

POLITECNICO DI MILANO

V Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica



**SVILUPPO E REALIZZAZIONE DI
PROCESSI DI INTEGRAZIONE DI
VENDITA IN AMBIENTE SAP
ATTRAVERSO SAP SOLUTION
MANAGER**

Relatore: Professor Barbara Pernici

Elaborato di Laurea di:
Davide Pasini Matr. 739300

Anno Accademico 2010-2011

Indice

Abstract	1
1 Introduzione	2
Struttura della tesi	3
2 Progetti IT nelle imprese	4
2.1 Panoramica delle organizzazioni IT	7
2.1.1 Fattori che influenzano le decisioni	13
2.1.2 Investimenti e tendenze	15
2.2 Rischi e problemi di un progetto	16
2.2.1 Aspettative degli stakeholders	20
2.2.2 Strategie sbagliate	21
2.2.3 Comportamenti non corretti	22
2.3 Possibili soluzioni	23
3 Application lifecycle management	26
3.1 Concetti di Software Process Engineering	28
3.2 Concetto di ALM	34
3.3 Caratteristiche ALM	37
3.3.1 Principali elementi	38
3.3.2 Ruoli e processi coinvolti	40

INDICE

3.4	Possibili approcci alla tecnologia	44
3.5	Soluzioni ALM in commercio	45
3.6	Sviluppi futuri : ALM e Software as a Service	48
4	Attività svolte in azienda	49
4.1	Processo di vendita	50
4.2	Sistema SAP CRM e Solution manager	53
4.3	Attività svolte	60
4.3.1	Scenario Telesales	62
4.3.2	Scenario Campaign Management	75
4.3.3	Configurazione e testing degli scenari	82
	Conclusioni	84
	Glossario e acronimi	88
	A Customer Relationship Management	92
A.1	Customer Relationship Management	92
	Bibliografia	95

Elenco delle figure

2.1	Mutamento dei progetti IT	8
2.2	Previsione dispositivi mobili venduti	9
2.3	Tipi di Service Provider	13
2.4	Previsioni di vendita mondiali per settore	16
3.1	Ciclo di vita dello sviluppo software	28
3.2	Modelli in software engineering	31
3.3	Application lifecycle management	37
3.4	Ruoli e processi coinvolti	41
3.5	Tre aree in cui agisce ALM	42
3.6	Governance in ALM	43
3.7	Development in ALM	43
3.8	Operations in ALM	44
4.1	SAP CRM - Schema funzionale	54
4.2	Solution Manager platform	57
4.3	Solution Manager folder	58
4.4	Inbound flow diagram	63
4.5	Outbound flow diagram	71
4.6	Campaign Management flow diagram	78
4.7	IC Outbound Telesales flow diagram	79

Elenco delle tabelle

2.1	Statistiche sul successo dei progetti informatici	18
-----	---	----

Abstract

Il presente lavoro di tesi é collegato all'attività di sviluppo di alcuni scenari standard in SAP CRM. In modo particolare vengono analizzati e realizzati i processi di vendita in ambito telesales e campaign management.

Il scenari realizzati nel sistema SAP CRM saranno un valido supporto all'analisi e allo studio dei processi di vendita.

Le conclusioni del lavoro hanno infine mostrato il punto della situazione e hanno evidenziato le prospettive di sviluppo future del progetto realizzato, indicando le prossime operazioni da svolgersi.

Capitolo 1

Introduzione

Il presente elaborato deriva da una esperienza di stage svolta in una grossa società di consulenza. Anche se tale esperienza é stata di breve periodo, ho comunque potuto affrontare una serie di tematiche sulla gestione del ciclo di vita del software e, in particolar modo, nello sviluppo e configurazione di processi di vendita in sistemi SAP CRM.

L'essere inserito in una grande organizzazione mi ha dato ovviamente una grossa opportunità di sperimentare e utilizzare un numero considerevoli di software per il coordinamento delle attività e la gestione del mio lavoro. Le dimensioni, la varietà e le disposizioni geografiche del team impongono l'utilizzo di tali software e la continua ricerca di nuovi tool permette una maggior efficienza nelle attività di sviluppo.

Durante il mio stage ho potuto osservare che numerose sono le problematiche da affrontare nello sviluppo di software, in particolar modo quando si tratta di coordinare il lavoro di numerose persone con esperienze e compiti diversi. Soprattutto la gestione delle informazioni, della documentazione, della gestione delle risorse risultano argomenti molto interessanti e molto sentiti da chi deve gestire e guidare le attività in un progetto.

Oggi giorno l'abilità di produrre un prodotto di qualità in tempo ed ad un costo competitivo é molto importante per qualsiasi organizzazione industriale.

CAPITOLO 1. INTRODUZIONE

La produzione e lo sviluppo di tali prodotti sta diventando sempre più complicata anche a causa della globalizzazione che ha permesso certamente una miglior distribuzione dei costi, ma complicando le comunicazioni, coordinamento e il controllo.

