

Project Management

ANNO: 18/19

RROK GJINAJ – LORIS CANGINI

Introduzione	4
Tar Pit	4
Complessità	4
Il mitico Mese-Uomo	5
Definizioni	6
Project Management Body of Knowledge (PMBOK)	6
Progetto	6
Programma	6
Project Management	7
Requisito	8
Team	9
tradizionale (surgical team)	9
agile (scrum)	9
Integrità concettuale	10
L'effetto Secondo sistema	10
Comunicazione	11
Informale (telefono, instant messaging, ...)	11
Meeting	11
Workbook (documentazione scritta)	11
Manuale	12
Usare un programma	12
Modificare un programma	13
Documentazione per un progetto Software	13
Qualità	13
Ciclo di vita di un progetto	14
Triangoli dello Scope	15
Creeps	15
Classificazione dei progetti	16
Ciclo di vita della gestione di progetto	17
Lineare	18
Incrementale	18
Iterativo	18
Adattivo	19
Extreme	19
Software Agile	19
Agile Manifesto	19
Definizione dei Processi	20
5 gruppi di processi	20
Scoping / Initiating	20
Planning	20

Launching / Executing	20
Monitoring & Controlling	20
Closing	20
9+1 Area di Conoscenza:	20
Integration Management	20
Scope Management	20
Conditions of Satisfaction	20
Requirement Breakdown Structure (RBS)	20
Cost Management	20
Time Management	21
Duration and Labor	21
Stima della durata dei Task	21
Quality Management	21
Quality planning	21
Quality assurance	21
Quality control	22
Human Resources Management	22
Teoria di Herzberg (Teoria dei 2 fattori)	22
Fattori motivazionali di J.D.Couger	22
Communications Management	23
Risk Management	24
Identificazione (Risk Identification)	24
Valutazione (Risk Assessment)	24
Mitigazione (Risk Mitigation)	25
Monitoraggio & Controllo	25
Procurement Management	25
Sollecitazione dei vendori	25
Valutazione dei vendori	26
Selezione dei vendori	27
Contratto con i vendori	27
Gestione dei vendori	27
Stakeholder Management	27
Scoping	28
Tools, templates e processi	28
Gestione delle aspettative	28
Conditions of Satisfaction (CoS)	29
Project scoping meeting	30
Requirement Breakdown Structure (RBS)	31
Diagrammi	35
Process Flow Diagram	35
Swim-Lane	35
Context Diagramming Process	35
Use-Case	35
Project Overview Statement (POS)	36

Planning	37
Work Breakdown Structure (WBS)	43
Gantt	44
Lancio / Esecuzione	44
Team di progetto	46
Kick-off	47
Assegnazione delle responsabilità (RASCI)	47
Cambiamento di scope	48
Monitoraggio & controllo	49
Sistema di reporting	49
Analisi del valore guadagnato (EVA)	49
Schedule Performance Index (SPI)	49
Cost Performance Index (CPI)	50
Scope Bank	50
Issues Log	50
Project Status Meeting	50
Daily Status Meeting	50
Problem Management Meeting	51
Problem Escalation Strategy	51
Chiusura	52

Introduzione

Tar Pit

Perché sembra spesso che un gruppo di programmati in un garage possa scrivere software in grado di competere con quelli delle grandi aziende?

Non converrebbe allora sostituire i grandi team con un gruppetto di persone chiuse in un garage?

NO!

Non si possono camparare i piccoli sistemi con i grandi, in quanto per i piccoli non serve fare approfondite analisi, testing, documentazione, piani di manutenzione, ...

Fare accurate analisi è fondamentale per iniziare a sviluppare qualcosa, ed una volta ottenuto un progetto bisogna attenersi ad esso rigidamente.

Detto ciò, bisogna però tenere bene a mente che “**l'unica costante è il cambiamento**”, quindi bisogna includere nel progetto tale possibilità.

Complessità

La **progettazione**, per quanto accurata possa essere, **non rimuove la complessità, ma permette di gestirla**.

Un **programma** è un insieme di istruzioni che implementano delle funzionalità.

Un **prodotto** in più deve:

- essere testato
- poter essere usato in diverse situazioni
- poter essere eseguito, modificato e testato da chiunque
- essere documentato

Fare un prodotto costa di più che fare un programma.

Un **sistema** è un insieme di prodotti che interagiscono, deve:

- definire con precisione le modalità di interazione tra componenti
- testare l'integrazione tra i componenti
- far rispettare ad input ed output dei tracciati ben definiti
- tener conto del fatto che certe risorse sono limitate.

Fare un sistema costa di più che fare un prodotto.

La complessità può essere suddivisa in:

- **Accidentale:** riguarda gli aspetti tecnici ⇒ **può essere ridotta**.
- **Essenziale:** riguarda il problema in sé ⇒ **complesso ridurla**.

Il mitico Mese-Uomo

Perché un software fallisce?

- violazione dell'integrità concettuale
- stime sbagliate
 - I costi del testing sono sempre sottostimati, mediamente serve la metà del tempo complessivo.
 - Altro motivo per stime sbagliate è la mancanza di coraggio da parte degli sviluppatori, che cedono alle richieste del management, rimandando così il problema a fasi più avanzate del progetto quando sarà più complesso porre rimedio.
- troppo ottimismo
- la convinzione che impegnandosi si possa sistemare tutto (**Sforzo ≠ Avanzamento**)
- mancato monitoraggio
- aggiunta di personale a progetto avviato
 - **Legge di Masson:** esiste un tempo minimo di sviluppo che non può essere ridotto ulteriormente, anche aumentando le risorse.
 - Aggiungere persone potrebbe anche far ritardare il progetto per via delle interrelazioni tra i processi non parallelizzabili.

Definizioni

Project Management Body of Knowledge (PMBOK)

PMBOK è la guida standard del project manager. È uno standard, descrive norme, metodi, processi e pratiche consolidate. È in continuo cambiamento in quanto le **buone pratiche** sono soggette a cambiamento.

PMBOK si focalizza sui singoli progetti. Ciò che c'è scritto in PMBOK va riadattato ai casi specifici.

Progetto

Definizione da PMBOK

Un progetto è un'iniziativa **temporanea** (ha un inizio ed una fine) intrapresa per creare un prodotto con caratteristiche di **unicità** (qualcosa di nuovo).

Serve che ci sia una fine perché deve esserci un risultato da portare a termine.

Il risultato del progetto è solitamente durevole, non si conclude con la fine del progetto.

Definizione di Wysocki

Un progetto è una **sequenza unica, complessa e connessa di attività** con un **obiettivo** specifico e con **vincoli di tempo e budget** da rispettare.

Va considerato anche il **business value**, cioè il valore percepito dal cliente.

Un progetto è una sequenza finita di attività dipendenti tra loro, il cui completamento con successo si traduce nella fornitura del business value atteso.

Un progetto può creare:

- un **prodotto** finito
- un **servizio**
- un **miglioramento** di un prodotto o servizio esistente
- **conoscenza**

Programma

Un **programma** è un insieme di progetti correlati, gestiti in modo coordinato per ottenere benefici non possibili dalla gestione individuale dei singoli progetti.

Si parla di **portafoglio** quando la relazione tra i progetti è debole o assente.

Project Management

Definizione Project Management Institute

Il Project Management è l'applicazione di **conoscenze, capacità, strumenti e tecniche** alle attività di progetto per **soddisfare i requisiti**.

Definizione di Wysocki

Il Project Management è un insieme di **strumenti, templates e processi** per rispondere a:

- Quale situazione affronta il progetto? (di cosa ha bisogno l'azienda)
 - un problema da risolvere
 - quale soluzione fornire (deliverable)
 - opportunità non sfruttata
- Cosa hai bisogno di fare? (cosa serve fare per risolvere il problema)
 - dato il deliverable, vanno definite le modalità di esecuzione.
- Cosa farai? (quale soluzione sceglioamo)
 - conseguenza diretta dei precedenti.
 - Verrà definito il [**Project Overview Statement \(POS\)**](#) (o equivalenti)
- Come lo farai? (come implementiamo la soluzione)
 - definizione dell'approccio
 - piano delle attività
- Come saprai di averlo fatto? (monitoraggio e controllo)
 - aumento del fatturato (**Increased Revenue**)
 - riduzione dei costi (**Avoided Cost**)
 - miglioramento del servizio (**Improved Service**)
- Quanto bene lo hai fatto? (così da migliorare)
 - analisi retrospettiva per miglioramento continuo
 - Quanti criteri di successo sono stati soddisfatti?
 - Che performance c'è stata nel team?
 - L'approccio scelto ha funzionato bene?
 - Quale lezione è stata imparata utile per progetti futuri?

Definizione di Wysocki n.2

Il Project Management è un approccio basato sul buon senso che **coinvolge il cliente** al fine di soddisfare i **requisiti** fornendo l'**incremento di business value** atteso.

- business value definito dal cliente.
- project manager responsabile per soddisfacimento requisiti

Requisito

Definizione IEEE

Un requisito è:

- una condizione o capacità utile all'utente per risolvere un problema o raggiungere un obiettivo (**scope**).
- una condizione o capacità che deve essere raggiunta o posseduta da un sistema o da un componente per soddisfare un contratto, uno standard, una specifica o un altro documento formale (**solution**).
- una rappresentazione documentata di una condizione o capacità del tipo espresso nei primi 2 punti.

Definizione di Wysocki

Un requisito è uno stato finale desiderato che, una volta integrato, fornisce all'organizzazione un aumento misurabile del business value.

Una volta definiti i requisiti sarà possibile stabilire le attività da eseguire e le stime.

Team

- **tradizionale (surgical team)**

- **chirurgo**: dirige il team, prendendo tutte le decisioni
- **copilota**: stesse mansioni del chirurgo, ma meno esperienza.
- **amministratore**: esecutore delle direttive del chirurgo
- **editor**: corregge, organizza, aggiorna la documentazione
- **segretaria**: supporta amministratore ed editor, senza conoscenze tecniche
- **program clerk**: supporta l'amministratore, ha conoscenze tecniche
- **toolsmith**: mantiene tutti gli strumenti utili agli sviluppatori
- **tester**: prepara ed esegue test
- **language lawyer**: definisce come usare il linguaggio di programmazione

- **agile (scrum)**

- **product owner**: responsabile del prodotto, definisce il “cosa”
- **scrum master**: coach al servizio del team (non è il capo team)
- **dev team**: sviluppatori, definiscono il “come”

Il tipo di processo incide sulla scelta del tipo di team.

Integrità concettuale

Per produrre un sistema **efficace** (fare tutto ciò per il quale è stato progettato), **efficiente** (lo deve fare nel modo migliore) e user-friendly è necessario che sia rispettata la sua integrità concettuale, ossia è necessario che le funzionalità e la loro implementazione risponda solo e solamente ai requisiti definiti in fase di progettazione.

Separare l'architettura dall'implementazione (il cosa dal come)

- l'architetto decide il cosa (requisiti)
 - dialogando col cliente
 - tenendo conto degli utenti tramite feedback
 - includendo il product owner (che non è l'architetto, che normalmente fa parte del dev team)
 - commerciale (budget, ...)
 - ufficio legale (contratti, limiti legali, ...)
- il team decide il come
 - l'architetto non dovrebbe decidere il come (lascia poca libertà agli sviluppatori)
 - scarso coinvolgimento degli altri
 - decide tutto 1 persona (poca scalabilità)
 - se tutto viene deciso dal team
 - si potrebbe non arrivare mai ad una conclusione
 - deve esserci poco turnover
 - tutti sono bloccati nel decidere e non riescono a fare altro
 - bisogna trovare il giusto bilanciamento
 - solo parte del team decide

L'effetto Secondo sistema

Finito un primo progetto si scopre sempre che si poteva farlo meglio.

Iniziando il secondo, si vuole mettere ciò che non si era fatto bene, ma così facendo si tende a fare overdesign (troppa roba).

A volte è bene accontentarsi di un sistema con meno roba se motivato (costo più basso, rimanere nel budget, ...).

Comunicazione

Informale (telefono, instant messaging, ...)

