

UNIVERSITÀ POLITECNICA DELLE MARCHE
FACOLTÀ DI INGEGNERIA
Dipartimento di Ingegneria dell'Informazione
Corso di Laurea Magistrale in Ingegneria Informatica e dell'Automazione



TESI DI LAUREA

**Governance di un progetto IT: attività di supporto per la costruzione
e il mantenimento di una Data Platform**

**IT project governance: support activities for implementation and
maintenance of a Data Platform**

Relatore

Prof. Domenico Ursino

Correlatore

Dott. Paolo Lo Giudice

Candidata

Lisa Burini

ANNO ACCADEMICO 2022-2023

*Non è perché le cose sono difficili che non osiamo,
ma è perché non osiamo che sono difficili.*

Lucio Anneo Seneca

Sommario

Nell'ambito dei progetti IT moderni, caratterizzati da una crescente complessità, l'adozione di tecniche di Project Management diventa essenziale per guidare e coordinare efficacemente tutte le fasi del progetto, assicurando l'allineamento con gli obiettivi strategici dell'organizzazione. Questa tesi si colloca nel contesto dello sviluppo di una Data Platform per un prestigioso cliente multinazionale nel settore automotive, con l'obiettivo di porre l'accento sull'importanza di una governance progettuale mirata e strutturata, finalizzata al raggiungimento di standard qualitativi elevati. Nel dettaglio, la tesi mette in luce le attività intraprese per la raccolta e l'organizzazione della documentazione tecnica, processo cruciale per poter far assumere al cliente la gestione e la manutenzione autonoma del sistema nella fase post-sviluppo. Ci si focalizza, poi, sulle attività di supporto operativo al team di Application Management Service, per cui si definisce un processo strutturato per il deploy delle applicazioni, si monitorano le performance della piattaforma attraverso dashboard e analisi dettagliate, e ci si concentra sul tracciamento e la risoluzione definitiva dei problemi attraverso la root cause analysis.

Keyword: Project Management, Project Manager, Governance, Information Technology, Digital Transformation, Automotive Industry, Data Platform

Introduzione	1
1 Introduzione al Project Management	3
1.1 Definizione di Project Management	3
1.2 Storia del Project Management	3
1.3 Ruoli e Responsabilità nel Project Management	5
1.3.1 Il Project Manager	5
1.3.2 Il team di progetto	5
1.3.3 Lo sponsor di progetto	6
1.3.4 Gli stakeholder	7
1.4 Ciclo di vita del progetto	8
1.5 Metodologie di Project Management	10
1.6 Strumenti e tecniche di Project Management	12
1.7 Importanza del Project Management	18
2 Contesto e ambito della tesi: governance di un progetto IT nel settore Automotive	19
2.1 Governance di progetti IT	19
2.2 Il ruolo della Digital Transformation nel settore automotive	21
2.3 Introduzione all'Enterprise Data Platform	23
2.4 Storia ed evoluzione del progetto	25
2.5 Requisiti tecnici di progetto	26
2.6 Attori in gioco e criticità di gestione	27
3 Documentazione di progetto e gestione delle attività correlate	29
3.1 Il ruolo della documentazione	29
3.1.1 Tipologie di documentazione	30
3.1.2 Best practice	32
3.1.3 Benefici della documentazione	32
3.1.4 Sfide della creazione e gestione della documentazione	33
3.2 Attività svolte per la raccolta e la gestione della documentazione	34
3.2.1 Censimento e organizzazione della documentazione	34
3.2.2 Definizione di un template	35
3.2.3 Pianificazione attività	37
3.2.4 Controllo e monitoraggio dello stato avanzamento	38
3.2.5 Creazione di una pagina web per la governance di progetto	39

4	Supporto operativo al team dedicato ai servizi di gestione delle applicazioni	40
4.1	Introduzione all'Application Management Service (AMS)	40
4.2	Strutturazione del processo di deploy front-end	41
4.3	Monitoraggio dei ticket	43
4.4	Tracciamento degli issue e analisi delle root cause	46
5	Controllo e monitoraggio delle attività	50
5.1	Fase di controllo e monitoraggio nel Project Management	50
5.2	Tracking delle attività	51
5.3	Lesson learned	53
	Conclusioni	57
	Bibliografia	58
	Ringraziamenti	62

Elenco delle figure

1.1	Funzioni e responsabilità del project manager (Bezak e Nahod [2011])	6
1.2	Rete di stakeholder (Larson e Gray [2010])	7
1.3	Cicli di vita del progetto (PMI [2021])	9
1.4	Fasi del ciclo di vita di un progetto (Larson e Gray [2010])	10
1.5	Approccio waterfall per lo sviluppo del software (Larson e Gray [2010]).	11
1.6	Approccio agile (Larson e Gray [2010])	12
1.7	Esempio di Work Breakdown Structure (Alutbi [2020])	13
1.8	Esempio di diagramma di Gantt con applicazione del CCPM (Larson e Gray [2010])	14
1.9	Baseline dei costi (PMI [2021])	15
1.10	Diagramma di Ishikawa (causa-effetto) (Coccia [2017])	15
1.11	Esempio di matrice RACI (Suhanda e Pratami [2021])	16
1.12	Matrice probabilità-impatto (Dumbravă e Iacob [2013])	17
1.13	Analisi SWOT	17
2.1	Ciclo di vita di un progetto generico (Marchewka [2016])	20
2.2	Ciclo di vita di un progetto IT (SDLC) (Marchewka [2016])	21
2.3	Potenziale impatto economico della Digital Transformation (Siebel [2019])	22
2.4	Architettura di alto livello dell'Enterprise Data Platform	25
2.5	Evoluzione del progetto	26
2.6	Architettura EDP di alto livello con specifica dei flussi outbound	26
2.7	Architettura EDP di dettaglio	27
3.1	Estratto di censimento della documentazione	35
3.2	Template dell'analisi tecnica: indice	36
3.3	Template dell'analisi tecnica: gestione del versioning	37
3.4	Pianificazione delle attività di raccolta e revisione della documentazione	38
3.5	Creazione di un tracker per il tracciamento degli open point	39
3.6	Pagina web per la governance di progetto	39
4.1	Template per le richieste di deploy front-end	42
4.2	Inserimento nel tracker di una nuova pubblicazione front-end	43
4.3	Tracker per i deploy front-end	44
4.4	Vincoli per le richieste di deploy	44
4.5	Dashboard Qlik per il monitoraggio dei ticket	45

4.6	Report settimanale per l'analisi dei principali KPI	46
4.7	Issue tracker	48
4.8	Root Cause Analysis	49
5.1	Kanban Board per il tracciamento delle attività ad alto livello	52
5.2	Report di allineamento	54
5.3	Tracker di attività in dettaglio	55
5.4	Lesson learned	56

Questo progetto nasce dalla necessità di governare un progetto IT consistente nell'implementazione e manutenzione di una Data Platform, cioè di un'infrastruttura tecnologica integrata che consente di raccogliere, gestire, conservare, elaborare ed analizzare grandi volumi di dati provenienti da fonti differenti. I moderni progetti IT non sono sostenibili senza figure professionali dedicate alla governance di progetto, ovvero i project manager IT. Questi permettono di attuare tecniche di Project Management con l'obiettivo di trarre i requisiti di progetto. Infatti, il Project Management influisce sul successo di un progetto garantendo una pianificazione e un'allocazione efficiente delle risorse, attenuando i rischi, migliorando la comunicazione e la collaborazione tra i team, mantenendo il controllo dell'ambito, dei tempi e del budget del progetto e assicurando la qualità dei risultati. Applicando tecniche di Project Management si promuove un approccio strutturato che si allinea con gli obiettivi organizzativi e delinea un approccio per raggiungere costantemente gli obiettivi contribuendo, così, in modo significativo al successo complessivo del progetto.

La governance di una Data Platform è un elemento cruciale che assicura l'efficacia, la sicurezza e la sostenibilità della piattaforma stessa. In un'epoca caratterizzata da enormi volumi di dati e regolamenti stringenti, la governance diventa fondamentale per gestire i dati in modo che siano un asset piuttosto che un onere; con una corretta gestione ci si assicura che la Data Platform non solo funzioni efficacemente, ma si allinei anche agli obiettivi aziendali e alle normative vigenti. Inoltre, una governance adeguata è fondamentale per stabilire chiari confini per il progetto, aiutando a definire le priorità e a gestire le aspettative degli stakeholder.

La governance fornisce un quadro per la gestione efficiente delle risorse, sia umane che tecniche, assicurando che siano allocate in modo da massimizzare l'efficienza e l'efficacia del progetto; essa consente l'identificazione proattiva dei rischi e la pianificazione delle mitigazioni adeguate, riducendo la probabilità e l'impatto degli eventi negativi sul progetto. Con un processo strutturato di governance, si possono affrontare le issue che emergono, garantendo che siano risolte in modo tempestivo e non compromettano la consegna del progetto.

Risultano fondamentali il controllo e il monitoraggio delle attività per ottenere standard elevati in tutte le fasi del progetto, assicurando che il prodotto finale soddisfi o superi le aspettative. Grazie alle tecniche di Project Management, si stabiliscono canali e piani di comunicazione per garantire che tutte le parti interessate siano informate, coinvolte e in grado di contribuire in modo significativo al successo del progetto. Inoltre, si gestiscono tutti gli stakeholder identificando e settando le loro aspettative, assicurando che i loro interessi siano rappresentati e che qualsiasi conflitto venga risolto in modo costruttivo.

Non meno importante, il fatto che una corretta gestione incoraggi la revisione e l'apprendimento continuo dal progetto, consentendo miglioramenti iterativi e l'adattamento delle pratiche di governance in base alle lezioni apprese. Dunque, grazie ad una governance robusta, il progetto può navigare con successo attraverso le sfide, adattarsi ai cambiamenti e realizzare i benefici previsti per l'organizzazione.

Declinando le attività di Project Management nel presente lavoro di tesi, si sono svolte attività incentrate sulla gestione della raccolta della documentazione di progetto ed attività a supporto del team di AMS (Application Management Service) con il ruolo di Project Manager.

Il lavoro di tesi si inserisce nel progetto di Data Platform nei mesi conclusivi di sviluppo, per cui, in questa fase cruciale, è stata ritenuta fondamentale l'attività di raccolta e organizzazione della documentazione tecnica. Ci si è focalizzati sul fornire al cliente tutte le risorse, le conoscenze e gli strumenti indispensabili per poter gestire e mantenere in modo efficace e autonomo il sistema implementato, anche successivamente al distacco dal progetto del team di sviluppo afferente all'azienda di system integrator.

Possedere una documentazione chiara, completa e facilmente accessibile è un fattore necessario per garantire una transizione fluida per la gestione e il mantenimento del sistema e, quindi, per mantenere un'efficacia operativa.

Per quanto riguarda il supporto operativo al team di AMS, si è stati coinvolti nella definizione di un processo strutturato per il deploy delle applicazioni front-end, fondamentale per coordinare tutti gli sviluppatori, sia interni che appartenenti a fornitori esterni. Per il monitoraggio e la valutazione delle performance della Data Platform ci si è concentrati sul monitoraggio dei ticket ricevuti dal team di AMS; grazie alla creazione di dashboard opportune si è potuto monitorare lo stato della piattaforma. Infine, risultano fondamentali il tracking degli issue e l'analisi delle root cause per identificare i problemi della piattaforma ed indagare le relative cause.

Per coordinare tutte le attività seguite è stato necessario monitorarle con due differenti livelli di astrazione; in particolare, si è utilizzato un approccio ad alto livello per avere una panoramica chiara degli stream in corso ed un approccio più di dettaglio per ogni singola area seguita per tracciare le micro attività relative. Infine, si è prodotto un registro delle lesson learned durante tutto il lavoro così da ottenere benefici per il progetto fin dall'immediato e poter partire con una marcia in più in eventuali progetti futuri.

La presente tesi è composta da cinque capitoli strutturati come di seguito specificato:

- Nel Capitolo 1 si fornisce una panoramica sul Project Management, partendo dalla definizione e dalla storia, fino alla declinazione dei ruoli e delle responsabilità. Si approfondisce, poi, il ciclo di vita di un progetto; infine, si illustrano le varie metodologie di Project Management e i relativi strumenti.
- Nel Capitolo 2 si introduce il contesto del lavoro di tesi, quindi si illustra il ruolo della Digital Transformation nel settore dell'automotive per poi spiegare nel dettaglio in cosa consiste la Data Platform e gli attori in gioco nel progetto.
- Nel Capitolo 3 si approfondisce l'importanza di una documentazione esaustiva in un progetto IT e, quindi, saranno presentate tutte le attività svolte per la raccolta e la gestione della documentazione di progetto.
- Nel Capitolo 4 si dettagliano tutte le attività svolte a supporto del team di AMS, ovvero la strutturazione del processo di deploy, il monitoraggio dei ticket e il tracking degli issue, con la relativa analisi delle root cause.
- Nel Capitolo 5 si mostrerà una panoramica degli strumenti utilizzati per il controllo ed il monitoraggio delle attività, e saranno presentate le lesson learned raccolte.

Introduzione al Project Management

Il Project Management è la disciplina che consiste nell'avviare, pianificare, eseguire, controllare e concludere il lavoro di un team per raggiungere gli obiettivi definiti e soddisfare criteri di successo specifici. In questo capitolo si introduce tale tematica, approfondendo innanzitutto l'evoluzione storica della disciplina, dai tempi antichi fino al giorno d'oggi. Dopo una breve introduzione di contesto, si analizzano nel dettaglio gli attori principali coinvolti nella gestione di un progetto con le relative responsabilità. In particolare, si approfondiscono i ruoli del project manager, del team di progetto, dello sponsor e degli stakeholder. Successivamente si propone una panoramica dei cicli di vita dei progetti, che possono declinarsi in predittivi, adattivi o ibridi. Vengono, quindi, descritte le metodologie di Project Management, distinguibili principalmente in waterfall e agile. In conclusione, si descrivono le aree di conoscenza riconosciute dal Project Management Institute, e si mettono in evidenza, per ogni area, gli strumenti principali utili al conseguimento degli obiettivi di gestione.

1.1 Definizione di Project Management

Il Project Management è la disciplina che consiste nell'avviare, pianificare, eseguire, controllare e concludere il lavoro di un team per raggiungere gli obiettivi definiti e soddisfare criteri di successo specifici. Esso consiste nell'applicazione di conoscenze, competenze, strumenti e tecniche alle attività di progetto per soddisfare i requisiti dello stesso. Un progetto è un'impresa unica e transitoria intrapresa per raggiungere gli obiettivi pianificati, che possono essere definiti in termini di output, risultati o benefici. Un progetto è, solitamente, considerato un successo se raggiunge gli obiettivi secondo i criteri di accettazione, entro i tempi e il budget concordati (PMI [2013]).

1.2 Storia del Project Management

La storia del Project Management può essere tracciata fin dai tempi antichi, anche se la disciplina come la conosciamo oggi si è sviluppata più chiaramente nel corso del XX secolo. Le fondamenta di una cultura del Project Management si sono sviluppate fin da tempi remoti. Ad esempio, la costruzione delle piramidi in Egitto intorno al 2500 a.C. mostra che esistevano figure responsabili per la gestione di progetti di grande scala, ma anche opere come il Colosseo e i grandi acquedotti romani riflettono principi di Project Management (Kozak-Holland [2011], Seymour e Hussein [2014]).

Il vero sviluppo del Project Management come disciplina distinta inizia nel XIX secolo, con la crescente complessità del mondo degli affari e la necessità di gestire progetti su larga scala, come la costruzione della ferrovia transcontinentale negli Stati Uniti, la cui costruzione

iniziò nel 1860. Improvvisamente, i leader aziendali si trovarono di fronte all'arduo compito di organizzare il lavoro manuale di migliaia di operai e la lavorazione e l'assemblaggio di quantità senza precedenti di materie prime. I progetti governativi su larga scala sono stati lo stimolo per prendere decisioni importanti che sono diventate la base della metodologia di Project Management (Kozak-Holland [2011]).

Verso la fine del secolo, Frederick Taylor (1856-1915) iniziò i suoi studi applicando il ragionamento scientifico al lavoro dimostrando che il lavoro può essere analizzato e migliorato concentrandosi sulle sue parti elementari. Egli applicò il suo pensiero alle mansioni che si svolgevano nelle acciaierie, come spalare la sabbia, sollevare e spostare pezzi. Prima di allora, l'unico modo per migliorare la produttività era quello di chiedere ai lavoratori di lavorare di più e più a lungo: in contrapposizione a questa filosofia, Taylor introdusse il concetto di lavoro efficiente (Taylor [1911]).

Il collaboratore di Taylor, Henry Gantt (1861-1919), studiò in dettaglio l'ordine delle operazioni nel lavoro. I suoi studi si concentrarono sulla costruzione di navi da guerra durante la Prima Guerra Mondiale. I suoi diagrammi, completi di barre per le attività e milestone, delineano la sequenza e la durata di tutti i compiti in un processo. I diagrammi di Gantt si sono rivelati uno strumento analitico così potente per i manager che sono rimasti praticamente invariati per quasi cento anni (Kozak-Holland [2011], Lock [2013]).

Durante la Seconda Guerra Mondiale, i complessi progetti governativi e militari e la riduzione della manodopera in tempo di guerra richiedevano nuove strutture organizzative. Vennero, quindi, introdotti i diagrammi PERT e il metodo del percorso critico, che permettevano ai manager di avere un maggiore controllo su progetti massivi e complessi. Ben presto, queste tecniche si diffusero in tutti i tipi di industrie, poiché i dirigenti aziendali cercavano nuove strategie e strumenti di gestione per direzionare la loro crescita in un mondo competitivo e in rapida evoluzione (Kozak-Holland [2011], Lock [2013]).

Il periodo moderno del Project Management inizia intorno al 1965 con la creazione dell'International Project Management Association (IPMA) e, successivamente, nel 1969, del Project Management Institute (PMI) nel 1969. Negli anni '70, il Project Management ha iniziato a essere applicato in settori diversi, come l'aerospazio, la difesa e la costruzione, e si sono sviluppate strategie importanti, come la Work Breakdown Structure (WBS). Dal 1970 al 1979 si è vista una rapida crescita della tecnologia dell'informazione (IT). La gestione dei progetti industriali è continuata come prima, ma con una maggiore disponibilità di software di gestione dei progetti e un più ampio riconoscimento del ruolo. Tuttavia, la diffusione dell'Information Technology portò sulla scena un altro tipo di project manager, ovvero il project manager IT, una figura professionale dedicata a guidare i team di sviluppo di progetti informatici (Lock [2013]).

Negli anni '80 e '90, ci sono state notevoli innovazioni nel campo del Project Management, inclusi lo sviluppo di metodi come Scrum e Lean, e l'introduzione del PMBOK (Project Management Body of Knowledge) da parte del PMI. In particolare, negli anni '90, si è assistito all'ascesa delle metodologie Agile nel Project Management, segnando un cambiamento significativo rispetto alle pratiche tradizionali. Il Manifesto Agile, pubblicato nel 2001, ha definito i principi chiave di questa metodologia, enfatizzando l'adattabilità, la collaborazione di squadra e la risposta rapida ai cambiamenti (Seymour e Hussein [2014]).

Oggi, il Project Management è riconosciuto come una competenza fondamentale in molteplici settori e continua a evolversi con l'incorporazione di nuove tecnologie e metodi di gestione.

1.3 Ruoli e Responsabilità nel Project Management

1.3.1 Il Project Manager

Il Project Manager è la figura centrale nel processo di gestione dei progetti. Secondo il Project Management Institute (PMI), il Project Manager ha la responsabilità di guidare il progetto dall'inizio alla fine (PMI [2013]). Ciò comprende la pianificazione, l'organizzazione, la gestione delle risorse, la supervisione dell'esecuzione e il monitoraggio del progresso del progetto. Il Project Manager deve anche gestire le relazioni con gli stakeholder e assicurare che il progetto sia allineato con gli obiettivi strategici dell'organizzazione (Bezak e Nahod [2011]). A prima vista, può sembrare che i project manager svolgano le stesse funzioni degli altri manager, cioè pianificano, programmano, motivano e controllano. Tuttavia, ciò che li distingue è che gestiscono attività temporanee e non ripetitive, per completare un progetto di durata limitata. A differenza dei manager funzionali, che si occupano di processi esistenti, i project manager creano un team di progetto e un'organizzazione dove prima non esisteva nulla. Devono decidere cosa fare e come fare le cose, invece di limitarsi a gestire processi prestabiliti. Devono affrontare le sfide di ogni fase del ciclo di vita del progetto, e persino supervisionare la dismissione delle attività una volta completato il progetto.

I project manager devono lavorare con una serie di attori differenti per portare a termine i progetti. In genere sono il collegamento diretto con il cliente e devono gestire la tensione tra le aspettative del cliente e ciò che è fattibile e ragionevole. I project manager forniscono direzione, coordinamento e integrazione al team di progetto, che spesso è composto da partecipanti part-time afferenti ai loro dipartimenti funzionali. Spesso devono lavorare con un gruppo di esterni (venditori, fornitori, subappaltatori) che non condividono necessariamente la loro fedeltà al progetto, ovvero posseggono interessi differenti e divergenti.

I project manager sono, in ultima analisi, responsabili delle prestazioni, anche se spesso con un'autorità troppo limitata. Devono assicurarsi che vengano effettuati adeguati compromessi tra i requisiti di tempo, costo e prestazioni del progetto. A differenza delle loro controparti funzionali, i project manager possiedono generalmente minori conoscenze tecniche e di contesto, tali per cui sarà necessario un supporto esterno per determinate decisioni. Ciò che, invece, è completamente a carico del project manager è l'orchestrazione del completamento del progetto inducendo le persone giuste, al momento giusto, ad affrontare le questioni giuste e a prendere le decisioni giuste (Larson e Gray [2010]). Le funzioni svolte dalla figura del project manager sono riassunte e visualizzabili in Figura 1.1.

1.3.2 Il team di progetto

La maggior parte delle persone si avvicina per la prima volta al Project Management quando lavora come parte di un team incaricato di portare a termine un progetto specifico. A volte questo lavoro è a tempo pieno, ma nella maggior parte dei casi si lavora part-time su uno o più progetti. I dipendenti devono imparare a destreggiarsi tra gli impegni quotidiani e le responsabilità aggiuntive del progetto. Possono entrare a far parte di un team con una lunga storia di collaborazione, nel qual caso i ruoli e le norme sono ben definiti. In alternativa, il team può essere composto da estranei provenienti da reparti e organizzazioni diverse. In questo caso, devono sopportare le difficoltà di crescita di un gruppo che si evolve in una squadra.

Le persone devono imprimere una forza positiva nell'aiutare il team a coagularsi in un team di progetto efficace. Non ci sono solo problemi di persone, ma i membri del progetto devono anche utilizzare strumenti e concetti di Project Management. Il project charter sarà fondamentale per poter condividere con tutti i membri del team la dichiarazione dell'ambito e, quindi, gli obiettivi e i parametri del progetto. I membri del team devono comprendere le

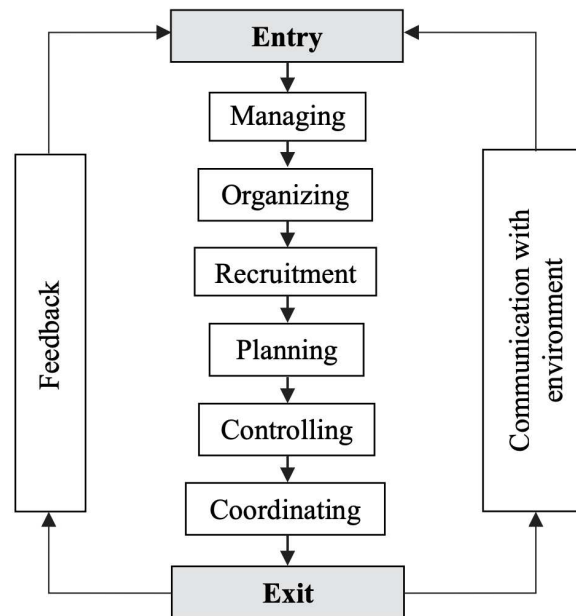


Figura 1.1: Funzioni e responsabilità del project manager (Bezak e Nahod [2011])

priorità del progetto in modo da poter prendere decisioni indipendenti; essi devono sapere come monitorare e riportare i progressi del progetto (Larson e Gray [2010]).