In questo scenario le aziende cercano comunque modi sistematici e più efficaci per sostenere lo sviluppo e la gestione dei prodotti sempre più complessi in un ambiente di sviluppo globale. Una risposta a queste sfide risiede nel concetto di Application Lifecycle Management (ALM).

In questo elaborato si pone l'obiettivo di dare una visione generale delle problematiche del settore IT e in particolare su alcuni strumenti per la gestione dei progetti. Nella seconda parte verrà presentato il lavoro svolto durante il periodo di stage in azienda, basato sullo sviluppo e la configurazione di alcuni scenari in ambito Sales Management su sistema SAP CRM.

La mia analisi svolta in questo documento si basa, oltre che su attività svolte in azienda sotto la supervisione di un tutor, anche da interviste a persone d'esperienza e ricerche svolte su motori di ricerca specializzati.

Struttura della tesi

I contenuti dell'elaborato di tesi sono strutturati nel seguente modo:

Capitolo 2: si descrivono una panoramica delle organizzazioni IT, i maggiori problemi nei progetti e vengono fornite delle possibili soluzioni o metodologie per risolverli

Capitolo 3: viene introdotto il concetto di Application Lifecycle Management, sue caratteristiche e i principali elementi che lo compongono

Capitolo 4: viene presentato il lavoro svolto durante lo stage in azienda

Capitolo 5: si riassume quanto svolto riportando possibili sviluppi futuri e alcune considerazioni

Capitolo 2

Progetti IT nelle imprese

Molto spesso la società moderna viene definita come società dell'informazione. A partire dagli anni 80 sino ad oggi, le informazioni hanno rivestito un ruolo sempre più alto nell'importanza strategica per lo svolgimento delle attività economiche e commerciali. Inoltre, anche la produzione, le trasmissioni, il consumo e la dipendenza dalle informazioni acquistano, col passare del tempo, una posizione sempre più rivelante, rispetto al passato. La potenza dei moderni processori è in continuo incremento e la quantità di spazio disponibile per i desktop e laptop oggigiorno era quasi inconcepibile 15 anni fa. Difatti l'epoca odierna era alimentata da un rapido sviluppo digitale delle tecnologie dell'informazione, ovvero dell'**Information Technology** (IT).

Questo rapido sviluppo dell'IT aiuta a incrementare l'efficienza ed l'efficacia dei business esistenti, ma permette anche di aprire nuove aspettative a nuovi scenari di mercato. Infatti, aziende si sono formate proprio sulla base delle informazioni, e hanno focalizzato il loro business solamente sull'uso e l'elaborazione delle informazioni. Si pensi solamente a come Google è riuscita a sfruttare informazioni e dati, derivati dall'analisi dei siti internet presenti nel web, e come il suo profitto derivi completamente da un asset intangibile come appunto lo sono le informazioni.

CAPITOLO 2. PROGETTI IT NELLE IMPRESE

Definizione di Information Technology L'*Information Technology* indica l'uso della tecnologia nella gestione e nel trattamento dell'informazione, specie nelle grandi organizzazioni. In particolare il settore IT riguarda l'uso di apparecchi digitali e di programmi software che consentono di creare, memorizzare, scambiare e utilizzare informazioni o dati nei più svariati formati: dati numerici, testuali, comunicazioni vocali e molto altro. Inoltre, l'*IT* comprende sempre un maggior numero di beni e servizi che permettono la connessione tra consumatori e produttori, in modo precedentemente non realizzabili. Tali servizi contribuiscono anche a una maggiore produttività in numerosi settori dell'industria dei servizi, come ad esempio il settore finanziario, della comunicazione, assicurativo e della vendita al dettaglio. Gli enormi cambiamenti e le forti crescite hanno portato a considerare l'*informazione*, non più come elemento di supporto, ma come base di valore di per sé.

Quando si fa riferimento al significato di Information Technology è giusto considerare anche il contesto in cui se ne parla. Infatti IT è un termine comune che però cambia significato in base al contesto di utilizzo. Una buona classificazione è rappresentata dai seguenti ambiti:

- *IT come Organizzazione*: unità interna o funzionale di un fornitore di servizi commerciali o d'impresa. IT in questo scenario è considerata come una organizzazione con al suo interno capacità e risorse. Organizzazione IT può essere suddivisa in basi a vari sottoinsiemi: funzioni aziendali, unità di servizi o a livello aziendale.
- *IT come un Elemento*: elemento di sistema o di processo. Sistemi, applicazioni e infrastrutture IT sono elementi o sottoinsiemi di più larghi prodotti e permettono o sono incorporati in processi o servizi.
- *IT come un Servizio*: in questo ambito l'IT è considerato come una categoria di servizi condivisi utilizzati da unità di business. Con il termine *Servizio* si indica la capacità di distribuire o consegnare valore al cliente,

CAPITOLO 2. PROGETTI IT NELLE IMPRESE

o nel nostro caso a unità di business, facilitando i risultati che si vogliono raggiungere. In questo modo si cerca di evitare costi e rischi ulteriori. Tali servizi sono tipicamente applicazioni o infrastrutture che sono presentati o offerti come servizio ad organizzazioni IT interna o esterna all'organizzazione.