- ✓ immediato
- ✗ rimane poca traccia (o non rimane)
 - dimenticanze
 - l'informazione non sarà usabile in futuro
 - possibilità di rinnegamento di ciò che si è detto
- ✗ informazioni non strutturate
 - facile non trovare ciò che serve

Meeting

- ✗ serve un sacco di tempo da parte di tutti i partecipanti
- ✗ vanno pianificati a modo
- ✓ c'è (o dovrebbe esserci) un verbale
 - c'è per riunioni formali
 - non c'è per riunioni rapide (stand-up meeting)

Workbook (documentazione scritta)

- ✓ oramai si fanno con strumenti appositi sul web
- ✓ essendo scritta, ne rimane traccia

Manuale

Quando si fa un sistema, poi qualcuno dovrà imparare ad usarlo, il manuale torna utile da questo punto di vista.

Il manuale andrebbe scritto quanto prima possibile, possibilmente assieme al progetto stesso, non alla fine.

Esistono:

- manuali utente (quello che troviamo nella scatola)
 - estetica
 - funzionalità generali
 - istruzioni passo-passo
- manuali tecnici
 - dettagli (per espandere, sistemare, ...)

Serve documentare bene per:

- indicare il funzionamento
- ricordarsi ciò che si è fatto
- condividere il proprio lavoro (un altro deve sapere ciò che ho fatto)
- tener traccia dei problemi e delle loro soluzioni

Normalmente grandi aziende documentano a modo, nelle piccole (la maggioranza!) c'è tendenza a farne meno.

Usare un programma

Nella documentazione bisogna chiarire:

- **Scopo:** a cosa servirà il software
- **Ambiente:** dove girerà il software (quali macchine, quali configurazioni, ...)
- **Dominio e Range:** quali input ed output sono validi
- **Funzioni realizzate ad algoritmi usati:** cosa permette di fare il software e come
- **Formato di input ed output:** qual è il mezzo
- **Istruzioni operative** (manuale): come si usa il software
- **Opzioni:** cosa si può scegliere e come lo si fa
- **Tempo di esecuzione:** quanto ci mette in una determinata configurazione con determinati input.
- **Accuratezza e Checking:** quanto preciso è e che controlli ci sono per verificare la correttezza

Modificare un programma

Chi modifica necessita di tutti i dettagli implementativi e della struttura dei vari componenti.
Si possono individuare diversi elementi:

- rappresentazione grafica dei componenti
- strutture dati e input/output
- algoritmi usati
- tecnologie usate e modalità di utilizzo
- possibili miglioramenti non attualmente apportati

Un modo efficace per documentare è inserire la documentazione nel codice stesso

- nomi delle variabili con senso!
- indentazione
- commenti (magari seguendo uno stile per farli capire in automatico e convertirli in formati leggibili (ed es. via web))

Documentazione per un progetto Software

Non si fa all'inizio e basta e poi la usa.

La si inizia a scrivere all'inizio e si continua ad aggiornare.

Sarebbe necessario produrre:

- **What: obiettivi:** i bisogni da soddisfare, gli obiettivi, i desiderata, i vincoli, le priorità.
- **What: specifiche prodotto:** in cosa consiste il prodotto. Può essere inizialmente sintetico, ma evolverà nel manuale e in una descrizione dell'implementazione.
- **When: schedula:** un calendario che indichi quando fare le cose e quanto tempo dovranno durare le attività. ([GANTT](#))
- **How much: budget:** correlato al when, come spendere (pagare i lavoratori, comprare parte della soluzione, ...).
- **Where: allocazione spaziale:** dove faccio le varie parti. Col software il problema non è dei più grossi, ma va considerato.
- **Who: organizzazione:** chi è coinvolto e le le varie mansioni, per determinare le responsabilità.

Fare documentazione è fondamentale per:

- mettere per iscritto le decisioni
 - ci si accorge dei limiti
 - si prendono diverse mini-decisioni che non si erano prese
- comunicare le decisioni agli altri
 - se non si scrive qualcosa, poi non bisogna meravigliarsi se qualcuno ignora certi aspetti.
- seguire ciò che si era deciso
 - controllo continuo per scoprire eventuali ritardi o problemi

Qualità

Vanno considerate:

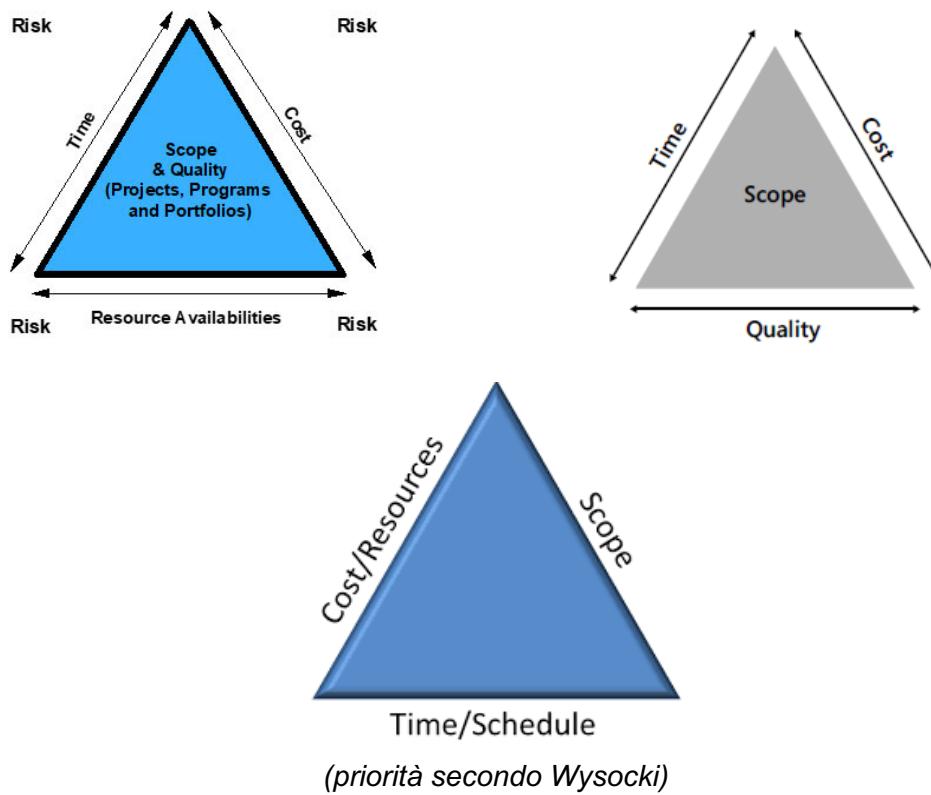
- **Qualità del prodotto:** efficienza, affidabilità, design, ...
Può essere concordata col cliente per stare entro certi vincoli (principalmente di budget).
- **Qualità del processo:** inherente ai processi di gestione.
Verificata da enti appositi per raggiungere la **certificazione di qualità**.

Ciclo di vita di un progetto

1. Idea
2. Charter: documento che definisce cosa dovrà esserci nel progetto
3. Definizione scope
4. Costruzione del piano (Gantt) ⇒ **baseline**
5. Avanzamento: monitorando rispetto alla baseline
6. Accettazione: verifica col cliente
7. Approvazione: conferma del risultato dal cliente
8. Consegnna

Normalmente il costo di un progetto si ha, per la maggior parte, nella fase intermedia. I cambiamenti grossi andrebbero fatti quanto prima possibile, in quanto i costi sarebbero minori.

Triangoli dello Scope



Priority Variable	Critical (1)	(2)	(3)	(4)	Flexible (5)
Scope				X	
Quality			X		
Time	X				
Cost					X
Resource Availability		X			

Ci sono casi dove le priorità dipendono molto dal progetto, dall'organizzazione, dal periodo
...

Il triangolo è un sistema che deve restare in equilibrio.

Le lunghezze dei lati indicano le risorse disponibili e limitano lo scope e la qualità.

Una modifica può portare il sistema fuori equilibrio, nel caso bisogna intervenire definendo:

- l'impatto del cambiamento sul progetto ([Project Impact Statement](#))
- le strategie per risolvere i problemi ([Problem Escalation Strategy](#))

Creeps

Sono tanti piccoli cambiamenti durante il progetto che alla fine portano a grandi problemi.

- **Scope Creep:** cambiamento del progetto rispetto a quanto previsto.
 - è evitabile mantenendo una certa formalità anche per i piccoli (apparentemente!) cambiamenti.
- **Hope Creep:** situazioni nascoste nella convinzione/speranza di poter recuperare.
 - va evitato tramite monitoraggio frequente (non troppo!) del PM.
- **Effort Creep:** un membro del team non è abbastanza produttivo.
 - Il PM deve monitorare le varie attività e membri del team.
- **Feature Creep:** comportamento scorretto di un membro che fa qualcosa di non concordato (violando l'integrità concettuale).

Classificazione dei progetti

Serve classificare i progetti per aiutare il PM a scegliere quale approccio utilizzare.

La classificazione può avvenire per:

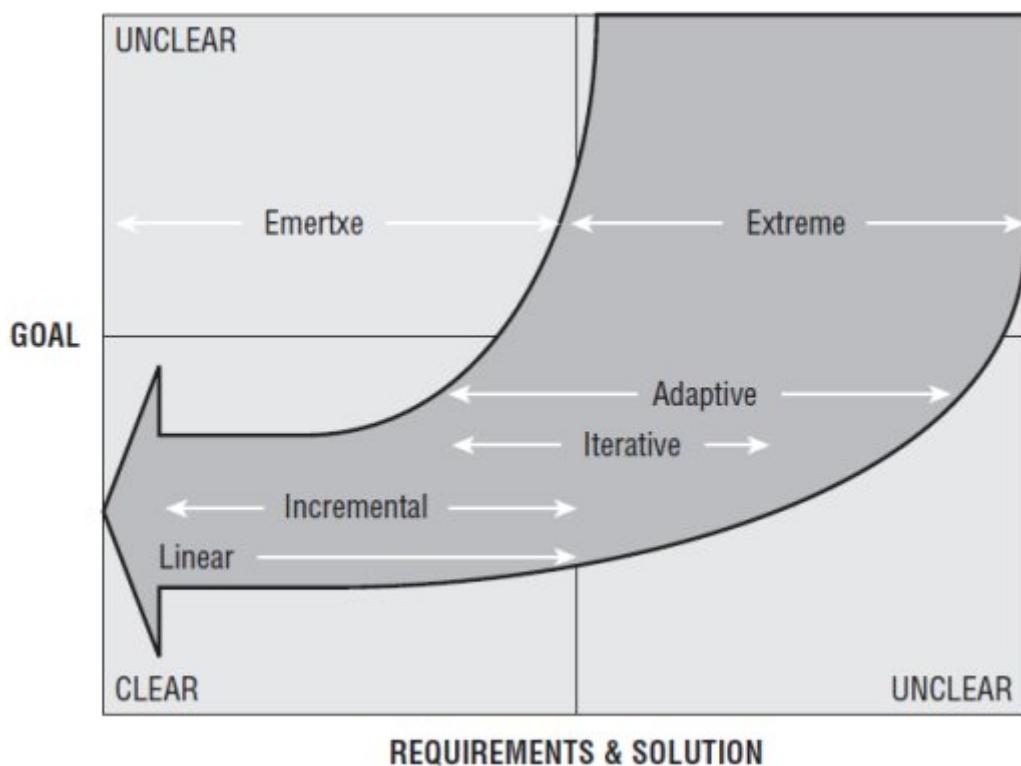
- **dimensione:** costi, durata, business value, ...
- **applicazione:** installazione, sviluppo software, ...
- **tipo:** nuovo, manutenzione, aggiornamento, ...
- **rischio:** alto, medio, basso
- **business value:** alto, medio, basso
- **durata:** a categoria (1-3 mesi, 3-6 mesi, ...)
- **complessità:** alta, media, bassa
- **tecnologia:** ben conosciuta, usata raramente, nuova
- **reparti impiegati:** 1, pochi, molti, tutti
- **costi:** alti, medi, bassi

Ciclo di vita della gestione di progetto

Il ciclo di vita della gestione di un progetto è una sequenza di processi che possono essere raggruppati in:

- **Scoping**: definizione dell'ambito (cosa dobbiamo fare)
- **Planning** (Pianificazione)
- **Execution / Launching** (Esecuzione/Lancio)
- **Monitoring & Controlling** (Monitoraggio & Controllo)
- **Closing** (Chiusura)

	Soluzione chiara	Soluzione non chiara
Goal non chiaro	Emertxe Project Management (MPx)	Extreme Project Management (xPM)
Goal chiaro	Traditional Project Management (TPM)	Agile Project Management (APM)



Il processo da utilizzare non è possibile deciderlo all'inizio in quanto non sappiamo il rischio del progetto.

