

- Analisi della variazione della schedulazione, Budget, Staff
- Feedback dei partecipanti, Questionario di soddisfazione del cliente
- Questionario relativo al team di progetto
- Caratteristiche del prodotto software

Riepilogo

Dopo tanta fatica per giungere in porto scopriamo che c'è ancora da lavorare per disarmare la nave. Potrebbe sembrare eccessivo ma solo un approccio rigoroso in chiusura consente all'Organizzazione di ereditare il lascito più importante del progetto: il patrimonio delle nuove conoscenze acquisite in corso d'opera. Nell'epoca del WEB le uniche cose certe sono incertezza e turbolenza della tecnologia ed una risposta metodologica sensata è un'accorta gestione delle conoscenze aziendali. Sono disponibili paradigmi fin troppo sofisticati per la gestione strategica della qualità e della tecnologia. Siamo convinti che le aziende, anche quelle meno dotate, possano entrare in tali circuiti virtuosi attraverso una strada più semplice e meno costosa: una buona gestione del progetto inteso non solo come istituto finalizzato alla soddisfazione delle Parti interessate ma anche come canale per l'acquisizione delle nuove tecnologie e come laboratorio per sperimentazione-consolidamento-rinnovamento delle pratiche di lavoro. Dunque solo dopo una interpretazione puntuale e precisa dei risultati possiamo dire di avere concluso.

Finalmente.

Ora si può ricominciare con un nuovo progetto.

Appendici

- a. Studio di fattibilità
- b. Metodologie di ingegneria del software
- c. Utilizzo di Rational Rose
- d. Pattern
- e. Schede prodotto
- f. Elenco dei principali tools
- g. Metodi di stima e conteggio del software
- h. Il modello a spirale
- i. Riferimenti bibliografici
- j. Collegamento ai principali siti web
- k. Glossario

“La maggiore e più importante avvertenza che debba avere uno capitano, è di avere appreso di sé uomini fedeli, peritissimi della guerra e prudenti, con gli quali continuamente si consigli e con loro ragioni delle sue genti e di quelle del nimico.”
Dell’arte della guerra – Niccolò Machiavelli

Le appendici a,b,c,d,e,f sono state curate dal dr. Gianfranco Sansone.

Studio di fattibilità

Template di uno studio di Fattibilità per la Pubblica Amministrazione secondo le specifiche AIPA (Autorità Informatica per la Pubblica Amministrazione).

Per maggiori dettagli riferirsi al sito dell’Autorità della Pubblica Amministrazione AIPA “www.aipa.it”.

Il seguente template mostra come dovrebbe essere organizzato il documento (capitoli, paragrafi e sottoparagrafi) e relativi argomenti, descrivente lo studio di fattibilità sopra citato.

Capitolo 1:

1. La situazione attuale
- 1.1. Il contesto dello studio
 - Ripresa della visione strategica dell'amministrazione in termini di servizi, organizzazione, tecnologia
 - 1.1.2. Ripresa dei principali passaggi che hanno portato all'individuazione del progetto
 - 1.1.3. Collocazione del progetto all'interno del piano triennale
- 1.2. Descrizione della problematica
 - 1.2.1. Descrizione del problema/opportunità
 - 1.2.2. Rilevanza del problema/opportunità
 - 1.2.3. Esigenze da soddisfare (rispetto a utenti interni e esterni)
- 1.3. Descrizione della situazione attuale del sistema informativo
 - 1.3.1. Individuazione e rappresentazione dei processi coinvolti
 - 1.3.2. Individuazione e rappresentazione dei flussi informativi
 - 1.3.3. Individuazione e rappresentazione della struttura organizzativa e dell'utenza coinvolta
 - 1.3.4. Attuale livello di automazione
- 1.4. Analisi e diagnosi della situazione attuale
 - 1.4.1. Individuazione dei fenomeni che costituiscono le cause del problema
 - 1.4.2. Collocazione di tali fenomeni sulle diverse componenti del processo di servizio
 - 1.4.3. Individuazione di metriche atte a rappresentare i fenomeni critici e la loro evoluzione
 - 1.4.4. Misurazione della situazione attuale
- 1.5. Identificazione dei vincoli
 - 1.5.1. Quadro normativo di riferimento
 - 1.5.2. Vincoli temporali e altri vincoli (economici, organizzativi..)
- 1.6. Definizione degli obiettivi del progetto

Capitolo 2:

- 2. Progetto di massima della soluzione
- 2.1. Requisiti della soluzione
 - 2.1.1. Dettaglio del processo previsto dopo la reingegnerizzazione
 - 2.1.2. Interventi previsti sulle componenti non informative del processo (flusso, organizzazione, personale, logistica...)
 - 2.1.3. Necessità di modifica della normativa
 - 2.1.4. Requisiti del sistema informativo da realizzare
 - 2.1.4.1. Informazioni trattate
 - 2.1.4.2. Funzioni informatizzate
 - 2.1.4.3. Modalità di lavoro
 - 2.1.4.4. Requisiti architettonici
 - 2.1.4.5. Requisiti di qualità
- 2.2. Specifiche generali del sistema
 - 2.2.1. Specifiche applicative
 - 2.2.1.1. Architettura dati (con esame e valutazione delle eventuali alternative)
 - 2.2.1.2. Architettura applicativa (con esame e valutazione delle eventuali alternative)
 - 2.2.1.3. Interfaccia utente
 - 2.2.2. Specifiche tecnologiche
 - 2.2.2.1. Architettura tecnologica (con esame e valutazione delle eventuali alternative)
 - 2.2.2.2. Ambiente e strumenti di sviluppo (con esame e valutazione delle eventuali alternative)
- 2.3. Modalità di realizzazione
 - 2.3.1. "Make or buy" (con esame e valutazione delle eventuali alternative)
 - 2.3.2. Riutilizzo di componenti esistenti (con esame e valutazione delle eventuali alternative)
 - 2.3.3. Avvio del sistema
 - 2.3.4. Esercizio e manutenzione del sistema (con esame e valutazione delle eventuali alternative)
 - 2.3.5. Formazione ed assistenza utenti

Capitolo 3:

- 3. Analisi del rischio
- 3.1. Fattori di rischio del progetto
 - 3.1.1. Complessità
 - 3.1.1.1. Complessità gestionale
 - 3.1.1.2. Dimensioni del progetto
 - 3.1.1.3. Altri fattori
 - 3.1.2. Incertezza
 - 3.1.2.1. Incertezza dei requisiti
 - 3.1.2.2. Innovazione tecnologica
 - 3.1.3. Analisi del rischio di progetto
 - 3.1.4. Modalità di gestione del rischio

Capitolo 4:

- 4. Il progetto proposto
- 4.1. Segmentazione del progetto
- 4.2. Specifiche globali del sistema informativo da realizzare
- 4.3. Riepilogo delle acquisizioni e realizzazioni previste
- 4.4. Piano di massima del progetto
 - 4.4.1. Piano dei rilasci
 - 4.4.2. Punti di controllo
 - 4.4.3. WBS, Pert, Gantt

Capitolo 5:

- 5. Analisi costi-benefici
- 5.1. Valutazione dei benefici attesi
 - 5.1.1. Individuazione e descrizione dei benefici attesi
 - 5.1.2. Individuazione ed esplicitazione delle metriche e dei valori attesi
 - 5.1.3. Correlazione obiettivi-benefici
- 5.2. Stima dei costi
 - 5.2.1. Individuazione delle principali voci di costo
 - 5.2.2. Esplicitazione delle metriche utilizzate
 - 5.2.3. Stima dell'impegno di risorse umane
 - 5.2.4. Stima dei costi di impianto e di esercizio
- 5.3. Analisi dell'investimento

Capitolo 6:

- 6. Raccomandazioni per le fasi realizzative
- 6.1. Indicazioni per l'approvvigionamento
 - 6.1.1. Criteri per la determinazione della tipologia di fornitore
 - 6.1.2. Criteri di selezione delle offerte
 - 6.1.3. Indicazioni sulle modalità di approvvigionamento
- 6.2. Indicazioni per la gestione del progetto
 - 6.2.1. Indicazioni per la gestione del piano di qualità
 - 6.2.2. Indicazioni sul project management

- 6.2.3. Esigenze di negoziazione delle varianti
- 6.3. Riepilogo degli elementi utili alla stesura del capitolato

Introduzione

In generale tutte le metodologie di ingegneria del software sono costituite da:

- un **modeling language** cioè la notazione, spesso grafica, che il metodo usa per la progettazione, ad esempio lo "Unified Modeling Language" (UML);
- il **processo** descrivente gli step da compiere dalla prima analisi al compimento del progetto.

Di seguito un elenco, con un breve cenno descrittivo, di alcune delle principali metodologie oggi disponibili sul mercato.

Metodologia "EXTREME PROGRAMMING"

Extreme Programming, o anche detta XP, fa parte delle Metodologie Leggere (lightweight methodologies) contrapposte alle metodologie di ingegneria del software tradizionale (heavyweight methodologies) e parzialmente ai processi basati su UML. In generale tutte le metodologie leggere sono calibrate per funzionare su progetti di media e piccola dimensione dove contano la qualità finale del software e la sua facilità di manutenzione e modifica. XP viene consigliata solo per i piccoli team, questo perché essa non implica l'utilizzo/creazione di documentazione e analisi scritta. Teoricamente è possibile utilizzare XP anche in caso di team numerosi, in realtà, i grandi progetti generalmente devono rispettare obbligati livelli di burocrazia per la gestione di fondi, risorse umane etc. e ciò impedisce di fatto l'uso di metodologie leggere. XP è neutra rispetto ai linguaggi di diagrammi (ad esempio UML) così come è neutra rispetto ai linguaggi di programmazione. Generalmente UML è associato a pagine e pagine di documentazione e questo è l'opposto di XP e di tutte le metodologie leggere. In XP ad es., la documentazione del codice è il codice stesso, infatti si parla di "CodeThatSpeaks". Questo porta grossi vantaggi nei costi di manutenzione ma obbliga ad avere solo codice estremamente chiaro. Lo stato dell'arte indica diverse pratiche per arrivare ad ottenere codice "CodeThatSpeaks".

Una ulteriore caratteristica di XP è quella di usare un proprio vocabolario specializzato con molti acronimi. Lo scopo è di evitare confusione con altre metodologie.

Metologia "SCRUM"

Si veda il sito www.controlchaos.com per una trattazione completa.

Lo Scrum e' una metodologia di gestione delle attivita' finalizzate alla produzione.

Tipicamente è usato per la gestione di gruppi tecnici ma e' molto generico rispetto alle competenze ed ai ruoli.

Lo Scrum non e' uno strumento di organizzazione aziendale né di coordinamento di gruppi ma fa parte delle metodologie di produzione.

All'interno del processo produttivo si distinguono: il modello del processo ovvero "il come" ed il modello del prodotto ovvero "il cosa".

"Il come" indica le metodologie di gestione delle attivita' di produzione cioè come arrivare dai requisiti al sistema.

"Il cosa" indica le specifiche cioè la descrizione dei requisiti e del sistema.

Metologia di Booch, Jacobson, Rumbaugh

Si veda il sito della ditta Rational www.rational.com per una trattazione completa e le schede prodotto di APPENDICE E (prodotti Rational Rose, RUP, etc.).

La metodologia in oggetto prevede l'uso di "Rational Unified Process" (RUP) per la gestione del processo e l'uso di "Unified Modeling Language" (UML) come linguaggio di modellazione.

Altre metodologia

Merise:

Metodologia di sviluppo sw molto diffusa in Francia che è stata aggiornata recentemente per poter usare UML. Ha alcune similitudini con Rational Unified Process (RUP).

Open:

Un altro modello di processo di provenienza australiana, che ha alcuni aspetti in comune con Rational Unified Process (RUP).

Experience Based Approach:

Processo sviluppato da IBM. E' Iterativo, Incrementale, Object Oriented, guidato da scenari e dal rischio.

DSDM (Dynamic System Development Method):

Può essere visto come una introduzione a Rational Unified Process (RUP) sebbene utilizzi una terminologia differente.

Esempio di utilizzo di Rational Rose

Il tool Rational Rose della ditta Rational (vedi www.rational.com) è il principale strumento di generazione dell'UML oggi sul mercato e può utilizzare modelli di processo così come utilizzati da RUP, VisualBasic, Java etc.;

Rational Rose è costituito dalle seguenti **viste logiche**:

- Use Case View
- Logical View
- Component View
- Deployment View

Use Case View

La vista logica Use Case View contiene "Use Case Diagram". Un "Use Case Diagram" è una sequenza di transazioni correlate che vengono svolte dagli attori e dal sistema e serve a definire un pattern comportamentale del sistema.

In generale un "Use Case" è utile per:

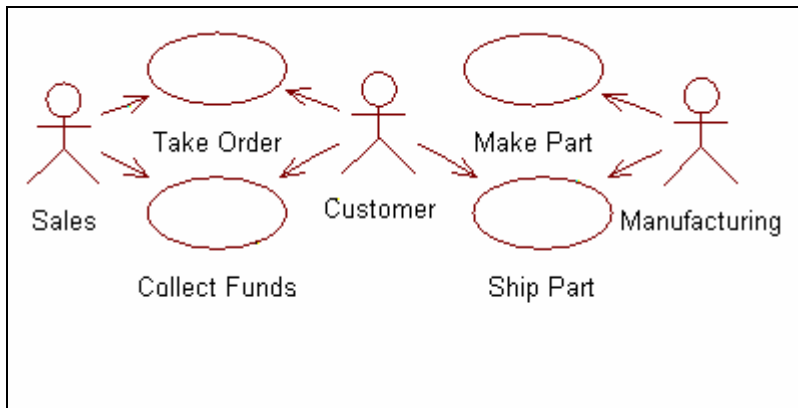
catturare i requisiti di sistema;
comunicare con gli utenti finali e gli esperti del dominio applicativo;
testare il sistema.