- *IT come un Asset*: IT viene vista come un insieme di capacità e risorse che forniscono un serie di benefici affidabile. Sotto questo aspetto l'IT è considerata come una categoria di asset economici, che forniscono un insieme di benefici per i possessori, includendo entrate, redditi e profitti. In questa classificazione i costi dell'IT derivati dalle attività economiche vengono gestiti come investimenti e non come costi aggiuntivi.

Riprendendo il concetto di servizio in ambito IT è importante stabilire quali sono gli Outcomes o risultati che da esso possono derivare. I risultati sono il frutto dell'esecuzione di task o compiti specifici e sono limitati e contrastati dalla presenza di alcuni vincoli. Servizi IT possono facilitare questi outcomes migliorando le performance e allentando la stretta dei constraints. Certamente per far ciò è necessario avere un buona visione generale di tutti gli aspetti e le conseguenze che riguardano questo settore; aspetti che si possono ricercare e inquadrare attraverso l'utilizzo di modelli ricorrenti. Infatti i modelli utilizzati sono spesso adottati per gestire complesse attività, riducendo i costi e mantenendo una buona flessibilità dei processi.

Un servizio quindi è qualsiasi attività o vantaggio che una parte può scambiare con un'altra, la cui natura risulta essenzialmente intangibile. Secondo la citazione del professore Theodore Levitt

"People want a quarter inch, not a quarter inch drill"

si può capire infatti la naturale intangibilità dei servizi. Anche al settore IT è possibile applicare lo stesso principio: un cliente richiede uno specifico servizio, come ad esempio E-Mail Server interno, per supportare le comunicazioni azien-

2.1 Panoramica delle organizzazioni IT

dali e migliorando l'affidabilità del sistema. Però lo stesso cliente non vuole essere annoiato da informazioni che sono relative al funzionamento dell'hardware o alla manutenzione del software o ancora dagli aspetti riguardanti la banda della rete.

2.1 Panoramica delle organizzazioni IT

Come accennato a inizio capitolo, il rapido sviluppo delle tecnologie e del settore dell'Information Technology in generale hanno portato a numerosi cambiamenti sia nel modo di lavorare che nel concepire i nuovi e più complicati problemi che ogni organizzazione deve affrontare. Sempre dagli inizi degli anni 80, ogni organizzazione si muoveva in favore di investimenti nel settore IT allo scopo di aumentare la velocità di produzione e ridurre i rischi dell'impresa. Nell'ultimo periodo invece le aziende svolgono le attività di investimento nel settore IT in modo più focalizzato e con maggiore cautela rispetto al passato. A causa anche di un minor budget aziendale, le compagnie focalizzano le proprie spese in modo da ottenere il maggior rendimento ed la maggior efficacia da ogni loro investimento, cercando di avere un maggior controllo sulle attività svolte. D'altro canto però, le aziende devono comunque attivare nuovi processi che permettano di innovare e rimodernizzare le proprie attività. Quindi si è sempre alla ricerca di un equilibrio tra innovazione e riduzione dei costi.

Allo stesso tempo le organizzazioni devono affrontare una serie di problematiche maggiori e più frequenti, molte delle quali sono intersecate direttamente con la gestione degli aspetti riguardanti l'IT. Per cercare di superare questi ostacoli, alle aziende è richiesto di attuare una serie di processi di controllo e tracciabilità delle informazioni. In questa modo le informazioni raccolte sono essenziali per lo sviluppo di processi che permettano di impedire costi operativi troppo flessibili e onerosi ed evitando l'aggiunta di rischi non calcolati.

Le sfide di oggi sono sicuramente più complicate rispetto al passato per chi è responsabile e deve gestire progetti IT. Soprattutto se si tratta di progetti