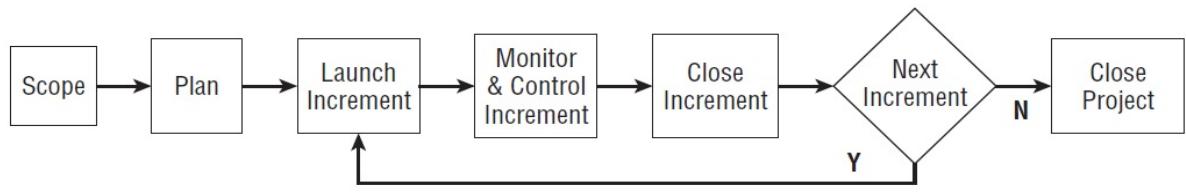
Vanno esplicitati in pochi documenti i requisiti, il loro valore ed i criteri di successo del progetto, così da permettere la decisione dell'approccio migliore da parte del senior management.

Lineare



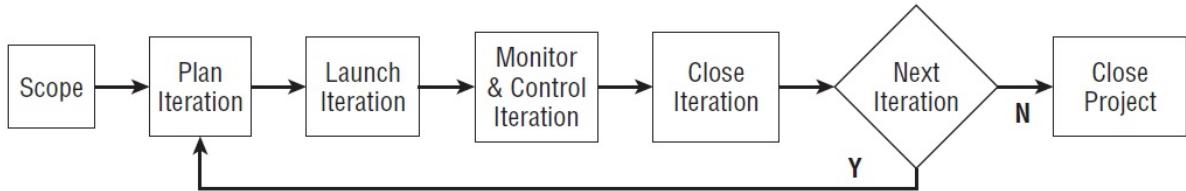
- Soluzione e requisiti chiaramente definiti.
- Non ci sono modifiche rilevanti dello scope.
- Progetti di routine e ripetitivi.
- Uso di templates consolidati.

Incrementale



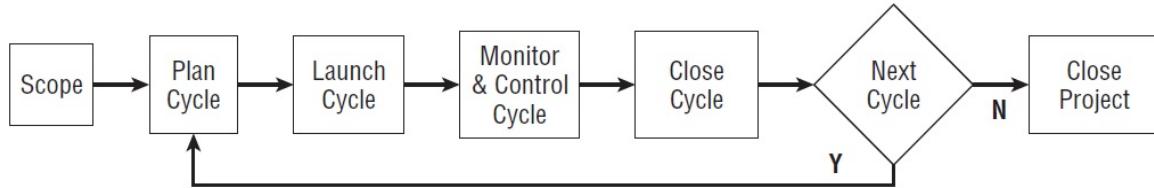
- Come lineare, ma parti del sistema verranno rilasciate a pezzi, incrementalmente.
- Lo scope ha una qualche probabilità di cambiare.

Iterativo



- Requisiti e funzionalità instabili o incompleti.
- Si apprende mentre si fa.

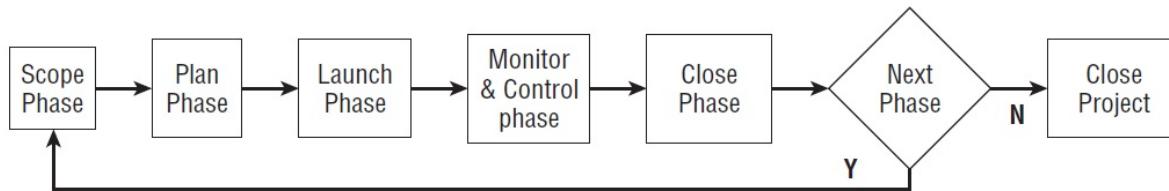
Adattivo



(Possibile mettere la retroazione anche fino allo scope, dipende dai punti di vista).

- *L'obiettivo è noto, ma non la soluzione.*
- *Soluzione influenzata dai cambiamenti attesi.*

Extreme



- *Obiettivo e soluzione non noti.*
- *Tramite le iterazioni si definiscono obiettivo e soluzione.*

Software Agile

Agile Manifesto

1. Soddisfazione del cliente dandogli continuamente qualcosa
2. il cambiamento è benvenuto, anche quando avviene tardi nello sviluppo
3. consegna frequentemente (sprint di poche settimane)
4. gli utenti e gli sviluppatori devono lavorare insieme giornalmente
5. Gli individui vanno supportati nelle loro necessità, credere che possano completare i task loro assegnati
6. comunicazione faccia a faccia, è + efficiente ed efficace
7. Il software funzionante è la misura primaria del progresso del progetto
8. I processi agile promuovono sviluppo sostenibile (mantenimento di un basso debito tecnico).
9. mantenere continua attenzione all'eccellenza tecnica ed al buon design
10. La semplicità (massimizzare il lavoro non fatto (non fare nulla di non necessario)) è essenziale.
11. Le migliori architetture emergono dai team che si auto-organizzano
12. riflettere ad intervalli regolari su come diventare + efficaci

Non si è anti-metodologia, ed è bene fare documentazione, ma non centinaia di pagine poco manutenute.

Definizione dei Processi

5 gruppi di processi

- [Scoping / Initiating](#)
- [Planning](#)
- [Launching / Executing](#)
- [Monitoring & Controlling](#)
- [Closing](#)

9+1 Arene di Conoscenza:

Integration Management

Gestione dei processi che collegano diversi deliverables, che integrati risolvono il progetto.

- Sviluppo del Project Overview Statement
- Sviluppo dello Scope Statement
- Sviluppo del piano di progetto
- Esecuzione
- Monitoraggio e controllo
- Controllo dei cambiamenti
- Chiusura del progetto

Scope Management

Conditions of Satisfaction

Requirement Breakdown Structure (RBS)

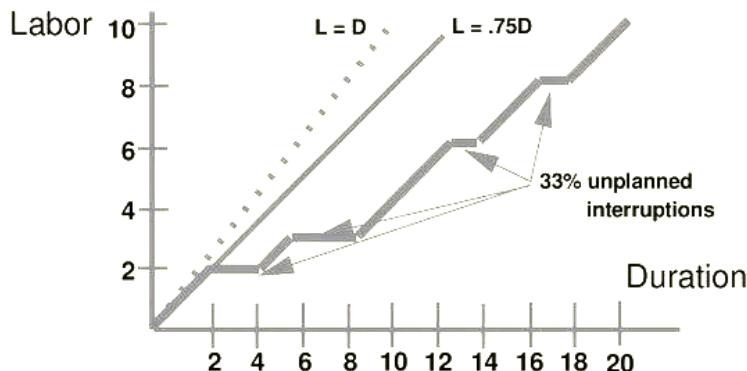
Cost Management

Da applicare durante pianificazione ed esecuzione.

- Costo della manodopera
- Beni strumentali
- Servizi appaltati
- Spese di trasferta

Time Management

- Duration and Labor



MODEL ASSUMPTIONS

- * Individuals work at 75 percent efficiency rate.
- * Unplanned interruptions account for 33 percent of clock time.

- Stima della durata dei Task
 - somiglianza con altre attività
 - dati storici
 - pareri di esperti
 - tecnica Delphi
 - tecnica Three-point
 - tecnica Delphi a banda larga

Quality Management

- Qualità del prodotto
- Qualità del processo
- Vanno considerati 3 processi fondamentali:
 - Quality planning

Determina quali sono gli standard di qualità e come soddisfarli.

Gli input sono:

- Fattori ambientali esterni
- Fattori ambientali interni
- Project Overview Statement
- Project Management Plan

Il risultato è il **piano di qualità**, che documenta quali politiche devono essere soddisfatte, quali metriche usare per misurare la qualità e come migliorare il processo.

- Quality assurance

Verifica che il piano di qualità sia rispettato.

- Quality Audits (verifica della qualità)
- Process Analysis
- Project Quality Management Tools
- Quality control
 - Definisce azioni da intraprendere in caso di non conformità con alcuni standard decisi.

Human Resources Management

- Profili chiave:
 - Developer co-manager
 - Client co-manager
 - Core Team
 - Task Leaders
 - Team Members
- Per ogni membro del team devono essere definite le abilità e le competenze.

Teoria di Herzberg (Teoria dei 2 fattori)

- Fattori motivazionali
 - Realizzazione
 - Riconoscimento
 - Carriera
 - Responsabilità
 - il lavoro stesso
- Fattori di igiene (devono esserci)
 - Politiche aziendali: regole di funzionamento dell'azienda
 - Pratiche amministrative: come verranno gestite le situazioni
 - Condizioni di lavoro
 - Supervisione tecnica
 - Relazioni interpersonali
 - Sicurezza del lavoro
 - Salario

Fattori motivazionali di J.D. Couger

(con priorità decrescente)

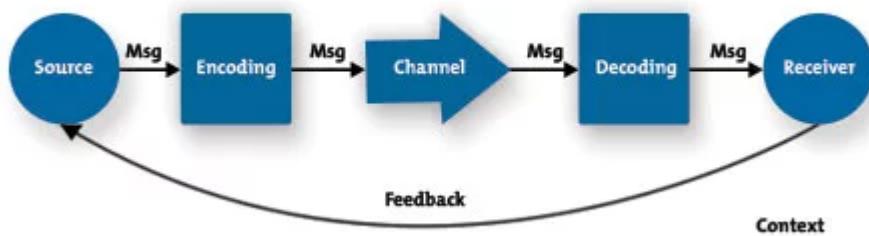
1. il lavoro stesso
2. Realizzazione
3. Carriera
4. Salario
5. Riconoscimento
6. Responsabilità
7. Supervisione tecnica
8. Relazioni interpersonali
9. Sicurezza del lavoro
10. Condizioni di lavoro
11. Politiche aziendali

Il Project Manager controlla o influenza:

- Challenge
- Recognition
- Design del lavoro
 - Varietà delle skills
 - Identità del task
 - Importanza del task
 - Autonomia
 - Feedback

Communications Management

- L'informazione deve essere **trasmessa correttamente**
- **Un messaggio inviato è un messaggio ricevuto**
- L'informazione deve essere **compresa ed accettata** dal destinatario



- Tipi di comunicazione:
 - 1 a 1
 - Conversazioni
 - Riunioni
 - Elettronica
 - E-mail
 - Siti web
 - Database
 - Scritta
 - Lettere
 - Documenti
 - Report
 - Interfacce



Risk Management

1. Identificazione (Risk Identification)

- Rischi tecnici
- Rischi di gestione
- Rischi organizzativi
- Rischi esterni

2. Valutazione (Risk Assessment)

- Calcolare la probabilità che l'evento accada.
 - Se la probabilità è vicina al 100%, allora non è un rischio.
- Trovare l'impatto atteso moltiplicando la probabilità per l'impatto effettivo.
 - Se l'impatto atteso è irrilevante allora il rischio può essere ignorato.
- Si può procedere in 2 modi:
 - **Valutazione statica:** fatta all'inizio del progetto, durante il planning.
 - **Valutazione dinamica:** fatta durante il planning ed aggiornata durante l'avanzamento.