In sintesi un "Use Case" serve sia durante l'analisi per catturare i requisiti che durante la progettazione per specificare il comportamento del sistema.

Un "Use Case Diagram", ovvero la notazione grafica di uno Use Case, contiene:

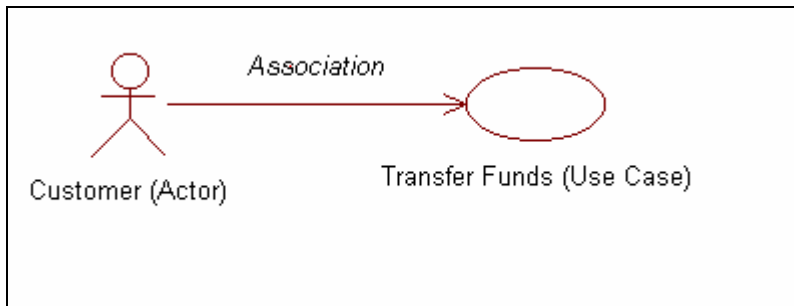
- Attori (notazione grafica: stick man)
- singoli Use Case (notazione grafica: ellisse)
- interazioni tra Attori e Use Case che possono essere **associazioni, dipendenze, generalizzazioni** (notazione grafica: frecce).

Il diagramma seguente rappresenta un es. di "Use Case Diagram" descrivente un certo "Scenario":



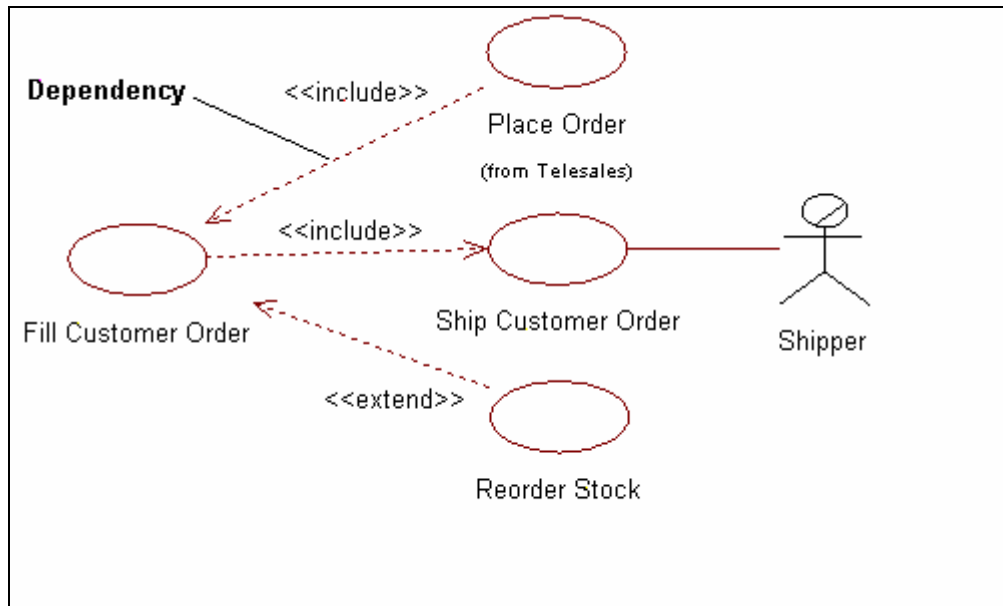
Sales è un Attore, "Take Order" è un Use Case.

Lo schema di seguito raffigurato rappresenta una interazione che di fatto è una **associazione uni-direzionale** (una associazione può anche essere bi-direzionale etc):

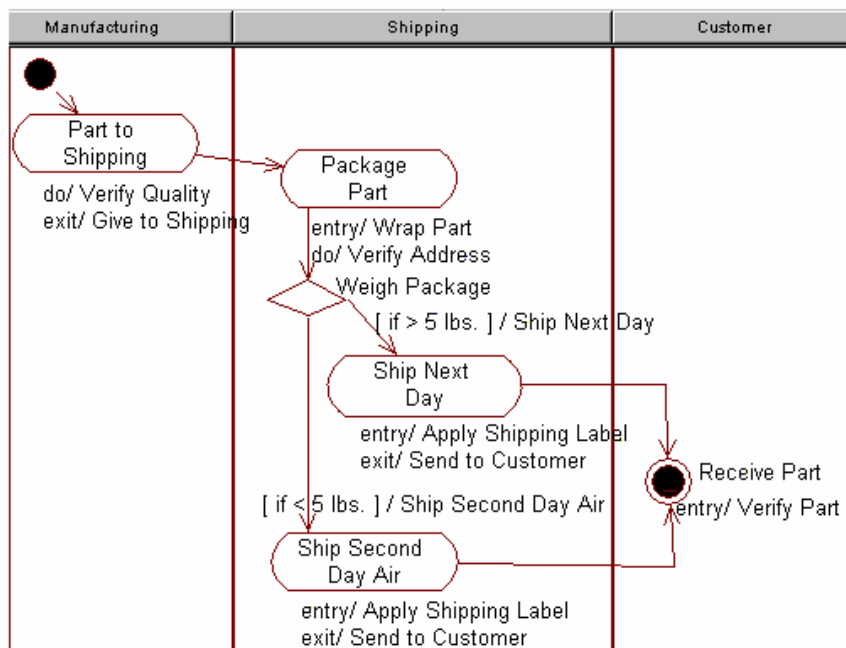


Lo schema di seguito raffigurato rappresenta delle **dipendenze**.

Le dipendenze possono essere di **<include>** (ad es. nel caso di comportamento simile tra due Use Case: "Place Order" e "Fill Customer Order" sono simili ed il secondo include il primo) o di **<extend>** (vedi "Fill Customer Order" e "Reorder Stock" in fig. sottostante):



Le **generalizzazioni** sono definibili quando un Use Case è simile ad un altro ma uno dei due è più generale:
 Da uno scenario di Use Case è possibile dedurre l'**Activity Diagram** ovvero il diagramma delle attività.
 Di seguito è illustrato un es. di Activity Diagram:



Come precedentemente scritto, oltre alla Use Case sopra in parte descritta, Rational Rose è costituito da Logical View, Component View e Deployment View di cui viene data di seguito una brevissima descrizione.

Logical View

Contiene "Analysis Model", "Business Object Model" e "Design Model"

Component View

Contiene "Implementation Model"

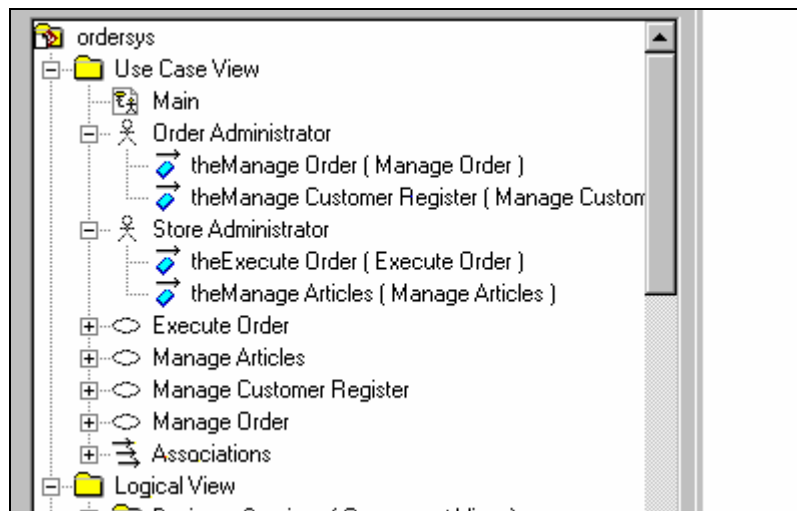
Deployment View

Contiene le informazioni relative alla installazione.

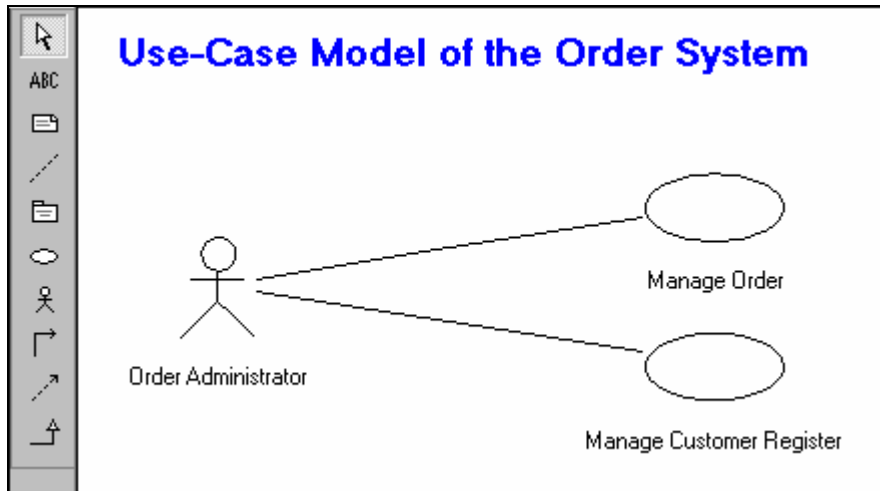
Esempio

es. di utilizzo di rational rose che mostra parte dell'interfaccia grafica del tool, relativamente ad un sistema di gestione ordini sviluppato in ambiente visual basic con architettura a tre livelli.

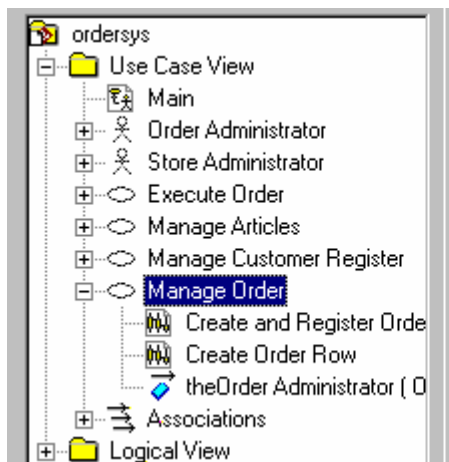
Di seguito è mostrato il pannello di Rational Rose contenente le voci/sottovoci relative alle funzionalità necessarie per definire tutto ciò che attiene alla creazione/gestione della vista logica "**Use Case View**".



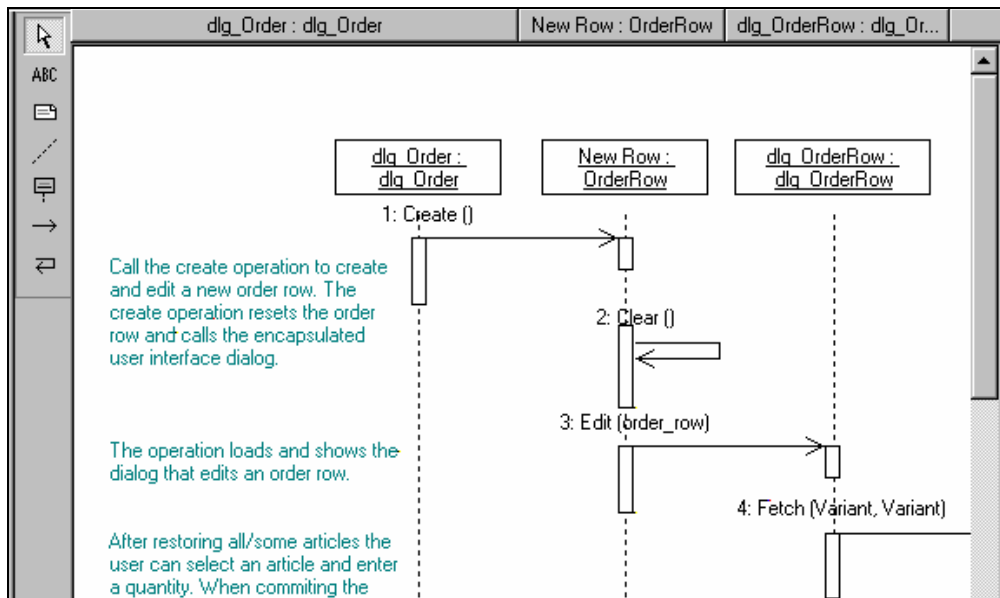
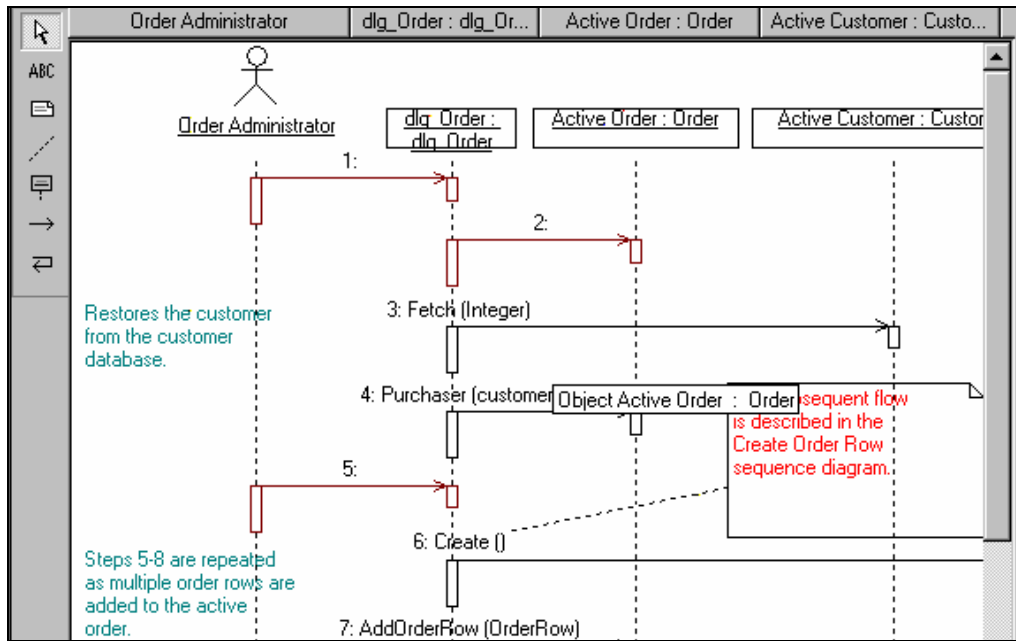
Dall'item Main del pannello sopra disegnato, si accede al modulo che permette di realizzare la rappresentazione grafica dello Use Case Diagram. La figura sotto rappresentata mostra un **attore** (Order Administrator) e due **Use Case** (Manage Order, Manage Customer Register).