2.1 Panoramica delle organizzazioni IT

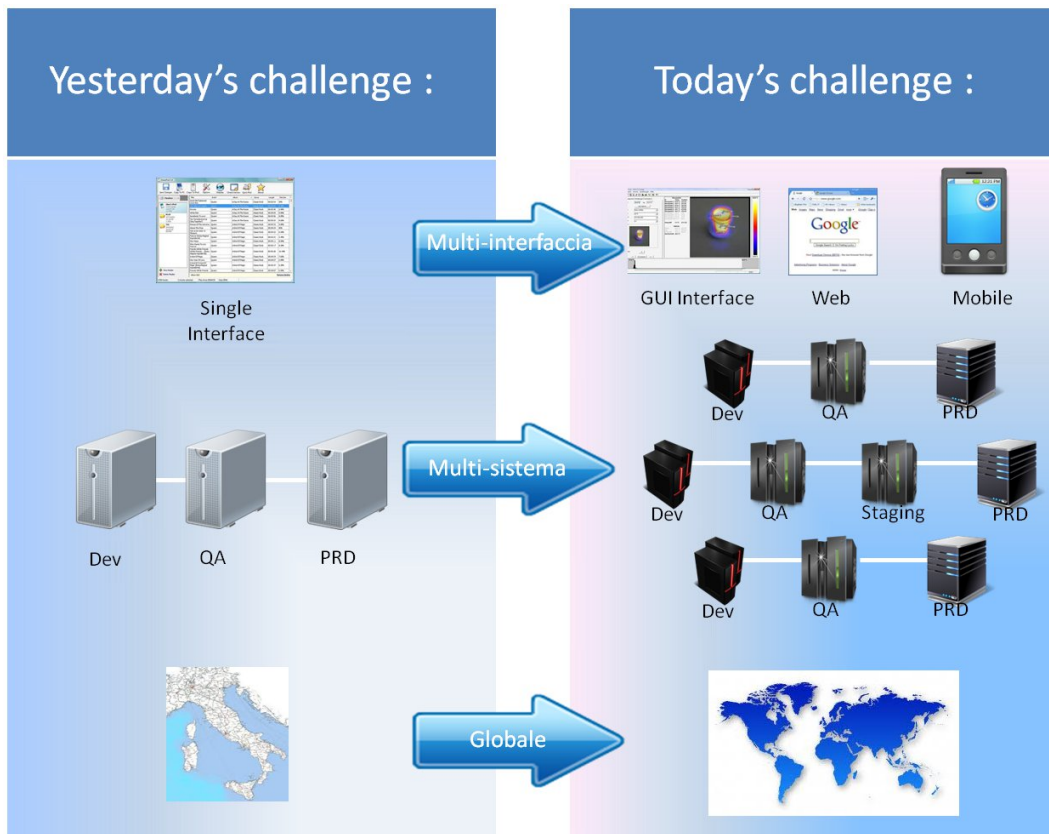


Figura 2.1: Mutamento dei progetti IT

che interessano e si intrecciano con una moltitudine di problematiche anche a livello internazionale. Oggigiorno si è davanti a sfide che in passato non erano previste : multi-interfacce per le applicazioni, la gestione di più sistemi e una globalizzazione che spinge sempre più verso una maggiore espansione.

Lo schema riportato in figura 2.1 analizza in maniera generale una visione odierna di un progetto IT. Il mondo dell'informatica è molto cambiato negli ultimi anni e la concorrenza sempre più agguerrita di nuove imprese nel settore IT ha spinto le aziende ad attuare politiche di espansioni in ogni parte del mondo. Il decentramento delle industrie nell'informatica ha sempre dato una serie di aspetti positivi come, ad esempio, maggiori guadagni e profitti derivati dallo spostare della produzione industriale in paesi dove la manodopera ha un costo inferiore. Intuitivamente in questo scenario globale sorgono anche molti

2.1 Panoramica delle organizzazioni IT

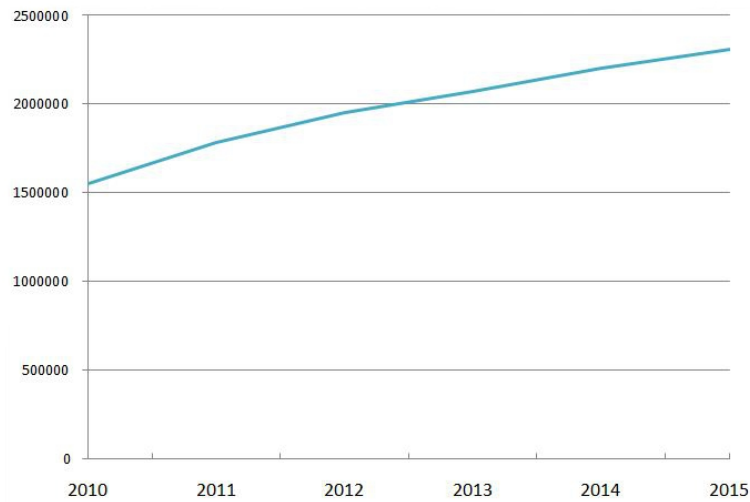


Figura 2.2: Previsione dispositivi mobili venduti

problemi tra cui un incremento delle difficoltà di gestione e di comunicazione che solo attraverso un più sistematico controllo è possibile in parte risolvere. La globalizzazione non è quindi solo un termine limitato al sociale, all'ambiente o alla cultura ma comprende un'infinità di aspetti non secondari che intervengono anche nella gestione delle attività dei processi industriali.

Un altro cambiamento in atto è lo sviluppo di nuove applicazioni su multi-interface. Questo aspetto risulta ormai molto interessante alla luce delle nuove statistiche sull'uso dei dispositivi mobili e portatili; una sfida ulteriore che project manager e responsabili di progetti devono affrontare sin dall'inizio.