- Templates:

- Risk Matrix

		High	Medium	High	High
	IMPACT	Medium	Low	Medium	High
		Low	Low	Low	Medium
			Low	Medium	High
					LIKELIHOOD

- Risk Identification & Assessment

Risk Category	Scope Triangle	Event #	Event	Y/N	Prob.	Impact	Priority	Mitigate Y/M/N
Tech	Scope	TS01	Available HW/SW technology limits scope					
		TS02	New technology does not integrate with old					
Tech	Time	TT01	Integrating technologies impacts schedule					
Tech	Cost	TC01	Unexpected need to acquire hardware					
Tech	Cost	TC02	Unexpected need to acquire software					
Tech	Quality	TQ01	Technology limits solution performance					
Tech	Res	TR01	New/unfamiliar technology					
Tech	Res	TR02	Inadequate software sizing					
Tech	Res	TR03	Inadequate hardware sizing					
Proj Mgt	Scope	PS01	Senior scope change request too significant					
Proj Mgt	Time	PT01	Schedule too aggressive					
Proj Mgt	Time	PT02	Interproject dependencies compromise schedule					
Proj Mgt	Time	PT03	Task duration estimates too optimistic					
Proj Mgt	Time	PT04	Difficulty scheduling meetings					
Proj Mgt	Quality	PQ01	Inaccurate assumption					

- Risk Assessment Worksheet

Project Activity	A	B	C	D	E	F	G	H	I	J	Score
Rqmnts Analysis	2	3	3	2	3	3	2	2	1	1	22
Specifications	2	1	3	2	2	2	1	2	2	3	20
Preliminary Design	1	1	2	2	2	2	1	2	2	2	17
Design	2	1	2	2	2	3	1	2	2	1	18
Implement	1	2	2	3	3	2	1	2	2	1	19
Test	2	2	2	2	2	3	2	2	2	2	21
Integration	3	2	3	3	3	3	2	3	3	2	27
Checkout	1	2	2	3	3	3	2	3	2	2	23
Operation	2	2	3	3	3	3	3	3	1	1	24
Score	16	16	22	22	23	24	15	21	17	15	191
Maximum score is 270. Risk level for this project is 191/270 = 71%.											

3. Mitigazione (Risk Mitigation)

- Accettare il rischio senza preparare contromisure
- Evitare il rischio (non faccio certe scelte per non doverlo affrontare)
- Preparare un piano di contingenza (stabilisco cosa farò quanto l'evento accadrà)
- Mitigare (modifico il progetto in modo tale che il rischio abbia un impatto minore)
- Trasferire (faccio fare a qualcun altro, predispongo assicurazioni, ...)

4. Monitoraggio & Controllo

- Risk Log Entry

ID #	Risk Description	P	I	Risk Owner	Action to be Taken	Outcome

Procurement Management

1. Sollecitazione dei vendori

- Sviluppare la RBS
- Decidere se usare una terza parte
- Identificare i vendors
- Selezionare il PMT (Procurement Management Team)
- Determinare la relazione con il venditore
- Determinare la strategia di acquisizione
- Stabilire i criteri di valutazione delle offerte
- Sviluppare i processi di gestione dei contratti

- Preparare e distribuire il **RFP (Request for Proposal)**
 - Il documento che verrà spedito ai fornitori che specifica tutto il necessario. Deve contenere:
 - Introduzione
 - Business profile: chi siamo e cosa facciamo
 - Descrizione del problema/opportunità
 - POS e/o RBS
 - Stime costi, tempi e prezzo previsto
 - Criteri di valutazione delle offerte
 - Descrizione responsabilità del vendor
 - Dettagli amministrativi
 - Istruzioni per i vendors
 - Riferimenti e contatti
- Rispondere alle domande degli offerenti
 - Come selezionare i vendors da sollecitare?
 - **Request for Information (RFI)**: chiedo informazioni sulle capacità del vendor, anche con possibili test tecnici.
 - **Pubblicità**: cerco vendor tramite le pubblicità in canali specifici, per poi mandare loro un RFI.
 - Chiedere ai fornitori abituali (**passaparola**)
 - **Fiere o business meeting**

2. Valutazione dei vendori

- Valutare i vendors
 - Forced Ranking

Vendor	Consultant A	Consultant B	Consultant C	Consultant D	Rank Sum	Forced Rank
1	2	3	2	4	11	3
2	4	1	1	2	8	1
3	6	2	5	5	18	6
4	1	5	3	1	10	2
5	3	4	4	3	14	4
6	5	6	6	6	23	6

- Paired Comparison

	1	2	3	4	5	6	SUM	RANK
1	X	1	1	0	1	1	4	2
2	0	X	1	0	1	1	3	3
3	0	0	X	0	0	1	1	5
4	1	1	1	X	1	1	5	1
5	0	0	1	0	X	1	2	4
6	0	0	0	0	0	X	0	6

- Valutare le risposte al RFP

- Ridurre la lista dei fornitori
 - Richiedere presentazioni/demo
3. Selezione dei venditori
- Selezionare il vincitore
 - Il risultato può essere:
 - Nessun vincitore: nessuno soddisfa i requisiti minimi (bisogna rifare tutto dall'inizio)
 - Un solo vincitore
 - Vincitori multipli: un insieme di fornitori soddisfa i requisiti.
4. Contratto con i venditori
- Negoziazione con i vendori vincitori dei termini contrattuali
5. Gestione dei venditori
- Definire i rapporti con i fornitori
 - Monitorare i progressi e le performance
 - Fare test
 - Gestire la transizione tra fornitori e cliente

Stakeholder Management

Stakeholders: gli interessati al progetto (cliente)

- *Fino alla 4^a edizione del PMBOK era incluso nel Communications Management.*
- Identificare gli stakeholders
 - Planificare la gestione degli stakeholders
 - Gestire il coinvolgimento degli stakeholders
 - Controllare il coinvolgimento degli stakeholders

Scoping

Lo scope (ambito) definisce **ciò che deve e non deve essere fatto**.

Una volta definito, bisogna focalizzarsi per mantenere i cambiamenti compatibili con esso, mantenendo l'integrità concettuale.

Se una modifica è in contrasto con lo scope, allora la situazione va risolta applicando strategie apposite.

Il processo per definire lo scope viene chiamato **Conditions of Satisfaction**.

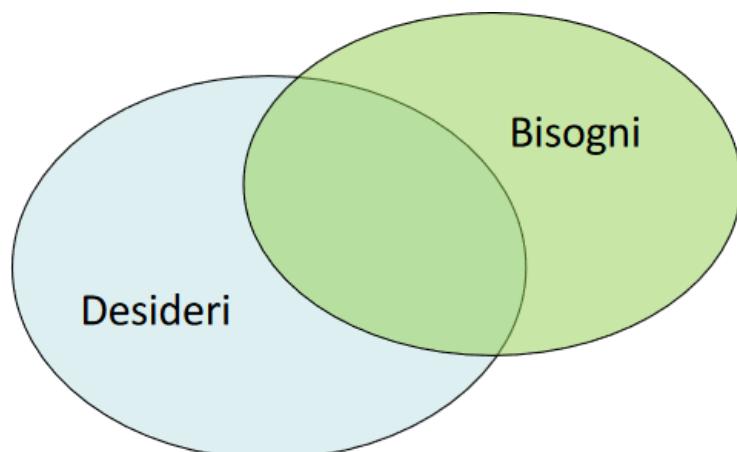
Definizione degli obiettivi e dei business values da acquisire (RBS).

- Ottenerne dal cliente i suoi veri bisogni e documentarli.
- Negoziare col cliente le modalità di soddisfacimento dei suoi bisogni.
- Descrivere sinteticamente il progetto.
- Ottenerne l'approvazione da parte del senior management.

Tools, templates e processi

- Conditions of Satisfaction
- Project Scoping Meeting
- Raccolta requisiti
- Diagrammi di processi
- Prototipazione
- Convalida dei casi aziendali
- Project Overview Statement
- Approvazione del POS e conferma a poter procedere con la pianificazione

Gestione delle aspettative



Ciò che un cliente chiede non è detto che sia ciò di cui ha bisogno.

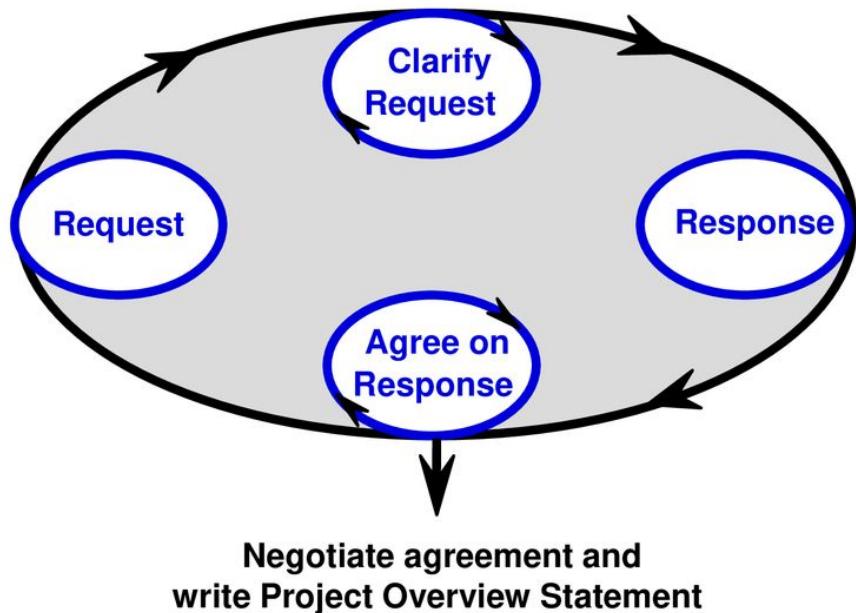
È necessario:

- Assicurarsi di aver compreso i desideri del committente
- Assicurarsi che il committente abbia compreso cosa verrà fatto
- Assicurarsi che i desideri del cliente corrispondano ai bisogni
- Coinvolgere e rendere il committente partecipe attivo dello scoping
- Mantenere il committente informato

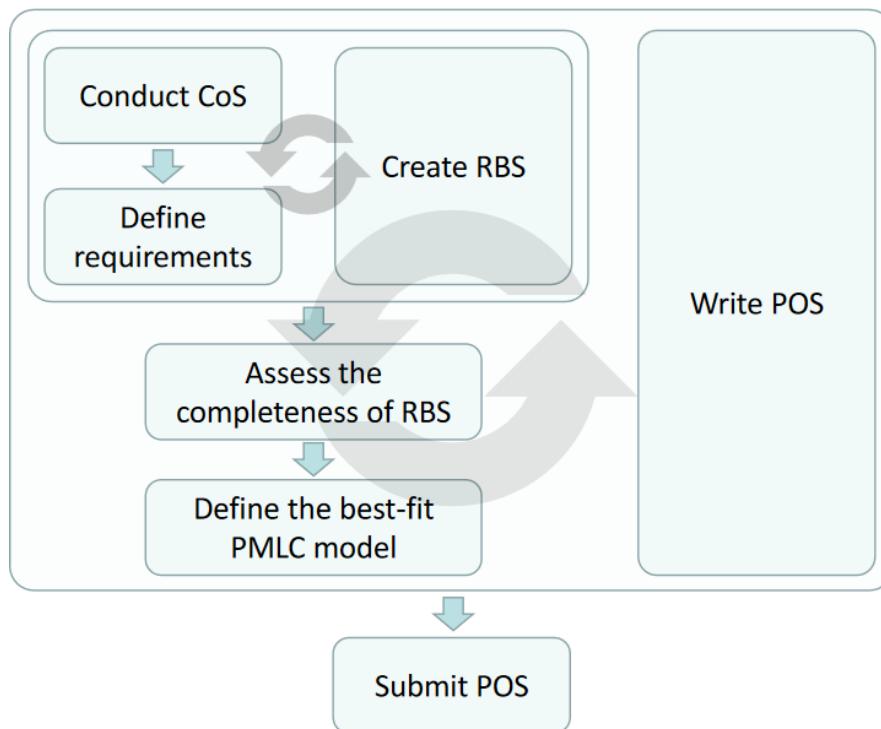
Conditions of Satisfaction (CoS)

I Criteri di accettazione non sono la stessa cosa.

I CoS servono per determinare cosa deve essere il risultato, mentre i criteri determinano se il risultato può essere accettato o meno.



Project scoping meeting



Serve per incontrare il committente ed iniziare a delineare lo scope del progetto.