La selezione dell'item "Manage Order" come evidenziato nella figura sottostante

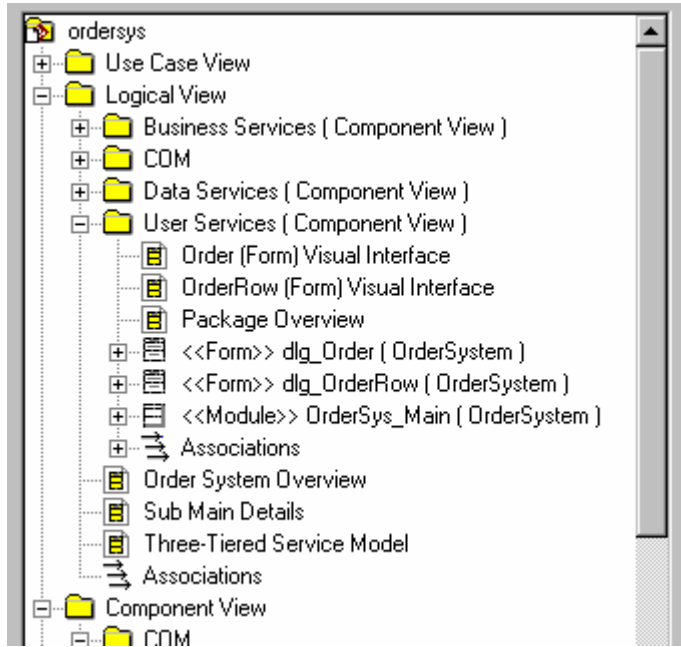


permette la visualizzazione dell'**Activity Diagram**.
Di seguito la rappresentazione di parte di esso.



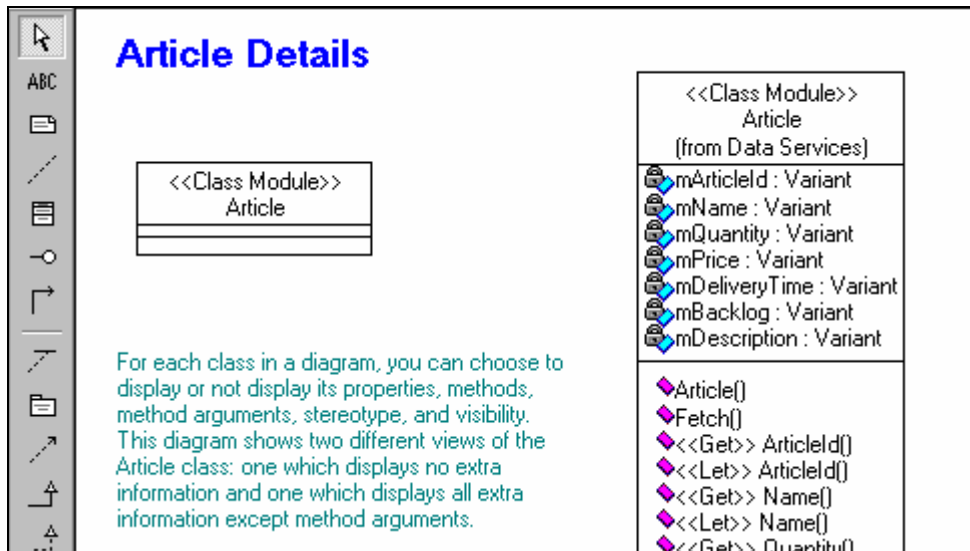
Come precedentemente scritto, oltre alla "Use Case View" esiste la vista logica "**Logical View**".

La figura sottostante mostra il pannello contenente le voci/sottovoci relative alle funzionalità necessarie per definire tutto ciò che attiene alla creazione/gestione della vista logica "Logical View".

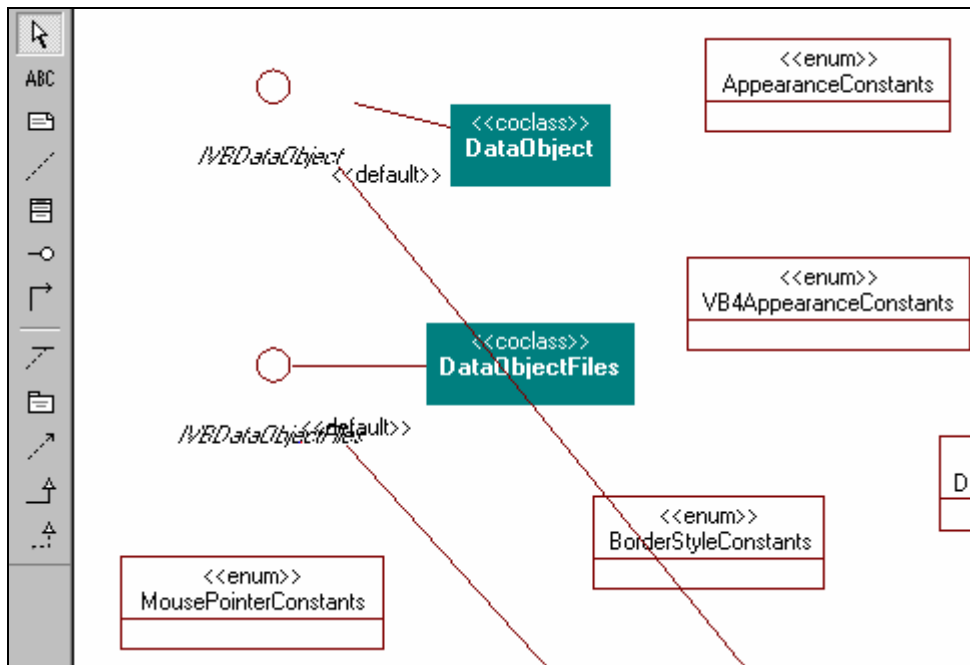


A partire dall'item Business Services del pannello sopra disegnato, si giunge al modulo che permette la visualizzazione della Logical View e cioè delle classi con relativi metodi e proprietà del diagramma.

La figura sotto rappresentata mostra i metodi ed i dati membro della classe "Article Details".



Di seguito è mostrata la Logical View di alcuni componenti distribuiti COM:



Di seguito è in parte mostrata la Logical View per la modellazione dei dati:

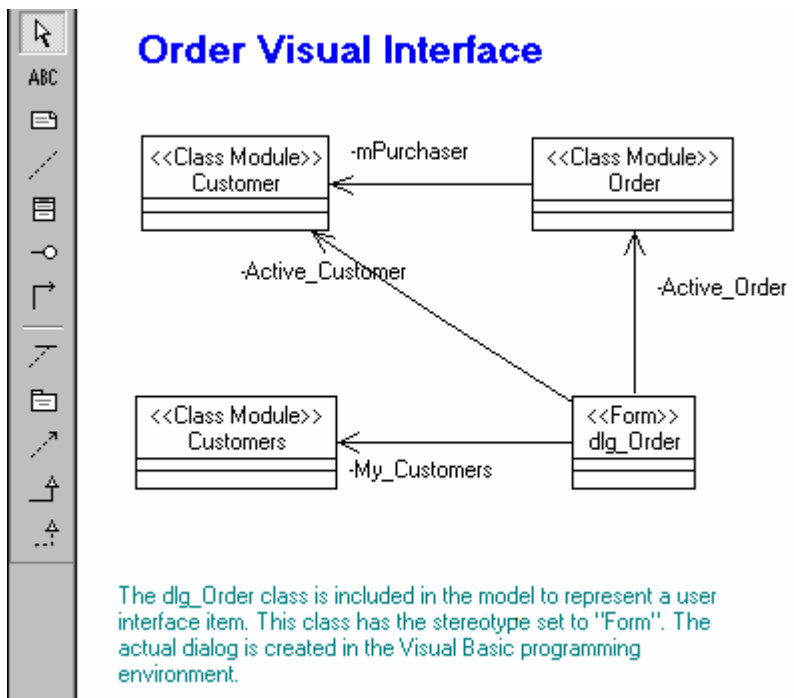
Db Details

<<Module>> Db
OrderSys_Db : Database
Db_Open() Db_Table_Article() Db_Table_Orderrow() Db_Table_Order() Db_Table_Customer()

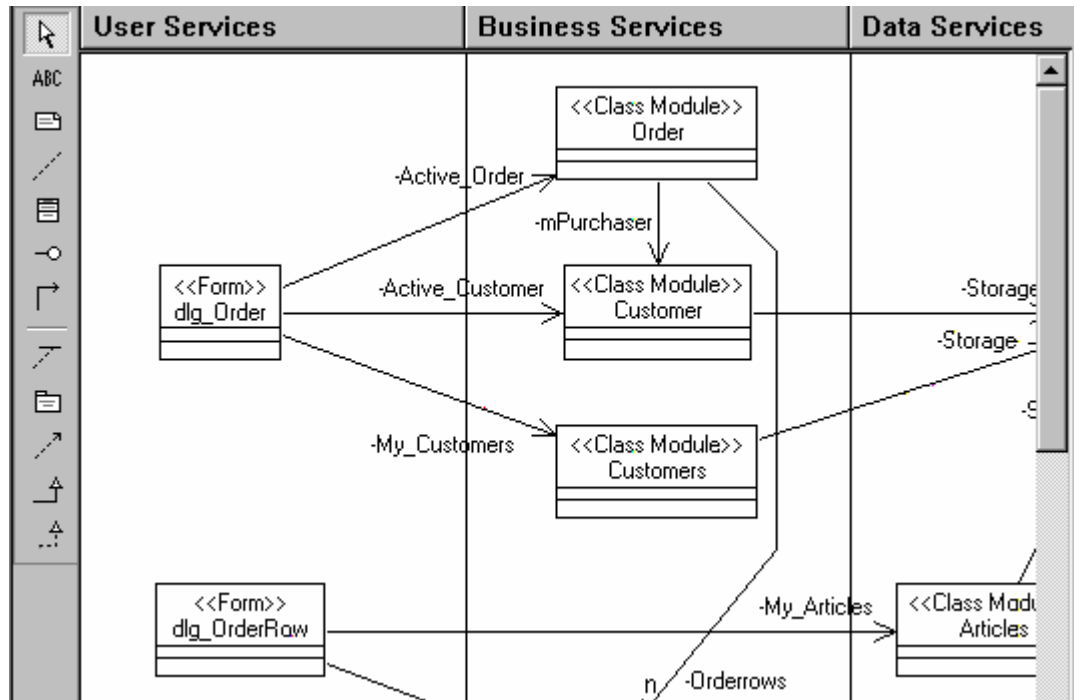
■ The Db module stores the database tables: tbl_customer, tbl_article, tbl_order, and tbl_order_row. Note that only the tbl_article and tbl_customer are implemented in the OrderSys_Db database.

The various persistent objects in the Order System use, conceptually, the OrderSys_Db database to preserve the object states. All business objects use the Persistence class as a bridge to the data services.