1.3.3 Lo sponsor di progetto

Gli sponsor hanno il compito di promuovere i progetti nelle organizzazioni per creare valore in linea con la strategia organizzativa. Lo sponsor definisce i benefici del progetto, li allinea alla strategia organizzativa e ne assicura la realizzazione. Gli sponsor devono essere responsabilizzati e tenuti in considerazione per creare un valore significativo per l'organizzazione. Nessuna azienda può sopravvivere senza creare valore per i clienti e, quindi, la sponsorizzazione dei progetti è una funzione essenziale delle organizzazioni.

La gestione dei progetti si è spesso concentrata sulla realizzazione di un progetto nel rispetto dei costi, della qualità e dei tempi corretti; non si concentra, invece, sulla garanzia che un'organizzazione realizzi il progetto corretto per realizzare la propria strategia. È qui che si è sviluppato il ruolo dello sponsor, che ha il compito di garantire che un'organizzazione porti a termine i progetti giusti. Lo sponsor può essere definito come l'assuntore del rischio critico di un progetto. Le responsabilità dello sponsor, dunque, possono essere così riepilogate:

- definire i benefici/requisiti aziendali;
- stabilire una strategia di progetto con le relative priorità;
- concordare la definizione del progetto, compresi gli obiettivi;
- definire i criteri di successo del progetto;
- monitorare continuamente l'ambiente di business del progetto e la realizzazione dei benefici;
- prendere in carico il progetto fino al suo completamento e, in casi estremi, prendere la decisione di cancellare il progetto (Opoku e Tallon [2019]).

1.3.4 Gli stakeholder

Gli stakeholder sono individui o gruppi che hanno interesse nel progetto. Possono includere clienti, fornitori, sponsor, la comunità, e altri influenzatori esterni. La gestione degli stakeholder è una componente cruciale del Project Management, poiché le loro esigenze e aspettative devono essere comprese e gestite efficacemente. Tra le ragioni che influenzano i risultati dei progetti, la gestione degli stakeholder e, soprattutto, la loro comprensione sono identificati come la chiave del successo del progetto. Ciascuno di questi gruppi di parti interessate porta al progetto diverse competenze, standard, priorità e agende.

Le parti interessate sono persone e organizzazioni attivamente coinvolte nel progetto, o i cui interessi possono essere positivamente o negativamente influenzati dal progetto. L'ampia gamma e complessità delle relazioni con le parti interessate distingue la gestione dei progetti dalla gestione ordinaria. Per essere efficace, un responsabile di progetto deve comprendere come le parti interessate possano influenzare il progetto e sviluppare metodi per gestire la dipendenza. Una possibile rete di stakeholder è rappresentata in Figura 1.2.

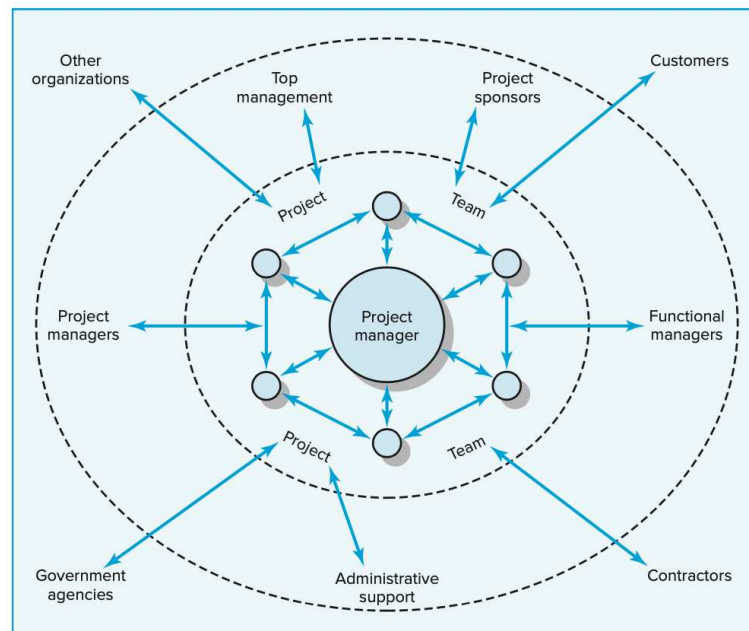


Figura 1.2: Rete di stakeholder (Larson e Gray [2010])

L'influenza degli stakeholder sul progetto può essere definita attraverso sei variabili indipendenti: potere, interesse, urgenza, legittimità, prossimità e rete di relazioni.

- I. *Potere*: rappresenta la capacità degli stakeholder di influenzare il progetto o le decisioni. Gli stakeholder con alto potere possono avere un impatto significativo sugli obiettivi, sulle risorse o sui risultati del progetto. Il potere può derivare da vari fattori, come autorità formale, controllo delle risorse, conoscenza esperta, o posizione influente all'interno o all'esterno dell'organizzazione.
- II. *Interesse*: si riferisce al livello di attenzione, preoccupazione o investimento degli stakeholder verso il progetto. Gli stakeholder con un alto interesse sono più propensi a prestare attenzione e a partecipare attivamente al progetto. Questo interesse può essere motivato da benefici personali, professionali o organizzativi che gli stakeholder percepiscono di poter ottenere dal successo del progetto.
- III. *Urgenza*: indica la necessità di azioni immediate o tempestive da parte degli stakeholder. Gli stakeholder con questioni urgenti richiedono una risposta rapida e possono esercitare

- pressione per accelerare determinate decisioni o azioni nel progetto. L'urgenza può essere dovuta a scadenze, problemi critici, o a situazioni in rapido cambiamento che richiedono attenzione immediata.
- IV. *Legittimità*: si riferisce al grado con cui le pretese o le azioni degli stakeholder sono percepite come appropriate, giustificate o conformi alle norme sociali, legali o professionali. Gli stakeholder legittimi hanno, generalmente, una maggiore influenza sul progetto, poiché le loro richieste o preoccupazioni sono viste come valide e meritano considerazione.
- V. *Prossimità*: riguarda la vicinanza fisica o relazionale al progetto da parte degli stakeholder. Gli stakeholder con una maggiore prossimità possono essere più coinvolti e influenti, poiché la loro vicinanza può permettere un maggior coinvolgimento e un accesso più diretto alle informazioni o alle persone chiave del progetto.
- VI. *Rete di relazioni*: si riferisce alla rete di collegamenti e relazioni che uno stakeholder ha con altri individui o gruppi, sia all'interno che all'esterno dell'organizzazione. Gli stakeholder con una vasta o influente rete di relazioni possono utilizzare questi collegamenti per influenzare il progetto, mobilitare risorse, o facilitare la comunicazione e la cooperazione tra diversi gruppi (Rajablu *et al.* [2015]).

1.4 Ciclo di vita del progetto

La natura unica del lavoro di progetto può essere espressa attraverso il suo ciclo di vita. Quest'ultimo riconosce che i progetti hanno una durata limitata e che ci sono cambiamenti prevedibili nel livello di impegno e di attenzione nel corso della vita del progetto. Nella letteratura del Project Management esistono diversi modelli di ciclo di vita. Quest'ultimi hanno alcune caratteristiche in comune:

- i costi e l'impegno sono bassi all'inizio, hanno un picco nella fase intermedia e scendono rapidamente alla fine;
- il livello d'incertezza e il livello di rischio sono alti all'inizio e diminuiscono con il procedere del progetto;
- la possibilità d'influenzare il prodotto finale del progetto da parte degli stakeholder è alta all'inizio e poi decresce;
- il costo dell'esecuzione di modifiche è basso all'inizio e poi cresce.

I cicli di vita del progetto possono essere catalogati nelle seguenti famiglie:

- *Cicli di vita predittivi (Predictive Life-cycles)*: questo approccio, noto anche come approccio tradizionale o a cascata (Waterfall), si basa sulla pianificazione dettagliata all'inizio del progetto. Il ciclo di vita del progetto è sequenziale e ogni fase deve essere completata prima di passare alla successiva. Ad esempio, non si passa allo sviluppo fino a quando la progettazione non è completa. Questo approccio funziona meglio per progetti con requisiti ben definiti e stabili, dove i cambiamenti sono limitati e prevedibili. È comunemente usato in progetti di costruzione, manifattura e altri settori, dove le modifiche in corso sono costose o tecnicamente impraticabili.
- *Cicli di vita adattivi (Adaptive Life-cycles)*: questo approccio, spesso associato all'Agile, si adatta ai cambiamenti rapidi e ai requisiti in evoluzione. Invece di pianificare tutto in anticipo, il progetto procede attraverso iterazioni o sprint, con team che lavorano in

cicli brevi e si adattano in base al feedback continuo. L'approccio adattivo è ideale per progetti in ambienti di rapido cambiamento, come lo sviluppo di software o progetti in settori in rapida evoluzione. Questo approccio enfatizza la collaborazione del team, il coinvolgimento del cliente, la risposta ai cambiamenti e la consegna incrementale di prodotti funzionanti.

- *Cicli di vita ibridi (Hybrid Life-cycles)*: come suggerisce il nome, questo approccio combina elementi sia predittivi che adattivi. In un approccio ibrido, alcune parti del progetto possono essere gestite con un processo sequenziale, mentre altre parti possono adottare un approccio più flessibile e iterativo. Questo modello è utile per progetti che hanno sia elementi ben definiti che aree con requisiti più incerti o soggetti a cambiamenti (PMI [2021]).

Ogni approccio ha i suoi punti di forza e le sue sfide, e la scelta dell'approccio più appropriato dipende dalla natura del progetto, dalla sua complessità, dai requisiti degli stakeholder, dall'ambiente di mercato e da altri fattori organizzativi e contestuali. Una schematizzazione degli approcci appena descritti è mostrata in Figura 1.3.

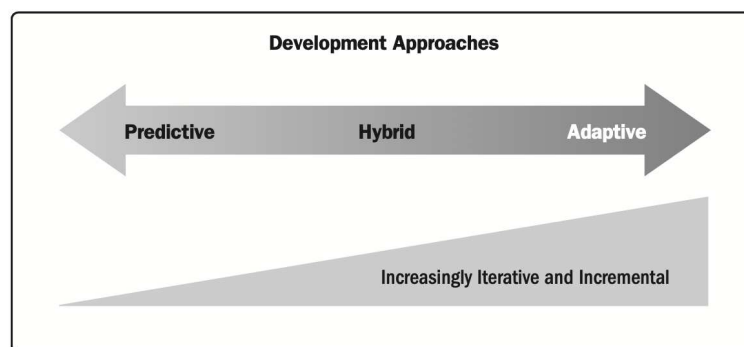


Figura 1.3: Cicli di vita del progetto (PMI [2021])

I progetti si caratterizzano per la loro specificità: sono un'impresa che ha un inizio e una fine, un obiettivo preciso e unico, realizzato da un team creato appositamente per il progetto. Lo scopo del ciclo di vita di un progetto è trovare qualcosa di comune a tutti i progetti, nonostante le differenze. Poiché i cicli di vita dei progetti sono modelli, il loro scopo è spiegare il mondo reale in modo più semplice e comprensibile. Poiché il mondo dei progetti non consiste in un numero molto piccolo di variabili, è difficile immaginare, e forse impossibile realizzare, un modello che possa integrare tutte le variabili che rendono specifici tutti i progetti. Per comprendere il progetto in cui sono coinvolti, i project manager e i membri del team devono condividere una visione comune dei loro obiettivi, e in particolare il modo in cui un progetto progredisce. Lo scopo dei modelli di ciclo di vita dei progetti è quello di illustrare semplicemente la "filosofia di progresso" dei progetti, al fine di promuovere una migliore comprensione e una migliore comunicazione all'interno degli stessi (Bonnal *et al.* [2002]).

Un nuovo progetto di sviluppo software può consistere in cinque fasi: definizione, progettazione, codice, integrazione/test e manutenzione. Un ciclo generico è rappresentato nella Figura 1.4. Il ciclo di vita del progetto passa in genere in sequenza attraverso quattro fasi: definizione, pianificazione, esecuzione e chiusura. Il punto di partenza inizia nel momento in cui viene dato il via libera al progetto. L'impegno inizia lentamente, cresce fino a raggiungere un picco e poi diminuisce fino alla consegna del progetto al cliente. Le quattro fasi del ciclo di vita di un progetto sono le seguenti:

- I. *Fase di definizione*: si definiscono le specifiche del progetto, si stabiliscono gli obiettivi, si formano i team e si assegnano le principali responsabilità.
- II. *Fase di pianificazione*: il livello di impegno aumenta e vengono sviluppati piani per determinare cosa comporterà il progetto in termini di ambito realizzativo, quando sarà programmato, chi ne beneficerà, quale livello di qualità dovrà essere mantenuto e quale sarà il budget.
- III. *Fase di esecuzione*: si svolge una parte importante del lavoro del progetto, viene realizzato il prodotto fisico. È necessario monitorare che il progetto rispetti i tempi, il budget e le specifiche.
- IV. *Fase di chiusura*: comprende tre attività, ovvero la consegna del prodotto del progetto al cliente, la redistribuzione delle risorse del progetto e la revisione post-progetto. La consegna del progetto può includere la formazione del cliente e il trasferimento dei documenti. La redistribuzione di solito comporta il rilascio di attrezzature/materiali del progetto ad altri progetti e la ricerca di nuovi incarichi per i membri del team. Le revisioni post-progetto comprendono non solo la valutazione delle prestazioni, ma anche l'acquisizione delle lezioni apprese.

In pratica, il ciclo di vita del progetto viene utilizzato da alcuni gruppi di progetto per rappresentare la tempistica delle attività principali nel corso della vita del progetto. Per esempio, il team di progettazione potrebbe pianificare un grosso impegno di risorse nella fase di definizione, mentre il team della qualità si aspetterebbe che il suo impegno maggiore aumenti nelle ultime fasi del ciclo di vita del progetto. Poiché la maggior parte delle organizzazioni ha un portafoglio di progetti in corso contemporaneamente, ciascuno in una fase diversa del ciclo di vita, è indispensabile un'attenta pianificazione e gestione a livello di organizzazione e di progetto (Larson e Gray [2010]).

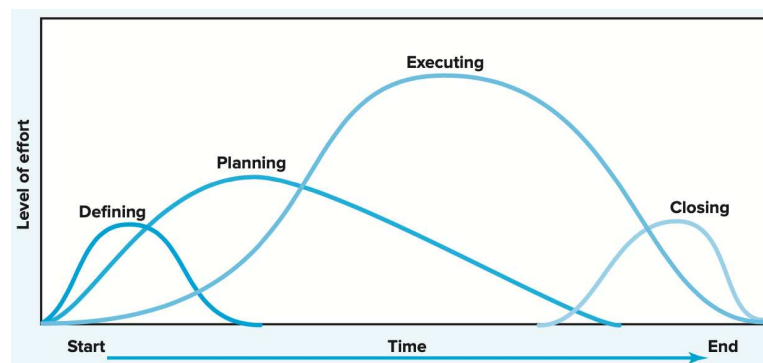


Figura 1.4: Fasi del ciclo di vita di un progetto (Larson e Gray [2010])

1.5 Metodologie di Project Management

In tutti i settori industriali, la gestione dei progetti è diventata un elemento essenziale per il loro successo. Indipendentemente dal settore industriale o dalle dimensioni del progetto, le metodologie di Project Management possono essere applicate per migliorare la probabilità di raggiungere gli obiettivi del progetto Chin *et al.* [2012].

Diversi tipi di progetti richiedono modelli procedurali differenti per essere eseguiti con successo. Un modello procedurale organizza i metodi e gli strumenti di Project Management in fasi o processi di progetto in modo standardizzato. I modelli procedurali per la gestione dei

progetti possono essere suddivisi in metodi plan-driven, che seguono un classico processo a cascata (waterfall), e metodi agili (come Scrum e Kanban), che seguono un approccio iterativo e test-driven.

Nella gestione classica dei progetti che segue un processo di pianificazione ed esecuzione "a cascata", i risultati attesi sono comunicati in modo relativamente chiaro dal cliente all'inizio del progetto. Affinché il progetto possa essere orientato agli obiettivi e ai piani, viene pianificato in modo dettagliato, dall'avvio al completamento, con workpackage, responsabilità e scadenze. L'attenzione si concentra sull'attuazione del piano iniziale nel modo più preciso possibile. Ciò garantisce stabilità e struttura, risorse prevedibili e una pianificazione documentata (Thesing *et al.* [2021]).

Il metodo a cascata prevede una serie di fasi logiche in cui il processo scorre da una fase all'altra (Figura 1.5). Nel caso di progetto IT, il presupposto fondamentale è che i requisiti essenziali possano essere definiti in anticipo, in modo da poter progettare, costruire e testare il software. I progetti software coinvolgono, in genere, molti clienti diversi con esigenze differenti; tali esigenze cambiano spesso e sono spesso difficili da articolare. In molti casi, i clienti iniziano a capire cosa desiderano realmente solo quando ricevono una bozza di ciò che vogliono. In queste condizioni, risulta difficile sviluppare un elenco dettagliato dei requisiti dell'ambito all'avvio del progetto. Questo è uno dei motivi principali per cui i progetti software che utilizzano l'approccio a cascata hanno una storia di ritardi e/o cancellazioni e, quindi, spesso si preferisce utilizzare una metodologia agile (Larson e Gray [2010]).

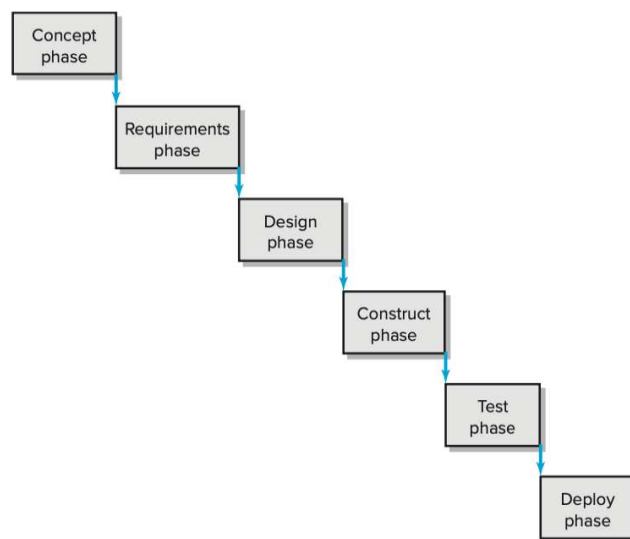


Figura 1.5: Approccio waterfall per lo sviluppo del software (Larson e Gray [2010]).

La metodologia agile, nata nell'ambito dello sviluppo del software, è ora utilizzata in un numero crescente di settori. I metodi agili, come Scrum o Kanban, non si concentrano su una pianificazione avanzata e completa e sulla esecuzione lineare ed esatta di un piano. Al contrario, un team di progetto sviluppa una soluzione passo dopo passo e coordina i rispettivi risultati intermedi con il cliente in cicli molto brevi. Una ragione per adottare questo metodo è che il cliente o l'utente del risultato del progetto specifica i requisiti generali, ma non può specificarli in dettaglio nelle prime fasi del progetto. D'altra parte, i passaggi necessari per raggiungere gli obiettivi possono essere poco chiari. L'approccio agile definisce anche gli obiettivi o la visione del progetto, ma lo fa a un livello di dettaglio relativamente basso e con un orizzonte di pianificazione più breve (ad esempio, da due a quattro settimane). La flessibilità, nel caso di richieste di modifica, è più importante della rigida aderenza al piano iniziale, poiché le aspettative sul risultato si concretizzano durante l'attuazione del progetto.

Il processo di progetto non è lineare in fasi successive, come nel classico processo a cascata. Piuttosto, si possono utilizzare più iterazioni per avvicinarsi al risultato desiderato in modo guidato dai test. I metodi di gestione agile dei progetti offrono flessibilità nella gestione dei progetti, consentendo alle aziende di reagire rapidamente alle mutevoli esigenze dei clienti (Thesing *et al.* [2021]).

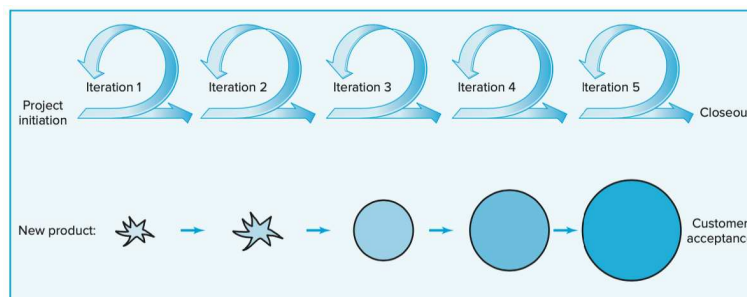


Figura 1.6: Approccio agile (Larson e Gray [2010])

Nella realtà possono essere utilizzati anche approcci ibridi, con lo scopo di arricchire il modello di processo plan-driven con i principi agili, combinando, così, i vantaggi di entrambi i metodi; in questo caso il "quadro generale" viene pianificato mediante un processo a cascata, ma i sottoprogetti vengono gestiti in modo agile. Utilizzando una comunicazione ad alta frequenza e cicli di feedback brevi, i sottoprogetti sfruttano le caratteristiche agili, come la trasparenza e l'adattabilità, ma seguono la struttura generale di un piano di progetto classico di livello superiore. La scelta di un modello procedurale adatto a un progetto è spesso una sfida importante per i professionisti (Thesing *et al.* [2021]).

1.6 Strumenti e tecniche di Project Management

Per aiutare i project manager ad avere un quadro organizzato del progetto e quindi a comprendere e gestire tutti gli aspetti dello stesso, nel PMBOK (PMI [2021]) vengono definite dieci aree di conoscenza. Attraverso la definizione di aree di conoscenza specifiche, i project manager possono assicurarsi di aver considerato tutti gli elementi chiave necessari per il successo del progetto, dalla gestione dei tempi e dei costi alla qualità, risorse umane, rischi, e oltre. La chiara definizione delle aree di conoscenza aiuta nella comunicazione all'interno dei team di progetto e con gli stakeholder, fornisce un linguaggio comune e assicura che tutti gli aspetti del progetto siano adeguatamente comunicati e compresi. Inoltre, le aree di conoscenza aiutano a standardizzare le pratiche di Project Management all'interno di un'organizzazione o di un settore, garantendo che i progetti siano gestiti con un approccio coerente e collaudato. Si noti che l'utilizzo delle aree di conoscenza aiuta a decomporre la complessità del progetto in parti più gestibili, consentendo ai project manager di focalizzarsi su aspetti specifici del progetto in modo più dettagliato. Dunque, attraverso l'applicazione di queste aree di conoscenza, i project manager possono identificare e mitigare rischi, gestire risorse in modo più efficace e assicurare che tutti gli aspetti del progetto siano adeguatamente controllati e coordinati, aumentando, così, la probabilità di successo del progetto. Di seguito un elenco dettagliato delle suddette aree e dei principali strumenti di Project Management a supporto.

- *Gestione dell'integrazione di progetto:* riguarda le attività necessarie per assicurare che i vari elementi del progetto siano adeguatamente coordinati. Include lo sviluppo del project charter, la definizione del piano di gestione del progetto, e la supervisione dell'esecuzione del progetto.

- *Gestione dell'ambito di progetto*: si concentra sulla definizione e sul controllo di ciò che è e non è incluso nel progetto. Comprende la pianificazione, la definizione, la verifica e il controllo dell'ambito del progetto. Un importante strumento di supporto per queste attività è la Work Breakdown Structure (WBS), che permette di scomporre il lavoro da svolgere in componenti più piccole, organizzate in forma gerarchica. La scomposizione del lavoro in una struttura gerarchica codificata è un caposaldo del Project Management, in quanto essa consente di fare chiarezza sul lavoro da svolgere, facilita una chiara assegnazione delle responsabilità, migliora l'accuratezza della stima di costi, durate e risorse necessarie e definisce la baseline dell'ambito per migliorare la misurazione e il controllo delle prestazioni di progetto durante l'esecuzione. La WBS è una rappresentazione gerarchica del lavoro del progetto ed è caratterizzata da elementi detti WBE (Work Breakdown Element), ciascuno con una codifica univoca. I Work Package rappresentano gli elementi finali della WBS non ulteriormente scomponibili, su cui verranno aggregate informazioni su schedulazione, costi, e risorse. A titolo esemplificativo, in Figura 1.7 è riportato un prototipo di WBS.