Devono partecipare:

- Project Manager
- Rappresentanza del cliente
- Membri del team
- Facilitatore
- Tecnografo

Ogni meeting ha un'**agenda**, formata da:

- Introduzione
- Scopo del meeting (conduce il facilitatore): chiarisce cosa dobbiamo fare
- Discussione delle CoS
- Descrizione dello stato corrente (conduce il rappresentante del committente)
- Descrizione del problema/opportunità (conduce il rappresentante del committente)
- Descrizione dello stato finale desiderato (conduce il rappresentante del committente)
- Definizione dei requisiti e documentazione (conduce il facilitatore)
- Discussione del gap tra stato corrente e stato desiderato (conduce il project manager)
- Scelta dell'approccio migliore per gestire il progetto (conduce il project manager)
- Abbozzamento ed approvazione del POS
- Aggiornamento ad eventuale riunione successiva

I deliverables di questa fase devono essere:

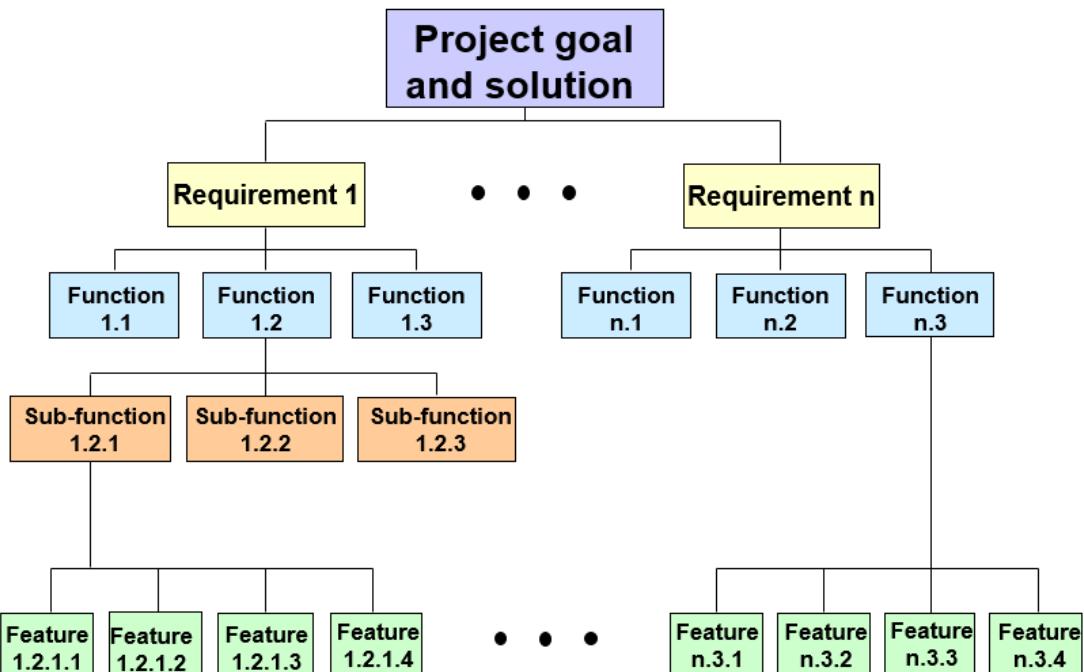
- [CoS](#)
- [RBS](#)
- Il modello di gestione migliore ([PMLC model](#))
- [POS](#)

Requirement Breakdown Structure (RBS)

I requisiti possono essere raccolti in uno dei seguenti modi:

- **Facilitated Group Session:** riunioni ristrette a certi partecipanti
 - ✓ problemi risolti tramite un facilitatore
 - ✓ requisiti dettagliati, documentati e verificati immediatamente
 - ✓ Eccellente per processi interfunzionali
- **Interviste**
 - ✓ Partecipano gli utenti finali
 - ✓ Descrizione ad alto livello delle funzioni e dei processi
 - X Gli stakeholder potrebbero non sapere quali informazioni fornire
 - X Si potrebbero ignorare i veri bisogni del committente
 - X Le descrizioni non sono necessariamente accurate
- **Osservazioni**
 - ✓ Fornisce specifiche e descrizioni complete dei processi osservati
 - ✓ Molto efficace quando le attività sono difficile da descrivere
 - X Osservare e documentare può essere molto difficile
 - X Ciò che si è osservato può essere male interpretato
- **Riuso dei requisiti**
 - ✓ Requisiti generati e rifiniti velocemente
 - ✓ La raccolta può essere ridotta
 - ✓ Il committente è soddisfatto dalla precedente esperienza positiva
 - ✓ Qualità maggiore
 - ✓ Non si reinventa la ruota
 - X Il mantenimento di archivi richiede investimenti
 - X Può essere violato il diritto d'autore
 - X Le similitudini tra funzionalità può essere male interpretata
- **Diagrammi di processi**
 - ✓ Eccellente per processi interfunzionali
 - ✓ Comunicazione visuale
 - ✓ Verifica del "cos'è/cosa non è"
 - X Il committente deve essere aperto ai miglioramenti proposti
 - X Bisogna essere esperti nel fare queste analisi
 - X Serve molto tempo
- **Prototipazione**
 - ✓ Generazione di nuove idee innovative
 - ✓ Gli utenti chiariscono cosa vogliono
 - ✓ Gli utenti identificano requisiti mancanti

- ✓ Focus sul committente
- ✓ Verifica anticipata della fattibilità (proof of concept)
- ✓ Stimola il processo mentale
- X Il committente potrebbe volere l'implementazione del prototipo (va spiegato bene prima cos'è un prototipo col committente)
- X Risulta difficile decidere quando fermarsi
- X Richiede skill specializzati
- X Assenza di documentazione
- Casi d'uso
 - ✓ Sistema descritto prima dell'interazione del committente
 - ✓ Si usano tutti i possibili scenari
 - ✓ Flusso eventi ed eccezioni rivelato facilmente
 - ✓ Miglioramento del soddisfacimento e del design
 - X Incongruenze
 - X ...



Se un requisito è presente in più di un posto, si duplica e si inseriscono delle note esplicitando la cosa.

- ✓ consistente col PMBOK
- ✓ approccio basato sui deliverables
- ✓ intuitivo e significativo per il committente
- ✓ rende possibile mantenere il contatto con il cliente
- ✓ descrizione ad alto livello della WBS
- ✓ Non serve un facilitatore esperto
- ✓ Non richiede di imparare l'uso di approcci nuovi
- ✓ È Intuitivo
- ✓ Permette al committente di lavorare col team
- ✓ Rappresenta quanto la soluzione è chiara
- ✓ Permette di scegliere il PMLC migliore
- ✗ I requisiti non sono sempre ovvi
- ✗ Fonti diverse e potenzialmente in conflitto
- ✗ Requisiti complessi da esprimere a parole
- ✗ Presenza di tanti requisiti
- ✗ Requisiti interdipendenti che generano conflitti
- ✗ Molti stakeholders interessati/responsabili
- ✗ Cambiamento dovuto alle “business conditions”
- ✗ Dipendenza dei requisiti dal tempo

In un RBS, vanno verificati i seguenti attributi per i requisiti:

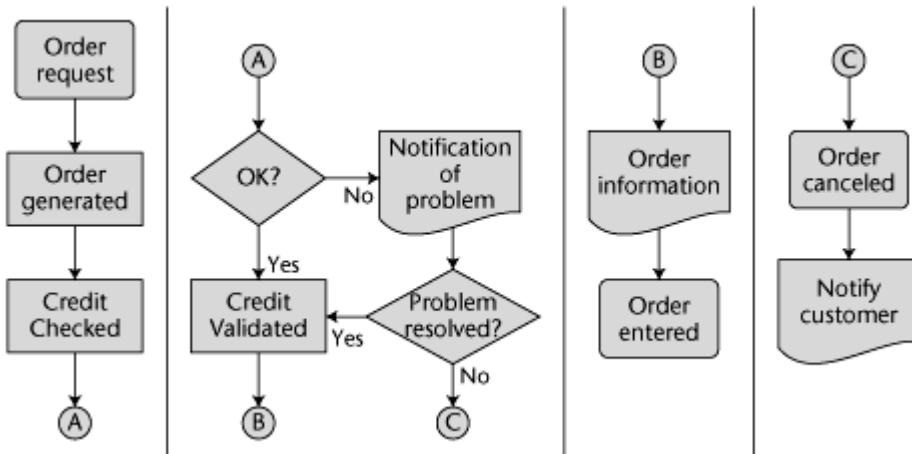
- **Completezza:** è stato dimenticato qualcosa?
- **Chiarezza:** c'è ambiguità?
- **Validità:** vengono riflesse le intenzioni del committente?
- **Misurabilità:** esiste un modo per definire quanto le richieste vengono soddisfatte?
- **Testabilità:** esiste un modo per verificare la soluzione?
- **Manutenibilità:** l'implementazione è facile da capire e manutenere?
- **Affidabilità:** i requisiti di affidabilità e disponibilità possono essere soddisfatti?
- **Look and feel:** i fattori relativi alla percezione della soluzione da parte dell'utente sono stati soddisfatti?
- **Fattibilità:** possono essere implementati?
- **Precedenti:** sono simili ad altri già implementati in passato?
- **Scala:** sono articolati e complessi?
- **Stabilità:** quanto possono cambiare?
- **Performance:** le prestazioni possono essere sempre soddisfacenti?
- **Specifiche:** la documentazione è adeguata?
- **Sicurezza:** i requisiti di sicurezza possono essere dimostrati?

Esistono diverse categorie di requisiti:

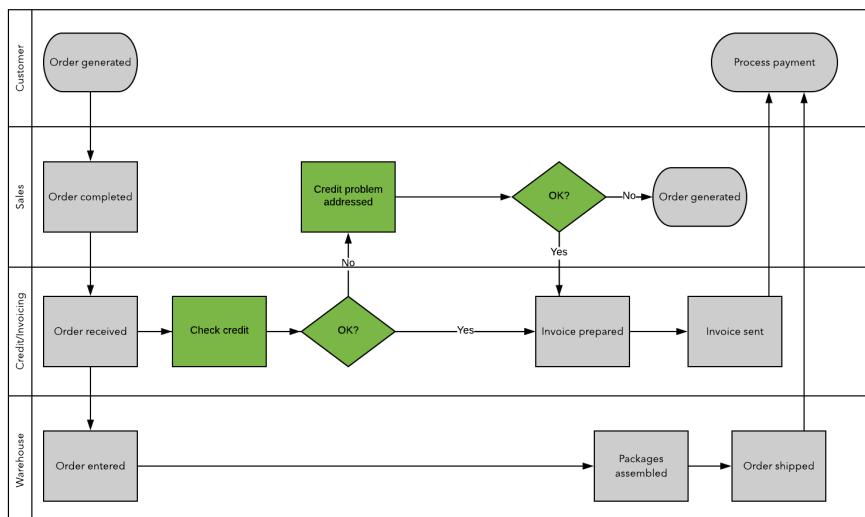
- **Funzionali:** cosa deve fare il prodotto
- **Non funzionali:** quali proprietà devono esserci
- **Globali:** requisiti di alto livello
- **Di prodotto/progetto:** vincoli di progettazione