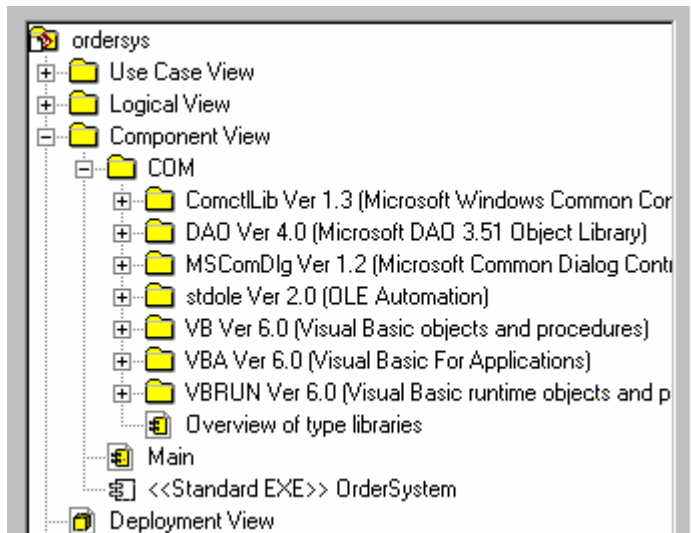
Di seguito è mostrata la Logical View per la modellazione dell'interfaccia visuale degli Ordini:



Di seguito è mostrata la Logical View per la overview del sistema a tre livelli (3 tiered Service Model):

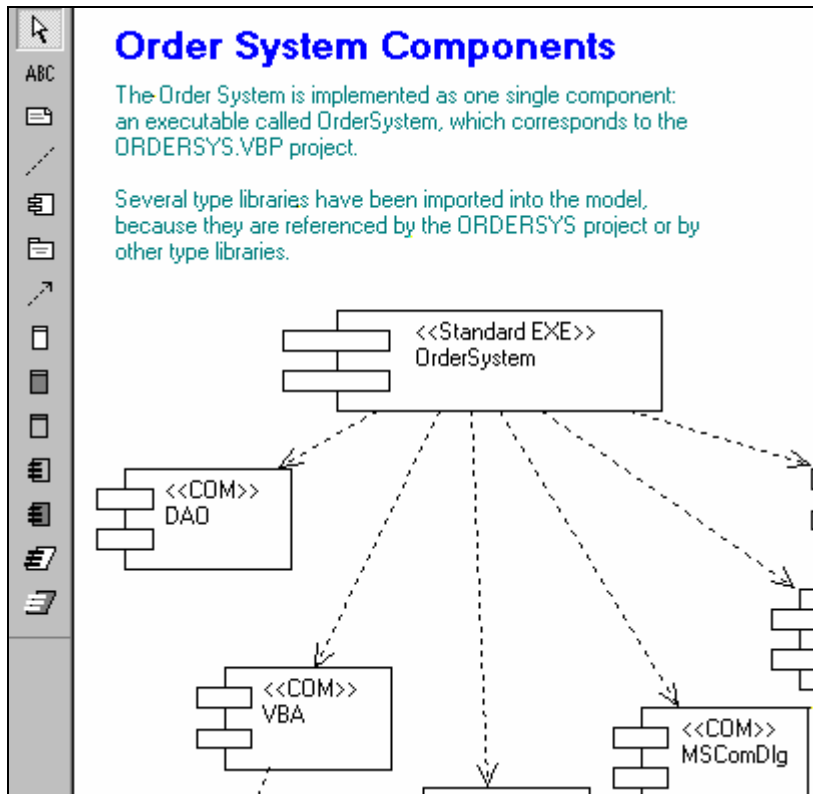


La terza vista logica di Rational Rose è la "**Component View**".
 La figura sottostante mostra il pannello contenente le voci/sottovoci relative alle funzionalità necessarie per definire tutto ciò che attiene alla creazione/gestione della vista logica "Component View".



Con Component View è possibile definire/visualizzare i componenti del sistema.

Di seguito è mostrata una parte dei componenti del sistema oggetto di questo esempio.



I PATTERN NELL'INGEGNERIA DEL SOFTWARE

Nel 1979 l'architetto software Christopher Alexander scrisse "The Timeless Way of Building" e introdusse il concetto di **pattern** che è stato successivamente utilizzato nel design di sistemi Object Oriented.

Infatti nell'OOPSLA 1987 (Object-Oriented Programming Systems, Languages and Applications; convegno di tecnologia OO "vedi sito <http://www.acm.org/sigplan/oopsla>") Cunningham e Beck ne fanno una presentazione alla comunità della Object Programming, e nel 1995, anno di pubblicazione di "Design Patterns: Elements of Reusable Object-Oriented Software", i pattern vengono formalmente consacrati a importante innovazione per la progettazione OO.

Alcune definizioni di pattern

Ogni autore e ogni ambiente di lavoro può avere una propria definizione, in particolare:

- An abstraction from a concrete form which keeps recurring in specific, non-arbitrary contexts. [la definizione più ampia possibile]
- A recurring solution to a common problem in a given context and system of forces. [definizione di Alexander]
- A named "nugget" of instructive insight, conveying the essence of a proven solution to a recurring problem in a given context amidst competing concerns.
- A successfully recurring "best practice" that has proven itself in the "trenches".
- A literary format for capturing the wisdom and experience of expert designers, and communicating it to novices

Scheda descrittiva di un pattern

- **Nome:** Il nome (cioè un "conceptual handle") è utile per memorizzare il pattern
- **Contesto:** Descrive in quale ambito e problema si può applicare il pattern ed inoltre quando e se la soluzione risulta valida
- **Problema:** Definisce il problema e gli intenti della soluzione
- **Forces:** Indica, nella eventualità che lo fosse, per quale motivo il problema è difficile da risolvere. In altri termini indica i cosiddetti "trade-offs, goals+constraints, motivating factors/concerns".
- **Soluzione:** Specifica come generare la soluzione (la struttura della soluzione, chi vi partecipa, le collaborazioni etc)
- **Contesto del risultato:** Descrive i risultati finali, i benefici, le conseguenze, come deve funzionare la soluzione
- **Razionale (opzionale):** Descrive i principi che giustificano la soluzione

- **Altri Pattern collegati a questo:** Elenca i pattern che risultano simili, che possono precedere o seguire il pattern trattato
- **Usi conosciuti del pattern:** Indica su quante problematiche è stato utilizzato il pattern

Elementi essenziali costituenti un pattern

Un pattern serve essenzialmente a risolvere una classe di problemi. Nella introduzione a "Design Patterns: Elements of Reusable Object-Oriented Software"(autori: Gamma, Helm, Johnson, Vlissides; editore: Addison Wesley, 1995) si legge che un pattern è composto essenzialmente da 4 elementi:

Nome:

Questa componente del pattern è significativa ed importante in quanto permette alla "comunità dei pattern" di comunicare rapidamente e in modo non ambiguo, riferendosi esclusivamente al Nome del pattern.

Problema:

Questa componente del pattern serve a far intendere se il problema reale da risolvere può essere risolto o meno dal pattern. In caso affermativo, il "Problema" ci fornisce un modello astratto che verrà usato nella sezione successiva per fornire la soluzione.

Soluzione:

Dato il modello della classe di problemi, in questa sezione se ne descrive una soluzione abbastanza astratta da "essere applicata molteplici volte", ma abbastanza precisa da poter essere immediatamente codificata da chiunque conosca un linguaggio OO (C++, Java, Smalltalk, ...).

In "Design Patterns: Elements of Reusable Object-Oriented Software" gli autori usano la notazione OMT.

Conseguenze:

Un pattern spesso può stimolare una riflessione su come migliorare il proprio design anziché fornire una soluzione definitiva. Questa sezione è importantissima per capire quali sono i vantaggi e i possibili svantaggi nell'uso del pattern; spesso, vi si trovano citati pattern che affrontano lo stesso problema privilegiandone aspetti diversi (per esempio la flessibilità contro l'efficienza).

Perché i pattern?

Gamma, Helm, Johnson, Vlissides ci vengono in aiuto nell'introduzione di "Design Patterns: Elements of Reusable Object-Oriented Software" spiegandoci il perché dei pattern. Da esperti progettisti OO, essi si sono accorti della difficoltà del riutilizzo del codice OO. Essi ritengono inoltre che il "gap" tra un progettista esperto ed uno alle nuove armi, è proprio la memoria di soluzioni che il progettista esperto, ha applicato con successo, magari dopo qualche tentativo fallito in progetti precedenti. Il designer esperto riconosce le similitudini salienti (e le differenze) rispetto al problema di cui ricorda la soluzione, ed è in grado di intuire che cosa può "riusare" (in senso ampio, non solo di riutilizzo di componenti software) di essa nel contesto attuale, e che cosa no.

I pattern sono certamente un modo "to record experience in designing object-oriented software".

Che cosa sono i pattern

I Software Patterns sono :

- Soluzioni ricorrenti a comuni problemi di progetto software
- Soluzioni pratiche e concrete per problemi del mondo reale
- Specifici ad un contesto
- La soluzione che meglio si adatta ai requisiti
- Provengono dalla esperienza degli esperti
- Un modo per documentare le "best practise"
- Un glossario condiviso per discutere di quel problema
- Un modo effettivo per il ri-uso della esperienza e della expertise di altri

Che cosa non sono i pattern

Software Patterns non sono :

- Ristretti al progetto software o OOD (Object-Oriented Design)
- Idee non testate o teorie
- Soluzioni che hanno funzionato una volta sola
- Principi astratti o euristici
- Qualcosa di applicabile ad un qualunque contesto

Perché i pattern sono utili

I Pattern possono aiutare a:

- Risolvere problemi del mondo reale
- Catturare l'esperienza relativa al dominio di conoscenza di quel particolare problema
- Documentare le decisioni progettuali
- Usare l'esperienza di chi ha già affrontato il problema
- Fornire consigli a chi affronta quel dato problema per la prima volta
- Formare un glossario specifico per discutere della risoluzione del problema
- Mostrare più della soluzione: contesto (quando e dove) "forces" ("trade-off alternatives, misfits, goals+constraints") risoluzione (come e perché questa soluzione bilancia le "forces")

I differenti tipi di pattern

- Design Patterns (per il software design; spesso object-oriented):
 - architecture (systems design)
 - design (component interactions)
 - programming idioms (language-specific techniques/style)
- Analysis Patterns (descrivono "recurring & reusable analysis models")
- Organization Patterns (descrivono "structure of organizations/projects")
- Process Patterns (descrivono "software process design")
- Domain-specific

Esempi di pattern

Si supponga di avere un sistema di data-entry e di voler realizzare una validazione di dati in input, quale pattern è possibile usare?

Per la risoluzione di questo problema si hanno a disposizione diversi patterns:

- il "Flyweight pattern"
- il "Mediator"
- etc

Bisogna ovviamente scegliere quello che si ritiene sia il più adatto alla risoluzione del problema.

Nella letteratura specializzata si ha la possibilità di vedere quali problemi sono stati risolti e con quali pattern.

Ad esempio:

Parser, Strategy, Lexical Analyzer, Composite, Interpreter, Builder, State sono i pattern utilizzabili per un "Interprete Software".

Iterator, Visitor, Façade,

sono i pattern utilizzabili per un "Software Tool" generico (ad esempio un generatore di codice, un gestore di stampanti, ecc.).

Processing, Command, Memento, Proxy sono i pattern utilizzabili per un software di "Transaction".

Abstract Factory, Singleton, Factory Method

sono i pattern utilizzabili per un "Interfaccia Utente".

Prototype, Adapter sono i pattern utilizzabili per un "driver di interfaccia verso un componente hardware".

Mediator, Observer sono i pattern utilizzabili per un "software di sincronizzazione tra macchine differenti".

Flyweight, è il pattern utilizzabile per un "software di compressione dati".

Decorator è il pattern utilizzabile per un "software di grafica".

Chain of Responsibility è il pattern utilizzabile per un "help context sensitive".

Riferimenti bibliografici

- Writings of architect Christopher Alexander (1977-1979)
- Documentation of best practices and handbooks for engineering and architecture
- Literate programming (Don Knuth), ca. 1984
- Kent Beck and Ward Cunningham, Textronix, OOPSLA'87
- Erich Gamma, Ph. D. thesis, 1988-1991
- James Coplien, Advanced C++ Idioms book, 1989-1991
- Gamma, Helm, Johnson, Vlissides (detta "Gang of Four") Object-Oriented Design Patterns book, 1991-1994

- PLoP Conferences and books, 1994-ad oggi

Libri sui pattern

- A Pattern Language: Towns, Buildings, Construction (APL) Christopher Alexander; Oxford University Press, 1977
- The Timeless Way of Building (TTWoB) Christopher Alexander; Oxford University Press, 1979
- Design Patterns: Elements of Reusable Object-Oriented Software (GoF) Gamma, Helm, Johnson, Vlissides; Addison-Wesley, 1994
- Pattern-Oriented Software Architecture: A System of Patterns (POSA) Buschmann, Meunier, Rohnert, Sommerlad, Stal; Wiley and Sons, 1996
- Pattern Languages of Program Design (PLoPD1) Coplien and Schmidt (editors); Addison-Wesley, 1995
- Patterns of Software: Tales from the Software Community Richard Gabriel; Oxford University Press, 1996
- Analysis Patterns: Reusable Object Models Martin Fowler; Addison-Wesley, 1996
- Pattern Languages of Program Design 2 (PLoPD2) Vlissides, Coplien, and Kerth (editors); Addison-Wesley, 1996
- Pattern Languages of Program Design 3 (PLoPD2) Martin, Riehle, and Buschmann (editors); Addison-Wesley, 1997
- Object Models: Strategies, Patterns, & Applications Peter Coad with David North & Mark Mayfield; Prentice Hall, 1995

In Internet

- Patterns Home Page, <http://www.hillside.net/patterns/>
- Patterns Discussion FAQ, <http://g.oswego.edu/dl/pd-FAQ/pd-FAQ.html>
- Ward Cunningham's WikiWikiWeb, <http://c2.com/cgi/wiki?WelcomeVisitors>
- Portland Pattern Repository, <http://www.c2.com/pp/>
- AGCS Patterns Page, <http://www.agcs.com/patterns/>
- Jim Coplien's OrganizationPatterns Front Page (a WikiWikiWeb work-alike), <http://www.www.bell-labs.com/cgi-user/OrgPatterns/OrgPatterns>
- Patterns Mailing Lists, <http://www.hillside.net/patterns/Lists.html>
- Cetus Links: Patterns, http://www.objenv.com/cetus/oo_patterns.html
- Brad's Pattern Links, <http://www.enteract.com/~bradapp/links/sw-pats.html>
- Brad's Patterns Intro, <http://www.enteract.com/~bradapp/docs/patterns-intro.html>
- Luke Hohmann's Patterns Intro, <http://members.aol.com/lhohmann/papers.htm>

- Doug Lea's OOD Patterns Intro,
<http://gee.cs.oswego.edu/dl/ca/ca/ca.html>

Scheda prodotto RATIONAL ROSE:

E' considerato il miglior tool di ingegneria del software oggi disponibile sul mercato. Unifica gli sforzi per lo sviluppo del software attraverso la modellazione UML ("Unified Modeling Language") ovvero la notazione standard per l' architettura del software.