Come si può capire dalle previsioni di vendita riportate in figura 2.2, le vendite a livello mondiale dei dispositivi mobili è destinata ad aumentare nei prossimi anni. Quindi come si può intuire l'uso degli smartphone continuerà ad aumentare. Lo si capisce ulteriormente anche dalle statistiche che le aziende fanno dell'utilizzo da parte dei consumatori di questi dispositivi. Infatti, secondo una recente ricerca di Google, il cui obiettivo è quello di capire il comportamento dei consumatori, si evidenzia come ormai le persone utilizzano gli smartphone sempre più in maniera assidua e costante. Infatti, questo è un tema che è al

2.1 Panoramica delle organizzazioni IT

centro dell'interesse di molte aziende IT che stanno proponendo nuovi prodotti mobile e creando nuove soluzioni per clienti che vogliono aumentare il proprio marketshare attraverso, appunto, gli smartphone. Le grandi aziende hanno già intuito questo potenziale ed è per questo motivo che, ormai, il mobile sta diventando sempre un maggior punto di forza per molte applicazioni. Anche durante la mia esperienza di stage in azienda ho potuto comprendere meglio come questo cambiamento sia in atto anche a livello enterprise. Lo sviluppo di nuove applicazioni su dispositivi mobili (smartphone, palmari, ecc.) è ormai un fenomeno accertato anche in ambito CRM e ERP. Come conoscenza diretta posso riportare l'esempio di SAP per il mobile. L'azienda leader nel mercato del software gestionale propone ai clienti nuove versioni multi-piattaforma per il canale mobile, interfacciate con i sistemi CRM e ERP. Queste nuove applicazioni sono dedicate al Sales Force Management e usate nella vendita ai punti vendita.

Un altro aspetto rivelante nello sviluppo di progetti rispetto al passato è il numero di sistemi e macchine coinvolte. I progetti possono risultare davvero complessi ed è necessario attivare un numero considerevole di server o macchine per poter portare a termine le varie operazioni di sviluppo. Il numero di sistemi è giustificato dal fatto che in progetti complicati e che si protraggono nel tempo anche per anni, molte applicazioni devono essere sviluppate e testate su ambienti diversi prima di essere portati a lavorare insieme.

Secondo l'esperienza lavorativa effettuata durante lo stage, le applicazioni che si andavano a sviluppare erano destinate al cliente stesso, e non per uso interno. In questo caso lo sviluppo di software è supportato generalmente da diversi tipi di ambienti di "lavoro" :

- *Development* sulla quale gli sviluppatori possono programmare le applicazioni ed effettuare prove o brevi test. Su questi calcolatori sono presenti un numero limitato di utenti, solitamente solo programmatori e tester. Per questo motivo tali calcolatori risultano generalmente poco potenti e, non di rado, si assiste a una virtualizzazione completa del sistemi su sin-

2.1 Panoramica delle organizzazioni IT

gole macchine fisiche. La virtualizzazione in questo caso permette, sotto certi aspetti, di ridurre i costi e le spese necessarie per attivare tutte le macchine. Tuttavia può capitare che a causa di prove non corrette si rovini l'integrità dei dati e risulti necessario ripristinare il database; con la virtualizzazione tutto ciò risulta più semplice e veloce e si ottiene risparmio di tempi e di costi. Detto ciò anche i database di solito hanno prestazioni non particolarmente veloci e i dati presenti sono adatti principalmente allo sviluppo di applicazioni specifiche sulle quali si sta lavorando. I database sono popolati dagli stessi sviluppatori in modo da avere un insieme di dati semplici e disponibili alla verifica veloce delle applicazioni sviluppate.

- *Quality* sulla quale vengono trasferite le applicazioni che hanno passato i primi test di validazione sulla sistema di sviluppo. Ogni applicazione che viene portata su questo calcolatore è testata successivamente anche da parte del cliente o da un team specializzato. Il software solitamente viene provato con dati più completi e diversi dai test precedenti, così da assicurare una migliore qualità del codice e scoprire casi particolari che sono sfuggiti agli sviluppatori. In questo ambito i server sono usati da un numero più elevato di persone e hanno un maggior numero di utenze nel sistema; questo fatto implica una potenza di calcolo maggiore sia da parte della macchina per l'elaborazione dell'applicazione sia da parte del database. Capita sovente che particolari errori vengano riconosciuti e trovati in questa fase; nel caso in cui l'errore sia di facile comprensione, gli sviluppatori cercano di risolverlo direttamente sul sistema di Quality stesso; in questo modo si tenta di ridurre i tempi e i costi che deriverebbero dal replicare l'errore sulla macchina di sviluppo e la sua successiva correzione. Se invece l'errore risultasse particolarmente importante e difficoltoso da risolvere, si ritornerebbe alla prima fase sulla macchina di sviluppo.
- *Production* sulla quale l'applicazione viene eseguita, dopo aver superato tutti i test e essendo stata autorizzata dal cliente al trasporto dal sistema

2.1 Panoramica delle organizzazioni IT

di quality a quello di produzione. Questa è l'ultima fase del ciclo di produzione del software, infatti in questo ambito si entra in produzione, cioè l'azienda che ha richiesto l'applicazione comincia a utilizzarla per le proprie attività commerciali ed economiche. Il numero di utenti del sistema risulta notevolmente maggiore rispetto ai casi precedenti e questi possono variare da poche decine a qualche decina di migliaia di unità. Quindi, nella quasi totalità dei casi, ogni server software si appoggia su una o più macchine fisiche. Per esperienza dei colleghi, che ho conosciuto durante lo stage e che svolgono progetti per clienti, è sconsigliato o quasi impossibile riuscire a testare su questo sistema le applicazioni software. I dati presenti risultano essere di elevato valore per l'azienda la quale non permetterebbe mai di testare qualsiasi tipo di applicazione direttamente in questa fase. Le conseguenze, come per perdite di tempo per sistemi fermati o perdita di dati, possono causare danni davvero incalcolabili. Infatti, questa fase è talmente delicata che spesso i clienti stessi, in generale quelli che dispongono di un reparto IT nel proprio organico, non autorizzano l'accesso al sistema agli sviluppatori esterni.