Diagrammi

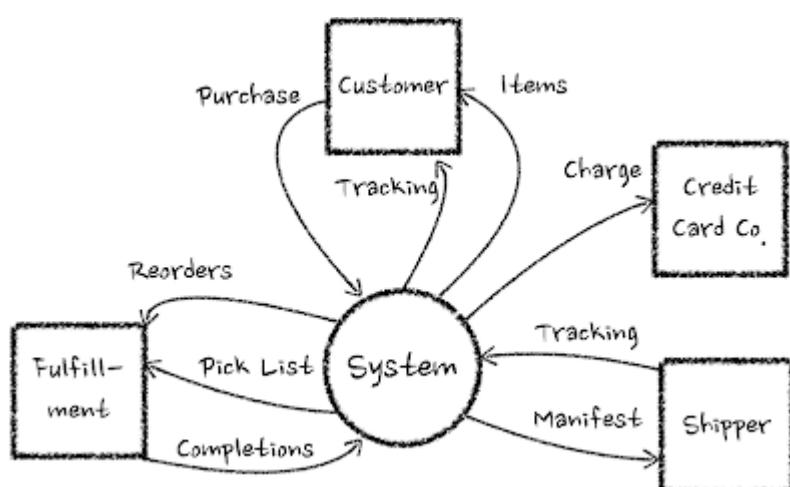
- Process Flow Diagram



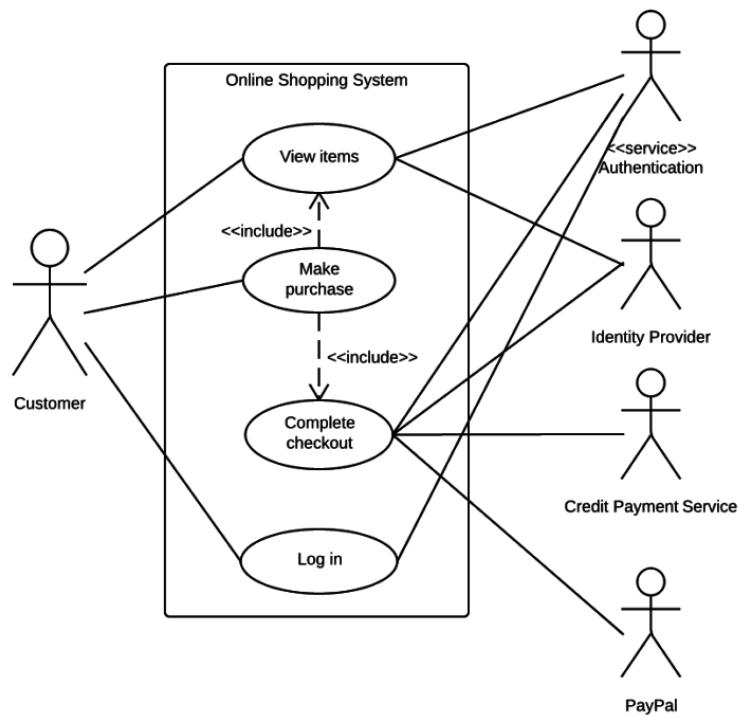
- Swim-Lane



- Context Diagramming Process



- Use-Case



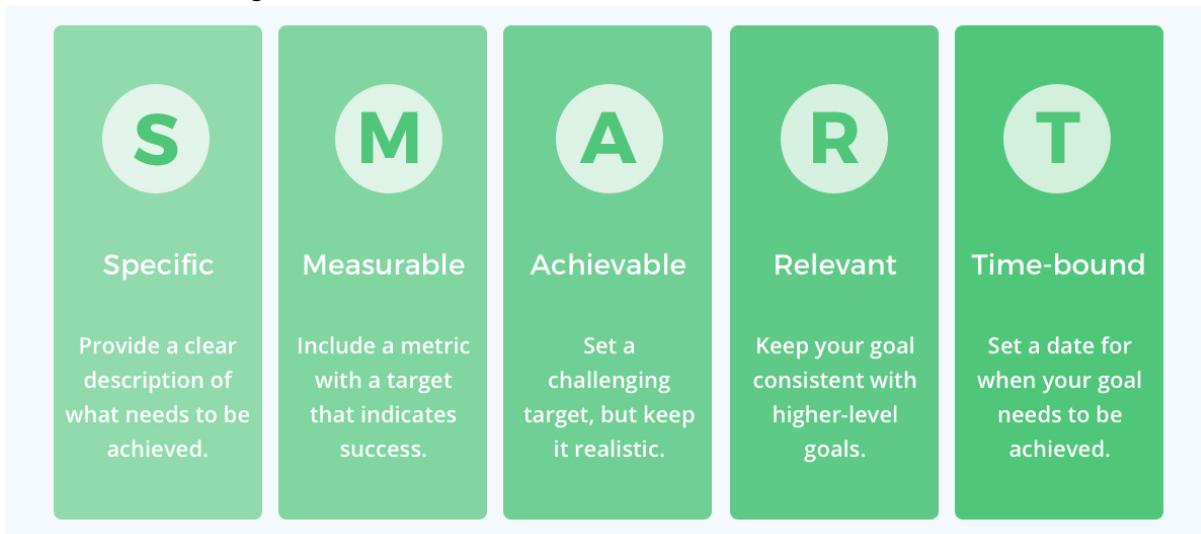
Project Overview Statement (POS)

È una singola pagina di descrizione, rappresentante:

- in cosa consiste il progetto
- un riferimento per il team
- un aiuto per le decisioni
- un documento per ottenere l'approvazione

Le informazioni che devono essere contenute sono:

- Descrizione del **problema / opportunità**
- **Goal** da raggiungere
 - Poche frasi che indicano cosa si vuole ottenere
 - Vengono definiti sfruttando i criteri S.M.A.R.T.:



- **Obiettivi:** le azioni da progettare per raggiungere il goal
- **Criteri di successo**
 - Increased Revenue (IR)
 - Avoided Costs (AC)
 - Improved Services (IS)
- **Assunzioni** (cose che impongo al progetto)
Rischi (solo quelli più importanti)
Ostacoli (ciò che non si ha)
 - Tecnologici
 - Ambientali
 - Interpersonali
 - Culturali
 - Relazioni causali

Ulteriori dettagli possono essere inclusi nel **Project Definition Statements** (PDS), un POS più denso.

Ci possono anche essere degli allegati:

- Analisi dei rischi
- Analisi finanziaria

Planning

Definizione delle attività per fare le cose decise nello scoping e le loro stime del tempo, risorse, costi.

- Definire tutte le attività del progetto, con relative stime di tempi di esecuzione, risorse necessarie e costi.
- Sequenziare le attività.
- Costruire il calendario iniziale.
- Analizzare e correggere il calendario.
- Scrivere il piano di gestione del rischio.
- Documentare il piano di progetto.
- Ottenerne l'approvazione da parte del senior management per eseguire.

È importante pianificare perché, se si fa fatica all'inizio, le cose risulteranno semplici nel procedere del progetto.

Consente di:

- Ridurre l'incertezza
- Aumentare la comprensione
- Migliorare l'efficienza

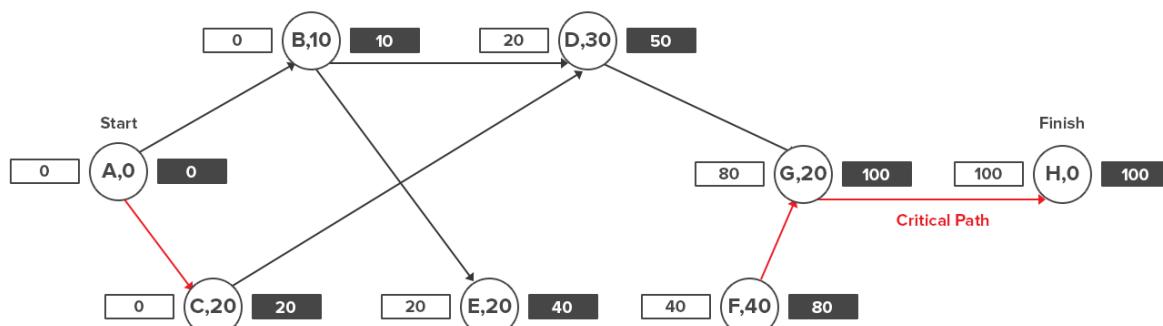
È consigliato usare software di supporto alla pianificazione quando il progetto è complesso e ci sono molti team coinvolti.

Non lo è quando il progetto è semplice o breve, quando la progettazione non porterebbe molto valore aggiunto o quando i requisiti variano troppo frequentemente.

In realtà dipende dal software, in qualche modo del buon software porta comunque qualche beneficio.

I tools + tradizionali sono:

- Post-it
 - Identificazione dell'attività
 - Durata dell'attività
 - Descrizione dell'attività
 - Risorse richieste
 - Responsabile dell'attività
 - Earliest Start, Earliest Finish, Latest Start e Latest Finish.
 - Percorso critico



- Pennarelli
 - Definizione delle dipendenze

- Identificazione percorso critico
- Lavagne
 - POS
 - WBS
 - Diagramma delle dipendenze
 - Initial Project Schedule
 - Final Project Schedule
 - Resource Schedule
 - Issues Log
 - Updated Project Schedule

Dimensione progetto	Durata pianificazione
molto piccoli	< ½ giornata
piccoli	< 1 giorno
medi	2 giorni
grandi	3-4 giorni
molto grandi	???

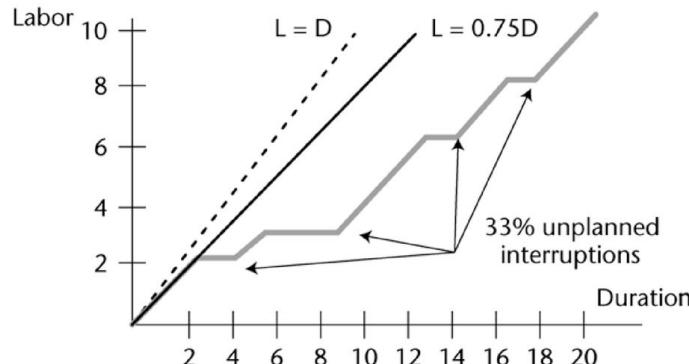
Per progetti non tradizionali (es. Scrum) il planning si ripete ad ogni iterazione.

La pianificazione avviene in sessioni chiamate **Joint Project Planning Sessions (JPPS)**.

Per ogni sessione serve definire:

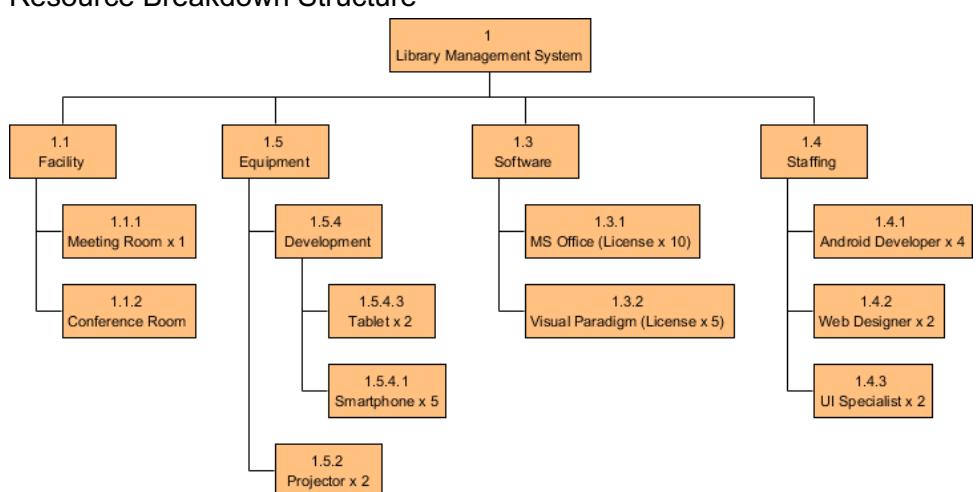
- **Partecipanti**
 - **Facilitatore**: mediatore imparziale tra le parti.
 - **Project Manager**: responsabile del successo del progetto.
 - **un altro Project Manager**: rappresentante del cliente per controllare il PM. Può anche assumere il ruolo di facilitatore nel caso non se ne disponga.
 - **Consulente JPP**: esperto della metodologia di gestione che si vuole usare.
 - **Tecnografo**: esperto dei software da usare per il project management.
 - **Core Team**: coloro che eseguiranno il progetto. Fondamentale includere per la stima dei tempi.
 - **Rappresentanza del cliente**: utile per avere feedback a fronte di cambiamenti.
 - **Manager delle risorse**: coloro che allocano le risorse al progetto.
 - **Manager funzionali**: responsabili di aree del progetto
 - **Project owner**: detentore dei processi da supportare con la soluzione.
 - **Project champion**: non coinvolto direttamente, ma interpellabile in caso di necessità. Entità al di sopra delle parti che viene ascoltata da tutti.
- **Strutture** (sale riunioni, ...)
 - Comode, confortevoli
 - Lontane da possibili interruzioni
 - breakout rooms: ci si chiude dentro e ci si rimane fino al raggiungimento dell'obiettivo.
- **Equipaggiamento**

- **Agenda:** cose da fare nella riunione
 - I partecipanti alle varie riunioni cambiano in base ad essa.
Solo gli interessati agli argomenti in agenda parteciperanno.
- **Deliverables:** quali devono essere i risultati della riunione
 - Project Definition Statement (PDS)
 - Versione estesa del POS, con le stesse sezioni ma + dettagliate.
 - Destinato al team.
 - WBS
 - Stima della durata delle attività
 - **Durata** (elapsed time) vs **quantità di lavoro** (working time)