Utile a sviluppatori, project manager, ingegneri ed analisti per implementare software migliore e più rapidamente.

Sistema Operativo: Windows, UNIX

Ambiente: Client / server; Stand Alone

Classe di costo: elevato

Link per il download della versione trial: www.rational.com

Scheda prodotto RATIONAL UNIFIED PROCESS (RUP):

E' un set di processi "web-enabled" di ingegneria del software che permettono di guidare il team attraverso le molteplici attività di sviluppo. RUP permette di scegliere facilmente il set di componenti per il proprio progetto. Il fine è quello di avere risultati migliori unificando il team di sviluppo; questo si ottiene migliorando la comunicazione e creando una base comune in tutti i task, in tutti i compiti ed in tutti i documenti. Si crea pertanto un unico sito web centralizzato per i programmatori software, i redattori di documenti, gli esperti di dominio e così via.

Sistema Operativo: Windows, UNIX

Ambiente: Intranet

Classe di costo: elevato (esiste una versione trial)

Link per il download della versione trial: www.rational.com

Altri prodotti RATIONAL:

Prodotti relativi a "Requirements & Analysis"

- Rational Rose
- Rational Suite AnalystStudio per il visual modeling, management dei requisiti, dei use case, dei difetti, del tracking delle richieste di modifica (change request),per le linee guida/ best-practice di processo
- Rational RequisitePro: Tool di management dei requisiti per promuovere una migliore comunicazione e ridurre i rischi di progetto

Prodotti relativi allo sviluppo software

Set di prodotti relativo a use-case e data modeling, architectural modeling, component modeling, code construction e analisi run-time.

- Rational Rose

- Rational Suite DevelopmentStudio: Soluzione full-lifecycle per i software architects, i progettisti, gli analisti e gli sviluppatori (disponibile in diverse edizioni: Windows, UNIX e RealTime)
- Rational PurifyPlus per la produzione di software affidabile
- Rational QualityArchitect: è una estensione di Rational Rose
- Rational Test RealTime: toolset per il software embedded e real-time
- Rational ClearCase per la gestione delle modifiche in corso d'opera (si parla di software configuration management ovvero SCM)
- Rational ClearQuest per il tracking dei difetti e dei cambiamenti
- Rational Apex per il progetto, lo sviluppo e la manutenzione del software (Ada, C, e C++)

Prodotti relativi al Test del sistema

- Rational Suite TestStudio per il load testing di sistemi client/server, Web, e ERP
- Rational Test RealTime soluzione per il test del software embedded, real-time and networked

Prodotti relativi al "Collaborative Project Management"

Sono prodotti finalizzati alla produzione di software di alta qualità ed entro il budget previsto ("on time" e "on budget").

- The Rational Unified Process o RUP
- Rational SoDA per la documentazione di progetto
- Rational ProjectConsole per il monitoraggio del progetto

Prodotti relativi al "Team Infrastructure"

- Rational RequisitePro
- Rational TestManager framework per l'unificazione di tutti i tool di test
- Rational ClearQuest
- Rational ClearCase
- Rational Suite ContentStudio per il management dei siti Web

Scheda prodotto MICROSOFT VISIO

Casa produttrice: Microsoft

A cosa serve: gestisce alcuni aspetti dell'UML

Sistema Operativo: Windows

Ambiente: Stand-Alone

Classe di costo: bassa

Link per il download della versione trial: www.microsoft.com

Scheda prodotto MICROSOFT PROJECT

Casa produttrice: Microsoft

A cosa serve: gestione del progetto, analisi dei tempi e della distribuzione dei task fra i componenti del team

Sistema Operativo: Windows

Ambiente: Stand-Alone

Classe di costo: bassa

Link per il download della versione trial: www.microsoft.com

Scheda prodotto COMPUTER ASSOCIATED ERWIN

Casa produttrice: Computer Associated

A cosa serve: modellazione ("data modeling") ovvero creazione e manutenzione di database e di datawarehouse; in particolare è composto dai seguenti moduli:

- Erwin for DB design
- Erwin Examiner per la validazione
- Bpwin per il progetto del Business Design
- ModelMart per il Model Management
- Paradigm Plus per il Component Modeling

Sistema Operativo: Windows, UNIX

Ambiente: Stand-Alone, Client/Server

Classe di costo: elevata

Link per il download della versione trial: www.ca.com

Elenco dei principali tool e delle principali metodologie di Project Management

- Action Plan
- Activity Network Diagram
- Advanced Analogies
- Affinity Diagram
- AHP – Analytic Hierarchy Process
- ANP – Analytic Network Process
- Arrow Diagram
- Assumptions Checklist
- Authority Template
- Balanced Scorecard
- Brainstorming
- Brainwriting
- Cause and Effect Diagram
- Compliance Audit
- Constructive Criticism Guidelines
- Control Chart
- Critical Path Analysis
- Cross Functional Management
- Different Reports
- DOE
- Electronic Expense Report
- Electronic Time Sheet
- Flow Chart
- Force Field Analysis
- Gantt Chart
- Histogram
- Hoshin Management
- Interrelationship
- Is/is-not matrix
- Matrix diagram
- Memory Jogger Plus
- Morphological Chart
- Nominal Group Technique
- Pareto Chart
- PDCA Cycle
- Prioritization Matrices
- Process Capability Chart
- Process Decision Program Chart (PDPC)
- Product Audit
- Project Calendars
- Project Management Software
- Project Planning Summary Template

- Quality Function Deployment (QFD)
- Quick Assessment Survey (QAS)
- Resource Field
- Resource Graph
- Resource Leveling
- Resource Usage
- Resources
- Risk Analysis Template
- Risk Management
- Run Chart
- Scatter Diagram
- Schedule Management
- Six Thinking Hats
- Spider Chart/Radar
- Stratification/Progressive Analysis
- SWOT Analysis
- System Audit
- Systems Thinking
- Task and Schedules
- Task Summary
- Task Usage
- The Memory Jogger Plus+Software
- The Project Implementation Profile (PIP)
- Theory of Constraints
- Tree Diagram/Structure Diagram
- TRIZ
- W3
- Work Breakdown Structure Chart (WBS)

Introduzione ai Function Point

I Function Point (FP) sono una tecnica per misurare la dimensione attraverso il concetto di funzionalità di un prodotto software.

La tecnica FP è indipendente dal linguaggio a differenza della tecnica basata su LOC.

In ogni caso esistono delle tabelle di conversione tra FP e LOC per linguaggio utilizzato.

È possibile trarre dal libro AIPA una di queste tabelle di conversione.

La tecnica FP sta avendo un buon successo, soprattutto fra gli utenti, che possono effettuare in autonomia il conteggio.

Essa presenta inoltre i seguenti vantaggi:

- Misurazione delle funzionalità dal punto di vista utente
- Indipendenza dalla tecnologia
- Disponibilità di una stima sin dalle prime fasi del ciclo di vita del software

Uno dei principali svantaggi risiede nel fatto che è adatta ad applicazioni gestionali, mentre per applicazioni che non fanno molto uso di funzioni di accesso ai dati si rileva inefficace.

La tecnica (conta o) stima il software sulle seguenti tipologie di funzionalità:

codice	Definizione	Descrizione
ILF	Internal Logical Files	raggruppamento logico di informazioni (tabelle DBMS, Flat file, etc..) memorizzate internamente al sistema e modificate da uno o più processi
EIF	External Interface Files	raggruppamento logico di informazioni memorizzate all'esterno del sistema e letti dal sistema
EI	External Input	funzioni che elaborano dati provenienti dall'esterno del sistema
EO	External Output	funzioni che inviano dati all'esterno del sistema
EQ	External Inquiries	funzioni che reperiscono dati esistenti all'interno del sistema mediante combinazioni di input e output

Figura 73

In fig. sono illustrati le principali tipologie di funzionalità e le relazioni tra di essi.

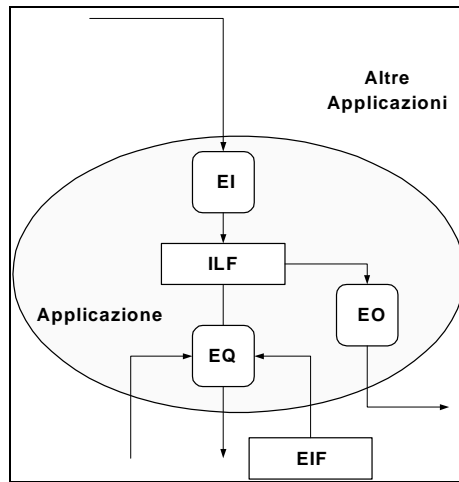


Figura 74

La procedura per (il conteggio o) la stima del software avviene nel seguente modo:

- Individuazione del tipo (Sviluppo, Manutenzione, Applicazione)
- Identificazione del confine (separazione tra l'applicazione che si vuole misurare e altre applicazioni)
- Calcolo dei Function Point di tipo dati (ILF, EIF)
- Calcolo dei Function Point transazionali (EI,EO,EQ)
- Calcolo dei Function Point non pesati
- Determinazione del valore del fattore di aggiustamento
- Calcolo finale dei Function Point pesati

Calcolo degli ILF e EIF

Per ogni File sia di tipo ILF che di tipo EIF si contano i DET (Data Element Type , ovvero i campi riconoscibili dall'utente e non ricorsivi e i RET (Record Element Type) cioè un sottogruppo di elementi dati in un ILF o EIF.

Il calcolo finale in Function Point è dato considerando la seguente tabella:

	ILF		
	1-19 DET	20-50 DET	>50DET
1 RET	7	7	10
2-5 RET	7	10	15
>5 RET	10	15	15

Figura 75 - Calcolo degli ILF

	EIF		
	1-19 DET	20-50 DET	>50DET
1 RET	5	5	7
2-5 RET	5	7	15
>5 RET	7	10	10

Figura 76 - Calcolo degli EIF

Calcolo dei Function Point Transazionali (EI,EO,EQ)

Per ogni EI,EO,EQ individuato si conta il numero dei file referenziati (FTR) e degli elementi di tipo dati (DET) trattati.

Il calcolo finale in Function Point per ogni tipo di funzionalità è dato considerando le seguenti tabelle:

	EI		
	1-4 DET	5-15 DET	>15 DET
0-1 FTR	3	3	4
2 FTR	3	4	5
>2 FTR	4	5	5

Figura 77 - Calcolo degli EI

	EO		
	1-5 DET	6-19 DET	>19 DET
0-1 FTR	4	3	5
2-3 FTR	4	5	7
>3 FTR	5	7	7

Figura 78 - Calcolo degli EO

	EQ Lato input		
	1-4 DET	5-15 DET	>15 DET
0-1 FTR	3	3	4
2 FTR	3	4	6
>2 FTR	4	6	6

Figura 79 - Calcolo degli EQ lato input

	EQ Lato output		
	1-5 DET	6-19 DET	>19 DET
0-1 FTR	3	3	4
2-3 FTR	3	4	6
>3 FTR	4	6	6

Figura 80 - Calcolo degli EQ lato output

Calcolo dei Function Point non pesati

Il calcolo dei FP non pesati si effettua sommando i FP determinati per ogni tipo.

Calcolo del valore del fattore di aggiustamento

Il valore del fattore di aggiustamento rappresenta le funzionalità generali dell'applicazione ed è utilizzato per aggiustare i Function Point.

Esso si basa su 14 caratteristiche ognuna delle quali ha un grado di influenza che va da 0 a 5.

Le 14 caratteristiche sono date nella tabella seguente:

Caratteristica	Grado di influenza	Caratteristica	Grado di influenza
Comunicazione dati		Aggiornamento interattivo	
Distribuzione dell'elaborazione		Complessità elaborativi	
Prestazioni		Ricusabilità	
Utilizzo intensivo della configurazione		Facilità di installazione	
Frequenza delle transazioni		Facilità di gestione operativa	
Inserimento dati interattivo		Molteplicità dei siti	
Efficienza per l'utente finale		Facilità di modifica	

Figura 81 - Caratteristiche

Il calcolo del valore del fattore di aggiustamento è dato da:

$$VAF = (TDI \times 0,01) + 0,65$$

dove TDI è la somma dei gradi di influenza.

Calcolo finale dei Function Point

Il calcolo finale per un progetto di sviluppo è dato dalla seguente formula:

$$DFP = (UFP + CFP) \times VAF$$

dove UFP sono i Function Point non pesati -aggiustati,
CFP sono i Function Point di conversione e
VAF è il valore del fattore di aggiustamento.

Il calcolo finale per un progetto di manutenzione evolutiva è dato dalla seguente formula:

$$EFP = [(ADD + CHGA + CFP) \times VAFA] + (DEL \times VAFB)$$

dove ADD sono i Function Point aggiunti,
CHGA sono i Function Point modificati,
DEL sono i Function Point cancellati,
VAFA e VAFB sono rispettivamente i valori del fattore di aggiustamento
prima e dopo l'intervento di manutenzione evolutiva.

Il calcolo finale per il conteggio iniziale di un'applicazione è dato da:

$$AFP = ADD \times VAF$$

COCOMO Cenni.

Il COCOMO (CONstructive COSt MOdel) è un modello per la stima dei costi dei progetti software.

La stima dei costi è derivata in funzione della dimensione del software e di fattori di aggiustamento che ne influenzano il calcolo.

Esistono due versioni del modello: la versione originale e il COCOMO II.

La più utilizzata quella originale è suddivisa in COCOMO base, intermedio e dettagliato ed è applicata a progetti che utilizzano cicli di sviluppo a cascata.