Ovviamente non ci esclude la possibilità da parte del cliente di richiedere un numero di sistemi maggiori per lo stesso progetto. Infatti, per assicurare una buona continuità del processo di sviluppo dell'applicazione possono essere installati più sistemi di quality, o in generale viene richiesto un sistema intermedio tra la quality e la produzione stessa. Chi sceglie queste modalità sa che andrà incontro a una maggiore spesa in termini di manutenzione e gestione dei vari macchinari, ma per progetti importanti e particolarmente tortuosi, una spesa iniziale leggermente più alta può portare a benefici di risparmio di tempo in caso di blocco dei sistemi. Tutto ciò va calcolato precedentemente durante le fasi iniziali del progetto e, tenuto conto delle priorità dello sviluppo dell'applicazione, affrontare una spesa maggiore potrebbe risultare più conveniente e sicuro.

2.1 Panoramica delle organizzazioni IT

2.1.1 Fattori che influenzano le decisioni

Ma quali sono i fattori che influenzano le decisioni da parte di un cliente nelle sue decisioni?

Rispondere a questa domanda è molto complicato; i fattori possono essere molteplici e i più disparati. Partiamo dal presupposto che un cliente è "qualcuno che acquista un servizio o un bene", e nel caso del settore informatico, il bene è un servizio IT o un prodotto software. Tipicamente nelle grandi aziende i clienti possono essere rappresentati anche da una business unit o da un'organizzazione interna della azienda stessa. I diversi clienti infatti hanno differenti necessità di servizi. Un esempio di fattore decisionale, a cui ho personalmente assistito e ho avuto modo di approfondire durante il mio stage è il seguente: quando un cliente decide di acquistare un servizio IT, lo fa sulla base di un budget di spesa stabilito in precedenza e un primo fattore di decisione risulta essere proprio da chi acquistare il servizio. Questo soggetto è generalmente chiamato "Service Provider". Si possono identificare principalmente tre tipologie di "Service Provider" :

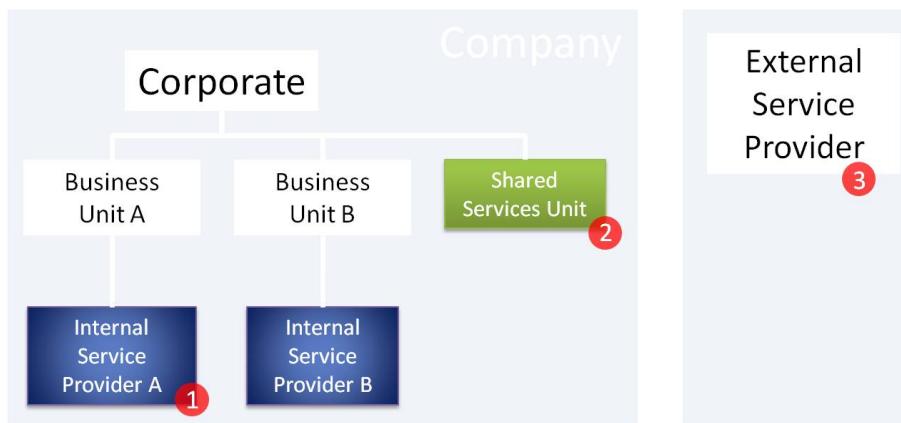


Figura 2.3: Tipi di Service Provider

1. *Internal Service Provider*: Un Internal Service Provider è una parte della

2.1 Panoramica delle organizzazioni IT

stessa organizzazione del cliente. Come mostrato nel punto numero 1 della figura 2.3, le unit A e B sono due esempi di internal Service Provider che forniscono i loro servizi alle altre unita di business. In questo caso ogni business unit ha il proprio Internal Service Provider.

2. *Shared Service Unit* : come altri servizi tipo finanza, Human Resource, ecc., l'IT in questo caso non è parte del business core dell'azienda, ma risulta come un'unità autonoma chiamata appunto Shared Service Unit. Il punto 2 della figura 2.3 riporta questo tipo di struttura. Tutto ciò porta certamente a una riduzione dei costi e permette anche un minor rischio nella gestione delle attività. Questa forma di service provider è quasi obbligatoria per le piccole e medie realtà aziendali.
3. *External Service Provider* : è definita come "Un service provider che è parte di una differente organizzazione ". Ciò significa che il service provider è un fornitore esterno all'azienda e il servizio è dato in outsourcing. Il provider in questo caso fornisce molte organizzazioni o clienti allo stesso tempo. Questo tipo di organizzazione è riportata nel punto 3 della figura.