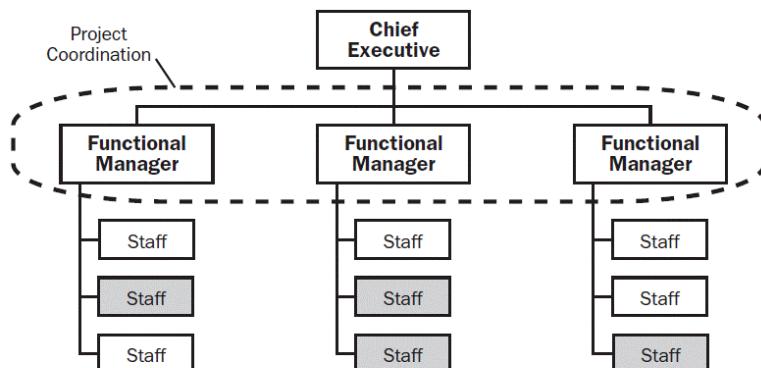


- Si deve stimare la durata per determinare la schedula.
- Si deve stimare la quantità di lavoro per valutare i costi ed assegnare risorse.
- Risorse vs durata
- La durata di un task dipende da:
 - esperienza e competenza
 - eventi inattesi
 - uso efficiente del tempo
 - errori
- Si possono fare stime usando 6 metodi:
 - Estrapolazione da attività simili
 - Analisi dati storici
 - Giudizio esperti
 - Uso di tecniche basate sul consenso
 - Delphi
 - Un gruppo di esperti fa stime in più round
 - Ad ogni round ognuno fa la sua stima anonima, dando anche la motivazione.
 - Ascoltando le stime degli altri e le loro motivazioni, si rivota.
 - Ci si ferma dopo un certo numero di round o quando il gap tra le stime si riduce oltre una soglia.
 - Three-point
 - Si prendono 3 stime, la + ottimistica, la + pessimistica e la media.
 - La stima usata è la media pesata $\frac{O + 4M + P}{6}$

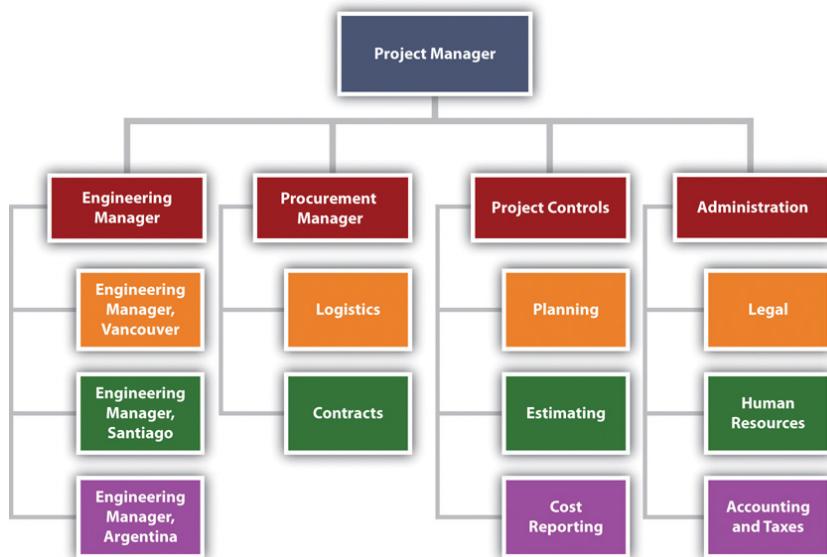
- Delphi a banda larga
- Anche le stime hanno un ciclo di vita, in quanto col passare del tempo si diventa più bravi nel farle.
- Risorse necessarie
 - Resource Breakdown Structure



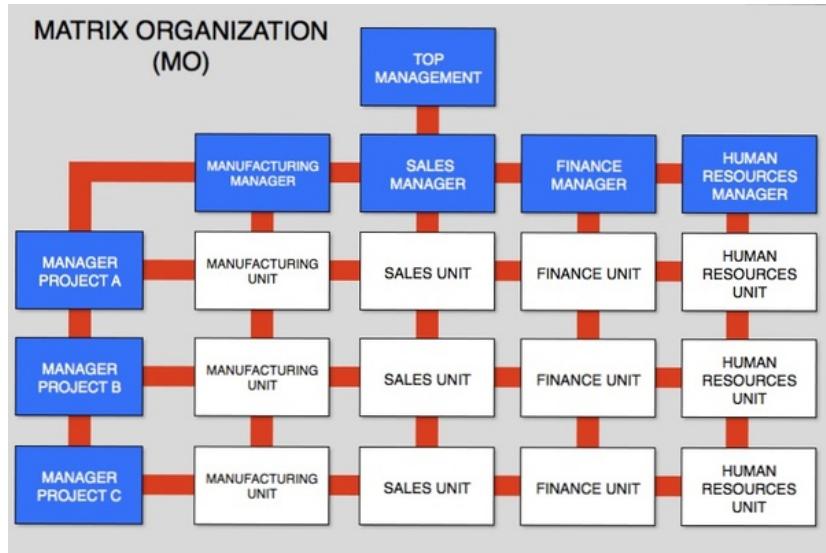
■ Organizzazione funzionale



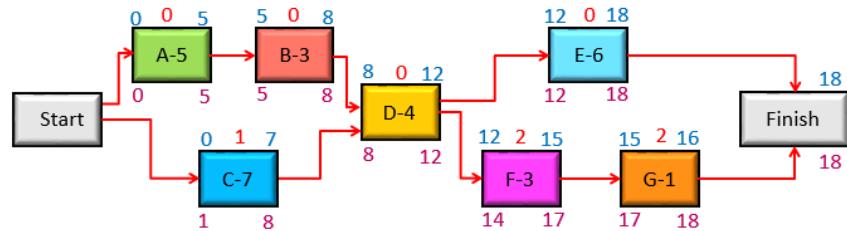
■ Organizzazione di progetto



- Organizzazione a matrice



- Project Network schedule (**PERT/CPM**)
 - Si deve arrivare a stabilire il Gantt
 - Project Network Diagram (2 alternative)
 - Le attività sono gli archi
 - Le attività sono i nodi ⇒ gli archi sono le dipendenze tra attività



- Dipendenze di 4 tipi:
 - Start to start (SS): le attività possono partire assieme.
 - Start to finish (SF): un'attività non può finire se quelle precedenti non sono iniziate.
 - Finish to start (FS): Dipendenza classica. un'attività non può iniziare se quelle precedenti non sono finite.
 - Finish to finish (FF): un'attività non può finire se quelle precedenti non sono finite.
 - Possibile inserire dei lag per ritardare la dipendenza.
- Si parte dalla data di partenza e si cerca di costruire la schedula, tenendo conto di eventuali vincoli tra le attività, scoprendo la data di fine al più presto.
- Tornando indietro dalla fine si scopre qual è la data di partenza al più tardi.
- Schedulazione delle attività
- Assegnamento delle risorse
- Project Notebook

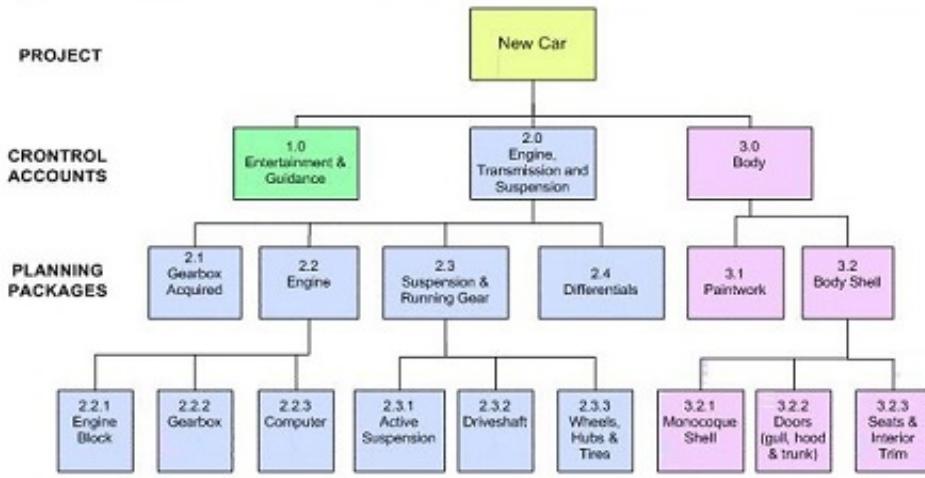
- Proposta di progetto
- **Proposta di progetto**
 - Background
 - Obiettivi
 - Approccio da usare
 - Definizione dettagliata del lavoro da fare
 - Tempi e costi
 - Appendici

Il committente è utile in un JPPS per:

- Validare il POS
- Prioritizzare i requisiti
 - Approccio **MoSCoW**
 - **Must**: requisiti da soddisfare senza meno (indispensabili)
 - **Should**: requisiti che dovrebbero essere inclusi (fortemente consigliati)
 - **Could**: requisiti che potrebbero essere inclusi (consigliati)
 - **Would**: requisiti che sarebbero da includere (non indispensabili)
- Generare e validare la WBS
- Accordarsi sulla pianificazione
- Avere il possesso del piano (così non può lamentarsi se lo si rispetta!)
- **Coinvolgerlo**

Work Breakdown Structure (WBS)

Descrizione gerarchica del lavoro da fare nel progetto per soddisfare la RBS.



Si può rappresentare anche come elenco numerato.

4 impieghi fondamentali:

- Processi mentali: la WBS aiuta a visualizzare come il lavoro deve essere definito e gestito.
- Design architetturale
- Pianificazione
- Reporting

Data una WBS, un **task** è l'attività di massimo dettaglio (le foglie dell'albero) per la quale vanno definiti tempi, costi e risorse necessarie.

La **WBS può essere generata dalla RBS** secondo 2 approcci:

- **Team:** tutto il team definisce l'intera WBS
- **Subteam:** il team viene suddiviso in gruppi ed ognuno definisce una parte della WBS partendo da certi requisiti. L'integrazione viene fatta coinvolgendo tutto il team.

La WBS si costruisce usando lavagne, pennarelli e post-it

Il POS deve essere tenuto ben presente, mantenendo l'integrità concettuale con esso.

Se si rilevano incongruenze o lacune è necessario affrontare subito il problema in quanto l'impatto di modifiche allo scope o all'RBS durante la pianificazione è minore rispetto a quanto lo saranno durante l'esecuzione.

Una WBS è **completa** se:

- Stato e completamento sono misurabili.
- Inizio e fine delle attività sono ben definiti.
- Le attività sono associate ad un deliverable.
- Tempi e costi sono facilmente stimabili.
- La durata delle attività è limitata (attività troppo lunghe sono complesse da controllare).
- Un task non richiede interruzioni e deve essere indipendente da altri durante la sua esecuzione.

Si può stabilire che la WBS è incompleta, ma che al momento non è possibile avere nulla di meglio.

È possibile fare eccezioni e considerare una WBS completa anche se non soddisfa tutti i criteri, l'importante è essere consapevoli dell'eccezione fatta.

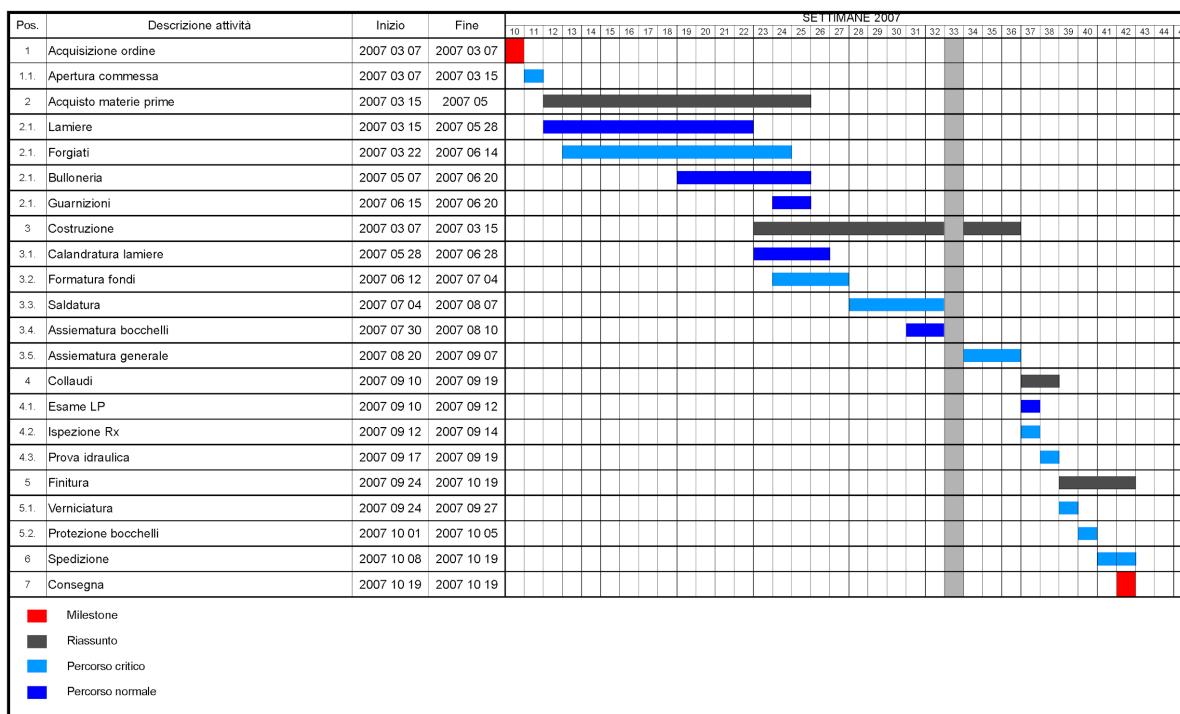
Una WBS può essere costruita secondo diversi approcci:

- Nominali
 - componenti fisici
 - componenti funzionali
- Verbali
 - Design-Build-Test-Implement
 - Obiettivi
- Organizzativi
 - Geografico
 - Dipartimentale
 - Business process

Nella costruzione della WBS si usano templates per risparmiare tempo.