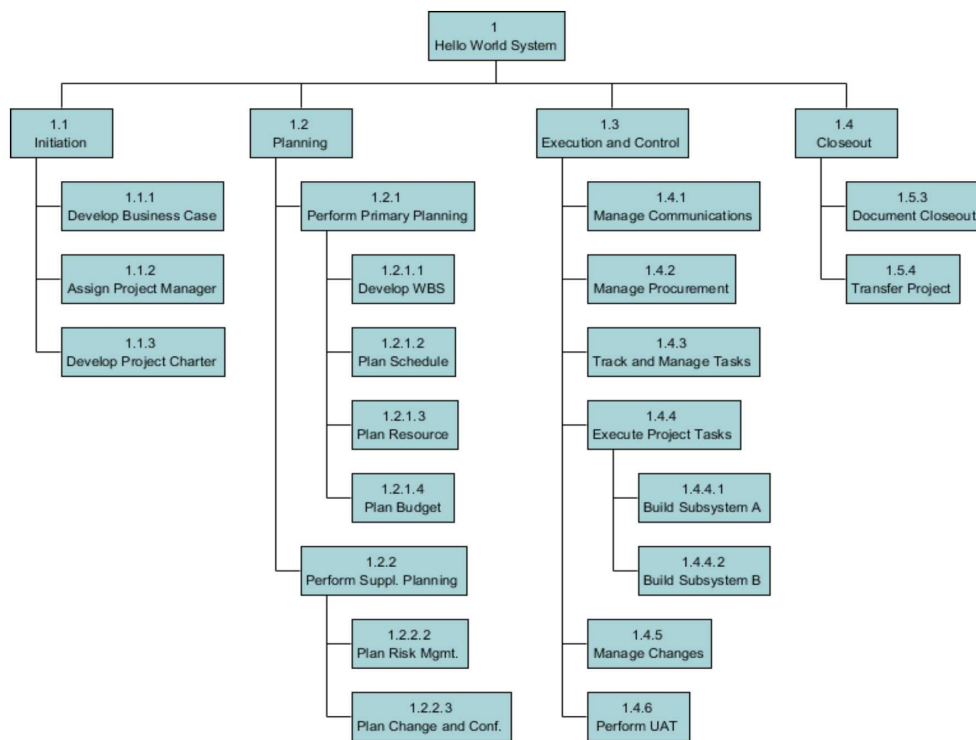


Figura 1.7: Esempio di Work Breakdown Structure (Alutbi [2020])

- *Gestione dei tempi di progetto*: include le procedure necessarie per completare il progetto nel tempo stabilito. Implica la definizione delle attività, la loro sequenzializzazione, la stima delle risorse e della durata, lo sviluppo e il controllo del progetto. La schedulazione di progetto è spesso rappresentata come diagramma di Gantt, dove in un calendario sviluppato orizzontalmente si rappresentano le attività con delle barre e si mostrano date di inizio, date di fine, durate attese e scorrimenti (James e Wilson [2000]). In combinazione a questo strumento, è importante considerare una metodologia focalizzata sull'identificazione e la gestione delle attività critiche per il completamento tempestivo del progetto, come il Critical Path Method (CPM) o il Critical Chain Project Management (CCPM). Il CPM è spesso usato in progetti dove la durata delle attività è relativamente prevedibile e il focus è sulla gestione del tempo e sulle sequenze di

attività; il CCPM, invece, è particolarmente utile in ambienti dove le risorse sono limitate o soggette a variazioni. Infatti mentre il CPM si concentra principalmente sulla sequenza temporale delle attività e sul percorso critico, il CCPM integra la gestione delle risorse e l'ottimizzazione dei buffer per migliorare l'efficienza complessiva e la consegna tempestiva del progetto (PMI [2021]). In Figura 1.8 è possibile visualizzare un esempio di diagramma di Gantt abbinato alla metodologia CCPM per la stima delle attività critiche.

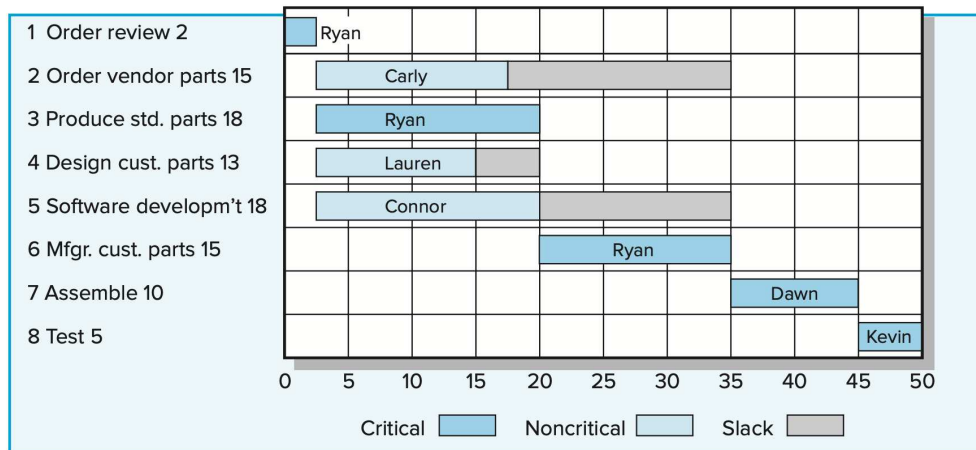


Figura 1.8: Esempio di diagramma di Gantt con applicazione del CCPM (Larson e Gray [2010])

Per determinare la sequenza di tutte le attività necessarie per completare un progetto è fondamentale stimare nel modo più corretto possibile la durata delle stesse. Per questo è possibile utilizzare la metodologia PERT (Program Evaluation and Review Technique). Per ciascuna attività, PERT considera tre stime di tempo:

- *Ottimistica (O)*: il tempo minimo necessario per completare un'attività, supponendo che tutto proceda meglio del previsto.
- *Pessimistica (P)*: il tempo massimo che potrebbe essere richiesto, supponendo che tutto vada male.
- *Più Probabile (M)*: il tempo più probabile necessario per completare l'attività, supponendo che tutto proceda normalmente.

Dopodiché, queste tre stime sono utilizzate per calcolare una durata stimata per ogni attività, utilizzando la seguente formula per bilanciare le incertezze e fornire una stima più realistica (Nafkha [2016]):

$$Durata\ Stimata = \frac{O + 4M + P}{6} \quad (1.6.1)$$

PERT è particolarmente efficace in progetti di ricerca e sviluppo, progetti di ingegneria e altri progetti di grande scala, dove il tempo è un fattore critico e dove esiste una notevole incertezza riguardo alla durata delle attività. La metodologia è usata per la pianificazione e il controllo del progetto, permettendo ai manager di identificare le aree che necessitano di maggiore attenzione e di allocare le risorse in modo più efficiente.

- *Gestione dei costi di progetto*: è fondamentale per assicurare che un progetto venga completato entro il budget previsto. Comprende la definizione della baseline dei costi. Quest'ultima è la distribuzione nel tempo del valore cumulato dei costi previsti del progetto e viene usata per misurare e controllare l'andamento dei costi di progetto durante

la sua evoluzione. Viene sviluppata sommando i costi stimati di progetto per periodo e assume una forma di curva a S. La baseline dei costi integra tempi e costi previsti per il progetto e si ottiene grazie alla schedulazione delle attività di progetto e alla distribuzione nel tempo dei costi a esse associate. In Figura 1.9 è possibile visualizzare un possibile andamento: si noti che, per rispettare il budget, i costi pianificati devono essere inferiori ai finanziamenti.

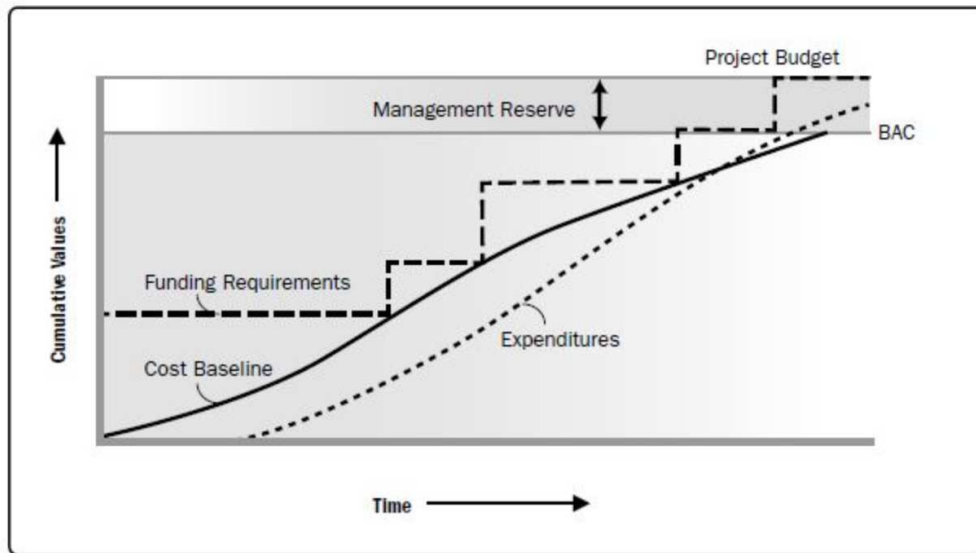


Figura 1.9: Baseline dei costi (PMI [2021])

- *Gestione della qualità di progetto:* assicura che il progetto soddisfi le esigenze per cui è stato intrapreso. Include la pianificazione, l'assicurazione e il controllo della qualità. Uno strumento ampiamente utilizzato nel controllo di qualità è il diagramma causa-effetto di Ishikawa, chiamato anche diagramma a lisca di pesce. Questo è uno strumento visivo utilizzato per identificare, esplorare e visualizzare le possibili cause di un problema specifico. Il diagramma ha una forma che ricorda una lisca di pesce: al centro si trova una linea orizzontale (la "spina dorsale"), con varie "liche" o rami che si diramano ad angolo. All'estremità della spina dorsale si trova il problema da analizzare (Figura 1.10) (Coccia [2017]).

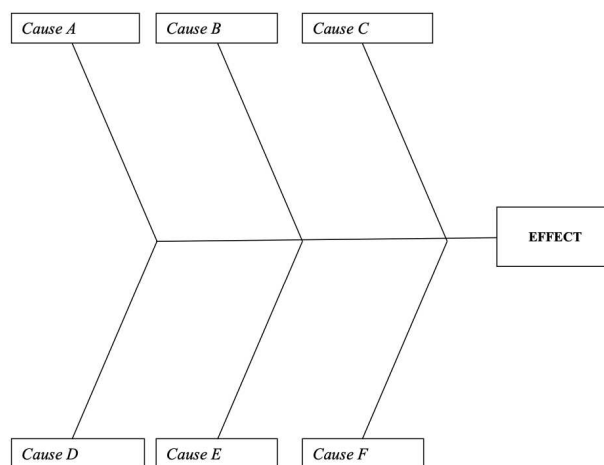


Figura 1.10: Diagramma di Ishikawa (causa-effetto) (Coccia [2017])

- *Gestione delle risorse di progetto*: riguarda l'organizzazione, la gestione e la guida del team di progetto. Comprende la pianificazione delle risorse umane, l'acquisizione, lo sviluppo e la gestione del team. Per chiarire le responsabilità e i ruoli nel team può essere utilizzata la matrice RACI, così da contribuire ad una gestione efficace delle risorse umane. RACI è un acronimo che sta per Responsible, Accountable, Consulted e Informed, che rappresentano i quattro ruoli chiave assegnati alle persone o ai gruppi per ogni attività. Più specificatamente:
 - *Responsible*: è chi effettua il lavoro per completare l'attività. Ogni attività deve avere almeno un responsabile, ma può averne anche di più.
 - *Accountable*: è chi è, in ultima istanza, responsabile per il completamento corretto e tempestivo dell'attività. Deve esserci un solo Accountable per ogni attività.
 - *Consulted*: è chi fornisce input o consigli durante il completamento dell'attività. Sono persone spesso coinvolte nelle discussioni o nelle decisioni prima che l'attività sia completata.
 - *Informed*: è chi deve essere tenuto informato dello stato delle attività e dei risultati. Sono persone che vengono aggiornate ma non hanno un ruolo attivo nelle discussioni o nelle decisioni (PMI [2021], Suhanda e Pratami [2021]).

Grazie alla definizione di questa matrice si eliminano ambiguità su chi fa cosa, riducendo il rischio di compiti trascurati o duplicati. Inoltre, ci si assicura che tutte le parti rilevanti siano adeguatamente consultate e informate. Un esempio di matrice RACI è mostrato in Figura 1.11.

NO	Activity	Manager	Manager Assistant	Site Coordinator
1	Government Licensing	A	C, I	R
2	Citizen Licensing	A	C, I	R
3	HDPE Material Procurement	A	C, I	I
4	ODC Material Procurement	A	C, I	I
5	Manhole Construction	A	I	C, I
6	HDPE Installation	A	I	C, I
7	Feeder Cable Installation	A	I	C, I
8	ODC foundation development	A	I	C, I
9	ODC Box Installation	A	I	C, I
10	Feeder cable Installation to ODC	A	I	C, I

Figura 1.11: Esempio di matrice RACI (Suhanda e Pratami [2021])

- *Gestione delle comunicazioni di progetto*: definisce il piano di gestione delle comunicazioni di progetto; implica la generazione, la raccolta, la diffusione e l'archiviazione finale delle informazioni di progetto in modo tempestivo ed efficace.
- *Gestione dei rischi di progetto*: si occupa dell'identificazione, dell'analisi e della risposta ai rischi del progetto. Include la pianificazione della gestione dei rischi, la loro identificazione, la loro analisi qualitativa e quantitativa, la pianificazione delle risposte ad essi e il loro monitoraggio e controllo. Essenziale per la corretta gestione dei rischi è la matrice probabilità-impatto, che definisce come aggregare omogeneamente tutte le possibili combinazioni dei valori della probabilità e dell'impatto, in modo tale da categorizzare i rischi in fasce. L'aver definito queste categorie consente di raggruppare

rischi di diversa natura in insiemi omogenei dal punto di vista della valutazione (è un rischio nero o grigio o bianco) e della politica di gestione (per esempio, per i rischi bianchi si opera con un’accezzazione passiva, mentre per gli altri si interviene) (PMI [2021]). Una possibile implementazione è rappresentata in Figura 1.12.

		IMPACT		
		Low (insignificant, just note)	Medium (reasonable impact, to be monitored)	High (will have a significant impact)
LIKELIHOOD	Low (unlikely to occur)	E	D	C
	Medium (may occur at a time)	D	C	B
	High (likely to occur)	C	B	A

Figura 1.12: Matrice probabilità-impatto (Dumbravă e Iacob [2013])

Altro strumento strategico utile all’identificazione dei rischi (e non solo) è l’analisi SWOT. Questo è uno strumento strategico utilizzato per valutare i punti di forza (Strengths), i punti di debolezza (Weaknesses), le opportunità (Opportunities) e le minacce (Threats) di un progetto. Questo tipo di analisi fornisce una panoramica chiara dei fattori interni ed esterni che possono influenzare la riuscita di un’iniziativa (Figura 1.13). Gli elementi di una matrice SWOT sono i seguenti:

- *Strengths (Punti di Forza)*: sono gli attributi interni positivi dell’organizzazione o del progetto che offrono un vantaggio nel raggiungere gli obiettivi.
- *Weaknesses (Punti di Debolezza)*: sono i fattori interni che limitano o impediscono il raggiungimento degli obiettivi; ne sono un esempio la mancanza di esperienza o le risorse finanziarie limitate.
- *Opportunities (Opportunità)*: sono i fattori esterni che l’organizzazione può sfruttare a suo vantaggio, come i cambiamenti di mercato o modifiche legislative.
- *Threats (Minacce)*: sono i fattori esterni che possono danneggiare l’organizzazione o il progetto, come la concorrenza o le crisi economiche (Helms e Nixon [2010]).

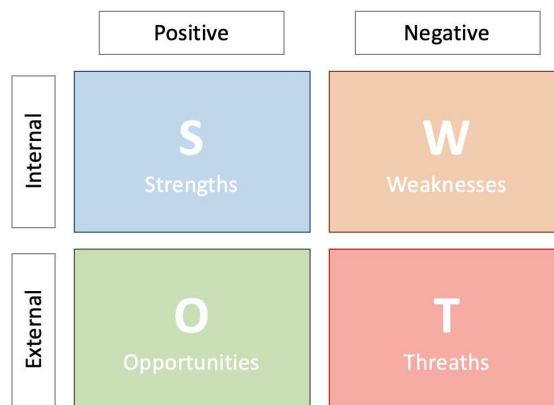


Figura 1.13: Analisi SWOT

- *Gestione dell'approvvigionamento del progetto*: comprende l'acquisizione di beni e servizi necessari al team del progetto dall'esterno dell'organizzazione esecutrice. Include la pianificazione degli acquisti, la selezione dei fornitori, l'amministrazione e la chiusura dei contratti.
- *Gestione degli stakeholder del progetto*: si concentra sulle relazioni con gli stakeholder, compreso il riconoscimento delle loro esigenze e aspettative, la gestione delle influenze sul progetto e l'adeguamento delle strategie e dei piani per coinvolgerli in modo efficace (PMI [2021]).

1.7 Importanza del Project Management

In conclusione, il Project Management influisce sul successo di un progetto garantendo una pianificazione e un'allocatione efficiente delle risorse, attenuando i rischi, migliorando la comunicazione e la collaborazione tra i team, mantenendo il controllo dell'ambito, dei tempi e del budget del progetto e assicurando la qualità dei risultati. Applicando tecniche di Project Management si promuove un approccio strutturato che si allinea con gli obiettivi organizzativi e delinea un approccio per raggiungere costantemente gli obiettivi contribuendo, così, in modo significativo al successo complessivo del progetto (Munns e Bjeirmi [1996], PMI [2021], Larson e Gray [2010]).

Contesto e ambito della tesi: governance di un progetto IT nel settore Automotive

L'obiettivo di questo capitolo è fornire un contesto chiaro in riferimento al lavoro di tesi. Prima di introdurre nel dettaglio il progetto su cui si sono svolte attività di governance, si approfondisce il tema della governance nell'ambito dell'Information Technology. Successivamente, si evidenzia il ruolo fondamentale della Digital Transformation nel settore dell'automotive. Dopo questa panoramica, utile a contestualizzare il lavoro, si elencano gli obiettivi che si vogliono realizzare attraverso l'implementazione della nuova piattaforma. Il progetto non è evoluto come previsto inizialmente; per tale motivo si è scelto di descrivere la storia degli approcci utilizzati per gli sviluppi. Si passa, quindi, a descrivere i requisiti tecnici del progetto attraverso un'architettura di alto livello a cui fa seguito un'architettura più di dettaglio. Infine, si analizzano gli attori in gioco e le motivazioni che hanno portato all'ingresso di BIP con l'incarico della governance di progetto.

2.1 Governance di progetti IT

Quando si parla di progetto IT ci si riferisce ad un'iniziativa in ambito Information Technology. Questi progetti possono variare notevolmente in termini di scala e complessità, ma, generalmente, includono elementi come lo sviluppo di software, l'implementazione di sistemi hardware, l'integrazione di sistemi IT, la sicurezza informatica, l'analisi dei dati o la gestione dell'infrastruttura IT.

Sebbene l'IT nel tempo sia diventato sempre più affidabile, veloce e meno costoso, i costi, le complessità e i rischi dei progetti IT continuano ad aumentare. Con la diffusione dell'Information Technology è nata l'esigenza di avere figure professionali formate per la gestione dei relativi progetti. Infatti, nel 1995, una società di consulenza chiamata Standish Group ha condotto un'indagine su 365 manager IT: il rapporto, chiamato CHAOS, era sorprendente. Sebbene gli Stati Uniti spendessero ogni anno oltre 250 miliardi di dollari per progetti di sviluppo di applicazioni IT, il 31% di questi progetti veniva cancellato prima del completamento. Quasi il 53% è stato completato, ma ha sforato il budget e la tabella di marcia, e non ha rispettato le specifiche originali. Il superamento medio dei costi per un'azienda di medie dimensioni intervistata è stato del 182% rispetto alla stima originale, mentre il superamento medio dei tempi è stato del 202%. In altre parole, i risultati dell'indagine suggeriscono che un progetto di medie dimensioni, il cui costo originario era stato stimato in circa 1 milione di dollari e che avrebbe richiesto un anno di sviluppo, è costato in realtà 1.820.000 dollari, ha richiesto poco più di due anni per essere completato e ha incluso solo il 65% delle caratteristiche e delle funzioni previste. Il 48% dei responsabili IT intervistati riteneva che in quel momento ci fossero più fallimenti rispetto a cinque e dieci anni prima.

Era evidente che nuove tipologie di progetti richiedessero nuove competenze e metodologie di gestione; da qui la diffusione del concetto di governance di progetti IT.

Il successo di un progetto IT non dovrebbe dipendere soltanto dalla composizione del team, ma piuttosto dall'adozione di principi e strumenti di gestione del progetto a livello organizzativo. È importante definire attività, processi, standard di qualità, controlli e deliverable per l'intero progetto. L'impiego di una metodologia comune assicura un linguaggio unificato tra i progetti e permette il confronto interno all'organizzazione. In questo modo, il successo del progetto è sostenuto da processi e infrastrutture in atto, piuttosto che essere condizionato unicamente dal team di progetto.

I progetti dovrebbero essere suddivisi in fasi per renderli più gestibili e ridurre i rischi. Anche se le consegne alla fine di una fase o di uno stadio dovrebbero essere, di solito, approvate prima di procedere alla fase successiva, l'avvio della fase successiva prima di ottenere l'approvazione può, talvolta, ridurre la tempistica del progetto. Sebbene i cicli di vita dei progetti possano variare a seconda del settore o del progetto, tutti hanno un inizio, una parte centrale e una fine. La Figura 2.1 fornisce un ciclo di vita generico che descrive le fasi o gli stadi comuni alla maggior parte dei progetti (Marchewka [2016]).

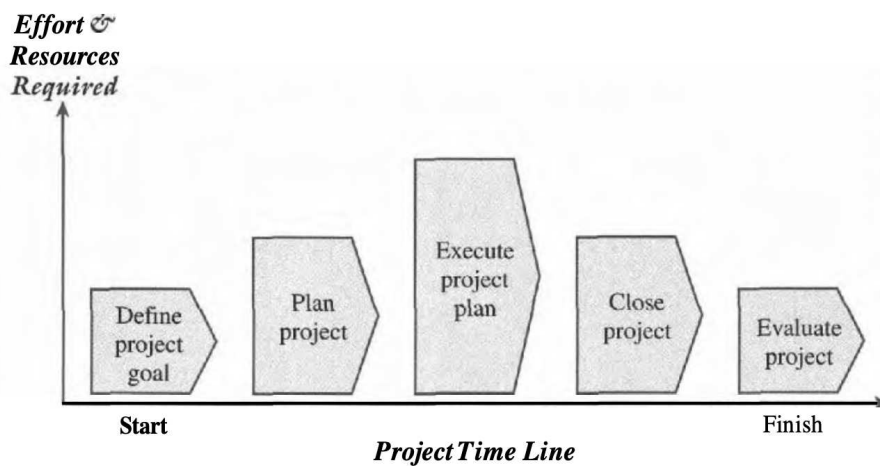


Figura 2.1: Ciclo di vita di un progetto generico (Marchewka [2016])

Il ciclo di vita appena citato è comune ad ogni tipologia di progetto, ma può essere specializzato in base all'area di applicazione. Il ciclo di vita più comune nell'IT è il ciclo di vita dello sviluppo dei sistemi (*System Development Life Cycle - SDLC*), che rappresenta gli stadi sequenziali che un sistema informativo segue durante la sua vita utile. L'*SDLC* stabilisce un ordine logico in cui si svolgono le attività di sviluppo del sistema e indica se procedere da un'attività di sviluppo a quella successiva. Sebbene non esista una versione definitiva e comunemente accettata dell'*SDLC*, il ciclo di vita illustrato nella Figura 2.2 comprende le fasi generalmente accettate associate allo sviluppo di sistemi, ovvero: pianificazione, analisi, progettazione, implementazione, manutenzione e supporto (Zhang *et al.* [2004], Marchewka [2016]). Di seguito una descrizione dettagliata di ciascuna di queste fasi:

- *Pianificazione (Planning)*: la fase di pianificazione prevede l'identificazione e la risposta a un problema o a un'opportunità e incorpora i processi e le attività di gestione del progetto e di sviluppo del sistema. Un processo formale e definito di pianificazione assicura che l'ambito, il budget e le tempistiche di progetto siano rispettati.
- *Analisi (Analysis)*: la fase di analisi cerca di approfondire il problema o l'opportunità. Il team di progetto può documentare il sistema attuale e sviluppare un modello "as is" per comprendere il sistema attualmente in uso. In generale, gli analisti di sistema si

incontrano con i vari stakeholder (utenti, manager, clienti, etc.) per saperne di più sul problema o sull'opportunità. Questo lavoro viene svolto per identificare e documentare eventuali problemi o colli di bottiglia associati al sistema attuale. In genere, l'analisi dello "stato attuale" è seguita da un'analisi dei requisiti. In questo caso vengono identificate e documentate le esigenze e i requisiti specifici del nuovo sistema. I requisiti possono essere sviluppati conducendo colloqui e indagini, osservando i processi di lavoro e analizzando i report aziendali. Quindi, vengono rappresentati e documentati il sistema attuale, i requisiti degli utenti e la progettazione logica del sistema futuro, detto "to be".