La versione II invece oltre ad utilizzare un mix di tecniche per il dimensionamento del prodotto software quali object point, function point utilizza modelli di riuso e re-engineering e aggiunge ulteriori fattori di aggiustamento.

Noi parleremo del cocomo II.

Le capacità di stima del cocomo II sono basate su due modelli:

- Early Design
- Post Architecture.

L'Early Design si applica alle fasi iniziali del progetto software quando poco è noto del prodotto.

Il Post Architecture viene applicato quando il disegno architeturale è noto.

Le formule per determinare l'effort nei due modelli sono rispettivamente:

$$1. PM_{\text{Aggiustato}} = PM_{\text{Nominale}} * \prod_{i=1}^7 EM_i$$

$$2. PM_{\text{Aggiustato}} = PM_{\text{Nominale}} * \prod_{i=1}^{17} EM_i$$

Dove

$$PM_{\text{Nominale}} = A * Size^B$$

con:

- $B = 0.91 + 0.01 * \sum_i w_i$,
- A è una costante
- Size è espresso in migliaia di linee di codice.

Di seguito in tabella vengono dati i fattori di aggiustamento per il modello Post Architecture e Early:

	Categoria dei fattori	Fattore	Descrizione
1.	Prodotto	RELY	Required Software Reliability
2.		DATA	Data base size
3.		CPLX	Product complexity
4.		RUSE	Required Reusability

Appendice G- Metodi di stima e conteggio del software

5.		DOCU	Documentation Match to Lyfe-cycle Needs
6.	Piattaforma	TIME	Execution Time Constrain
7.		STOR	Main Storage Constrain
8.		PVOL	Platform Volatility
9.	Personale	ACAP	Analyst Capability
10.		PCAP	Programmer Capability
11.		AEXP	Application Experience
12.		PEXP	Platform Experience
13.		LTEX	Language and tools experience
14.		PCON	Personnel Continuity
15.	Progetto	TOOL	Use of software tools
16.		SITE	Multisyte development
17.		SCED	Required Development Schedule

	Fattore	Controparte
	RCPX	RELY,DATA ;CPLX,DOCU
	RUSE	RUSE
	PDIF	TIME,STORE,PVOL
	PERS	ACAP,PCAP,PCON
	PREX	AEXP,PEXP,LTEX
	FCIL	TOOL,SITE
	SCED	SCED

Figura 82

Metodo del conteggio dei blocchi funzione

Le operazioni da effettuare quando si effettua la stima sono le seguenti:

- Determinare le funzioni che comporranno il sistema.
Per ogni funzione, definire sulla base di passate esperienze la dimensione (in LOC o in FP) del software i seguenti valori:
 1. numero minimo (NMIN) dei LOC o FP che costituiranno la funzione
 2. numero massimo (NMAX) dei LOC o FP che costituiranno la funzione
 3. numero desiderato dei (NDES) LOC o FP che costituiranno la funzione
- Calcolare il valore atteso (E) e la deviazione standard (DS) nel seguente modo per ogni funzione

$$E = (NMIN + 4 * NDES + NMAX) / 6$$

$$DS = (NMAX - NMIN) / 6$$

Appendici G- Metodi di stima e conteggio del software

- Calcolare il valore atteso totale e la deviazione standard per tutto il sistema da realizzare nel seguente modo:

$$E_{tot} = \sum_{i=1}^n E_i$$

$$D_{stot} = \left(\sum_{i=1}^n DS_i^2 \right)^{1/2}$$

Il modello a spirale

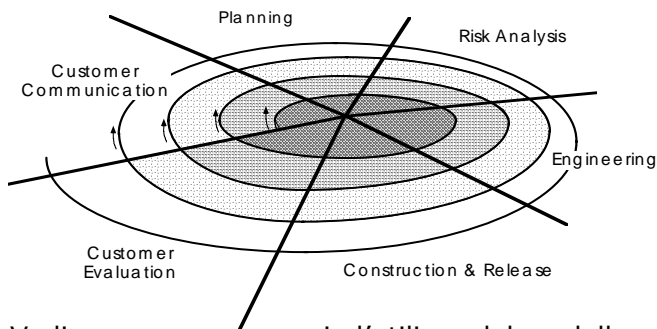
Il modello a spirale [Spiral Model – Bohem 1988] è un meta modello di ciclo di vita del software totalmente compatibile con il modello incrementale ed evolutivo, nonché con quello prototipale.

L'aspetto più importante del modello a spirale è l'esplicita analisi dei rischi e la fase di valutazione, inoltre il sottoprocesso (prototipale, a cascata, incrementale, etc..) utilizzato durante una iterazione dipende dal rischio.

L'obiettivo del modello a spirale è la riduzione dei rischi, ciò è fatto attraverso l'iterazione dei seguenti passi:

- Individuare gli obiettivi
- Determinare le alternative per gli obiettivi individuati in funzione dei rischi
- Eseguire l'approccio selezionato al passo 2
- Valutazione ciò che si è fatto

La dimensione a spirale del modello denota i costi cumulativi del progetto ad ogni iterazione.



Vediamo con un esempio l'utilizzo del modello.

Il progetto ha l'obiettivo di realizzare un software per il disegno dei rendiconti telefonici utilizzando un'interfaccia grafica basata su Java che viene eseguita su postazioni windows e collegamenti a base dati su mainframe.

Nel primo passo si individua che l'obiettivo principale è la fattibilità del progetto e il rischio associato riguarda le prestazioni, le connessioni alle basi dati su mainframe nonché le primitive grafiche per il disegno degli oggetti grafici.

Si decide di realizzare delle funzionalità che accedono alle basi dati che leggono i dati contenuti nelle tabelle e si rappresentano graficamente alcuni oggetti grafici (quali testo, rettangoli), per far ciò si utilizza il ciclo di vita a cascata.

Dopo aver realizzato ciò si è valutata la fattibilità e si passa alla seconda iterazione.

Alla seconda iterazione il rischio associato è l'interfaccia grafica per cui si decide di effettuare un prototipo nella speranza che il cliente capisca ciò che effettivamente desidera e dia un feedback.

Al termine della fase prototipale non tutti i requisiti sono chiari.

Si passa alla terza iterazione, è il rischio associato è l'incompletezza dei requisiti per cui si decide di utilizzare un modello di ciclo di vita incrementale, per cui si realizza l'incremento.

Alla quarta iterazione tutti i requisiti vengono forniti.

Bibliografia

- [Dyer 1984] Dyer, J. 'Team Research and Team Training: A State-of-the-Art Review' *Human Factor Review* (1984)
- [Humphrey 2000] Humphrey, Watts S. 'Introduction to the Team Software Process' *Addison-Wesley* (2000)
- [Humphrey 1997] Humphrey, Watts S. 'Introduction to the Personal Software Process' *Addison-Wesley* (1997)
- [PMBOK 1996] 'A Guide to the Project Management Body of Knowledge' *PMI* (1996)
- [ISO/IEC 12207] *ISO/IEC 12207:1995 Information Technology – Software Life Cycle Processes*
- [SWEBOK] 'Guide to the Software Engineering Body of Knowledge' *IEEE Computer Society* (trial version May 2001)
- [AICQCI 2001] "Quaderno n°11 – Linee guida per l'applicazione della norma ISO9001:2000 (Vision 2000) per lo sviluppo e la manutenzione del software" *AICQ-CI* (Aprile 2001)
- [DeMarco 1999] De Marco M., Salvo V., Lanzarani W., "Balanced Scorecard: dalla teoria alla pratica" *FrancoAngeli* (1999)
- [CORTI 2002] Corti, Eugenio "Gestione dell'innovazione – la piccola impresa innovativa" *ESI* (2002)
- [ISO 9001] "UNI EN ISO 9001 Sistemi di gestione per la qualità – requisiti" *UNI* (Dicembre 2000)
- [Roetzheim 2000] Roetzheim, William "Estimating Software Costs" - *Articolo In Software Development Magazine* (Ottobre 2000)
- [RADMCCO] Rapid Development: Taming Wild Software Schedules by Steve C McConnell
Managing Software Requirements: A Unified Approach (The Addison-Wesley)
- [MCCODE] Code Complete: A Practical Handbook of Software Construction by Steve C McConnell
- [SRWIEGER] Software Requirements
by Karl E. Wiegers
- [ROYCESPM] Software Project Management: A Unified Framework
by Walker Royce (Hardcover - September 1998)
- [PMMEREDITH] Project Management: A Managerial Approach, 4th Edition with Microsoft (r) Project 2000
by Jack R. Meredith, et al (Hardcover)
- [WYSOCKY] Effective Project Management, 2nd Edition
by Robert K. Wysocki, et al
- [AMATOCHIAPPI] Tecniche di Project Management - Pianificazione e controllo dei progetti, Rocco Amato - Roberto Chiappi
- [THAYERSPM] Software Engineering Project Management
by Richard H. Thayer (Editor), et al (Paperback - October 1997)
- [ARTROETZHEIM] [Articolo - Estimating Software Costs di William Roetzheim - In Software Development Magazine Ottobre 2000]
- [ZANICHELLI] Vocabolario della lingua italiana
- [BOEHM]
- [RAFFAZOLLO] Economia del software:Elementi introduttivi – Mario Raffa,

- [JACOBSON]** Giuseppe Zollo
Object Oriented Software Engineering - Jacobson - Addison Wesley.
- [ESAGUIDE]** Software Engineering Guide
Mazza, Fairclough, Melton, De Pablo, Schiffer, Stevens, Jones, Alvisi
Prentice Hall
- [YOURDON]** L'analisi strutturata dei sistemi - Ed. Yourdon, Jackson
[KERZNER] Project Management: A Systems Approach to Planning, Scheduling, and Controlling, 7th Edition-- by Harold Kerzner;
- [SEITAXONO]** Technical Report CMU/SEI-93-TR-6 ESC-TR-93-183
Taxonomy- Based Risk Identification Marvin J. Carr Suresh L. Konda Ira Monarch F. Carol Ulrich Clay F. Walker June 1993
- [FLEMING]** Earned Value
Q.W. Fleming, J.M. Koppelman
PMI
- [ZOLLO]** Zollo, G., Iandoli, L., Teta, V., Capaccio, M., (2001), "A fuzzy logic based method for the selection of the life-cycle model in software projects development", Proceedings VIII Congress of SIGEF "New logics for the new economy" 20-21 Sept 2001 Naples, Italy
- [ROYCE 1970]** Royce, W. W., "Managing the Development of Large Software System", *Proceedings of IEEE WESCON*, Agosto 1970.
- [CMMPAULK]** Paulk, M. *et al.* (1993), "Capability maturity model for software", *Software engineering institute*, Carnegie Mellon University, Pittsburgh.
- [CIRAIPA]** Circolare 28 dicembre 2001, n. AIPA/CR/38

Altri riferimenti bibliografici

Software engineering Project Management

Vari

IEEE Computer Society

Pianificazione, stima della dimensione del software, schedulazione e costi, Organizzazione, Staffing, Direzione, Controllo, metriche del software e visibilità dei progressi.

Software Project Survival Guide

Steve McConnell

Microsoft Press

Pianificazione, Disegno, Gestione, Assicurazione della qualità, testing..

Software Project Management - A Unified Framework

Walker Royce

Addison Wesley

Software Project Management e Processo di sviluppo del software unificato

Dynamics of software development

Jim McCarty
Microsoft Press

Organizzazione, La competizione, Il cliente, Il disegno, lo sviluppo, Tracking dei progressi, la distribuzione, lancio del prodotto.

Software Engineering Economics

B. Boehm
Prentice Hall

COCOMO, Analisi dei costi, Analisi decisionale, teoria statistica delle decisioni, metodi e procedure per la stima dei costi, stima dei costi e gestione dei cicli di vita del software.

Software Engineering Guide

Mazza, Fairclough, Melton, De Pablo, Schiffer, Stevens, Jones, Alvisi
Prentice Hall

Definizione dei requisiti utente, Analisi dei requisiti software, Disegno Architettonico, Disegno di dettaglio, Trasferimento del software, Esercizio e Manutenzione del software, Verifica e Validazione, Gestione della configurazione, Gestione della qualità, Software Project Management

Patterns of Software Systems Failure and Success

Capers Jones
Thompson Computer Press

Esplorazione dei successi e fallimenti nel software, Punti di forza e debolezza di sei classi di software, Minimare i rischi software e ottimizzazione dei successi, Ottimizzazione dei fattori di software Project Management, Ottimizzazione dei fattori relativi allo staff tecnico,

Software Engineering - A Practitioner's approach

Roger S. Pressman
Mc Graw Hill

Il prodotto e il processo, Gestione dei processi software, Metodi convenzionali per l'ingegneria del software, Ingegneria del software object oriented, argomenti avanzati di ingegneria del software.

The Mythical Man-Month

F. Brooks
Addison Wesley

Fatti e opinioni sui progetti software spiegati da F. Brooks Project Manager dell'OS390 dell'IBM

Managing the software process

Watts S. Humphrey
Addison Wesley

Struttura software matura, Processo ripetibile, Processo definito, Processo gestito, Processo ottimizzato.

Managing a programming project - Processi e persone.

Philip Metzger - John Boddie
Prentice Hall

Analisti e relative attività, Ruolo del manager, analista e<programmatori, il disegno del software, il processo di programmazione, e di test.

Debugging the development process

Steve McGuire

Microsoft Press

Strategie per rispettare tempi, costi, qualità dei prodotti e costruzione di team di successo.

Dynamics of software development

Jim McCarty

Microsoft Press

Organizzazione, La competizione, Il cliente, Il disegno, lo sviluppo, Tracking dei progressi, la distribuzione, lancio del prodotto.