La scelta di quale tipo di service provider adottare per i propri servizi è calcolata certamente sulla base dalla tipologia di struttura organizzativa e dalle dimensioni aziendali, ma aspetti come convenienza e efficienza produttiva risultano spesso determinanti per la decisione.

Questo esempio dimostra come la selezione di un service provider per un progetto IT è solo uno dei numerosi fattori che possono influenzare la progettazione e lo sviluppo, pur essendo un caso molto semplice e generale.

Infatti sotto vari aspetti è necessario cercare sempre di stabilire vari equilibri tra diversi fattori :

- Requisiti di business vs. Tecnologia : si deve trovare un bilanciamento tra una visione più attratta del servizio che si vuole offrire e una più pragmatica visione dei requisiti che si focalizzano sulla parte tecnica.

2.1 Panoramica delle organizzazioni IT

- Stabilità vs. reattività : esiste un tradeoff tra una gestione più conservativa basata sulla crescita con gli attuali sistemi e un crescita che si focalizza sulla crescita basata sulle nuove tecnologie.
- Qualità vs. Costi : con costi contenuti si raggiunge i requisiti che la parte finance del progetto pretende. Ma potrebbe essere possibile raggiungere un significativo aumento della qualità attraverso una quantità relativa di costo aggiuntivo. Raggiungere un buon equilibrio tra qualità e costi permette di assicurare una miglior riuscita del servizio IT.
- Reattivo vs. Proattivo : scelte diverse si possono fare per quanto riguarda le tempistiche dei requisiti. Si può rispondere alle necessità solamente quando il cliente lo richiede oppure anticipare i bisogni. Queste due possibilità portano a diverse conseguenze sia per i tipi di costi, sia per quanto riguarda la gestione delle attività.

Questi e molti altri sono i fattori e le scelte da tenere a mente durante una progettazione.

2.1.2 Investimenti e tendenze

In questo paragrafo viene dato un breve quadro dell'evoluzione degli investimenti in campo IT e nello sviluppo di software. Secondo le ultime ricerche gli investimenti nel settore IT non sono cambiati nell'ultimo anno nonostante le recenti vicende in Medio Oriente e in Giappone, anche se in quest'ultimo caso l'impatto sul mondo IT dovrebbe essere meglio calcolato.

Come mostrato nella figura 2.4, gli investimenti dopo il crollo del 2009 continuano ad aumentare, trainati dalle spese del "Computing Hardware". Comunque, le previsioni di crescita globale di tutto il settore IT si attesteranno attorno al 4% per i prossimi anni. Questo da certezze sulle possibilità da parte delle aziende del settore di poter ancora investire molto nell'immediato periodo.

2.2 Rischi e problemi di un progetto

	2008	2009	2010	2011	2012	2013	2014	2015	CAGR (%) 2010- 2015
Computing Hardware									
Spending (\$B)	379	333	374	409	438	469	505	554	8.2
Annual Growth (%)	-	-12.3	12.3	9.5	7.2	7.1	7.6	9.6	-
Software									
Spending (\$B)	228	222	237	255	273	291	309	328	6.7
Annual Growth (%)	-	-2.6	6.7	7.6	6.8	6.5	6.3	6.2	-
IT Services									
Spending (\$B)	804	763	785	824	862	901	944	989	4.7
Annual Growth (%)	-	-5.1	2.9	5.0	4.5	4.6	4.7	4.8	-
Telecommunications									
Spending (\$B)	1,977	1,905	2,011	2,110	2,186	2,271	2,355	2,430	3.9
Annual Growth (%)	-	-3.6	5.5	4.9	3.6	3.9	3.7	3.2	-
All IT									
Spending (\$B)	3,389	3,224	3,406	3,598	3,759	3,932	4,112	4,301	4.8
Annual Growth (%)	-	-4.9	5.7	5.6	4.5	4.6	4.6	4.6	-
CAGR = compound annual growth rate									

Figura 2.4: Previsioni di vendita mondiali per settore

Considerando anche il continuo sviluppo e diffusione degli smartphone tra le persone, si può pensare che molto denaro verrà investito anche in attività collegate all'utilizzo dei dispositivi mobili.

2.2 Rischi e problemi di un progetto

Dopo aver discusso di alcuni aspetti riguardanti lo sviluppo dei progetti IT, in questo paragrafo vengono introdotte tematiche che ho potuto riscontrare anche nella mia, seppur breve, attività lavorativa. Durante lo stage infatti, ho osservato che alcuni fattori con alto tasso di ripetitività, hanno un notevole impatto sulle performance del progetto. Questi fattori possono essere definiti come "fattori critici" per il successo di un progetto. In questa parte verranno analizzati, con diversi gradi di dettaglio, i fattori che più comunemente ostacolano la gestione di progetti informatici. I dati riportati sono estratti da conversazioni e discorsi effettuati con persone che hanno partecipato a grossi progetti per lunghi periodi

2.2 Rischi e problemi di un progetto

di tempo. Perciò le informazioni presentate fanno parte dell'esperienza lavorativa e sono direttamente collegate alle problematiche inerenti lo sviluppo del software.