- *Progettazione (Design)*: durante la fase di progettazione, il team di progetto utilizza i requisiti e i modelli logici "to be" come input per progettare l'architettura a supporto del nuovo sistema informativo. Tale architettura potrebbe comprendere, in base al progetto di riferimento, la progettazione della rete, della configurazione hardware, dei database, dell'interfaccia utente e dei programmi applicativi.
- *Implementazione (Implementation)*: l'implementazione comprende lo sviluppo o la costruzione del sistema, il collaudo e l'installazione. Inoltre, devono essere previsti la formazione, il supporto e la documentazione.
- *Manutenzione e supporto (Maintenance and support)*: sebbene la manutenzione e il supporto non siano una vera e propria fase del progetto in corso, sono comunque attività importanti da tenere in considerazione. Una volta che il sistema è stato implementato, si dice che è in produzione. Le modifiche al sistema, sotto forma di manutenzione e miglioramenti, sono spesso richieste per correggere eventuali errori (bug) scoperti all'interno del sistema, per aggiungere funzionalità che non erano state incorporate nel progetto originale o per adattarsi ad un ambiente aziendale in evoluzione. Può essere prevista anche un'assistenza, in termini di call center o help desk, per aiutare gli utenti in caso di necessità. Nel corso del tempo il sistema avrà bisogno sempre più di manutenzione, per cui l'organizzazione potrebbe decidere che è giunto il momento di sostituire il vecchio sistema con uno più recente, che sarà più affidabile, richiederà meno manutenzione e risponderà meglio alle nuove esigenze. Di conseguenza, inizia un nuovo ciclo di vita (Marchewka [2016]).

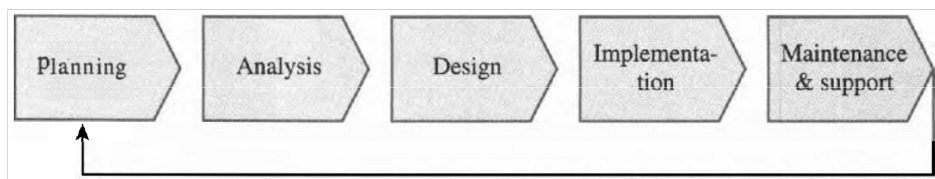


Figura 2.2: Ciclo di vita di un progetto IT (SDLC) (Marchewka [2016])

2.2 Il ruolo della Digital Transformation nel settore automotive

La Digital Transformation è un processo in continua evoluzione che coinvolge l'adozione di tecnologie digitali da parte delle organizzazioni per migliorare i processi, le esperienze dei clienti e generare nuove opportunità di business. Questo processo non si limita alla semplice automazione delle attività esistenti, ma riguarda una ristrutturazione profonda dei modelli di business, delle strategie aziendali e dell'infrastruttura IT. La Digital Transformation può comprendere varie aree, come l'analisi dei dati, il cloud computing, l'intelligenza artificiale,

l'Internet of Things (IoT) e la cybersecurity. I cambiamenti apportati dalla Digital Transformation sono vasti e non impattano solo nelle aziende, ma anche nella società nel suo insieme. Le aziende devono adattarsi rapidamente per rimanere competitive in un ambiente in costante cambiamento, e ciò richiede una leadership visionaria e una cultura aziendale che favorisca l'innovazione e l'adattamento al cambiamento (Siebel [2019], Drahokoupil [2020]).

Gli indici di trasformazione digitale si stanno diffondendo per capire quanto siano preparati (o impreparati) gli amministratori delegati e le loro aziende, ne sono un esempio il Dell Digital Transformation Index e l'Industry Digitization Index del McKinsey Global Institute. Questi indici confermano che il divario tra le aziende e i settori che si sono trasformati digitalmente e quelli che non lo hanno fatto è già ampio e aumenterà in modo esponenziale. Secondo Roland Berger Strategy Consultants, l'Europa potrebbe aggiungere 1250 miliardi di euro di valore industriale lordo o perdere 605 miliardi di euro di valore entro il 2025. McKinsey afferma che, considerando solamente tre grandi aree di potenziale - le piattaforme di talento online, l'analisi dei big data e l'internet delle cose - si stima che la digitalizzazione potrebbe aggiungere fino a 2200 miliardi di dollari al PIL annuale entro il 2025. Recentemente sono emersi diversi strumenti e risorse per spingere le organizzazioni sulla strada della Digital Transformation: le aziende possono, ora, avvalersi di solide piattaforme di cloud computing, come Amazon Web Services, Microsoft Azure, IBM Watson e Google Cloud, per attivare le iniziative di trasformazione. Le società di consulenza per la trasformazione digitale sono in piena espansione; il mercato della consulenza sulla trasformazione digitale vale da solo 23 miliardi di dollari. Dalla nascita di Internet, il mercato della consulenza digitale si è evoluto con le successive ondate di digitalizzazione. Durante la prima ondata le società di consulenza hanno iniziato ad aiutare i clienti a costruire la loro presenza digitale. Successivamente, con il Web 2.0, esse si sono concentrate sull'esperienza del cliente. Oggi, nell'attuale ondata di trasformazione digitale, esse aiutano i clienti a utilizzare i dati per reinventare i loro modelli di business (Siebel [2019]). In Figura 2.3 è mostrato il potenziale impatto economico della Digital Transformation.

Increase in Global Business and Social Value	Timeframe	Source
\$100 Trillion	2016–2030	World Economic Forum, 2016
Increase in Annual Global GDP		
\$15.7 Trillion (driven by AI)	By 2030	PwC, 2017
\$13.0 Trillion (driven by AI)	By 2030	McKinsey, 2018
\$11.1 Trillion (driven by IoT)	By 2025	McKinsey, 2015
\$3.9 Trillion (driven by AI)	By 2022	Gartner, 2018

Figura 2.3: Potenziale impatto economico della Digital Transformation (Siebel [2019])

L'industria dell'automotive è uno dei settori più significativi e in continua evoluzione nel panorama economico globale. Questo settore non solo è un potente motore di crescita economica, ma ha anche un impatto profondo sulla società e sull'ambiente. Originariamente focalizzata sulla produzione di massa di veicoli per il trasporto personale e commerciale, l'industria automobilistica si è evoluta per incorporare innovazioni tecnologiche avanzate, rispondendo, così, a esigenze ambientali, sociali e di mobilità in continua trasformazione (Paul Nieuwenhuis [2003]).

La Digital Transformation in questo settore ha un ruolo cruciale; essa rappresenta non solo un'evoluzione tecnologica, ma anche un cambiamento culturale e strategico. La trasformazione digitale nel settore automobilistico può essere esaminata sotto diversi aspetti:

- *Innovazione nel prodotto:* l'industria dell'automotive sta vivendo una rivoluzione grazie all'introduzione di veicoli connessi, elettrici e autonomi. Questi sviluppi sono resi

possibili dall'integrazione di tecnologie avanzate come l'Intelligenza Artificiale, l'IoT e i sistemi di guida assistita. Queste innovazioni non solo migliorano l'esperienza di guida, ma contribuiscono anche a ridurre l'impatto ambientale dei veicoli.

- *Ottimizzazione della produzione*: la digitalizzazione ha portato all'automazione avanzata delle linee di produzione. La robotica, i sistemi di visione artificiale e la realtà aumentata sono utilizzati per aumentare l'efficienza, ridurre gli errori e personalizzare i prodotti in modo più efficiente. Ciò si traduce in una maggiore flessibilità e in una riduzione dei costi di produzione.
- *Gestione della logistica*: la raccolta e l'analisi dei Big Data permette una gestione più efficiente della catena di fornitura. Ciò consente alle aziende automobilistiche di rispondere più rapidamente alle variazioni della domanda del mercato e di gestire in modo più efficace le risorse.
- *Esperienza del cliente*: la digitalizzazione sta trasformando il modo in cui i clienti interagiscono con i marchi automobilistici, dalle vendite online ai configuratori digitali, dalla realtà virtuale negli showroom alle app per la gestione dei veicoli. La tecnologia sta migliorando l'esperienza di acquisto e di possesso.
- *Sostenibilità e mobilità urbana*: la trasformazione digitale sta anche guidando il settore verso soluzioni di mobilità più sostenibili. Ciò include lo sviluppo di veicoli elettrici, la gestione intelligente del traffico urbano e l'integrazione di veicoli con sistemi di energia rinnovabile (Paul Nieuwenhuis [2003], Llopis-Albert *et al.* [2021], Riasanow *et al.* [2017]).

Il presente lavoro di tesi si inserisce in questo contesto con lo scopo di contribuire alla Digital Transformation di uno dei principali player italiani nel settore dell'automotive, dove si è lavorato con il ruolo di consulente esterno.

2.3 Introduzione all'Enterprise Data Platform

I fattori appena elencati comportano una crescente complessità nel modello di business da adottare per riuscire a rimanere competitivi. Per questo motivo l'azienda in questione (cliente), ha deciso di affrontare le nuove sfide di mercato definendo un modello di dati integrato, lo IOM (Integrated Operating Model). Lo sviluppo dello IOM ha lo scopo di realizzare i seguenti temi:

- *High Value People*: evoluzione dei ruoli verso una maggiore collaborazione e un approccio orientato al valore.
- *Integrated Processes*: revisione dei processi per garantire una visione end-to-end interfunzionale.
- *Digital Platforms*: nuove risorse tecnologiche e dati capaci di potenziare le persone e i processi.

La definizione dello IOM è alla base della nuova piattaforma, l'Enterprise Data Platform (EDP), che si è ritenuta necessaria per raccogliere i dati in un unico "point of truth" basato sui sistemi cloud. Attraverso tale raccolta diventa possibile:

- abilitare una visione end-to-end dei dati;
- integrare i dati provenienti da fonti diverse e abilitare reportistica cross-funzionale;

- utilizzare modelli predittivi basati sugli stessi dati;
- tracciare ed identificare chiaramente il dato dalla sorgente fino al report finale;
- abilitare reportistica self-service.

L'implementazione di questa Enterprise Data Platform ha come obiettivo il completamento di processo di Digital Transformation, già iniziato in precedenza. Infatti, il cliente ha effettuato i seguenti step con l'ottica di raggiungere la trasformazione digitale:

- I. *Dati*: adozione di una nuova cultura data-driven per anticipare i problemi. Ciò è stato reso possibile attraverso la creazione di KPI aziendali inter-funzionali, analisi near real-time con i dati provenienti dalle fabbriche e data discovery, funzioni facilmente fruibili grazie alle soluzioni di front-end implementate.
- II. *Data Warehouse*: creazione di interazione tra il data warehouse, le fabbriche e le soluzioni on-cloud, integrando, con il massimo livello di dettaglio, i dati provenienti dai diversi sistemi alimentanti.
- III. *Amazon Web Services*: terzo ed ultimo step, che si integra con la costruzione della nuova Enterprise Data Platform. Si è voluto implementare un nuovo framework architetturale basato su AWS, re-ingegnerizzando l'esistente data warehouse per mantenerne, estendere e ottimizzare le funzionalità attraverso la nuova EDP. Infatti, il nuovo sistema consiste in una implementazione di una soluzione cloud-based che costituisce una piattaforma integrata per la raccolta, all'interno di un data lake, di tutte le informazioni di massimo dettaglio dei sistemi alimentati e consente di fornire una soluzione relazionale per la realizzazione di KPI aziendali interfunzionali leggibili dalla piattaforma di front-end QlikSense.

Una prima architettura ad alto livello, rappresentativa dell'Enterprise Data Platform è mostrata in Figura 2.4. Di seguito una descrizione di dettaglio dei principali componenti.

1. *Data Lake integrato e centralizzato*: qui i dati possono essere salvati e resi disponibili per il reporting e per le analisi avanzate.
2. *Enterprise Data Model*: è la struttura che comprende tutti i dati aziendali e i loro attributi, necessari a tutti i dipartimenti aziendali per le loro attività.
3. *Data Lineage*: è uno strumento necessario per tracciare e identificare i dati, a partire dalla fonte dei dati fino all'EDP, garantendo la coerenza e la completezza dei dati; tale strumento aiuta la gestione del dizionario dei dati che contiene tutte le informazioni relative agli stessi.
4. *Advanced Modelling & AI Tools*: permette ai data scientist di sviluppare algoritmi avanzati e modelli predittivi.
5. *Data Visualization Tool*: permette agli utenti aziendali di creare rapporti certificati e visualizzazioni di dati, sfruttando i dati presenti nella piattaforma.

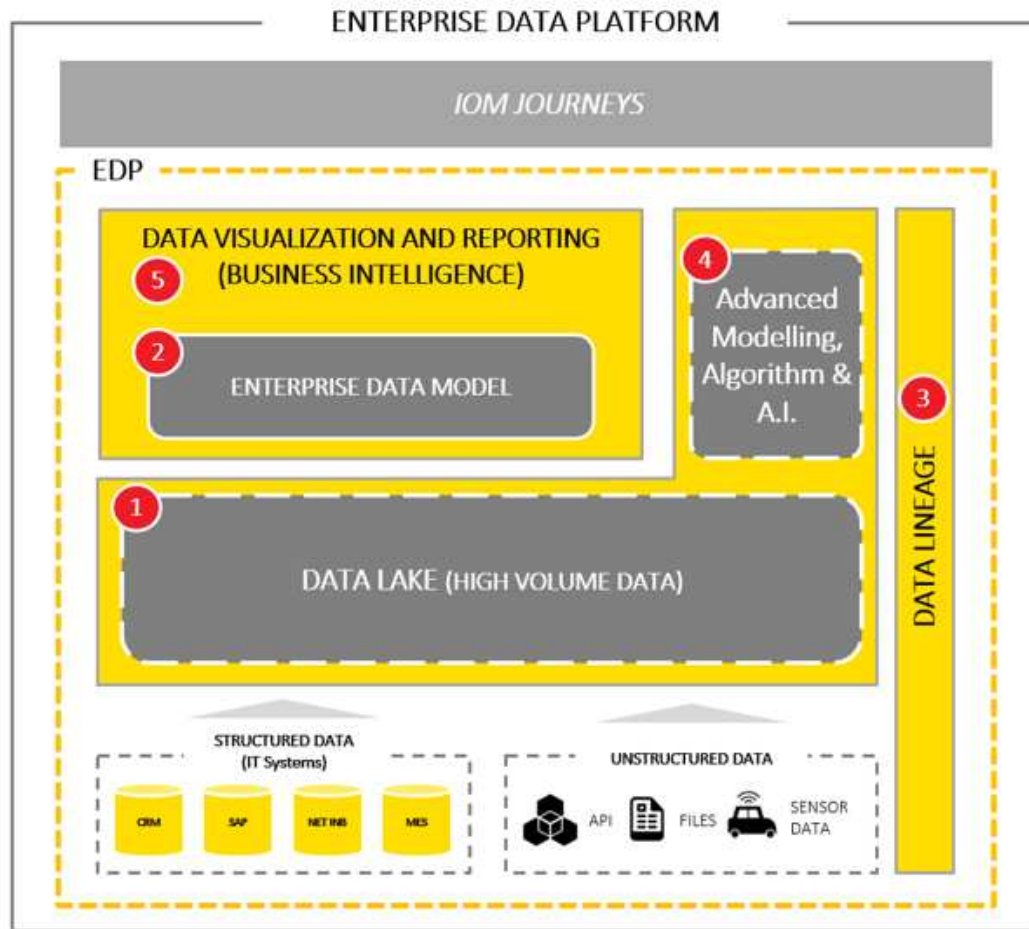


Figura 2.4: Architettura di alto livello dell'Enterprise Data Platform

2.4 Storia ed evoluzione del progetto

Come anticipato, inizialmente si prevedeva uno sviluppo del progetto con un approccio orientato al modello dei dati (Model Driven), dove un Integrated Operating Model avrebbe descritto tutte le entità di business da inserire nella piattaforma. Durante gli sviluppi ci si è resi conto che la complessità dei dati non era gestibile attraverso un unico modello integrato, per cui si è optato per un approccio Business Driven.

Attraverso questa metodologia si volevano porre gli utenti delle aree di business al centro degli sviluppi, per ottenere in output un risultato quanto più funzionale possibile alle loro analisi. Tale approccio prevedeva una ridefinizione del modello IOM attraverso interviste agli utenti chiave e, quindi, un'integrazione delle logiche già presenti nel data warehouse on premise. Anche questa linea di sviluppo ha fatto riscontrare notevoli problemi a causa di una raccolta di requisiti lenta e inconcludente.

Dunque, a novembre 2021 si è intrapresa una strada IT Driven, ovvero si è scelto di implementare l'Enterprise Data Platform con un approccio bottom-up, basato interamente sulla ricostruzione e integrazione delle logiche esistenti nel data warehouse già esistente. Una schematizzazione delle fasi è rappresentata in Figura 2.5.



Figura 2.5: Evoluzione del progetto

2.5 Requisiti tecnici di progetto

Per comprendere meglio il concetto di Enterprise Data Platform, nel seguito vengono esposti e mostrati dettagli tecnici relativi all’architettura dell’EDP. La Figura 2.6 mostra i moduli fondamentali della piattaforma¹, con una prospettiva di più alto livello. Si può notare come le sorgenti dati si suddividono principalmente tra fabbriche e sistemi relazionali (ad esempio, SAP). Tutti i dati vengono raccolti in un data lake implementato attraverso AWS. Successivamente si utilizza Redshift per la creazione di un modello di dati orientato alle esigenze del business; i dati vengono, quindi, mostrati agli utenti business attraverso la piattaforma di Business Intelligence Qlik oppure vengono demandati a sistemi esterni attraverso i cosiddetti flussi outbound (ad esempio, Salesforce).

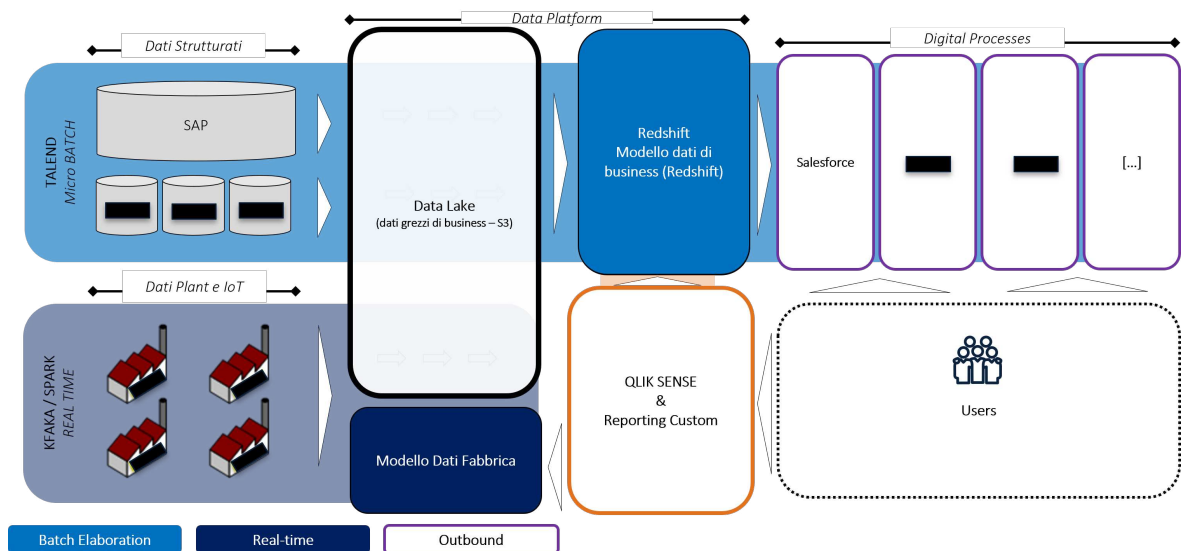


Figura 2.6: Architettura EDP di alto livello con specifica dei flussi outbound

Un’architettura più di dettaglio è riportata in Figura 2.7. Si evidenzia come la tipologia di sorgente dati influisca sulla modalità di raccolta degli stessi. Infatti, i dati provenienti dalle industrie vengono acquisiti in modalità real-time e vengono associati, quindi, allo "speed layer", mentre i dati provenienti da sistemi relazionali sono acquisiti in modalità "batch", cioè "a lotti". I flussi attraversati dai dati non sono gli stessi; infatti, i dati in real-time vengono raccolti ed elaborati attraverso il software Kafka, per poi, dopo varie elaborazioni, essere fruibili dagli utenti in output attraverso delle web app sviluppate con Angular. Questi dati vengono, in parte, anche demandati al data lake per analisi successive. I dati strutturati, invece, subiscono un processo di ETL attraverso il sistema Talend e vengono depositati nel data lake configurato attraverso i bucket di Amazon S3. Questo si compone di tre layer, ciascuno caratterizzato da un diverso grado di trasformazione del dato: da quello più grezzo a quello più raffinato. Come sopra accennato, il software Redshift entra in gioco per la creazione

¹Per motivi di riservatezza, alcuni dettagli nelle figure sono oscurati.

di modelli di dati a partire da ciò che è presente in S3. Infine, i dati sono visualizzabili a front-end grazie a software di Business Intelligence.

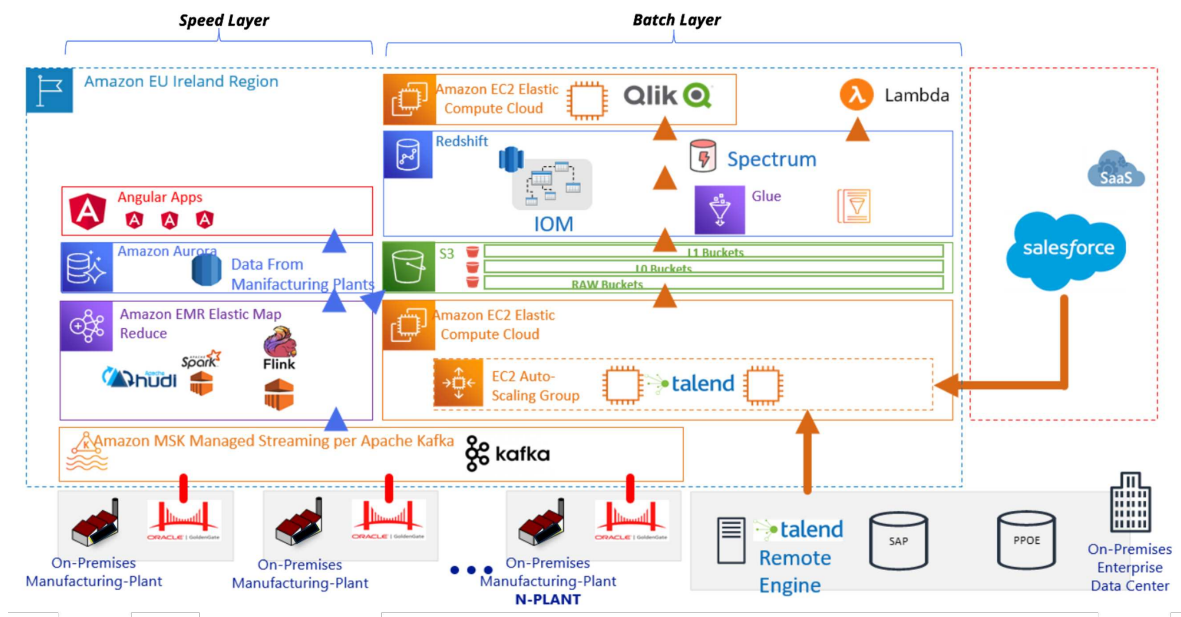


Figura 2.7: Architettura EDP di dettaglio

2.6 Attori in gioco e criticità di gestione

Inizialmente per la guida del progetto c'erano due attori principali in gioco, il team appartenente al cliente e un'azienda system integrator vincitrice del bando per lo sviluppo della piattaforma. Questo processo di Digital Transformation ha impattato fortemente sul team di gestione dei dati del cliente, che si è ritrovato ad essere un punto cardine per tutte le altre aree aziendali che necessitano dei dati forniti dalla piattaforma per gestire i relativi processi verticali; ciò ha generato, sin dal principio, tensioni e preoccupazioni. Dal lato del cliente i seguenti temi sono stati causa di problematiche interne:

- *Coinvolgimento degli utenti business:* per raggiungere gli obiettivi prefissati è necessario il coinvolgimento degli utenti business, che, però, inizialmente erano restii al cambiamento di infrastruttura.
- *Quadratura dei dati:* il vecchio DWH on premise, al momento dell'inizio della migrazione in Data Platform, era ancora un punto nevralgico delle attività business. Il tema principale della migrazione dei dati ruota attorno alla quadratura e al confronto con i dati contenuti nel DWH, considerato come punto di verità.
- *Dismissione del DWH:* era necessaria una dismissione incrementale del DWH sulla base delle aree di business già migrate in Data Platform; tuttavia nel frattempo, persone appartenenti ad altri team continuavano a sviluppare attività nel DWH, il che ha generato rallentamenti e confusione.
- *Knowledge cross-area:* il team di gestione dei dati inizialmente non aveva abbastanza conoscenza sull'implementazione di tutti i flussi, per cui il coordinamento e l'implementazioni ne hanno risentito.