Rapid development

Steve McConnell

Microsoft Press

Sviluppo efficiente, Sviluppo RAD, Best Practice

Project Management - La gestione di programmi e progetti complessi

Russell D. Archibald

Franco Angeli

La gestione di programmi e progetti nell'industria e negli enti pubblici, programmi e progetti, Organizzazione della funzione di project management, ruoli di integrazione di molteplici apporti nel project management, il team di progetto e i principali aspetti umani del project management, l'impegno per i singoli apporti nel team di progetto, strategia di superamento degli ostacoli al project management, l'ambiente multiprogetto, Organizzazione del project office e del team di progetto, Pianificazione del progetto, Pianificazione in team e avviamento del progetto, Controllo del lavoro, della schedulazione dei costi, il management delle interfacce di progetto, Sistemi informativi di project management, Valutazione e gestione del progetto, Chiusura o estensione del progetto

Multi Project Management MPM - Un'approccio innovativo per gestire la fabbrica dei nuovi prodotti

Paolo Manzoni in collaborazione con JMAC Consiel

Franco Angeli

Il vantaggio competitivo generato dalla gestione dei progetti, La fabbrica dei nuovi prodotti, Gli elementi del multiproject management, L'organizzazione, Il sistema di gestione, Gli standard, L'impostazione e la realizzazione dei sistemi di multi project management.

Tecniche di Project Management

Tecniche di Project Management - Pianificazione e Controllo dei Progetti

Franco Angeli

Il project management, La wbs di progetto, L'analisi dei tempi, il carico delle risorse, il database e l'approvvigionamento dei materiali, Il budget di progetto, L'avanzamento del progetto e il controllo dei costi, Il metodo lineare, cenni alle tecniche di controllo progetto nelle società di impiantistica, livelli di planning e misura dell'avanzamento nei progetti offshore, Una semplice procedura per l'analisi e la gestione del rischio.

Project & Program - Risk Management - A guide to managing project risks & opportunities

Max Wideman

PMI

Identificazione del rischio, finalità dell'accertamento del rischio e metodologia, Applicazioni i computer, Documentazione e Risposta al rischio, Gestione della contingenza, Gestione dei rischi e ambiente di progetto, Trattamento dei rischi nei contratti.

Quality Management

Lewis R. Ireland

PMI

Il movimento della qualità, Dimensione dei requisiti del cliente, Pianificare la qualità nei progetti, Costo della qualità, Concetti statistici e strumenti della qualità, Qualità e Persone nel Project Management, Raggiungere la qualità nei progetti

A guide to the Project Management Body of Knowledge

Project Management Institute Standards Committee

PMI

La struttura di Project Management, Le aree di conoscenza del Project Management

Project Management - Planning & Control Technique

Rory Burke

Wiley

Storia del Project Management, Ciclo di vita del progetto, Studio di fattibilità, Selezione del progetto, Stima del progetto, Pianificazione e ciclo di controllo, _Gestione dell'ambito, WBS, CPM, Barchart, Schedulazione degli approvvigionamenti, Pianificazione delle risorse, Costi di progetto, Controllo del progetto, Earned Value, Gestione della qualità, Gestione del rischio, Comunicazione di progetto, Struttura organizzativa, Team, Leadership.

Principles of Project Management

Vari

PMI

Organizzarsi per il Project Management, L'ambiente di lavoro del Project Manager, Ruoli e Responsabilità per il Project Manager, Team Building per i Project Manager, Gestione dei conflitti per i Project Manager, Negoziazione e Contrattazione, Amministrazione del Contratto per i Project Manager.

Effettive Project Management - How to plan, manage and deliver projects on time and within budget.

R.K. Wysocki, R.Beck Jr, D.B. Crane

Wiley

Competenze e conoscenze per il Project Management, Casi di studio, Che cos'è un progetto, Che cos'è il Project Management, Stili di gestione per i Project Managers, Ambito del progetto, Identificare le attività di progetto, Stimare la durata delle attività, Determinare i requisiti delle risorse, Construire/Analizzare il network di progetto, Preparare la proposta di progetto, Reclutare e organizzare il team di progetto, Livellare le risorse di progetto, Schedulare e documentare i work packages, Monitorare e Controllare i progressi, Chiusura del Progetto,.

Project Management a managerial approach

J.R. Meredith, J. Mantel

Wiley

Progetti nelle organizzazioni contemporanee, Selezione del progetto, Il Project Manager, Organizzazione del Progetto, Pianificazione del Progetto, Negoziazione e Risoluzione dei conflitti, Budgeting e assegnazione dei costi, Schedulazione, Assegnazione delle risorse, Monitoraggio e PMIS, Controllo di Progetto, Auditing del Progetto, Terminazione del Progetto, Ambienti Multiculturali.

Earned Value

Q.W. Fleming, J.M. Koppelman

PMI

Concetti, Genesi ed evoluzione dell'earned value, Il corpo della conoscenza dell'earned value, ambito del progetto, Pianificazione e Schedulazione del Progetto, Stima e budget delle risorse e centri di costo, Stabilire la baseline di progetto, Monitorare le performance rispetto alla baseline, Prevedere i costi finali e i risultati della schedulazione, Reingegnerizzazione del processo dell'earned value.

The fast forward MBA in Project Management

Eric Verzuh

Wiley

Il Project Management in un mondo in cambiamento, Gli attori del Progetto, L'ambiente di Progetto, Definire le regole, Gestione del rischio, WBS, Schedulazione realistica, Le dinamiche delle stime accurate, Bilanciare il progetto, Comunicazione, Misurare i progressi, Organizzarsi per il Project Management, Risoluzione di Problemi comuni.

The New Project Management

J. Davidson Frames

Jossey-Bass

Collegamenti ai principali siti web

PMI Project Management Institute

<http://www.pmi.org/>

Software Engineering Institute

<http://www.cmu.sei.edu/>

Osservatorio Project

<http://www.microsoft.com/italy/project/using/osservatorio/default.asp>

PM Forum

<http://www.pmforum.org/>

International Project Management Association

<http://www.ipma.ch/>

Microsoft Project User Group

<http://www.mpug.org/>

Software Project Managers Network

<http://www.spmn.com/>

R.S. Pressman & Associates Inc.

<http://www.rspa.com/>

AIPA

<http://www.aipa.it/>

AICQ Associazione Italiana Cultura della Qualità

<http://www.aicq.it/>

Data and Analysis center for software

<http://www.dacs.dtic.mil/>

IEEE Standards

<http://standards.ieee.org/>

Glossario

In questa sezione non abbiamo ritenuto opportuno sviluppare un glossario completo essendo disponibili eccellenti riferimenti quali:

[RAFFAZOLLO]	Economia del software:Elementi introduttivi – Mario Raffa, Giuseppe Zollo
[CORTI 2002]	Corti, Eugenio "Gestione dell'innovazione – la piccola impresa innovativa" <i>ESI</i> (2002)
[PMBOK 2000]	'A Guide to the Project Management Body of Knowledge' <i>PMI</i> (2000)

Di seguito riportiamo solo alcuni termini ricorrenti nel testo.

accordo di livello di servizio	Accordo scritto o contratto fra gli utilizzatori e il fornitore di servizi di Tecnologie dell'Informazione che documenta i livelli di servizio concordati per un servizio di Tecnologie dell'Informazione. Tipicamente esso copre: ore di servizio, disponibilità del servizio, livelli di supporto dell'utilizzatore, tempi di risposta del terminale e di passaggio, restrizioni, funzionalità e livelli di servizio da fornire in casi contingenti. Può anche comprendere procedure di sicurezza e di contabilità.
Accordo quadro	documento che disciplina gli eventuali rapporti Cliente – Fornitore sia nell'area Vendite, sia nell'area Acquisti
ACQ	ACQuisti
ADD	Architectural Design Document; documento di disegno architettuale; Specifiche di architettura; Specifiche della Progettazione di Sistema
affidabilità	Attitudine di un oggetto ad adempiere alla funzione richiesta nelle condizioni fissate e per un periodo di tempo stabilito.
assicurazione qualità; garanzia qualità	L'insieme delle azioni pianificate e sistematiche necessarie a dare adeguata confidenza che un prodotto o servizio soddisfi determinati requisiti di qualità.
Attore	ente aziendale incaricato della gestione di almeno un'attività in un processo aziendale
Attore primario	attore che interpreta un ruolo di responsabilità in un processo primario
Baseline	Insieme di configuration item (CI – elementi di configurazione) che sono di riferimento durante il ciclo di vita del progetto software
BUDGET	rappresenta in una opportuna unità di misura (es.: mesi/uomo oppure Lire) un preventivo di uno o più tipi di risorse da allocare sul progetto
capacità di processo	spettro dei risultati attesi di uno specifico processo aziendale
CapoCommessa	Specialista di TEC incaricato da D::T per: gestione del Cliente in assistenza, pianificazione e controllo commessa di produzione, pianificazione e controllo commessa di

	previdita, pianificazione e controllo intervento di azione correttiva
caratteristiche del processo SW	<p>classificabile – esistono input di commessa che consentono di inquadrare il processo sw in un modello di ciclo di vita praticabile in azienda e di classificare la commessa</p> <p>pianificabile – esistono input di commessa che consentono di pianificare il processo secondo il modello e la classe di commessa definiti</p> <p>attuabile – esistono input di commessa che consentono l’allocazione delle risorse sulle attività di processo in modo di ottenere prestazioni allineate con le aspettative del Cliente e con la capacità aziendale</p> <p>controllabile – per ogni commessa esiste ed è definito un piano di qualità ed un piano di misure attraverso i quali è possibile consolidare degli stati di avanzamento in corrispondenza di specifiche pietre miliari dell’iter di commessa</p>
caratteristiche del prodotto SW	definizioni della ISO9126: funzionalità, affidabilità, usabilità, efficienza, manutenibilità, portabilità.
CASE	Computer Aided Software Engineering
CI	<p>Configuration Item – Elemento di configurazione; definizioni: Entità in una configurazione che soddisfa una funzione d’uso finale e che può essere univocamente identificata [ISO/IEC 12207]</p> <p>sottoinsieme di una configurazione ovvero collezione organica di elementi sw trattati unitariamente per gli scopi della gestione della configurazione</p> <p>La definizione di CI è applicabile ricorsivamente alla struttura gerarchica di una configurazione: un CI può essere composto da più CI di livello inferiore</p>
CM	<p>Configuration Management – gestione della configurazione – sistema di rintracciabilità del prodotto sw; comprende: identificazione e definizione dei CI di una applicazione sw – struttura fisica del prodotto sw</p> <p>controllo delle release e delle modifiche della applicazione sw lungo l’intero ciclo di vita di commessa</p> <p>registrazione delle modifiche allo stato della configurazione</p> <p>rapporto sulla stato della configurazione</p> <p>verifica di completezza e di correttezza dei CI</p> <p>supporto ai processi di distribuzione e consegna</p>
concessione	Autorizzazione scritta a usare o consegnare una determinata quantità di materiali, componenti o scorte già prodotti, ma non conformi ai requisiti specificati.
conduzione aziendale per la qualità	La componente della conduzione aziendale rivolta alla definizione e all’attuazione della politica per la qualità.
Configurazione	<p>L’organizzazione di un sistema o di un componente, come definito per natura e per le interconnessioni delle sue parti costituenti [IEEE 610.12-90]</p> <p>Le caratteristiche fisiche e funzionali di un prodotto come espresse nella documentazione tecnica e ottenute nel</p>

	<p>prodotto [ISO/IEC 10007:1994] L'insieme organico di tutti gli elementi sw di una commessa per lo sviluppo di un'applicazione. Una configurazione definisce una specifica versione di un sistema applicativo o di un suo sottosistema in un preciso istante (data e ora di consolidamento di una baseline).</p>
contratto	<p>accordo commerciale tra Cliente e G. contenente requisiti tecnici e condizioni di fornitura e comunicato mediante un mezzo qualsiasi</p>
Contratto quadro	<p>accordo commerciale che riferisce forniture specifiche che il cliente ordinerà con successivi buoni d'ordine nel periodo di validità del contratto</p>
controllo della configurazione	<p>Il processo di valutare, approvare o disapprovare e coordinare modifiche agli elementi della configurazione di un sistema applicativo dopo la definizione formale della loro identificazione di configurazione (baseline di commessa).</p>
controllo qualità	<p>Le tecniche e le attività a carattere operativo messe in atto per soddisfare i requisiti di qualità.</p>
dato elementare di misura	<p>misurazione (quantitativa) con la quale si calcola il valore della metrica di riferimento</p>
DDD	<p>Detailed Design Document - documento di disegno di dettaglio - Specifiche di progettazione di dettaglio</p>
deroga	<p>Autorizzazione scritta, rilasciata prima della produzione (o della prestazione di un servizio), a scostarsi dai requisiti prestabiliti per una determinata qualità o per un determinato periodo di tempo.</p>
DFS	<p>Distribuzione Forza Specialistica – Registrazione di Qualità</p>
distribuzione	<p>processo di consegna dei workproduct di commessa (intermedi e finali) collegato a monte con la gestione della configurazione ed a valle con la installazione, avviamento e collaudo esterno dei risultati di commessa</p>
DM	<p>Document Management – gestione dei documenti e dei dati</p>
DOC	<p>Documenti</p>
DT	<p>classe di documenti tecnici di prodotto</p>
E.V.	<p>Ente Valutato in una Verifica Ispettiva Interna</p>
elemento di configurazione	<p>Una collezione di elementi hardware o software trattati come una unità al fine della gestione della configurazione.</p>
elemento software	<p>Parte identificabile di un prodotto software, ad uno stadio intermedio o allo stato finale di sviluppo.</p>
Escalation	<p>processo di escalation: aumento della capacità di processo attraverso una sovrallocazione delle risorse di commessa ed una eventuale redistribuzione di compiti e responsabilità di Cliente, Fornitore e Subfornitore</p>
Evidenza oggettiva	<p>Informazione, documentazione qualitativa/quantitativa o constatazione attinente alla qualità di un prodotto/servizio, ovvero all'esistenza e all'applicazione di un elemento del sistema qualità. Tale informazione, documentazione o constatazione può essere verificata ed è basata su osservazioni, misure o prove.</p>
fase	<p>segmento definito del ciclo di vita di commessa/progetto;</p>