- I templates sono costruiti sulla base delle esperienze passate.
- Permettono di standardizzare la pianificazione.
- Molto efficaci per gestire progetti ricorrenti.

Gantt



Lancio / Esecuzione

Definizione di chi farà parte del team e delle regole operative.

- Reclutare i membri del team

- Scrivere il **Project Description Document** (può essere una collezione di documenti, molto probabilmente derivanti dalla fase di planning)
- Stabilire le **regole operative**
 - Soluzione dei problemi
 - Presa delle decisioni ⇒ [RASCI](#)
 - Risoluzione dei conflitti
 - Costruzione del consenso
 - Non è detto che anche se si raggiunge una soluzione condivisa, essa sia corretta.
 - Brainstorming
 - È bene invitare solo i conoscenti del dominio
 - Enunciare e discutere tutte le idee
 - Riunioni
 - Da fare prima
 - Chiedersi se la riunione serve
 - Determinare lo scopo
 - Stabilire i partecipanti
 - Preparare note e presentazioni
 - Da fare durante
 - Raccogliere informazioni dai partecipanti
 - Riassumere i risultati raggiunti periodicamente
 - Da fare dopo
 - Fissaggio prossima riunione e relativi partecipanti
 - Assegnazione task per la prossima riunione
 - **Daily status meeting:** riunioni giornaliere per mantenere monitorato lo stato di avanzamento.
 - **Problem resolution meeting:** riunioni per risolvere problemi.
 - **Project review meeting:** eventi formali al raggiungimento di una milestone.
- Gestire le comunicazioni del team
 - timing
 - contenuti
- Stabilire come gestire il cambiamento di scope
- Finalizzare il calendario
- Scrivere i componenti in cui è decomposto il lavoro (work packages)

Team di progetto

- Core Team
 - Caratteristiche:
 - Impegno
 - Condivisione responsabilità
 - Flessibilità
 - Task-oriented
 - Abilità a lavorare con vincoli
 - Fiducia e supporto reciproco
 - Team-oriented
 - Apertura mentale
 - Rispetto delle gerarchie
 - Utilizzo di strumenti di Project Management
- Team Cliente
 - Devono comprendere i processi
 - Devono essere in grado di prendere decisioni
- Team a contratto
 - Potrebbero non essere sempre disponibili
 - Dovrebbero conoscere le relazioni tra task
 - Potenziale scarsa qualità

Bilanciamento del team, 4 tipologie di persone:

- **Assimilatori**
 - Abili a raccogliere dati ed a rappresentarli.
 - Il focus è sui concetti, non sono orientati ai risultati.
- **Divergenti**
 - Abili ad individuare alternative, fornendo punti di vista out-of-the-box.
 - Tendono ad osservare piuttosto che agire.
- **Accomodanti**
 - Orientati ai risultati, adattabili alle circostanze.
 - Si relazionano con gli altri per decidere il da farsi, poche analisi tecniche!
- **Convergenti**
 - Orientati ai risultati, preferiscono individuare soluzioni che implementarle.
 - Non sono collaborativi.
 - Date diverse alternative, sono abili ad individuare la migliore.

I fattori che rendono un team efficace sono:

- Qualità
- Flessibilità
- Coordinamento
- Soddisfazione dei membri
- Crescita professionale
- Possesso delle abilità necessarie
- Produttività

Il team può migliorare in:

- Metodi e procedure di lavoro
- Conoscenza delle tecnologie

- Uso di servizi e prodotti di qualità
- Presa delle decisioni
- Supervisione
- Supporto allo staff
- Attrarre e fidelizzare
- Tasso di uscita
- Flessibilità del personale

Ogni membro del team ha le seguenti responsabilità:

- Comunicazione
- Ascolto
- Condivisione
- Atteggiamento positivo
- Creatività
- Rispetto per gli altri
- Crescita

Kick-off

Il meeting di kick-off è il primo vero incontro del team che svilupperà il progetto.

Include:

- Presentazione dello sponsor
- Presentazione degli aspetti rilevanti
- Presentazione del progetto
- Presentazione dei membri del team
- Definizione PDS
- Stabilimento delle [regole operative](#)
- Integrazione nella schedula le disponibilità dei membri del team
- Scrivere i Work Packages

Assegnazione delle responsabilità (RASCI)

ATTIVITA'	DIRETTORE	CIO	FRONT OFFICE	PRODUZIONE	AMMINISTRAZIONE	COMMERCIALE
POLITICA AZIENDALE	A/R	C	I	I		R
RAPPORTI CLIENTI	C	I	C			A/R
ORDINI CLIENTI	I	I	I	C	R	A
ORDINI FORNITORI	I	I		A	R	C
BILANCIO	A				R	C

R	Responsible	The People (or person) who are expected to actively participate in the activity and contribute to the best of their Position working on the activity "The doer"
A	Accountable	The person who is ultimately accountable for the results Position with Yes/No authority "The buck stops here"
S	Support	People to support the responsible person (help on the work) NOT used all the time Position that is helping out at the direction of Responsible "Helper"
C	Consult	People who either have a particular expertise they can contribute to specific decisions (i.e., their advice will be used) Position involved prior to decision or action "In the Loop"
I	Inform	People who are affected by the activity/decision and therefore need to be kept informed, but do not participate in the decision or action Position that needs to know of the decision or action "Tell me after" "Notify me"

Cambiamento di scope

Se ci si accorge che è necessario cambiare lo scope, allora, a fronte di richieste si valuta se rigettare o accogliere.

Esistono templates per scrivere le richieste di cambiamento di scope: Scope Change Request Form.

Data una richiesta, va identificato l'impatto. Un modo per farlo è preparare il **Project Impact Statement**.

- Che benefici porta?
- Quanta differenza di costo?
- Come cambierebbe la schedula?
- Come cambia la qualità?
- Come andrebbero riallocate le risorse?
- Il cambiamento può essere rimandato?
- Come cambia il rischio del progetto?

Monitoraggio & controllo

Verifica di eventuali problematiche e risoluzione di esse.

- Stabilire le performance del progetto e il sistema di reporting (informazioni che servono per capire a che punto si è, KPI)
- Monitorare le performance
- Monitorare il rischio
- Rendicontare le stato del progetto
- Elaborare le richieste di modifica allo scope
- Scoprire e risolvere problemi

Sistema di reporting

Per effettuare in modo efficiente il reporting sullo stato di avanzamento del progetto è consigliabile dotarsi di un sistema di supporto intuitivo e che non introduca complessità di utilizzo.

Esistono 5 tipologie di report sullo stato del progetto:

- Current period reports: coprono i periodi più recenti.
- Cumulative reports: coprono l'intera storia del progetto.
- Exception reports: si concentrano sulle cause degli scostamenti rispetto al piano e sulle loro soluzioni.
- Stoplight reports: segnalazioni sintetiche dello stato di avanzamento.
- Variance reports: si concentrano sugli scostamenti rispetto al piano.

Analisi del valore guadagnato (EVA)

L'analisi relativa al valore guadagnato (Earned Value) è usata per misurare le performance del progetto.

Il lavoro effettivamente svolto è confrontato con il lavoro previsto dal piano, per determinare gli scostamenti rispetto alla schedula e ai costi previsti.

I valori usati sono:

- **Planned Value (PV)**: il budget approvato per un'attività del WBS.
- **Earned Value (EV)**: il valore del lavoro ad oggi completato in funzione del suo budget.
- **Actual Cost (AC)**: il costo sostenuto per completare un'attività del WBS.

Schedule Performance Index (SPI)

Fornisce una misura di come il progetto sta performando rispetto a quanto pianificato:

$$SPI = \frac{EV}{PV}$$

Cost Performance Index (CPI)

Fornisce una misura di come i costi del lavoro svolto evolvono rispetto ai costi di progetto stimati in fase di pianificazione:

$$CPI = \frac{EV}{AC}$$

Scope Bank

Si predispone un deposito iniziale di una certa percentuale del tempo complessivamente necessario.

Il tempo necessario per processare e integrare una richiesta di modifica dello scope è attinto dalla Scope Bank.

Se risparmiamo del tempo nell'esecuzione di qualche attività, lo possiamo aggiungere alla Scope Bank.

Issues Log

Documento che contiene tutti i problemi che sono emersi durante il progetto e che ancora non sono stati risolti.

Le informazioni contenute possono essere le seguenti:

- ID
- Data di scrittura
- Descrizione del problema
- Descrizione dell'impatto sul progetto se non sarà risolto
- Definizione del proprietario del problema
- Azione che deve essere intrapresa per risolvere il problema
- Stato
- Esito

Project Status Meeting

Meeting che devono essere effettuati con cadenza periodica per poter monitorare e controllare lo stato di avanzamento del progetto.

Daily Status Meeting

Partecipano responsabili dei task in lavorazione. Solitamente sono riunioni brevi in cui tutti stanno in piedi. Un facilitatore potrebbe essere molto utile e può essere interpretato da uno dei partecipanti.

Deve essere riportato sinteticamente lo stato di ciascun task:

- in schedula
- in anticipo rispetto alla schedula
- in ritardo rispetto alla schedula

Si devono anche aggiornare Scope Bank e Issues Log.

Problem Management Meeting

Devono partecipare solo i membri del team coinvolti.

- Concordare la definizione del problema.
- Concordare chi è il proprietario del problema.
- Brainstorming per la soluzione del problema.
- Assegnare una priorità alle soluzioni individuate.
- Aggiornare l'Issues Log.

Problem Escalation Strategy

Nella definizione della strategia di risoluzione dei problemi si segue una gerarchia ben precisa:

1. Non sono richieste azioni
2. Esaminare le dipendenze per identificare delle opportunità di compressione della schedula.
3. Riassegnare delle risorse per contrastare il ritardo.
4. Negoziare risorse aggiuntive.
5. Negoziare una strategia che preveda dei rilasci multipli.
6. Richiedere al cliente un'estensione della schedula.
7. Richiedere al cliente una modifica di scope.

Chiusura

Per ottenere una chiusura di progetto ordinata è necessario seguire i seguenti passi:

- Ottenere l'approvazione del cliente
- Assicurarsi che tutto il necessario sia installato
 - In fasi: deliverables consegnati in sequenza.
 - Per unità di business: deliverables installati uno alla volta.
 - Cut-over: sostituzione immediata del vecchio prodotto con in nuovo
 - Parallello: In nuovo deliverable viene installato mantenendo anche il vecchio.
- Assicurarsi che tutta la documentazione sia disponibile
 - [POS](#)
 - [RBS](#)
 - [WBS](#)
 - [Gantt](#)
 - Design
 - Prototipi
 - Rapporto finale
 - Sommario esecutivo
 - Livello di successo
 - Organizzazione/Amministrazione del progetto
 - Tecniche impiegate (con pregi e difetti)
 - Raccomandazioni
 - ...
- **Festeggiare!**