D'altra parte, oltre a queste tematiche interne al cliente, nei primi anni di progetto sono state riscontrate problematiche anche lato system integrator. Infatti, i seguenti punti hanno provocato disagi e conflitti tra l'azienda di consulenza e il cliente:

- *Turn over*: il team del system integrator è stato caratterizzato da un elevato turn-over. Dopo circa due anni di progetto, soltanto quattro persone sono rimaste per tutto il tempo di sviluppo. Questo continuo cambio di personale ha causato una bassa fidelizzazione degli utenti business.
- *Sotto dimensionamento dello staff*: questo tema ha impattato sulla consapevolezza sull'effettivo carico delle attività e ha generato inefficienze sulla pianificazione a breve termine delle attività del team di sviluppo.
- *Deliverable di bassa qualità*: i primi deliverable sono stati valutati come insufficienti, il che ha costretto il team di sviluppo ad effettuare continui rework sui deliverable consegnati, generando, quindi, ritardi e rendendo obsoleta l'iniziale pianificazione del progetto.
- *Clima di tensione*: tali inefficienze hanno generato un clima di tensione e disaccordi tra il system integrator e il cliente, i quali si sono trovati ad accusarsi l'un l'altro delle problematiche, rendendone complessa la risoluzione.

A causa di questi continui conflitti tra i due principali attori, dopo due anni di progetto, nel 2021, l'azienda di consulenza BIP viene incaricata dal cliente di occuparsi della governance di progetto, e quindi di fungere da mediatore tra le parti in gioco. Il seguente lavoro di tesi si basa sul lavoro svolto da settembre 2023 a dicembre 2023; dunque, saranno descritte attività di governance sviluppate in concomitanza degli ultimi mesi di progetto. Infatti, si prevede la chiusura degli sviluppi dell'Enterprise Data Platform nei primi mesi del 2024.

Documentazione di progetto e gestione delle attività correlate

In questo capitolo si approfondisce il ruolo fondamentale della documentazione in un progetto IT. In particolare, si distinguono le varie tipologie di documentazione, si descrivono le best practice per la scrittura di una documentazione di qualità, si riportano i benefici dell'aver una solida documentazione a supporto del progetto e si discutono le sfide da affrontare per la creazione e la gestione di una documentazione esaustiva ed affidabile. Nella seconda parte del capitolo si espongono le attività svolte durante il periodo di tirocinio a supporto della raccolta e della gestione della documentazione di progetto. Innanzitutto, si è proceduto con il censimento e l'organizzazione della documentazione già esistente; prima di procedere con la fase di richiesta delle integrazioni e di revisione, ci si è soffermati sulla creazione di un template di analisi tecnica e sulla pianificazione delle attività. In seguito, è stato fondamentale attuare il controllo e il monitoraggio delle attività in corso. In conclusione, si è progettata una pagina web per facilitare l'accesso alla documentazione di progetto e, quindi, supportare la governance.

3.1 Il ruolo della documentazione

Nel contesto dei progetti IT, la documentazione riveste un ruolo cruciale, fungendo da ponte tra le diverse fasi di sviluppo, implementazione e manutenzione. Nonostante sia spesso sottovalutata, la documentazione è una componente essenziale per la riuscita e la sostenibilità di qualsiasi progetto tecnologico.

La documentazione di un progetto software è costituita da informazioni sia all'interno dei listati del codice sorgente sia al di fuori di essi, di solito sotto forma di documenti separati o di cartelle di sviluppo unitario. Nei grandi progetti formali, la maggior parte della documentazione è esterna al codice sorgente. La documentazione esterna tende a essere di alto livello rispetto al codice e di basso livello rispetto alla documentazione delle attività di definizione dei problemi, dei requisiti e dell'architettura (McConnell [2004]).

Lo sviluppo della documentazione di un sistema informativo è una questione complessa, sia dal punto di vista delle componenti che dovrebbe includere e del momento in cui dovrebbe essere redatta, che dal punto di vista della responsabilità delle persone incaricate di raggiungere questo risultato. Si potrebbe discutere a lungo sulla documentazione del sistema, e probabilmente il problema non sarà mai completamente risolto, poiché ci sono diversi fattori da considerare, tra cui il più importante è il fattore umano, poiché è quello che ha la maggiore influenza. Si può affrontare liberamente la questione della qualità della documentazione, dei mezzi automatici utilizzati per la sua stesura e il suo aggiornamento, del suo ruolo nella valutazione della qualità dei sistemi sviluppati, ma alla fine spetta al project manager, al team di sviluppo e al beneficiario stabilire quanto dettagliata, precisa

e ben strutturata dovrà essere la documentazione alla fine del progetto (Oprea e Mesnita [2006]).

I progetti IT che falliscono spesso trascurano l'importanza della documentazione adeguata; una documentazione carente o inaccurata può portare a malintesi, errori e ritardi, compromettendo il successo del progetto sia nel breve che nel lungo termine. Infatti, la documentazione del codice e dell'architettura del sistema è cruciale per la manutenzione e l'aggiornamento del software. Dunque, una documentazione ben strutturata e dettagliata è cruciale per il successo di un progetto IT, poiché migliora la comunicazione, mantiene gli standard di qualità, facilita la manutenzione e supporta la gestione dei rischi (Nasution e Weistroffer [2009]). In conclusione, la documentazione è un pilastro fondamentale nella gestione dei progetti IT. La sua accuratezza, chiarezza e tempestività influenzano direttamente la qualità del prodotto finale e la soddisfazione di chi ne fa uso. In questa sezione, si approfondisce il ruolo e l'importanza della documentazione, analizzando come essa possa essere gestita efficacemente per massimizzare il successo di un progetto IT.

3.1.1 Tipologie di documentazione

La documentazione tecnica in un progetto IT è cruciale per la comprensione, lo sviluppo, il supporto e la manutenzione del sistema software. Di seguito un elenco di tipologie di documentazione tecnica.

- *Documentazione dei requisiti*: specifica ciò che il sistema dovrebbe fare, dettagliando i requisiti funzionali e non funzionali. È cruciale per garantire che il prodotto finale soddisfi le esigenze degli utenti. La documentazione dei requisiti è un componente essenziale in qualsiasi progetto IT, fungendo da ponte tra le esigenze degli stakeholder e la realizzazione tecnica del progetto. Dunque, con documentazione dei requisiti si intende il processo di raccolta, analisi, documentazione e gestione delle esigenze e delle aspettative degli stakeholder riguardo a un nuovo prodotto o sistema. L'obiettivo è assicurarsi che il prodotto finito soddisfi le esigenze degli utenti, evitando incomprensioni e costosi cambiamenti in fasi successive dello sviluppo. I requisiti si dividono nelle seguenti tipologie:
 - *Requisiti funzionali*: descrivono cosa deve fare il sistema, dal punto di vista delle funzionalità, dei comportamenti, dei servizi e delle operazioni.
 - *Requisiti non funzionali*: riguardano le qualità del sistema, come prestazioni, sicurezza, affidabilità, usabilità, e standard di conformità.
 - *Requisiti di business*: esprimono gli obiettivi di business che il sistema deve soddisfare, inclusi aspetti legati al budget, al mercato e agli impatti organizzativi (Robertson e Robertson [2012]).
- *Architettura del sistema*: la documentazione di architettura del sistema in un progetto IT è un componente critico che descrive la struttura e il design complessivo del sistema. Questa fornisce una visione chiara dell'organizzazione strutturale del software, inclusi i componenti principali, le loro relazioni, e il modo in cui interagiscono per formare il sistema. Serve a garantire che tutti i membri del team, così come le parti interessate, abbiano una comprensione comune dell'architettura del sistema. Inoltre, risulta un aiuto fondamentale nella manutenzione, nella scalabilità e nell'aggiornamento del sistema. In questa documentazione si trovano i diagrammi architeturali, come diagrammi UML, diagrammi di sequenza e di classe, che illustrano visivamente la struttura del sistema. Inoltre, è importante avere spiegazioni dettagliate dei vari componenti del sistema, compresi i moduli software, i database, e i servizi esterni. Altre informazioni

essenziali sono i protocolli di interfaccia, i parametri di configurazione e la spiegazione dei modelli architettureali scelti Bass *et al.* [2012].

- *Documentazione del codice*: la documentazione del codice è una parte fondamentale dello sviluppo del software. Essa fornisce una spiegazione chiara del codice sorgente, facilitando la comprensione, la manutenzione e l'estensione del software da parte di altri sviluppatori. La documentazione del codice comprende commenti inseriti direttamente nel codice sorgente e documenti esterni che descrivono l'architettura e il funzionamento del software. Questo, dunque, rende il codice più comprensibile e accessibile, facilita il processo di debugging e manutenzione, e supporta la formazione dei nuovi sviluppatori. Una documentazione del codice ben eseguita è vitale per la longevità e il successo di un progetto software. Essa funge da guida e riferimento per gli sviluppatori, migliorando l'efficienza e la qualità del lavoro di sviluppo (Martin [2008]).
- *Documentazione di implementazione*: la documentazione di implementazione è un aspetto cruciale in qualsiasi progetto software, poiché fornisce le informazioni necessarie per installare, configurare, integrare e far funzionare correttamente il software in un ambiente di produzione. Essa assicura che il software sia installato e configurato correttamente, riducendo il rischio di errori e problemi operativi. Come best practice la documentazione di implementazione dovrebbe includere immagini e diagrammi per aiutare a spiegare passaggi complessi o configurazioni. Dunque, una buona documentazione di implementazione rende il processo di installazione e configurazione più efficiente e meno soggetto ad errori e aiuta a diagnosticare e risolvere rapidamente eventuali problemi che possono verificarsi durante l'installazione o la configurazione. Quindi, la documentazione di implementazione è un componente essenziale per il successo del software, specialmente in ambienti aziendali e tecnici complessi. Essa non solo guida l'utente attraverso l'installazione e la configurazione, ma fornisce, anche, un supporto vitale per la manutenzione e l'aggiornamento del software nel tempo. Investire tempo e risorse per creare una documentazione di implementazione chiara, accurata e completa è fondamentale per garantire che il software venga installato e utilizzato correttamente, massimizzando il suo valore e la sua efficienza (Bass *et al.* [2012]).
- *Documentazione di test*: la documentazione di test è un elemento essenziale nel processo di sviluppo del software, che si concentra sulla pianificazione, esecuzione e registrazione dei risultati dei test per garantire la qualità e l'affidabilità del prodotto software. Questa include una serie di documenti che guidano e registrano il processo di testing del software con lo scopo di garantire che il software funzioni come previsto, di identificare bug e problemi, e di assicurare che i requisiti del software siano soddisfatti. In particolare, la documentazione di test potrebbe contenere i seguenti documenti:
 - *Piano di test*: un documento che definisce la strategia e l'approccio generale ai test, inclusi obiettivi, risorse, calendario e metodi di testing.
 - *Casi di test*: specifiche dettagliate per ciascun test, comprese le condizioni, gli input, l'azione da eseguire e l'esito previsto.
 - *Script di test*: istruzioni passo-passo per eseguire un test specifico, spesso utilizzate nei test automatizzati.
 - *Report di test*: documentazione dei risultati dei test; include successi, fallimenti, problemi riscontrati e comportamenti imprevisti del software.
 - *Registro dei bug*: un elenco dei bug identificati durante i test, con dettagli sul problema, la gravità, lo stato e le note sulla risoluzione (Jorgensen [2013]).

3.1.2 Best practice

Le best practice per la documentazione di progetti IT sono essenziali per garantire che la documentazione sia utile, accessibile e aggiornata. Queste pratiche sono vitali per una comunicazione efficace all'interno del team e per il successo complessivo del progetto. Di seguito un elenco delle best practice più importanti:

- *Chiarezza e concisione*: la documentazione dovrebbe essere chiara, diretta e priva di ambiguità. Dovrebbe evitare gergo complesso quando possibile e fornire una spiegazione approfondita dei termini tecnici che si impiegano.
- *Coerenza e standardizzazione*: nello stilare la documentazione si dovrebbero utilizzare formati, stili e terminologie coerenti. Tali accortezze aiutano a mantenere la documentazione ordinata e facile da seguire.
- *Aggiornamento e manutenzione*: la documentazione dovrebbe essere mantenuta aggiornata con le ultime modifiche del progetto, altrimenti questa diventerebbe presto obsoleta.
- *Accessibilità e organizzazione*: è importante rendere la documentazione facilmente accessibile a tutti i membri del team utilizzando sistemi di archiviazione condivisi e organizzarla in modo da rendere la navigazione chiara e veloce.
- *Includere esempi e casi d'uso*: dove possibile, è consigliato includere esempi pratici e casi d'uso, così da aiutare a chiarire concetti complessi.
- *Feedback e revisione continua*: è una norma incoraggiare il team a condividere feedback sulla documentazione ed essere disponibili a modificarla in base alle esigenze e alle evoluzioni del progetto.
- *Utilizzare strumenti e tecnologie adeguate*: è consigliato sfruttare strumenti e tecnologie moderne per la creazione e la condivisione della documentazione, come software di documentazione collaborativa o piattaforme di gestione dei documenti. Ad esempio, grazie alle tecnologie cloud, il team potrebbe lavorare sugli stessi documenti condivisi in real-time; ciò facilita sia la creazione che la revisione e l'aggiornamento della documentazione.

Seguendo queste best practice, la documentazione di un progetto IT può diventare uno strumento efficace per la gestione del progetto, la comunicazione all'interno del team e il successo a lungo termine del progetto (Shore e Warden [2008]).

3.1.3 Benefici della documentazione

Di seguito si ricapitolano i principali benefici che una documentazione di progetto ben redatta e correttamente gestita può portare al team e al progetto stesso.

- *Miglioramento della comunicazione*: una documentazione completa e chiara facilita la comunicazione tra team di sviluppo, stakeholder e utenti.
- *Incremento della qualità del prodotto finale*: una documentazione esaustiva e facilmente accessibile da tutti riduce significativamente errori ed incomprensioni, impattando, quindi, nella qualità dei deliverable.
- *Facilitazione della gestione del progetto*: la documentazione di progetto supporta la pianificazione e l'esecuzione del progetto, aiutando, in particolare, nella tracciabilità e nella gestione dei requisiti.

- *Supporto alla manutenzione e agli aggiornamenti del sistema:* per ogni tipologia di manutenzione e per eventuali miglioramenti da apportare al sistema, una documentazione correttamente redatta risulta essere un punto di partenza fondamentale per gli sviluppi. Senza tale documentazione risulterebbe molto più oneroso comprendere dove e come intervenire nel sistema.
- *Miglioramento dell'onboarding e della formazione:* è fondamentale possedere una documentazione di riferimento per gli sviluppi implementati per agevolare il processo di onboarding e di formazione di eventuali nuovi membri nel team.

In conclusione, la documentazione di progetto risulta un supporto essenziale per tutte le attività del team, dagli sviluppi fino alla manutenzione successiva del sistema. Sarà compito del project manager assicurarsi che venga redatta documentazione esaustiva e che questa sia accessibile a tutti i componenti del team (PMI [2021], April [2008]).

3.1.4 Sfide della creazione e gestione della documentazione

Il tema della necessità di una documentazione esaustiva in un progetto IT è contrastato dalle difficoltà nel realizzarla, in quanto il processo risulta complesso e critico. Le sfide che più impattano la creazione di una documentazione esaustiva ed aggiornata sono le seguenti:

- *Limitazioni di tempo e risorse:* spesso, i team di progetto sono sotto pressione per consegnare risultati rapidi, lasciando poco tempo per la scrittura di una documentazione approfondita.
- *Mancanza di competenze o formazione:* la mancanza di competenze specifiche nella scrittura tecnica può portare a documentazione di bassa qualità.
- *Modifiche continue nel progetto:* i progetti IT sono dinamici, con frequenti cambiamenti; ciò rende difficile mantenere la documentazione aggiornata.
- *Percezione del valore:* a volte la documentazione viene vista come un compito secondario rispetto allo sviluppo del software, per cui non si allocano risorse adeguate per la sua stesura.
- *Sovraccarico di informazioni:* trovare il giusto equilibrio tra completezza e sovraccarico di informazioni non è un compito semplice. Bilanciare questi due temi è essenziale per ottenere una documentazione facilmente fruibile e, allo stesso tempo, esaustiva in tutte le sue parti.

Le seguenti accortezze possono essere attuate per superare le criticità appena elencate e, quindi, raggiungere l'obiettivo di ottenere una documentazione affidabile a supporto di tutto il team:

- *Pianificazione della documentazione:* integrare l'attività di creazione della documentazione nella pianificazione del progetto sin dall'inizio.
- *Formazione e strumenti:* fornire formazione adeguata e strumenti efficienti per la scrittura della documentazione ai membri del team.
- *Revisione e aggiornamento Continuo:* stabilire processi per la revisione regolare e l'aggiornamento della documentazione.
- *Valutazione del valore della documentazione:* sensibilizzare il team sull'importanza della documentazione e sul suo impatto sul successo del progetto a lungo termine.

- *Delega e collaborazione*: coinvolgere tutti i membri del team nel processo di documentazione per distribuire il carico di lavoro e migliorare la completezza delle informazioni.
- *Utilizzo di standard di documentazione*: adottare standard e template per mantenere la coerenza e la chiarezza della documentazione.

Dunque, una documentazione completa ed esaustiva è, indubbiamente, essenziale per il successo di un progetto IT, ma le sfide nel suo sviluppo sono reali e significative. Superare queste sfide richiede un impegno cosciente, risorse dedicate e una strategia ben pianificata. Riconoscendo l'importanza della documentazione e affrontando proattivamente queste sfide, le organizzazioni possono migliorare significativamente l'efficacia e l'efficienza dei loro progetti IT (Clements [2003]).

3.2 Attività svolte per la raccolta e la gestione della documentazione

Il lavoro di tesi si inserisce nel progetto di Data Platform nei mesi conclusivi di sviluppo, per cui, in questa fase cruciale, è stata ritenuta fondamentale l'attività di raccolta e organizzazione della documentazione tecnica. Ci si è focalizzati sul fornire al cliente tutte le risorse, le conoscenze e gli strumenti indispensabili per poter gestire e mantenere in modo efficace e autonomo il sistema implementato, anche successivamente al distacco dal progetto del team di sviluppo afferente all'azienda di system integrator. Possedere una documentazione chiara, completa e facilmente accessibile è un fattore necessario per garantire una transizione fluida per la gestione e il mantenimento del sistema e, quindi, per mantenere un'efficacia operativa.

3.2.1 Censimento e organizzazione della documentazione

Come già evidenziato, la documentazione per essere efficace deve poter essere accessibile facilmente da tutto il team. Ciò si traduce nell'utilizzare strumenti idonei e nel salvare i file con un criterio lineare e coerente. Tra gli strumenti a disposizione, si è ritenuto più idoneo selezionare la suite di Microsoft Office per la memorizzazione del materiale. In particolare, si è creato un canale dedicato alla documentazione nel team di progetto in Microsoft Teams. Il path scelto risulta, così, facilmente accessibile da tutti gli utenti abilitati sia da Microsoft Teams che dal corrispondente sito nel dominio Microsoft SharePoint del cliente. Una volta stabilito che repository utilizzare, si è definito come organizzare i file di progetto. Vista la propensione business oriented delle attività del team di Data Management, si è optato per un primo livello di gerarchia corrispondente alle aree di business, un secondo livello gerarchico corrispondente agli stream Qlik afferenti alle corrispondenti aree di business ed un terzo livello gerarchico corrispondente alle applicazioni Qlik predisposte per la reportistica. Per stream particolarmente complessi, come quelli dell'area Supply Chain, si è scelto di utilizzare una suddivisione su quattro livelli.

Prima di spostare tutto il materiale già a disposizione del cliente, è stata necessaria effettuare una fase di censimento della documentazione. Per strutturare questo processo si è scelto di utilizzare il tool Microsoft Excel così da poter tracciare un elenco delle applicazioni di cui dover raccogliere materiale documentativo (si riporta un estratto di esempio in Figura 3.1) ed arricchire ogni voce con informazioni essenziali per una buona governance. In particolare, sono state tenute in considerazione i seguenti dettagli per ogni app:

- *Fornitore*: quale azienda si è occupata degli sviluppi dell'applicazione.
- *Perimetro*: indica se l'applicazione di riferimento è inclusa nel contratto di Data Platform.

- *Referenti*: chi sono le persone di riferimento all'interno del team del cliente.
- *Approvazione*: voce per indicare se la documentazione disponibile fosse ritenuta dal cliente sufficiente ed esaustiva.
- *Documentazione tecnica*: campo utile ad indicare se la documentazione fosse stata già rilasciata ed eventualmente a che link fosse disponibile.

Hier 1	Hier 2	Hier 3	Hier 4
Purchase	Purchase KPI	Purchase spending	
Purchase	Purchase KPI	Purchase spending with GR	
Purchase	Purchase Raw Material	Free analysis	
Purchase	Purchase Raw Material	Report 200/300	
Purchase	Vendor Rating		
Supply Chain	Kpi	NetKpi	stock ageing
Supply Chain	Kpi	NetKpi	stock (by factory)
Supply Chain	Kpi	NetKpi	overstock
Commercial	Repmarketing	Forecast accuracy	
Commercial	Repmarketing	Monitoring orders	
Commercial	Repmarketing	Portfolio Order Analysis	

Figura 3.1: Estratto di censimento della documentazione

3.2.2 Definizione di un template

Dopo la fase di censimento della documentazione già esistente, si è proceduto con una prima fase di revisione interna per valutare la chiarezza e la completezza del materiale già rilasciato. Nel controllare tale documentazione ci si è resi conto che questa non risultava essere uniforme e che spesso risultava carente di dettagli. Per questo motivo si è deciso di stilare un template di analisi tecnica (ATE) con un duplice obiettivo: fornire al system integrator un template da compilare per le varie aree così da ottenere una documentazione il più possibile uniforme, e fungere, allo stesso tempo, da checklist esaustiva per il team interno in modo da facilitare il controllo dei temi trattati in ogni file. Si è scelto di inserire nel template (Figura 3.2) sia una sezione dedicata al back-end che una dedicata al front-end, così da ottenere una descrizione completa del flusso dei dati relativi all'applicazione di riferimento, riportando il lineage del dato dall'estrazione fino all'utilizzo finale per la creazione di KPI. La sezione "Glossary" del template è stata preposta per la descrizione dei termini tecnici al fine ottenere i seguenti vantaggi:

- chiarezza del contenuto;
- facilità di comprensione;
- coerenza del linguaggio;
- risparmio di tempo per i lettori;
- aumento dell'accessibilità;
- prevenzione di errori causati da incomprensioni.

Infine, la sezione "Annex" è dedicata all'inserimento di eventuali file allegati, infatti, spesso, le logiche dei flussi Talend sono complesse da riportare interamente in modo discorsivo, per cui, per completezza, è necessario riportare anche file dedicati che riescono a dettagliare interamente i processi.