	<p>sono fasi di commessa: Definizione Pianificazione Esecuzione e Controllo Chiusura Alternativamente per fase si può intendere un segmento definito del ciclo di vita di prodotto: Requisiti Analisi Disegno Implementazione Test</p>
fase	Segmento definito di lavoro.
fidezza	Termine collettivo utilizzato per descrivere le prestazioni di disponibilità e i fattori che le influenzano: prestazioni di affidabilità, prestazioni di manutenibilità e prestazioni del supporto di manutenzione.
FORM	Formazione del Personale
GANTT	diagramma di Gantt – tecnica per la pianificazione del processo – diagramma a barre in cui sono indicate le attività e le relative durate
gestione della configurazione	Il processo di identificare e definire gli elementi della configurazione in un sistema, controllare il rilascio e la modifica di questi elementi lungo l'intero ciclo di vita del sistema, registrare e relazionare lo stato degli elementi della configurazione e le richieste di modifica e di verificare la completezza e la correttezza degli elementi della configurazione (vedi controllo di configurazione).
GPR	reparto di gestione del personale, dipende da D::P
guasto	La fine della capacità di un elemento hardware di eseguire la funzione richiesta.
HW	HardWare
implementazione	<p>fase di lavoro comprendente attività che trasformano le caratteristiche teoriche di una applicazione sw solo progettata attraverso uno o più documenti nelle caratteristiche reali (comportamento a run time) del codice eseguibile. Sinonimo di realizzazione e di realizzazione-integrazione</p>
IN	Input – elemento di un processo
INTER	INTERvento di azione correttiva o preventiva
Istruttore	Responsabile operativo della formazione
ISW	Ingegneria del SoftWare
KIT	insieme organico di sussidi didattici per un corso di formazione
Librarian	Responsabile operativo della gestione della configurazione (librerie statiche).
Libreria di base	Libreria usata per gestire le baselines attuali e per controllarne le modifiche ed è la libreria dove sono mantenute le unità e i componenti di un CI promossi per l'integrazione. Ogni elemento della libreria è controllato e le modifiche a unità o componenti in questa libreria devono

	essere autorizzate dal CapoCommessa responsabile.
Libreria dinamica	Libreria usata per contenere le entità software modificate o create ex novo, è usata dai programmatori quando sviluppano l'elemento ed è accessibile liberamente dal programmatore responsabile. Rappresenta lo spazio di lavoro dei programmatori ed è controllata dai programmatori.
Libreria statica	Libreria usata per archiviare le baselines
malfunzionamento	hardware: Condizione accidentale che causa il guasto di una unità funzionale nell'esecuzione della sua funzione richiesta. software: Manifestazione di un errore nel software. Un malfunzionamento, se riscontrato, può causare un guasto. Un sinonimo è baco (bug).
METRICA SW	metodo di misura quantitativa relativa ad un attributo che caratterizza un prodotto sw (ISO9126) o un processo sw (ISO12207)
MILESTONES	Pietra Miliare – indica un preciso punto nella traiettoria del progetto, generalmente caratterizzato da eventi determinanti per il progetto (es.: conclusione di una fase di produzione, input di commessa, scadenze di termini contrattuali, consegne)
modello del ciclo di vita di un prodotto	schema di comportamento aziendale contenente processi attività e compiti relativi all'ideazione, sviluppo, esercizio e manutenzione di una applicazione sw. Il modello copre l'intera vita dell'applicazione, dalla concezione della prima idea dell'applicazione alla uscita definitiva dell'applicazione dall'esercizio.
MTPD	Module Test Plan Document – documento di pianificazione del test di modulo – test unitario
N/A	Non Applicabile
NC	Non Conformità
non conformità	Non soddisfacimento di requisiti specificati. La definizione riguarda lo scostamento o l'assenza di una o più caratteristiche di qualità, o di elementi del sistema qualità, rispetto ai requisiti specificati.
Operation review	occasione di autocontrollo collettivo (riesame, verifica) nel ciclo di commessa
Osservazione	Constatazione di un fatto rilevato durante il processo di verifica ispettiva e supportata da evidenza oggettiva.
OU	Output, elemento di un processo
PDCA	Plan Do Control Act – rappresentazione procedurale di un processo – Schema che decompone un processo in 4 subprocessi (pianificazione, esecuzione, controllo, reazione); per ogni subprocesso sono definiti 5 elementi (IN: input; OU: output; RE: regole; AT: attori; S/A: sottoattività)
PERT	Program Evaluation Review Tech – tecnica per la pianificazione del processo – diagramma reticolare in cui sono indicate tutte le attività e le relazioni tra attività
piano della qualità	Documento che precisa le modalità operative, le risorse e le sequenze di attività che influenzano la qualità di un determinato prodotto, servizio, contratto o progetto.
PIANO	documento di pianificazione del processo di progettazione –

DI PROGETTO	<p>specifica la struttura delle attività del processo e responsabili, impegni, durate, date di inizio e date di fine di ciascuna attività</p> <p>NOTA 1: un piano di progetto dovrebbe includere o far riferimento ad un piano di qualità</p> <p>NOTA 2: un piano di progetto dovrebbe includere o far riferimento ad un piano dei costi, un piano di impiego delle risorse materiali, un piano di interazione con il Cliente, con le funzioni di supporto e con la Direzione</p>
politica per la qualità	<p>Gli obiettivi e gli indirizzi generali di una organizzazione per quanto riguarda la qualità, espressi in modo formale dall'alta direzione.</p>
prestazioni di processo	<p>collezione dei dati di misura dei risultati disponibili di una specifica istanza di processo.</p> <p>Le prestazioni di processo devono essere confrontabili con la capacità di processo.</p>
Processi di supporto	<p>documenti, Controllo di Gestione, personale, Configurazione, Assicurazione e controllo di Qualità, Formazione, ISW (Mercato e tecnologie), Sistema informativo e registrazioni di qualità</p>
Processi primari	<p>Vendite, Acquisti, Produzione (Sviluppo sistemi sw, Erogazione di servizi di consulenza informatica)</p>
PROCESSO	<p>insieme di risorse, attività, responsabilità e procedure tra loro interconnesse e coordinate in modo tale da trasformare secondo uno scopo predeterminato entità in ingresso (input) in entità in uscita.</p> <p>Entità in uscita, output, prodotti del processo, workproduct sono sinonimi.</p> <p>NOTA 1: le risorse includono gli attori del processo (responsabili, collaboratori, decisori su deroghe e concessioni) ed i sistemi automatici di supporto al processo</p> <p>NOTA 2: le attività si collazionano in un flusso ordinato di processo</p> <p>NOTA 3: le responsabilità sono predefinite ed assegnate agli attori definiti per il processo</p> <p>NOTA 4: le regole secondo cui evolve il processo sono definite in SIAQ, nel contratto, nella normazione di settore, nella legislazione corrente</p>
PRODOTTI DI PROGETTO	<p>workproduct - i prodotti intermedi e finali del processo di progettazione, definiti rispetto agli obiettivi di progetto e consegnati al Cliente (esterno e/o interno) secondo quanto contrattato e/o pianificato in apertura attività</p>
prodotto	<p>Risultato di attività o processi.</p>
prodotto software	<p>Insieme completo dei programmi per elaboratore, delle procedure, dei dati e della relativa documentazione, designati per la consegna a un utente.</p>
programma di fidatezza	<p>La struttura organizzativa, le responsabilità, le procedure, i processi e le risorse utilizzati per la gestione della fidatezza.</p>
PS	<p>Progetto Strategico</p>
qualità	<p>Insieme delle caratteristiche di un'entità che conferiscono ad essa la capacità di soddisfare esigenze espresse ed implicite.</p>

QUALITA' DI PROCESSO	caratterizza la maturità del processo di sviluppo del prodotto software. Le metriche relative alla qualità del processo possono essere misurate anche durante lo sviluppo del prodotto per fornire un feedback in tempo reale della gestione del processo stesso.
QUALITA' DI PRODOTTO	si riferisce a indicatori globali della qualità del prodotto finale; esempi di metriche appartenenti a questa categoria sono non conformità ed azioni correttive durante il collaudo di accettazione, l'esercizio provvisorio, l'esercizio a regime.
RAC	Richiesta di Azione Correttiva – gestione non conformità
RE	Regole – elemento di un processo
replica o duplicazione	E' la copiatura o il trasferimento di software su supporti fisici o altri mezzi per impieghi o distribuzione successivi.
RISCHIO	possibilità di occorrenza di conseguenze negative/indesiderate di un evento. I rischi possono essere tecnici, di costo, di risorse, di scheduling; le origini dell'evento rischioso possono essere legate all'organizzazione interna, ai Fornitori, ai Clienti.
S/A	Sotto Attività di un processo
SC	Structure Chart – carta di struttura
scalata	La scalata è il processo che permette di accedere a risorse aggiuntive, coinvolgimento di livelli superiori di direzione, ecc. Per assicurare che i requisiti contrattuali siano soddisfatti.
SCMP	Software Configuration Management Plan – Piano di gestione della configurazione software
SE	Struttura Elaborativa, riferita a Componente/Elemento sw strutturalmente e funzionalmente distinto (Sistema, Sottosistema, Modulo, Programma, Processo)
sistema qualità	La struttura organizzativa, le procedure, i processi e le risorse necessari ad attuare la gestione per la qualità.
SIZE	categoria di metriche relative alla dimensione del software prodotto; può essere misurata in numero di statement o in numero di Function Point
software	Creazione intellettuale che comprende i programmi, le procedure, le regole e la relativa documentazione, pertinenti al funzionamento di un sistema per l'elaborazione di dati. Il software è indipendente dal supporto fisico sul quale risiede.
Software critico	è un'applicazione sw i cui errori possono avere un impatto sulla sicurezza degli esseri umani o causare rilevanti perdite sociali o finanziarie
SPMP	Software Project Management Plan – piano di gestione del progetto sw – piano di sviluppo della commessa TK – ammette due livelli di complessità in funzione della classe (1 o 2) di commessa
SQAP	Software Quality Assurance Plan – piano di assicurazione della qualità del software – ammette tre livelli di complessità in funzione della classe (1, 2 o 3) di commessa
SRD	Software Requirements Document – Specifiche dei requisiti software System Requirements Document – Specifiche dei requisiti di

	sistema
STAKE-HOLDER	individuo o gruppo di individui che portano un interesse comune nella prestazione dell'organizzazione del fornitore anche in relazione all'ambiente (fisico, economico, sociale) in cui il fornitore opera NOTA: gli stakeholder possono includere cliente – committente utente finale persone ed enti del fornitore partner del fornitore alta direzione, proprietari e finanziatori del fornitore subfornitori ambiente internal stakeholder – dipendenti
STD	classe di documenti che definiscono lo Standard (Guide) di altri documenti
SUM	Software User Manual – Manuale dell'utente dell'applicazione sw
sviluppo	insieme di tutte le attività che devono essere effettuate per costruire un'applicazione sw. Sinonimo di processo di Produzione
Sviluppo incrementale	sviluppo di un'applicazione sw secondo il modello incrementale del ciclo di vita
SVVP	Software Verification and Validation Plan – piano di verifica e validazione del sw
SW	SoftWare
TK	Turn Key – chiavi in mano – tipo di commessa di produzione per sviluppo di sistemi SW secondo la modalità a forfait
TM	Time & Material – tempo e materiali – tipo di commessa di produzione per erogazione di servizi di consulenza SW secondo la modalità monte ore
TPD	Test Plan Document – documento di pianificazione del test – test di (sotto)sistema sw
Unità responsabile delle verifiche ispettive	Unità organizzativa, o funzione nell'ambito di un'organizzazione, che ha la responsabilità di pianificare e di eseguire serie programmate di verifiche ispettive di sistemi qualità.
URD	User Requirements Document – Specifiche dei requisiti dell'utente (di applicazione sw)
V&V	Verifica e Validazione
validazione	Il processo di valutazione dei prodotti per assicurarne la conformità con i requisiti d'uso.
Valutando	Organizzazione o Ente oggetto della valutazione.
Valutatore di sistemi qualità	Persona che ha la qualifica per eseguire verifiche ispettive della qualità.
verifica (del software)	Il processo di valutazione dei prodotti (software) in una data fase per assicurarne la correttezza e la coerenza rispetto ai prodotti e ai requisiti specificati forniti in ingresso a quella fase.
Verifica ispettiva della qualità	Esame sistematico ed indipendente per determinare se le attività svolte per la qualità ed i risultati ottenuti sono in

	accordo con quanto pianificato e se quanto predisposto viene attuato efficacemente e risulta idoneo al conseguimento degli obiettivi.
VII	Verifica Ispettiva Interna
WBS	Work Breakdown Structure - è la decomposizione gerarchica di un progetto in attività di livello sempre più basso
workproduct	prodotti di progetto
WP	Work Package - porzione di lavoro in un progetto individuabile nella decomposizione gerarchica delle attività di un progetto