Table of Contents

1.	SCOPO E AMBITO	3
1.1	SCOPO	4
1.2	AMBITO DI INTERVENTO	4
2	BACK-END	4
2.1	FREQUENZA E LOGICHE DI ESTRAZIONE	5
2.2	GESTIONE INITIAL LOAD	5
2.3	LO-RAW	5
2.4	TALEND	5
2.4.1	ANALISI DELLE DIPENDENZE	5
2.4.2	JOB EXECUTION ANALYSIS	5
2.4.3	ESECUZIONE MANUALE DEL JOB	5
2.5	L1	5
3	FRONT-END	5
3.1	ACCESSO	6
3.2	ETL QLIK	6
3.2.1	GRANULARITÀ APPLICAZIONE	6
3.2.2	PROFONDITÀ STORICA	6
3.2.3	SORGENTI	6
3.2.4	LOGICHE DI FILTRO E AGGREGAZIONE	6
3.3	SHEET	6
3.4	ELENCO KPI	6
3.5	SECTION ACCESS	6
3.6	REFRESH	7
3.7	DATA MODEL - PRESENTATION	7
3.8	MASTER ITEM	7
3.9	REPORT NPRINTING	7
3.9.1	DATA LOAD AND REFRESH	8
3.9.2	ACCESS	8
3.9.3	NPRINTING APPS AND CONNECTION	8
3.9.4	REPORT	8
3.9.5	REPORT FILTER	8
4	GLOSSARY	8
5	ANNEX	8

Figura 3.2: Template dell'analisi tecnica: indice

Nel template è stata inserita, inoltre, una sezione dedicata alla governance del documento (Figura 3.3); essa richiede di annotare, per ogni ATE, da chi è stato redatto, da chi è stato

revisionato, il numero di versione e un eventuale cronologia di versioni.

Document	EDP ATE
Prepared by	
Reviewed by	
Approved by	
Version	
Date	

Document history

Date	Version #	Author	Description

Figura 3.3: Template dell'analisi tecnica: gestione del versioning

3.2.3 Pianificazione attività

La documentazione censita è risultata non esaustiva e totalmente assente per alcune aree; per questo è stato necessario prevedere una fase di raccolta della documentazione mancante. Ci si è posti, quindi, come intermediari tra il cliente e il system integrator al fine gestire i rilasci delle integrazioni e le corrispettive revisioni da parte del team interno. Per poter effettuare una governance ottimale delle suddette fasi, si è proceduto pianificando con il diagramma di Gantt la raccolta della documentazione per area: in Figura 3.5 si riporta, a titolo esemplificativo, una parte delle aree pianificate. Si è scelto di riportare non solo la stima per la conclusione dell'attività, ma anche uno stato di avanzamento suddiviso tra i seguenti stati:

- 0%: per indicare che non è stata svolta alcuna attività per l'area di riferimento;
- 25%: la documentazione già presente è stata revisionata dal team interno del cliente;
- 50%: sono in corso le integrazioni lato system integrator e le relative revisioni lato cliente;
- 75%: il principale fornitore (system integrator) ha rilasciato tutta la documentazione e questa è stata confermata dal cliente, mancando dei documenti afferenti a fornitori differenti;
- 100%: la documentazione per tutte le voci dell'area è stata ritenuta sufficiente ed esaustiva.

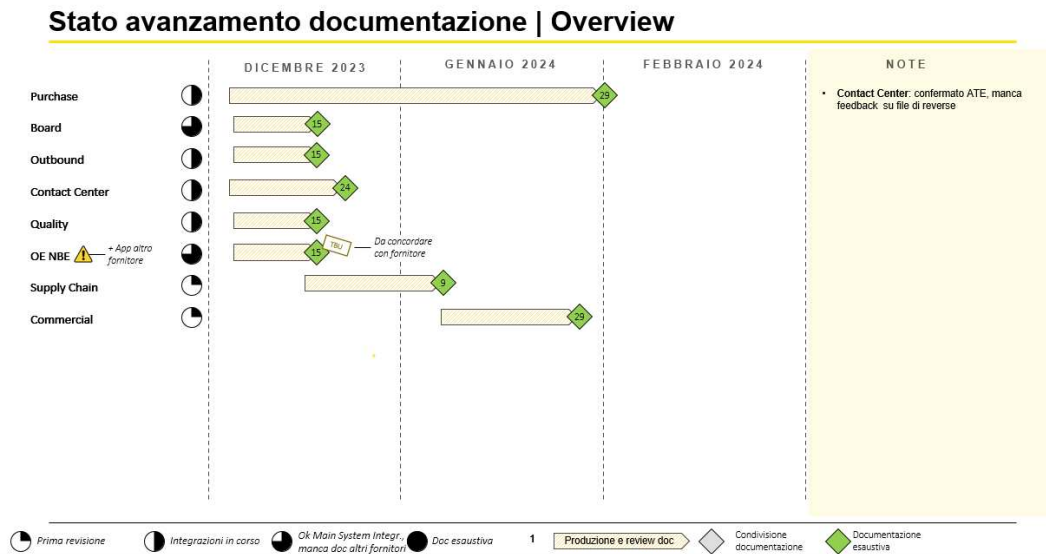


Figura 3.4: Pianificazione delle attività di raccolta e revisione della documentazione

Si noti che si sono prodotti dei diagrammi di Gantt anche ad un livello di dettaglio maggiore, per poter controllare lo stato di avanzamento anche a livello di app, ma, per motivi di riservatezza, questi non possono essere riportati nella tesi.

La fase di revisione della documentazione si è ritenuta fondamentale per poter approvare le analisi tecniche rilasciate, in quanto, una volta finito il progetto, il system integrator che si è occupato degli sviluppi si distaccherà dalla gestione della Data Platform e tutte le attività future di aggiornamento e manutenzione saranno in carico al team interno del cliente, il quale avrà la documentazione richiesta come punto di riferimento per modificare il sistema.

3.2.4 Controllo e monitoraggio dello stato avanzamento

Seguire le attività di raccolta e revisione della documentazione ha significato fungere da intermediario tra il fornitore ed il cliente e, quindi, gestire molteplici flussi di informazione. Per poter controllare le attività su più fronti, ovvero su aree differenti, si è implementato un tracker finalizzato al tracciamento dei punti aperti (Figura 3.5). Per ogni voce del tracker si è scelto di tenere traccia dei seguenti dettagli:

- *Attività*: descrizione dell'attività da tracciare.
- *Area*: riferimento all'area di business a cui appartiene l'applicazione per cui si è aperto il punto.
- *Stream/App*: applicazione o stream di riferimento.
- *Status*: stato di avanzamento del task; potrebbe essere, ad esempio, "Pending" o "Closed".
- *Owner*: persona che deve portare a compimento l'attività.
- *Due Date*: scadenza prefissata.
- *Note*: campo per eventuali annotazioni.

Attività	Area	Stream/App	Status	Owner	Due Date	Note
Integrazione doc con vecchia documentazione: fare proposta	Board	monthly - daily cockpit	Pending			Integrazioni vecchia doc non sono corrisposte da
Atteso feedback doc contact center	Contact Center		Pending		13/12/2023	Riscontro positivo per ATE, atteso feedback per file di reverse.
Attesa data per primo feedback	Outbound	Cross	Closed			Feedback entro 22/12/2023

Figura 3.5: Creazione di un tracker per il tracciamento degli open point

3.2.5 Creazione di una pagina web per la governance di progetto

In parallelo al lavoro di gestione della raccolta e revisione della documentazione, si è pensato che potesse essere utile la creazione di una pagina web per facilitare l'accesso alla documentazione. Si è scelto di utilizzare il servizio Microsoft Sharepoint per la creazione e la personalizzazione di una pagina attraverso moduli dedicati. La nuova pagina web è stata associata al team Teams di Data Management, cosicché solo le persone già autorizzate potessero avere un accesso diretto.

La pagina non ha come unico obiettivo l'accessibilità della documentazione di progetto, ma è stata progettata anche per poter fungere come punto di riferimento per la governance del progetto stesso. Infatti, oltre ai moduli dedicati all'accesso e alla navigazione dei file, sono state inserite delle sezioni predisposte alla consultazione delle best practice definite per le procedure di sviluppo e dei link rapidi.

In Figura 3.6 si riporta uno screen dimostrativo della pagina web realizzata. Oltre ai moduli visualizzabili, è presente una sezione per la navigazione delle cartelle embedded, un modulo per visualizzare i file recenti utilizzati dal team, altri due moduli per la visualizzazione dei propri file recenti e dei contenuti contrassegnati con un bookmark, ed, infine, una sezione dedicata ai contatti, cioè alle persone di riferimento per eventuali chiarimenti o consigli per la gestione della pagina stessa.

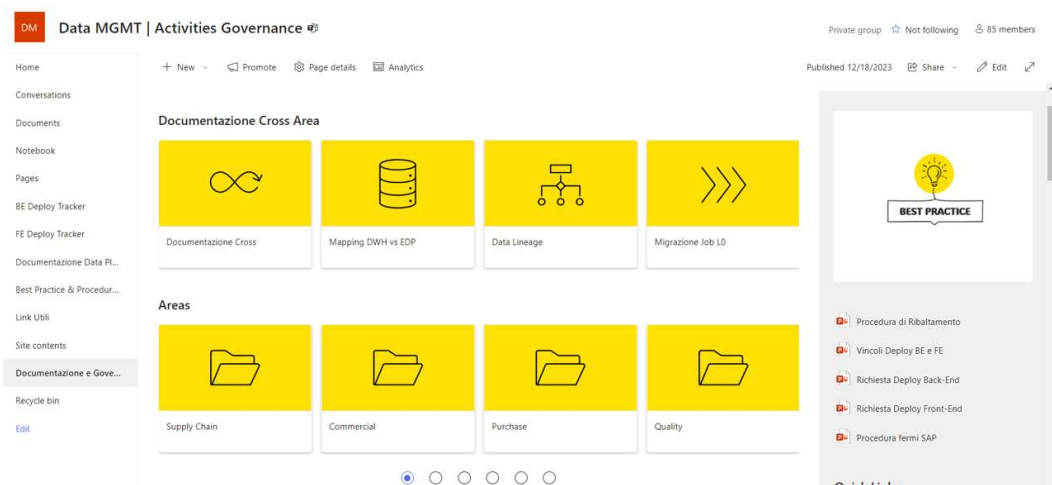


Figura 3.6: Pagina web per la governance di progetto

Supporto operativo al team dedicato ai servizi di gestione delle applicazioni

In questo capitolo si introduce il concetto di Application Management Service, un servizio cruciale per gestire e ottimizzare le applicazioni software nelle organizzazioni. Vengono, quindi, descritte in dettaglio le attività seguite durante il periodo di tirocinio a supporto del team di AMS. In prima istanza si descrive l'esperienza di definizione di un processo di deploy, con un focus particolare sulle fasi che lo caratterizzano. Dopodiché, si approfondisce il meccanismo di ticketing, enfatizzando l'importanza di una sua gestione efficiente al fine di migliorare il servizio agli utenti e l'operatività. Infine, il capitolo esplora il tracciamento degli issue e l'analisi delle root cause, processo fondamentale per identificare e risolvere i problemi del software, migliorando, così, la qualità del prodotto. Questo approccio critico e strutturato garantisce robustezza e solidità della Data Platform.

4.1 Introduzione all'Application Management Service (AMS)

L'Application Management Service (AMS) è un insieme di servizi e soluzioni forniti per gestire e ottimizzare le applicazioni software di un'organizzazione. Include attività come il mantenimento dell'applicazione, il supporto tecnico, l'aggiornamento e l'adattamento alle nuove esigenze aziendali. Questo servizio mira a garantire che le applicazioni funzionino in modo efficiente, affidabile e sicuro durante l'intero ciclo di vita, dalla progettazione alla dismissione. Inoltre, l'AMS si concentra sulla soddisfazione degli utenti finali, assicurando che le applicazioni soddisfino le loro esigenze e aspettative. L'evoluzione dell'AMS è strettamente legata agli sviluppi tecnologici e alle esigenze aziendali in cambiamento. Inizialmente, l'AMS si concentrava sulla manutenzione e il supporto di base delle applicazioni. Con l'avvento del cloud computing e delle tecnologie emergenti, l'AMS si è trasformato per offrire servizi più complessi e integrati, supportando la digitalizzazione delle aziende e l'integrazione di nuove soluzioni tecnologiche. L'AMS è vitale per le organizzazioni perché garantisce che le applicazioni aziendali siano affidabili, efficienti e sicure. Un AMS efficace riduce i tempi di inattività, migliora le prestazioni delle applicazioni e garantisce che le soluzioni software siano sempre allineate con le strategie aziendali e le normative del settore. Questo non solo riduce i costi operativi ma migliora anche la produttività e la competitività dell'azienda.

In dettaglio, gli obiettivi principali dell'AMS includono:

- *Migliorare la disponibilità dell'applicazione:* l'AMS si preoccupa di mantenere le applicazioni disponibili in modo continuativo, minimizzando eventuali interruzioni o downtime non pianificati.

- *Ottimizzare le prestazioni*: l'AMS mira a garantire che le applicazioni funzionino con le migliori prestazioni possibili, riducendo i tempi di risposta e ottimizzando l'efficienza delle risorse.
- *Assicurare la sicurezza*: l'AMS implementa misure di sicurezza per proteggere le applicazioni da minacce esterne e interne, garantendo la confidenzialità, l'integrità e la disponibilità dei dati.
- *Ridurre i costi operativi*: l'AMS cerca di ridurre i costi di gestione delle applicazioni attraverso l'automazione dei processi, la standardizzazione e l'ottimizzazione delle risorse.

In conclusione, si è evidenziata l'importanza dell'Application Management Service (AMS) nel panorama IT attuale. L'AMS non è solo un mezzo per mantenere e gestire le applicazioni software; esso è diventato un catalizzatore per l'innovazione aziendale, l'efficienza operativa e la competitività sul mercato. Attraverso la sua evoluzione, l'AMS si è adattato per rispondere alle sfide emergenti e sfruttare le opportunità presentate dalle nuove tecnologie (Györy *et al.* [2012], Brewster *et al.* [2012]).

4.2 Strutturazione del processo di deploy front-end

Durante il lavoro di tirocinio si sono svolte attività a supporto del team di AMS del cliente. Innanzitutto, si è contribuito alla strutturazione di un processo per la gestione dei deploy front-end, dove gli attori in gioco sono gli sviluppatori che richiedono un determinato rilascio e il team di AMS che provvede ad effettuare il deploy. Si è reso necessario un supporto a livello di governance per questo processo al fine di riuscire a tracciare in modo organico ed uniforme tutte le richieste di deploy ricevute ed elaborate dal team di AMS. Si è pensato, quindi, di inserire una fase intermedia tra la ricezione della richiesta di deploy e l'elaborazione della stessa dedicata al tracciamento delle attività. Dunque, per il processo di deploy si distinguono le seguenti tre fasi:

- I. *Richiesta di deploy*: gli sviluppatori, appartenenti sia al team interno IT del cliente, che a team di fornitori esterni, non possono effettuare direttamente rilasci a livello di front-end della Data Platform, ma devono richiedere il deploy al team di AMS che si occupa della gestione e manutenzione della piattaforma. Per uniformare le richieste di deploy, è stato definito un template (Figura 4.1) con una lista di voci necessarie per la pubblicazione.
- II. *Tracking della richiesta di deploy*: per questa nuova fase si è implementato un tracker attraverso lo strumento Microsoft List, così da fornire al team di AMS la possibilità di tracciare le richieste di deploy ricevute. In Figura 4.2 si possono visualizzare tutte le informazioni utili alla descrizione del rilascio ricevuto. Ogni richiesta sarà caratterizzata da:
 - ID;
 - Titolo;
 - Stream;
 - Soggetto richiedente;
 - Soggetto Esecutore;
 - Descrizione.

Una volta inseriti tutti i dettagli, la richiesta di deploy verrà inserita nel tracker (Figura 4.3).

Deploy Information	Request type	
	Source stream	
	Source app	
	Source app ID	
	Destination Stream	
	Destination App Name	
	Overwrite app with the same name?	
	Date and Time to deploy	
	Need reload before deploy?	
	Need KT?	
	Required by	
Request reason or version Information	Version number	
	Publish reason	
	Change with previous release	
App Information	Estimated app file size in Mb	
	Estimated app ram footprint in Mb	
	Total record count	
	Refresh frequency	
	Refresh duration	
User Information	Number of End Users	
	Estimated Concurrent users	
Dependencies	List of database connection used by the app	
	List of database tables read by the app	
	List of QVD files read by the app	
	Binary ID	
	Binary App Name	
	Binary App Stream	
ETL Information	Task Trigger (scheduled or event)	
	Table list for event trigger	
Reference	SI key person	
	IT key person	
	Business key person	
Additional Information	Provided here any additional information you want to share	

Figura 4.1: Template per le richieste di deploy front-end

III. *Effettuazione del deploy:* la fase effettiva di rilascio è portata a termine dal team di AMS.

In ultimo, sono stati condivisi con tutti i fornitori dei vincoli temporali per le richieste di rilascio. In caso di pubblicazione di nuove applicazioni front-end, la richiesta al team di AMS dovrà essere effettuata con tre settimane di anticipo rispetto alla data di deploy. Due settimane prima di questo sarà necessario fissare il periodo di Knowledge Transfer, per aiutare il team

Save Cancel Copy link

New Item

Ticket Richiesta Deploy

Enter value here

Ticket generato a valle della richiesta di deploy

Stream

—

Stream a cui afferisce l'applicazione pubblicata.

Title

Enter value here

Richiedente

Enter a name or email address

Nome della persona che ha richiesto la pubblicazione dell'applicazione.

Cambiamenti Versione

Enter value here

Descrizione dei cambiamenti effettuati con questa nuova versione dell'app.

Esecutore

Enter a name or email address

Nome della persona che ha eseguito il deploy in produzione

Attachments

Add attachments

Save Cancel

Figura 4.2: Inserimento nel tracker di una nuova pubblicazione front-end

di AMS nella gestione della nuova applicazione, e questo dovrà essere espletato entro una settimana prima del rilascio. Nel caso in cui le richieste di deploy front-end coinvolgano solo piccole modifiche, allora potranno essere effettuate fino a tre giorni prima della data di deploy, e la fase di Knowledge Transfer potrà essere ultimata fino ad una settimana dopo del rilascio. Per chiarezza, si riporta in Figura 4.4 una schematizzazione dei vincoli definiti.

4.3 Monitoraggio dei ticket

Nel campo dell'Application Management Service (AMS), la gestione efficiente dei ticket è un aspetto cruciale. Essa rappresenta il meccanismo primario attraverso il quale le richieste di supporto, gli errori, le richieste di funzionalità e gli altri compiti di manutenzione delle

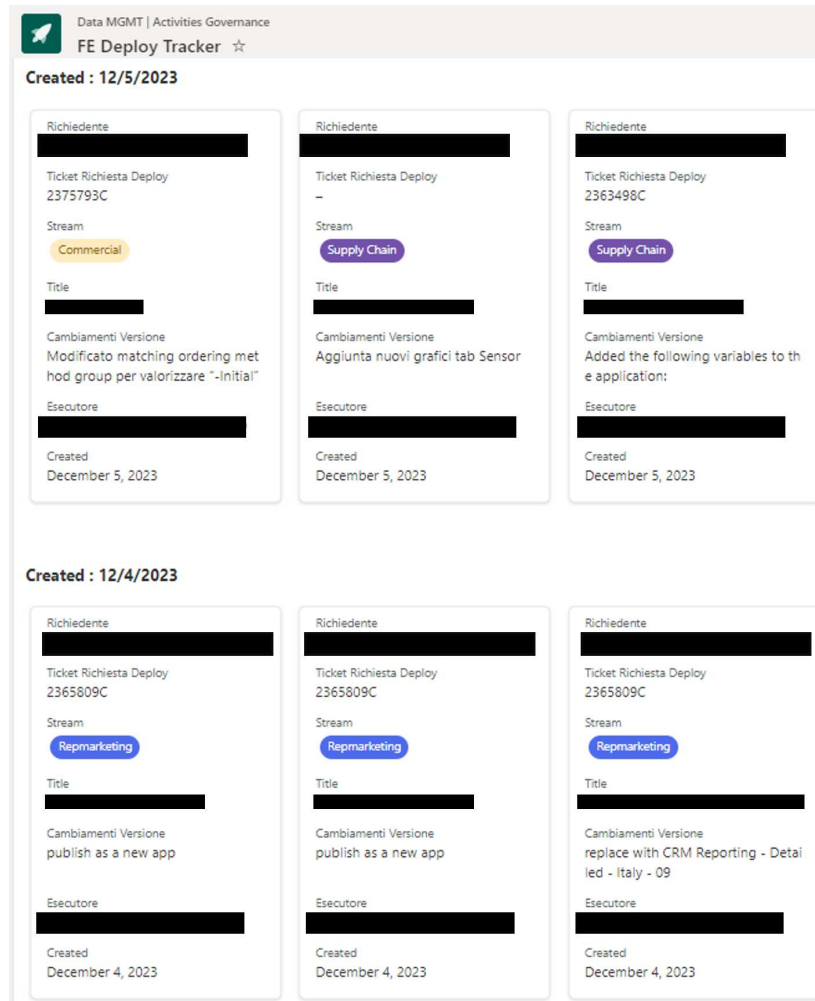


Figura 4.3: Tracker per i deploy front-end

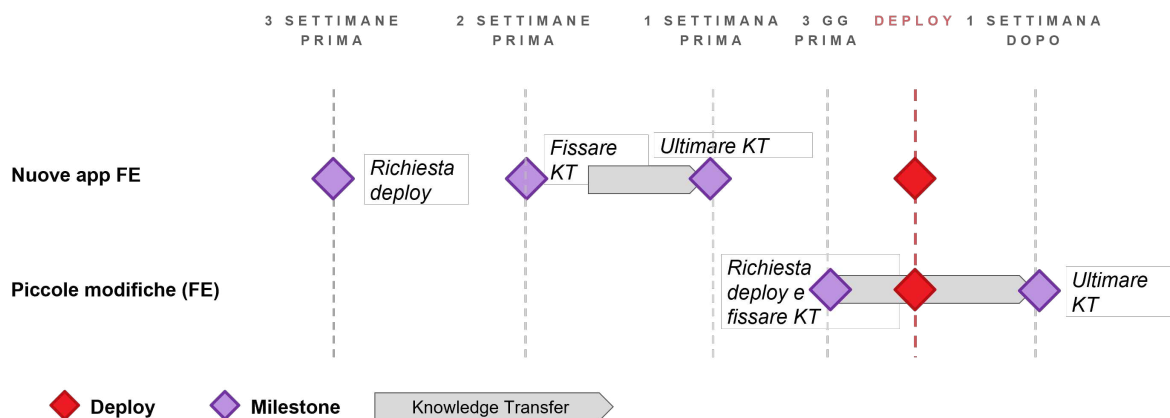


Figura 4.4: Vincoli per le richieste di deploy

applicazioni vengono tracciati, assegnati e risolti. Per un team di AMS, una gestione efficace dei ticket non solo migliora il servizio agli utenti, ma aumenta anche l'efficienza operativa e riduce il tempo di risoluzione dei problemi. I ticket in un ambiente AMS possono variare da semplici richieste di informazioni a complesse segnalazioni di malfunzionamenti del software. Possono includere anche richieste di nuove funzionalità o aggiornamenti. Ogni

ticket rappresenta una necessità specifica dell'utente o un problema da risolvere, e la sua gestione efficace è fondamentale per il successo del servizio offerto.

La gestione dei ticket inizia con la loro ricezione attraverso canali dedicati: dopo la ricezione, i ticket vengono classificati, prioritizzati e assegnati ai membri del team più adatti a risolverli. Il processo include, anche, la comunicazione con l'utente, il tracciamento del progresso del ticket, e la sua risoluzione finale. La gestione dei ticket presenta diverse sfide, tra cui la gestione di un alto volume di ticket, la priorità tra ticket urgenti e meno urgenti, e la comunicazione efficace con gli utenti. Le attività per una gestione efficiente dei ticket includono l'istituzione di chiare linee guida per la priorità dei ticket, la formazione continua del team, l'utilizzo di soluzioni tecnologiche avanzate e la valutazione regolare dei processi per identificare aree di miglioramento. Una gestione efficace dei ticket è essenziale per il successo di un team di AMS; essa garantisce che le richieste degli utenti siano soddisfatte in modo tempestivo ed efficiente, migliorando la soddisfazione dell'utente e la produttività del team Li *et al.* [2014].

Per il monitoraggio della gestione dei ticket da parte del team di AMS, il cliente utilizza una dashboard Qlik, dove confluiscono i dati e vengono, quindi, calcolati i principali KPI riguardanti i ticket elaborati, la loro prioritizzazione e classificazione (Figura 4.5).

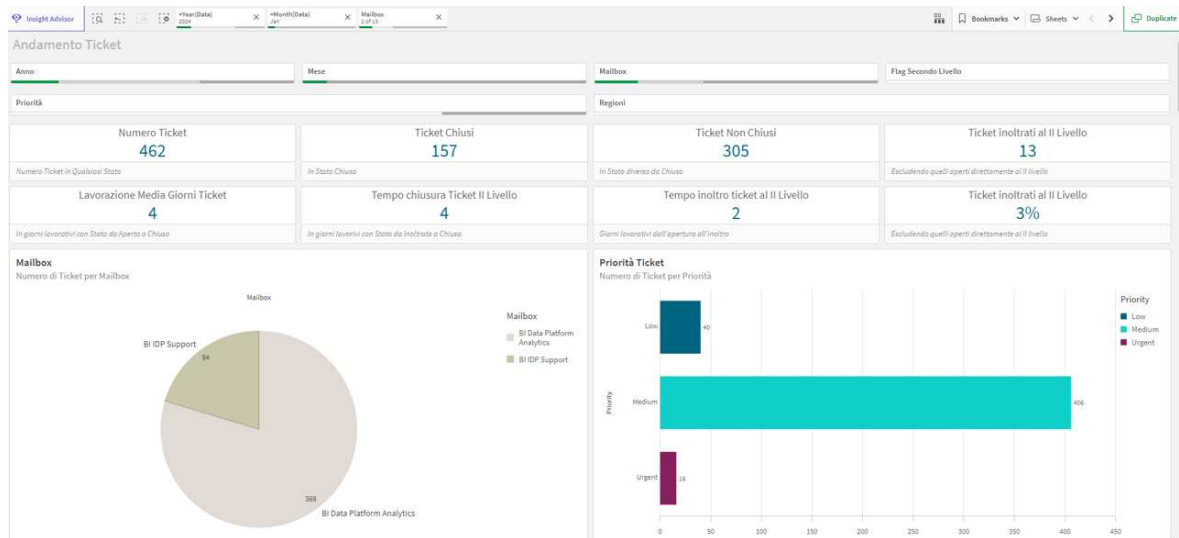


Figura 4.5: Dashboard Qlik per il monitoraggio dei ticket

Durante il periodo di tirocinio, è stato necessario monitorare questi KPI dalla dashboard suddetta ed è stata fondamentale l'estrazione settimanale dei principali indicatori, al fine di creare un report di interesse per il cliente. Il report (Figura 4.6) evidenzia tra i KPI i ticket totali ricevuti, i ticket aperti, i ticket chiusi e i giorni medi di lavorazione per ticket, sia in riferimento al mese corrente (month to date) che all'anno corrente (year to date). Con un grafico a linee viene mostrato il backlog annuale, ovvero l'andamento del numero di ticket aperti. Nei grafici ad anello sulla destra, invece, si riportano i ticket classificati e prioritizzati; si può notare che i ticket vengono classificati correttamente dal team di AMS, mentre non viene posta abbastanza attenzione sulla prioritizzazione; infatti quasi tutti i ticket hanno assegnata la priorità "medium", che coincide con quella associata di default al ticket nel momento della ricezione.

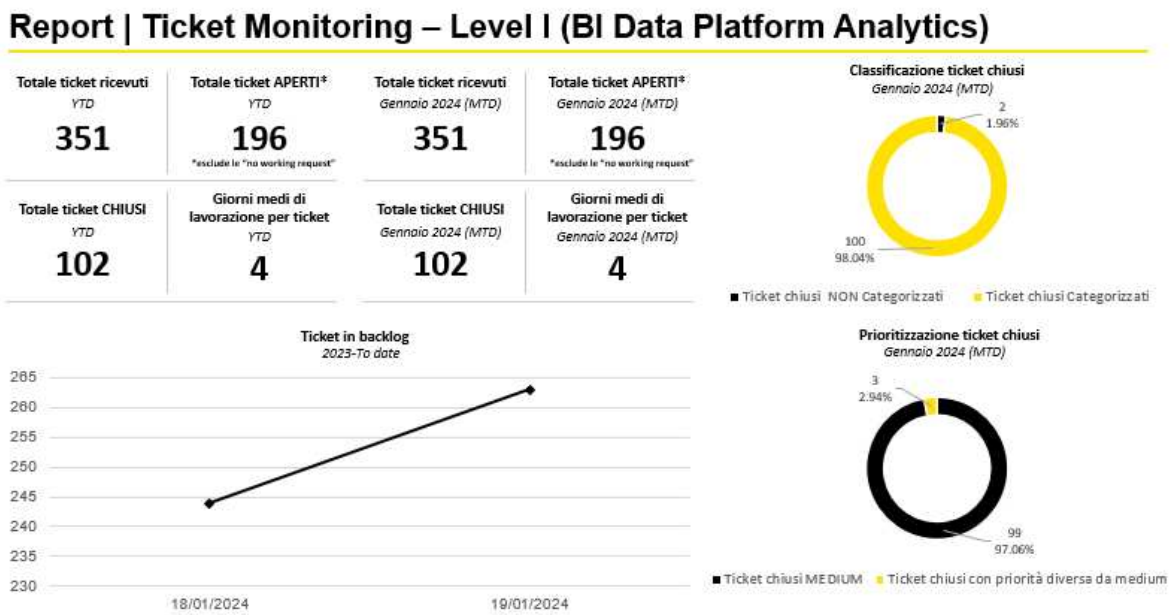


Figura 4.6: Report settimanale per l'analisi dei principali KPI

4.4 Tracciamento degli issue e analisi delle root cause

In ambito IT, un "issue" è qualsiasi problema, difetto, bug, o malfunzionamento che interrompe il normale funzionamento di un sistema software. Con il tracciamento degli issue ci si riferisce al processo di monitoraggio dei difetti del software e dei problemi che sorgono prima o dopo il deployment; dunque, ci si riferisce a tutte le attività correlate alla gestione di questi problemi, dalla loro iniziale scoperta alla loro risoluzione finale. Il tracking degli issue può essere distinto in tre fasi:

- *Identificazione e registrazione*: implica riconoscere e registrare gli issue, spesso tramite un sistema di ticketing, dove ciascun problema viene documentato con dettagli specifici.
- *Prioritizzazione e assegnazione*: gli issue vengono valutati e prioritizzati in base alla loro gravità, urgenza e impatto sul progetto. Successivamente, vengono assegnati ai membri del team per la risoluzione.
- *Monitoraggio e risoluzione*: comporta seguire il progresso degli issue dalla loro identificazione fino alla risoluzione, garantendo che vengano trattati in modo efficace e tempestivo.

Il tracciamento degli issue in un progetto IT è fondamentale per garantire che i problemi siano gestiti in modo efficiente e sistematico, contribuendo, così, al successo generale del progetto. Questo processo non solo aiuta a mantenere alta la qualità del prodotto software, ma facilita anche la collaborazione tra i team e migliora la soddisfazione degli utenti finali (Davis [2023]).

L'identificazione dei problemi è cruciale per l'analisi delle cause; infatti, il passo successivo al tracciamento degli issue è l'analisi delle root cause per identificare la causa alla radice del problema. Questo passaggio è fondamentale per evitare che lo stesso problema si ripresenti in futuro. La Root Cause Analysis (RCA) si avvale delle informazioni raccolte durante il tracciamento degli issue, come i log di sistema, le descrizioni degli errori e il contesto in cui l'issue si è verificato, per condurre un'analisi approfondita e sistematica. Dunque, il tracciamento degli issue in combinazione con la RCA punta a modifiche nella progettazione del sistema,

nel codice o nei processi per prevenire la ricomparsa del problema. Le soluzioni derivanti dalla RCA, dopo essere state implementate, vengono monitorate per valutarne l'efficacia, grazie al supporto continuo del sistema di issue tracking. Questo ciclo di feedback crea un processo di miglioramento continuo, dove le lezioni apprese da ogni issue sono integrate nei processi futuri per migliorare la qualità e l'efficienza complessive del progetto. In sintesi, il tracciamento degli issue e l'analisi delle cause principali sono processi interdipendenti in un progetto IT. Mentre il tracciamento degli issue si concentra sulla gestione e la risoluzione dei problemi attuali, la RCA approfondisce per identificare e correggere le cause fondamentali, prevenendo, così, la ricorrenza di problemi simili (Latino *et al.* [2019]).

Una delle attività focus di tirocinio, è stato proprio il monitoraggio dei problemi della Data Platform, correlato all'analisi delle root cause. Si è, quindi, implementato un sistema che potesse tracciare tutti gli issue e approfondire le cause di ognuno. Si è ritenuto fondamentale lavorare in questa direzione per offrire un supporto al mantenimento della Data Platform, con lo scopo di ridurre le problematiche grazie ad una continua analisi e alle attività in risposta agli issue. I dettagli che si sono scelti di registrare per il tracciamento degli issue sono visualizzabili in Figura 4.7.

Si noti che, ad ogni issue, è legata una root cause, poi tracciata ed approfondita in un sistema separato al fine di poter riportare più dettagli relativi ai next step e alle azioni da intraprendere in risposta al problema verificatosi. Si è ritenuto importante distinguere il tracciamento tra issue e root cause perché questi hanno spesso uno status differente; di solito l'issue viene risolto attraverso pratiche di work around, soluzioni manuali ed immediate al problema, mentre la risoluzione della root cause risulta più onerosa e prolungata nel tempo. Un esempio di tracciamento delle root cause è riportato in Figura 4.8.

ID	Multi-Import	Area	Issue-Description	Priority	Status	Assegnato a	Date Reported	Date Closed	Sign	Time-ETP-Officer (D)	Resolution	Root cause	Notes
11	10	Talend	Problema di connessione verso salesforce (job che prendono dati da salesforce vanno in errore). Cambio IP Address della macchina di Talend per cui non viene più gestita la Job Talend (dati da salesforce a causa di cambio di network job si presentavano con IP diversi. Errore eterno (infinite) di ip permessi.	Normal	Closed		08/11/2023	10/11/2023	2		Rischio lato salesforce sono stati aggiunti le nuove subnet (ip) alla whitelist.	Root Cause: 1	
12	11	Talend	Problema accodamento durante la finestra di manutenzione di sabato. La coda dei file è rimasta non processata.	High	Closed	AMS Level 1	12/11/2023	14/11/2023	2		Si è aspettato il caricamento delle file sono stati processati.	Root Cause: 1	Stesso problema rilevato con id 6 e id 7
13	12	SAP	File [redacted] sono stati inviati da SAP con dimensione 0 e senza messaggio SCS	High	Closed	SAP	24/11/2023	27/11/2023	3	38 h	Sono stati inviati dei file da SAP con dati non processati prima	Root Cause: 2	
14	13	SAP	File (VBK) (VBP) sono arrivati con una formattazione sbagliata (separatore)	High	Closed	SAP	24/11/2023	28/11/2023	4	4 h	File con il formato sbagliato sono stati modificati manualmente e rinviati sullo share. È stato inviato anche file di repair da parte di sap.	Root Cause: 3	Si deve contattare utente per modificare record file. In attesa di SAP.
15	14	Glissense	Errore sulle dati sales. Non ho girato task delle 15.30	High	Closed		30/11/2023	30/11/2023	0		Sono stati inviati PDF manualmente	Root Cause: 4	
16	15	Talend	Dalle 21.50 tutti i job che girano sul server di PRODUZIONE stanno andando in errore con il seguente messaggio: GC overhead limit exceeded	Critical	Closed	AMS Level 1	12/12/2023	13/12/2023	1	1 h	Ravviso del server [redacted]	Root Cause: 5	
17	16	Glissense	I task lato reporting giravano senza generare i report PDF	Critical	Closed		13/12/2023	13/12/2023	0		Ravviso Server il printing e il lavoro del server [redacted]	Root Cause: 6	
18	17	Talend	Problema server produzione [redacted] rilevato TCC in via straordinaria per risolvere	Critical	Closed		16/12/2023	16/12/2023	0	0 2 h	Si è aspettato il caricamento delle file sono stati processati.	Root Cause: 7	
19	18	Talend	Problema accodamento durante la finestra di manutenzione di sabato. La coda dei file è rimasta non processata.	High	Closed	AMS Level 1	17/12/2023	18/12/2023	1			Root Cause: 8	
20	19	Redditit	Il crawler [redacted] Job ha bloccato una tabella di redditit ed è rimasto in running, tutti gli altri sistemi che sono in questa tabella sono rimasti fermi	High	Closed		19/12/2023	20/12/2023	1			Root Cause: 9	
21	20	Talend	Problema accodamento per due cause: 1. Problema con lambda job non partivano 2. Messaggi duplicati per file arrivati da SAP	Critical	Closed	AMS Level 1 e Level 2	24/12/2023	26/12/2023	2	3 gg	Il 25 situazione iperincassa senza alcun problema. Il job è stato ricominciato a funzionare avendo ancora un backlog da gestire.	Root Cause: 8	
22	21	Glissense	Talend non vengono per nulla eseguiti causando uno stop della catena per un periodo prolungato.	High	Closed	AMS Level 1	27/12/2023	27/12/2023	0	0 5 h	Restaurato manualmente i task.	Root Cause: 3	
23	22	Talend	Problema accodamento per due cause: 1. Problema con lambda job non partivano 2. Messaggi duplicati per file arrivati da SAP	Critical	Closed	AMS Level 1 e Level 2	30/12/2023	01/01/2024	2	3 gg	Il 26 situazione iperincassa senza alcun problema. Il job è stato ricominciato a funzionare avendo ancora un backlog da gestire. Si sono isolati i messaggi che non sono stati processati. Il 27 il job è stato ricominciato e il 28 per evitare errore sul job di SAP evidenziamo un problema sui server Talend.	Root Cause: 8	
24	23	Talend	Talend non vengono per nulla eseguiti causando uno stop della catena per un periodo prolungato.	Critical	Closed	AMS Level 1 e Level 2	30/12/2023	01/01/2024	2	3 gg	Restaurato manualmente i task	Root Cause: 9	
25	24	Glissense	Errore sul case A che ha generato l'inefficienza dei tempi dei task	High	Closed	AMS Level 1	28/12/2023	28/12/2023	0	0 5 h			
26	25	Wolfrow	Dati di Daily Sales e aggiornamento dall'ERP di CRM [redacted] problema sul job SAP nell'ingestione dei file [redacted] l'errore è dovuto ad un problema di comunicazione fra la lambda e la TIC che ha mandato il messaggio SDS in FAILED.	High	Closed	AMS Level 1	11/01/2024	11/01/2024	0				Problema temporaneo, legato alla lambda. Non c'è root cause specifica da tracciare.
27	26	Wolfrow	Blocco catena nel tardo pomeriggio di sabato 13/01/24, ha causato ritardo sull'elaborazione dell'entità ZREB. PRI e quindi CRM ha subito ritardo	High	Closed	AMS Level 1	13/01/2024	13/01/2024	0		Sono stati rimandati in messaggi SDS tutti i task in attesa di essere sfilati per il job. In attesa di essere sfilati per il job. In attesa di essere sfilati per il job. In attesa di essere sfilati per il job.		Problema temporaneo, legato alla lambda. Non c'è root cause specifica da tracciare. Dipende dalle azioni a volte necessarie per il job. In attesa di essere sfilati per il job. In attesa di essere sfilati per il job. In attesa di essere sfilati per il job.
28	27	Glissense	Impartato glissense. Blocco sul DNGT, ETL, L template_Lt.studipanel.log.spd	High	Closed	AMS Level 1	21/01/2024	23/01/2024	1	1 gg	Renominazione del QVO	Root Cause: 10	

Figura 4.7: Issue tracker

ID Root Cause	Root Title	Priority	Date Reported	Start Date	Closed Date	Aging (days)	Root Cause Status	Root Cause	Owner Root Cause	Notes	Owner Root Steps	Due Date Root Steps
1	Buildlog loading chain - problema legato al nuovo sistema di manutenzione	Low	23/02/2023				In progress	Accadimenti causati da spaggiamento lambda per finestra manutenzione di nodi task che avviene nei weekend.		Il QoS di questo file, che hanno fatto il tempo che vedersi in cui ci sono stati problemi. La causa potrebbe essere lato lambda. E' stato maggiore il tempo interrogato con il vecchio sistema di manutenzione. - In contatto con AWS per capire se finestra di manutenzione può essere sposta meno di frequente (non tutti i weekend) e se può essere effettuato fine tuning sul modello per avere meno file in coda	1. Monitorare se problema si ripresenta o se isolato a seguito dei vari accorgimenti 2. Effettuare analisi quantitativa per capire quali e quanti job in accodamento, tempo di scodamento. 3. Verificare con AWS allineamento spaggiamento lambda e qik	
2	Causa de file arrivati con messaggio log	High	24/12/2023		27/12/2023		Closed	Problema lato sever BQDS (sap)				
3	Causa de file arrivati con formato sbagliato	High	24/12/2023		27/12/2023		Closed	Record sbagliati lato SAP	SAP			
4	Causa de task Qik non partiti	High	30/12/2023				In progress		Qik	Qik è stato contattato per indagare su causa problemi.	Si attende risposta Qik.	
6	Causa job in errore sever produzione	Critical	12/12/2023				In progress				Capire con Daniela se approfondire causa con toc	
7	Modifica ip esistente whitelister Saleforce	Normal	08/11/2023		10/11/2023		Closed	Cambio configurazione new old lato Data Management ed esistente whitelister lato salesforce (team data management non era a conoscenza di questa attività).	Saleforce	Proposta di nuovo ip fatto, con per risoluzione più nuovi issue/aggiornamenti instrumntali per arrivare a soluzione più velocemente.		
8	Problema accodamento e case A	High	24/12/2023	03/01/2023			In progress	Logica di entry	Debite		Verifica della logica di entry. Contattare Qik per analizzare causa	
9	Problema task qik case B	High	27/12/2023				Not started		Qik	da approfondire con qik.	AMS Level 1	
10	Blocco DVD	Normal	21/01/2024	23/01/2024			In progress	Analisi del file "ETL01_Template_Swinging_sif" viene utilizzato massivamente per creare i qid di primo livello in qik e su base di questo file vengono creati i task chiamati su estrazioni di configurazione di esecuzione. Tra tante configurazioni che ci sono (SET Data Connection, Cloud'yes o file e altro) e' anche la configurazione se andare in produzione o no. Il task che viene eseguito e' proprio la discriminante se scrivere o no nel qid di LOG. Quindi, la causa dell'errore che ha bloccato produzione e' che 2 task hanno scritto il qid al LOG nello stesso istante e il file e' stato corrotto. Questo file e' usato in tutti i task che hanno questo tipo di configurazione scrivono il LOG. E' un evento abbastanza raro perché quella operazione viene eseguita in meno di 1 secondo e quindi il rischio è estremamente basso, ma non 0.		1. Effettuare un controllo da subito su questo qid con il nuovo sistema di manutenzione. Verificare se il task è stato aggiornato e quando parte il suo estrattore. Questo perché non avendo più i qid di LOG parte e va in FULL. 2. Verificare le configurazioni duplicate nelle ed eventualmente toglierle		
11	App qik ha zalcato un		24/01/2024	24/01/2024			In progress					1. analizzare log dopo aggiornamento dell'app 2. aprire un ticket al livello 1 per aggiornare ogni 12h) 3. aprire un ticket al livello 1 per approfondire argomento

Figura 4.8: Root Cause Analysis

Controllo e monitoraggio delle attività

In questo capitolo si affronta l'importanza della fase di controllo e monitoraggio delle attività. In particolare, si esplora la rispettiva declinazione nel progetto Data Platform, dove per monitorare e coordinare le attività si sono utilizzati più strumenti a diversi livelli di dettaglio. Si presentano, quindi, una Kanban Board implementata per seguire le macro attività di progetto e un report costruito per focalizzarsi sullo stato di avanzamento di quest'ultime. Per il tracciamento dei task di dettaglio si riporta un esempio di tracker improntato per una gestione più specifica e puntuale di ogni attività. Infine, si illustra come si siano estrapolati in modo continuativo degli insegnamenti, poi inseriti in un registro delle lesson learned che permettesse di affrontare il progetto con una filosofia adattativa di miglioramento continuo, ed anche, di ottenere un vantaggio competitivo per progetti futuri.

5.1 Fase di controllo e monitoraggio nel Project Management

La fase di controllo e monitoraggio nel contesto del Project Management ha un ruolo cruciale nel garantire il successo dei progetti; infatti, risulta fondamentale tracciare ogni attività e il suo stato di avanzamento per traguardare gli obiettivi nei tempi e costi prefissati. Con "monitoraggio" ci si riferisce all'osservazione e al rilevamento dello stato del progetto, mentre con "controllo" si intende la correzione delle deviazioni rispetto al piano. Dunque, con questa fase si garantisce che il progetto proceda come previsto, identificando e risolvendo problemi e rischi in tempo reale. Di seguito si evidenziano i principali benefici di questa fase:

- *Allineamento con gli obiettivi:* il controllo e il monitoraggio contribuiscono a mantenere il progetto allineato agli obiettivi prefissati, evitando deragliamenti e minimizzando i rischi.
- *Ottimizzazione delle risorse:* un efficace monitoraggio può portare a un utilizzo più efficiente delle risorse, sia in termini di tempo che di budget.
- *Gestione degli stakeholder:* il controllo e il monitoraggio facilitano la comunicazione con gli stakeholder, permettendo di fornire trasparenza e costruendo fiducia attraverso aggiornamenti regolari e precisi.

Tra gli strumenti che si possono utilizzare per il monitoraggio delle attività si trovano:

- *Software di Project Management:* ne sono un esempio Trello, Asana, Microsoft Project, Microsoft Planner; questi offrono funzionalità per il tracciamento dei task, la gestione delle risorse, la pianificazione del tempo, e la comunicazione tra i membri del team. Con il loro utilizzo si ottengono l'automatizzazione dei processi di monitoraggio, la

centralizzazione delle informazioni, la facilità nell'aggiornamento dello stato e nella condivisione di aggiornamenti con il team e gli stakeholder.

- *Strumenti di reportistica e dashboard*: gli strumenti di reportistica e le dashboard permettono di visualizzare in tempo reale lo stato del progetto, evidenziando KPI, progressi, e aree di rischio. Grazie al loro utilizzo, si possono ottenere decisioni basate sui dati, visione d'insieme in tempo reale e facilità nel comunicare lo stato del progetto a stakeholder interni ed esterni (PMI [2021]).

Nel seguito si approfondiscono gli strumenti utilizzati nel progetto per monitorare le attività e, quindi, coordinarsi su più fronti.

5.2 Tracking delle attività

Per la gestione del monitoraggio e controllo per le attività del progetto Data Platform si è scelto di utilizzare una metodologia ibrida, che non comprendesse solamente software di Project Management, ma anche reportistica. Inoltre, vista la numerosità di stream differenti seguiti, non è stato possibile effettuare il tracciamento delle attività in dettaglio per ogni area seguita con lo stesso strumento, ma si è scelto di declinare il controllo e il monitoraggio dei task su due livelli, uno più di dettaglio ed uno più generale, per garantire che tutte le parti del progetto fossero monitorate in modo adeguato.

Per avere una panoramica chiara delle aree seguite, delle principali scadenze e dei next step fondamentali, si è scelto di utilizzare lo strumento Microsoft Planner (Figura 5.1), dove con l'implementazione di una Kanban Board è stato possibile ottenere un overview delle attività in backlog, in partenza, bloccate o chiuse. Una Kanban Board è uno strumento visivo per il monitoraggio e la gestione delle attività quotidiane; essa consente di visualizzare il flusso di lavoro, con le attività categorizzate in colonne. La Kanban Board aiuta a identificare i colli di bottiglia nel processo e consente una gestione flessibile delle priorità; ogni attività o compito è rappresentato da una scheda che si sposta attraverso la board in base al suo stato (Anderson [2010]). Ogni scheda può contenere le seguenti informazioni:

- titolo;
- data di inizio;
- data di scadenza;
- sotto attività;
- note;
- etichetta.

Questo approccio ha permesso di ottenere una visione di insieme delle attività seguite nel progetto e, quindi, di verificare costantemente l'allineamento con gli obiettivi complessivi.

A supporto della Kanban Board è stato prodotto un report quotidiano che potesse rappresentare le stesse macro attività, ma con dettagli aggiuntivi sullo stato di avanzamento e sulla pianificazione (Figura 5.2). Infatti, con il supporto grafico del diagramma di Gantt e con dei diagrammi a torta, si è potuto illustrare in modo chiaro lo stato di ogni attività. Questo report, con ancora meno dettagli della Kanban Board, ma con evidenze grafiche specifiche, è stato redatto allo scopo di riportare al cliente lo stato e, quindi, la gestione delle attività seguite in modo chiaro, conciso e immediatamente comprensibile.

Come anticipato, la visualizzazione del flusso di lavoro ad alto livello non è stata sufficiente per coordinare le attività quotidiane, poiché ogni area di lavoro è stata caratterizzata



Figura 5.1: Kanban Board per il tracciamento delle attività ad alto livello

da decine di micro attività assegnate a risorse differenti e con determinate date di scadenza. Inserire tutte queste attività di dettaglio in un'unica board avrebbe reso il planner poco

chiaro e difficilmente manutenibile; per tale ragione si è optato per gestire esternamente le attività di dettaglio per ogni area tracciata a livello macroscopico nello strumento Microsoft Planner. Sono state già in parte discusse le metodologie per il controllo di dettaglio delle attività; infatti, ad esempio per la documentazione, si è implementato un tracker ad hoc che permettesse di evidenziare le attività in sospeso, e quindi facilitare anche la comunicazione con il system integrator nella quotidianità (Figura 3.5). Un altro esempio già citato è quello del tracciamento delle root cause; riguardo a ciò si può notare che, nello sheet mostrato in Figura 4.8, si tiene traccia dei next step, dei relativi owner e delle date di scadenza. Un sistema ancora differente è stato implementato per gestire la raccolta della documentazione destinata ad un audit finalizzato al rilascio della certificazione TISAX per il cliente. Al fine di garantire il raccoglimento esaustivo della documentazione necessaria per poter certificare l'azienda, si è scelto di impostare un tracker come quello mostrato in Figura 5.3. Si noti come questo livello di complessità non possa essere integrato per ogni area monitorata, ma si ha la necessità di distinguere i vari sistemi per chiarezza e ottimizzazione della gestione delle attività.

5.3 Lesson learned

Durante la fase di monitoraggio e controllo delle attività, si sono anche raccolte le lesson learned, riportate in modo parziale in Figura 5.4. Raccoglierle in modo continuativo, anziché associare la loro raccolta alla fase di chiusura del progetto, ha permesso di ottenere i seguenti vantaggi:

- *Miglioramento continuo*: raccogliendo le lezioni apprese durante il progetto, si è potuto applicare immediatamente la conoscenza acquisita per migliorare le fasi successive del progetto. Questo approccio al miglioramento continuo ha consentito di correggere le rotte e di ottimizzare i processi in tempo reale.
- *Prevenzione dei problemi*: identificare e documentare le lezioni apprese durante il monitoraggio e il controllo ha aiutato a prevenire la ripetizione degli stessi errori o inefficienze nelle fasi successive del progetto.
- *Maggiore coinvolgimento del team*: la raccolta continua di feedback e osservazioni dal team ha promosso un ambiente di collaborazione e apprendimento. I membri del team sono più propensi a condividere le loro esperienze e soluzioni se vedono che il loro contributo viene valorizzato e applicato in modo tempestivo.
- *Documentazione accurata*: le lezioni apprese sono più precise e dettagliate quando vengono raccolte "sul campo", piuttosto che mesi dopo il completamento del progetto. La memoria è fresca, e i dettagli degli eventi e delle decisioni sono più chiari.
- *Miglioramento della pianificazione futura*: le lezioni raccolte durante il progetto hanno permesso il miglioramento della pianificazione e dell'esecuzione di fasi future del progetto stesso, contribuendo a una cultura aziendale di apprendimento e adattamento continuo.

Per integrare efficacemente la raccolta delle lesson learned nel processo di monitoraggio e controllo è stato utile stabilire un sistema strutturato per la documentazione e la revisione regolare di queste lezioni, coinvolgendo tutti i membri del team e garantendo che le conoscenze acquisite venissero condivise e, se possibile, implementate. In particolare, si è deciso di utilizzare lo strumento Microsoft List, mostrato in Figura 5.4, come sistema condiviso, accessibile facilmente da tutto il team, come punto di accesso comune per la condivisione di conoscenza e delle lezioni apprese.














% AVANZAMENTO	DESCRIZIONE	NEXT STEPS	NOVEMBRE	DICEMBRE
<p>Doc Commercial</p> 	<p>Revisione doc commerciale</p>	<ul style="list-style-type: none"> Continuare revisione doc (HQ, CR, Bari) 		
<p>Doc Supply Chain</p> 	<p>Revisione doc supply chain con HQ, CR, Bari</p>	<ul style="list-style-type: none"> Continuare revisione doc in settimana con Bari, CR, HQ 		
<p>Doc Supply Chain - File Planning</p> 	<p>Creazione file guida per indicare app da utilizzare tra NetKPI e New Planning</p>	<ul style="list-style-type: none"> Revisionare ultimi dati inseriti con Sara e Carlo Allineare Daniela Dashboard Qlik? 	 <p>Allineamento Daniela</p>	
<p>Doc Sharepoint e Alberatura</p> 	<p>Creare pagina sharepoint di accesso e riorganizzare doc</p>	<ul style="list-style-type: none"> Avvisare che dal 6/11 iniziamo a spostare file Fare traccia pptx contenuti sharepoint 	 	
<p>AMS Deploy Front-End</p> 	<p>Avvio nuova procedura per deploy front-end</p>	<ul style="list-style-type: none"> Aggiornare slide con vincoli temporali Email per avvertire di nuova procedura 	 <p>Allineamento Daniela</p>	
<p>AMS Issue Log Tracking</p> 	<p>Tenere traccia quotidianamente degli issue emersi</p>	<ul style="list-style-type: none"> Descrizione issue log comprensibile 		
<p>AMS Ticket Monitoring</p> 	<p>Tracciare quotidianamente priorità e categorie ticket chiusi e backlog idp support</p>			

Figura 5.2: Report di allineamento

DOCUMENTAZIONE	AUDIT	OWNER	DUE DATE	DONE	NOTES
Documentazione che riporta per ogni app lo stream owner	2.1/5.1		29/02/2024	<input type="radio"/>	Il mapping tra app TISAX e owner potremmo inviarlo come allegato alla documentazione.
Richiedere stream owner a ams level1		Elena		<input checked="" type="radio"/>	Ricevuto da Data Platform Analytics elenco owner di ogni stream.
Mapping lista stream owner con app TISAX		Lisa	30/01/2024	<input checked="" type="radio"/>	Utilizzato mapping stream owner contenuto in file "documentazione_Integrativa.docx" perché più aggiornato rispetto quello condiviso dal team di AMS.
Aggiornare Documentazione_Integrativa.docx con mapping aggiornato e condividerlo con		Luca		<input type="radio"/>	
Documentazione in cui si esplicitano i ruoli e cosa vedono e possono fare abbinato ad ogni ruolo	2.1		29/02/2024	<input type="radio"/>	
Produzione doc con		Luca	26/01/2024	<input checked="" type="radio"/>	Questo documento va in sostituzione del manuale di gestione utenze ruoli stream e ad?
Verificare che doc integrativa possa sostituire file "manuale gestione utenze ruoli, stream e ad"		Elena		<input type="radio"/>	
Documentazione di review del processo di pulizia delle utenze	5.1		31/03/2024	<input checked="" type="radio"/>	Processo di pulizia delle utenze documentato nel file "Documentazione_Integrativa.docx"
Produzione doc con		Luca	26/01/2024	<input checked="" type="radio"/>	
Documentazione applicazione tisax con relative fonti.	2.1		29/02/2024	<input type="radio"/>	Per le app per le quali c'è già una specifica documentazione oppure la documentazione è in corso di realizzazione, controlliamo che sia riportato il lineage completo, mentre per le altre la produrremo in versione ridotta solo per le sorgenti.
Mapping app tisax con doc ATE disponibili		Lisa	30/01/2024	<input checked="" type="radio"/>	Effettuato primo mapping tra doc disponibile in teams e lista app tisax. Da capire se ate disponibili sono aggiornati e sufficienti per estrarre fonti dati.
Recuperare fonti dati per app TISAX senza ATE aggiornato (tutte tranne board, oe homologations, europool)		Elena		<input type="radio"/>	

Figura 5.3: Tracker di attività in dettaglio

Elenchi personali		Lessons Learned ☆		
Titolo	Data ↑	Area	Evento	Raccomandazione
↳ Pianificazione documentazione	16/11/2023	Documentazione	Attesa indefinita per ricezione feedback su revisione documentazione, il che non permetteva una corretta pianificazione delle attività.	In concomitanza della condivisione della documentazione da revisionare internamente, richiedere sempre una data stimata entro la quale il team pensa di riuscire a dare un feedback.
↳ Solleciti non incisivi	21/11/2023	Monitoraggio Ticket	Problema mancata prioritizzazione non risolto con soli solleciti email, il tema è stato indirizzato correttamente solo con meeting ricorrenti a supporto del team di AMS per ripetere le analisi insieme ed evidenziare il problema.	Se al secondo sollecito via email non si notano miglioramenti procedere per vie differenti, non continuare a sollecitare invano con lo stesso metodo.
↳ Distinzione root cause da issue	01/12/2023	Issue Tracking	Ci si è accorti che non tracciando la root cause in un sistema di log differente da quello di tracking degli issue non	Distinguere il tracciamento della root cause da quello dell'issue
↳ Dare visibilità delle nuove iniziative	06/12/2023	Governance	Produzione di materiale di governance non utilizzato dal team poiché non diffuso e spiegato	Evidenziare con il team ogni nuova attività di governance e facilitare l'utilizzo di nuovi strumenti dandone la corretta visibilità.

Figura 5.4: Lesson learned

Il lavoro di tesi si è incentrato sulla governance di un progetto IT riguardante l'implementazione e il mantenimento di una Data Platform. La tesi si è inserita nella fase conclusiva di questo progetto pluriennale, per cui l'attenzione si è concentrata principalmente sulla raccolta meticolosa della documentazione, un passo fondamentale per l'indipendenza del cliente nella gestione autonoma della Data Platform, necessario per svincolarlo dalla dipendenza dal system integrator incaricato dell'implementazione. Per garantire una facile accessibilità alla documentazione, si è provveduto a sviluppare una pagina web Sharepoint dedicata, pensata come punto nevralgico per il reperimento di file cruciali e best practice legate al progetto. Inoltre, si sono svolte attività di supporto rivolte al team di AMS (Application Management Service), con un focus particolare sulla definizione di un processo di deploy strutturato e sulla creazione di linee guida chiare e uniformi per i fornitori, al fine di regolare con precisione i rilasci in ambiente di produzione.

Un altro aspetto chiave è stata la gestione del sistema di ticketing da parte di AMS, che ha richiesto un attento monitoraggio dell'andamento dei ticket con conseguente elaborazione di un report specifico. Il mantenimento della Data Platform, poi, è stato assicurato attraverso un rigoroso tracciamento dei problemi e un'analisi approfondita delle cause alla loro radice.

Per orchestrare tali attività, sono stati impiegati strumenti di Project Management all'avanguardia: una Kanban Board per la gestione a livello generale delle attività principali e tracker specializzati per il monitoraggio di compiti più specifici, configurati in base alle necessità emergenti.

In conclusione, questo lavoro di tesi ha apportato un contributo significativo al progetto. La produzione di una documentazione accurata e aggiornata ha permesso al cliente di acquisire una comprensione approfondita del sistema sviluppato e di gestirlo autonomamente, senza la necessità di ulteriore assistenza esterna. Parallelamente, le attività svolte in collaborazione con il team di AMS hanno giocato un ruolo cruciale nell'assicurare elevati standard qualitativi e nell'incorporare una filosofia di miglioramento continuo. Quest'ultima, in particolare, si rivela essenziale per garantire che un sistema IT possa evolvere costantemente, migliorando le sue prestazioni e aumentando la sua fruibilità per gli utenti.

Per ampliare ulteriormente l'efficacia del progetto, uno sviluppo futuro potrebbe riguardare la formalizzazione di un processo definito e strutturato per l'aggiornamento della documentazione. Infatti, è importante annotare ogni nuovo sviluppo del sistema per evitare che la documentazione diventi obsoleta. Un altro passo in avanti per la stabilizzazione della Data Platform, potrebbe coinvolgere il miglioramento della root cause analysis già in essere attraverso l'applicazione di una metodologia standard come 5 Whys, Fishbone (Ishikawa) Diagram, o Failure Mode and Effects Analysis (FMEA).

- ALUTBI, M. (2020), «WORK BREAKDOWN STRUCTURE (WBS)», . (Cited at pages iv e 13)
- ANDERSON, D. (2010), *Kanban: Successful Evolutionary Change for Your Technology Business*, Blue Hole Press, URL <https://books.google.it/books?id=RJ0VUkfUWZkC>. (Cited at page 51)
- APRIL, A. (2008), *Software Maintenance Management : Evaluation and Continuous Improvement*. (Cited at page 33)
- BASS, L., CLEMENTS, P. e KAZMAN, R. (2012), *Software Architecture in Practice: Software Architect Practice_c3*, SEI Series in Software Engineering, Pearson Education, URL <https://books.google.it/books?id=-II73rBDXCYC>. (Cited at page 31)
- BEZAK, S. e NAHOD, M.-M. (2011), «Project manager's role analysis as A project management concept», *Tehnicki Vjesnik*, vol. 18. (Cited at pages iv, 5 e 6)
- BONNAL, P., GOURC, D. e LACOSTE, G. (2002), «The Life Cycle of Technical Projects», *Project Management Journal*, vol. 33, p. 12–19. (Cited at page 9)
- BREWSTER, E., GRIFFITHS, R., LAWES, A. e SANSBURY, J. (2012), *IT Service Management: A Guide for ITIL Foundation Exam Candidates*, Bcs Series, BCS, URL <https://books.google.it/books?id=hBs7EunMctUC>. (Cited at page 41)
- CHIN, C., YAP, E. H. e SPOWAGE, A. (2012), «Project Management Methodologies: A Comparative Analysis», *Journal for the Advancement of Performance Information & Value*, vol. 4. (Cited at page 10)
- CLEMENTS, P. (2003), *Documenting Software Architectures: Views and Beyond*, SEI series in software engineering, Addison-Wesley, URL <https://books.google.it/books?id=ASc9HYPkr4sC>. (Cited at page 34)
- COCCIA, M. (2017), «The Fishbone diagram to identify, systematize and analyze the sources of general purpose technologies», vol. 4, p. 291–303. (Cited at pages iv e 15)
- DAVIS, S. (2023), *Issue Management: The Strategic Approach*, p. 1–7. (Cited at page 46)
- DRAHOKOUPIL, J., curatore (2020), *The challenge of digital transformation in the automotive industry : jobs, upgrading and the prospects for development*, Brussels : ETUI, URL <http://hdl.handle.net/11159/4875,1738745007>. (Cited at page 22)

- DUMBRAVĂ, V. e IACOB, V. S. (2013), «Using Probability – Impact Matrix in Analysis and Risk Assessment Projects», *Journal of Knowledge Management, Economics and Information Technology*, vol. 3 (6), p. 1–7, URL <https://ideas.repec.org/a/spp/jkmeit/spi13-07.html>. (Cited at pages iv e 17)
- GYÖRY, A., BRENNER, W., UEBERNICKEL, F., WULF, J., ZELT, S., HEYM, M. e WARNKE, A. (2012), «Strategies for Application Management Services», . (Cited at page 41)
- HELMS, M. e NIXON, J. (2010), «Exploring SWOT analysis – where are we now? : A review of academic research from the last decade», *Journal of Strategy and Management - J Econ Manag Strat*, vol. 3, p. 215–251. (Cited at page 17)
- JAMES, D. e WILSON, J. (2000), «Gantt Charts: A Centenary Appreciation», . (Cited at page 13)
- JORGENSEN, P. (2013), *Software Testing: A Craftsman's Approach, Fourth Edition*, An Auerbach book, Taylor & Francis, URL <https://books.google.it/books?id=6WlmAQAAQBAJ>. (Cited at page 31)
- KOZAK-HOLLAND, M. (2011), *The History of Project Management, Lessons from history*, Multi-Media Publications, URL <https://books.google.it/books?id=9eicuAAACAAJ>. (Cited at pages 3 e 4)
- LARSON, E. e GRAY, C. (2010), *Project Management: The Managerial Process*, McGraw-Hill/Irwin series operations and decision sciences, McGraw-Hill Irwin, URL <https://books.google.it/books?id=tNdqngEACAAJ>. (Cited at pages iv, 5, 6, 7, 10, 11, 12, 14 e 18)
- LATINO, M., LATINO, R. e LATINO, K. (2019), *Root Cause Analysis: Improving Performance for Bottom-Line Results, Fifth Edition*, CRC Press, URL <https://books.google.it/books?id=HTb3DwAAQBAJ>. (Cited at page 47)
- LI, T., LIU, R., SUKAVIRIYA, N., LI, Y., YANG, J., SANDIN, M. e LEE, J. (2014), «Incident Ticket Analytics for IT Application Management Services», p. 568–574. (Cited at page 45)
- LLOPIS-ALBERT, C., RUBIO, F. e VALERO, F. (2021), «Impact of digital transformation on the automotive industry», *Technological Forecasting and Social Change*, vol. 162, p. 120 343, URL <https://www.sciencedirect.com/science/article/pii/S0040162520311690>. (Cited at page 23)
- LOCK, D. (2013), *Project Management*, Gower, URL <https://books.google.it/books?id=mipKLwEACAAJ>. (Cited at page 4)
- MARCHEWKA, J. (2016), *Information Technology Project Management: Providing Measurable Organizational Value*, Wiley, URL <https://books.google.it/books?id=rsLlBQAQBAJ>. (Cited at pages iv, 20 e 21)
- MARTIN, R. (2008), *Clean Code: A Handbook of Agile Software Craftsmanship*, Robert C. Martin Series, Pearson Education, URL https://books.google.it/books?id=_i6bDeoCQzsc. (Cited at page 31)
- MCCONNELL, S. (2004), *Code Complete, Developer Best Practices*, Pearson Education, URL <https://books.google.it/books?id=LpVCAwAAQBAJ>. (Cited at page 29)
- MUNNS, A. e BJEIRMI, B. (1996), «The role of project management in achieving project success», *International Journal of Project Management*, vol. 14 (2), p. 81–87, URL <https://www.sciencedirect.com/science/article/pii/0263786395000577>. (Cited at page 18)

- NAFKHA, R. (2016), «The pert method in estimating project duration», *Information Systems Management*, vol. 5, p. 542–550, URL <https://api.semanticscholar.org/CorpusID:57983717>. (Cited at page 14)
- NASUTION, M. e WEISTROFFER, H. (2009), «Documentation in Systems Development: A Significant Criterion for Project Success», in «2009 42nd Hawaii International Conference on System Sciences», p. 1–9. (Cited at page 30)
- OPOKU, A. e TALLON, A. (2019), «The role of project sponsors in defining and realising project benefits», . (Cited at page 6)
- OPREA, D. e MESNITA, G. (2006), «The Information Systems Documentation - Another Problem for Project Management», . (Cited at page 30)
- PAUL NIEUWENHUIS, P. W., curatore (2003), *The Automotive Industry and the Environment*, Woodhead Publishing, URL <https://www.sciencedirect.com/science/article/pii/B9781855737136500019>. (Cited at pages 22 e 23)
- PMI, curatore (2013), *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, Project Management Institute, Newtown Square, PA, 5 ed. (Cited at pages 3 e 5)
- PMI (2021), *Guide to the Project Management Body of Knowledge (PMBOK Guide) and the Standard for Project Management, Pmbok(r) Guide*, Project Management Institute, URL <https://books.google.it/books?id=rp29zQEACAAJ>. (Cited at pages iv, 9, 12, 14, 15, 16, 17, 18, 33 e 51)
- RAJABLU, M., MARTHANDAN, G., FADZILAH, W. e WAN YUSOFF, W. F. (2015), «Managing for Stakeholders: The Role of Stakeholder-Based Management in Project Success», *Asian Social Science*, vol. 11. (Cited at page 8)
- RIASANOW, T., GALIC, G. e BÖHM, M. (2017), «Digital Transformation in the Automotive Industry: Towards a Generic Value Network», . (Cited at page 23)
- ROBERTSON, S. e ROBERTSON, J. (2012), *Mastering the Requirements Process: Getting Requirements Right*, Pearson Education, URL <https://books.google.it/books?id=yE91LgrpaHsC>. (Cited at page 30)
- SEYMOUR, D. T. e HUSSEIN, S. (2014), «The History Of Project Management», *International Journal of Management & Information Systems (IJMIS)*, vol. 18, p. 233. (Cited at pages 3 e 4)
- SHORE, J. e WARDEN, S. (2008), *The Art of Agile Development, Theory in practice*, O'Reilly Media, Incorporated, URL <https://books.google.it/books?id=2q6bAgAAQBAJ>. (Cited at page 32)
- SIEBEL, T. (2019), *Digital Transformation: Survive and Thrive in a Time of Mass Extinction*, RosettaBooks, URL <https://books.google.it/books?id=AlpsxAEACAAJ>. (Cited at pages iv e 22)
- SUHANDA, R. e PRATAMI, D. (2021), «RACI Matrix Design for Managing Stakeholders in Project Case Study of PT. XYZ», *International Journal of Innovation in Enterprise System*, vol. 5, p. 122–133. (Cited at pages iv e 16)
- TAYLOR, F. W. (1911), *The Principles of Scientific Management*, no. taylor1911 in History of Economic Thought Books, McMaster University Archive for the History of Economic Thought, URL <https://ideas.repec.org/b/hay/hetboo/taylor1911.html>. (Cited at page 4)

- THESING, T., FELDMANN, C. e BURCHARDT, M. (2021), «Agile versus Waterfall Project Management: Decision Model for Selecting the Appropriate Approach to a Project», *Procedia Computer Science*, vol. 181, p. 746–756. (Cited at pages 11 e 12)
- ZHANG, P., CAREY, J. e TE'ENI, D. (2004), «Integrating Human-Computer Interaction Development into SDLC: A Methodology.», p. 574. (Cited at page 20)

Siti web consultati

- What Is a Work Breakdown Structure (WBS) In Project Management? (projectmanager.com) – <https://www.projectmanager.com/guides/work-breakdown-structure>
- What is project management? | APM – <https://www.apm.org.uk/resources/what-is-project-management/>
- Managing Communications - The Digital Project Manager – <https://thedigitalprojectmanager.com/managing-communications/>
- I sette rischi di progetto più comuni e come evitarli [2022] | Asana – <https://asana.com/it/resources/project-risks>
- Gestione dei rischi di progetto | Project Management Center (humanwareonline.com) – <https://www.humanwareonline.com/project-management/center/gestione-rischi-progetto/>
- SWOT Analysis: How To With Table and Example (investopedia.com) – <https://www.investopedia.com/terms/s/swot.asp>
- Project team roles and responsibilities (with examples) (resourceguruapp.com) – <https://resourceguruapp.com/blog/project-team-roles-and-responsibilities>
- What is The Project Life Cycle & its 5 Main Phases? - Venngage – <https://venngage.com/blog/project-life-cycle/>
- What is digital transformation? | IBM – <https://www.ibm.com/topics/digital-transformation>
- What Is A Data Platform? | MongoDB – <https://www.mongodb.com/what-is-a-data-platform>
- The Evolution of Project Management (projectsmart.co.uk) – <https://www.projectsmart.co.uk/history-of-project-management/evolution-of-project-management.php>

Ringraziamenti

Vorrei dedicare questo spazio alle persone a me più care, le quali hanno reso possibile il raggiungimento di questo traguardo.

Un immenso grazie va al mio relatore, Domenico Ursino, che mi ha fatto scoprire ed appassionare alla disciplina del Project Management. La sua disponibilità e il suo supporto costante sono stati fondamentali per la realizzazione di questo lavoro di tesi.

Un ringraziamento sincero al mio correlatore, Paolo Lo Giudice, che mi ha accompagnato in questo percorso fungendo da colonna portante. Gli sarò eternamente grata per la fiducia che ha riposto in me.

Grazie ai miei amici, Anna, Chiara, Giulia, Matteo, Michelangelo, Samuele e Simone, con i quali ho potuto condividere traguardi e delusioni, gioie e tormenti; senza di loro non sarei qui ora a scrivere le conclusioni di questo percorso meraviglioso.

Un grazie speciale a Mattia, che non ha mai smesso di credere in me, e in tutti questi anni ha pazientemente accettato di ricevere più "devo studiare" che "ti amo".

Grazie ai miei zii e ai miei nonni, che mi hanno incoraggiata e sostenuta costantemente con amore incondizionato.

In ultimo, non posso che esprimere la mia infinita gratitudine verso mia sorella, Debora, e i miei genitori, a cui devo ogni parte di me.