Come evidenziato da alcune ricerche, i progetti informatici sono spesso iniziative che presentano un tasso di successo contenuto. Con il passare degli anni sono stati notati segnali di miglioramento, ma la percentuale di progetti fortemente in ritardo, con assorbimento eccessivo di risorse economiche o con importanti lacune qualitative, rimane ancora elevata. La tabella 2.1 riporta alcuni studi effettuati dalla seconda metà degli anni novanta in poi.

E' interessante notare come le problematiche relative alla gestione dei progetti siano, in un certo senso, "democratiche", cioè colpiscono tutti i tipi di aziende, a prescindere dalla dimensione e dal settore. Infatti nella letteratura sono riportati casi particolarmente importanti ed eclatanti di fallimento di progetti informatici. Si possono trovare casi di aziende che hanno bloccato l'installazione di sistemi gestionali ERP, dopo aver speso decine di milioni di euro, o ancora di eccezionali investimenti da parti di multinazionali del settore automobilistico costati oltre centinaia di milioni di euro e interrotti sbruscamente. La lista dei casi di fallimento potrebbe riempire molte pagine, e questi esempi citati sono solo alcuni.

Infatti, l'obiettivo di questo paragrafo non è elencare i problemi o le sventure delle aziende ma evidenziare, dove possibile, quali sono i fattori che normalmente contribuiscono alla problematicità dei progetti informatici.

Per ogni progetto è possibile ricercare una lista abbastanza ampia di fattori che ne hanno influenzato negativamente le performance. Dal punto di vista della singola azienda, la discussione di questi fattori è un metodo molto utile che permette di evidenziare su quali aree è possibile intervenire mantenendo comunque un collegamento con la situazione specifica.

La prima indicazione che emerge chiaramente dagli studi e dalle ricerche riguardanti i *Fattori Critici di Successo* è che le difficoltà raramente riguardano

2.2 Rischi e problemi di un progetto

Ricerca	Maggiori risultati
Standish Group-Chaos Report (1994 e 2004)	Nel 1994 il tasso di successo dei progetti informatici era il 16%, nel 2004 il 24%. Nel 2004 il 51% dei progetti erano considerati problematici, il 15% era fallito.
Robbins Gioia Survey (2001)	Il 51% degli intervistati considerava il progetto ERP non soddisfacente. Il 46% degli intervistati notava che l'azienda non comprendeva le potenzialità del sistema e quindi non c'erano stati miglioramenti sostanziali.
Conference Board Survey (2001)	Il 40% dei rispondenti dichiarava che dopo un anno dall'implementazione di un sistema ERP non si notava nessun beneficio.
Oxford University (2003)	Il 16% dei progetti informatici veniva considerato di successo, il 74% era problematico e il 10% abbandonato.
Royal Academy of Engineering and the British Computer Society (2004)	Solo il 16% dei progetti informatici si poteva ritenere di successo.

Tabella 2.1: Statistiche sul successo dei progetti informatici

2.2 Rischi e problemi di un progetto

le tecnologie adottate. Questo può apparire strano, poiché le tecnologie vengono spesso viste come complesse e fonti di problematiche. Sebbene a prima vista questo può essere corretto, è anche vero che la risoluzione di problematiche tecnologiche è maggiormente deterministica, ovvero, a fronte di un problema tecnologico, la raccolta di informazioni, la preparazione e la competenza dei professionisti sono le variabili determinanti per la loro risoluzione. Infatti le problematiche che insorgono nell'adozione nel proprio progetto di nuove tecnologie, risiede nel fatto di reperire risorse sia di tipo umano o infrastrutturale che aiutino a risolvere il problema. Inoltre, le interazioni tra le componenti tecnologiche, sebbene complesse, sono rappresentabili e schematizzabili oggettivamente e quindi risolvibili con metodo scientifico.

Sempre secondo l'autore dell'articolo, la stessa cosa non vale per altre variabili che influenzano l'andamento dei progetti. Le relazioni tra le persone, l'influenza dei cambiamenti tecnologici e le aspettative degli stakeholders, sono tutte variabili ad alto impatto che tuttavia, non possono mai essere rappresentate con certezza e oggettività. I margini di indeterminatezza sono sempre presenti e la gestione del progetto diventa, allora, la gestione dell'incertezza e della dinamicità.

Dagli studi emerge come i Fattori Critici di Successo dei progetti informatici possano essere classificati in due famiglie: i fattori legati alla modalità di organizzazione, gestione e controllo del progetto informatico, quindi alle competenze nell'applicazione del project management e i fattori relazionali, legati alla capacità di comprendere se stessi, gli altri e di gestire, nel migliore dei modi, le numerose relazioni che si instaurano in un progetto IT.

Per chi gestisce progetti informatici o è presente nel settore IT da molti anni, questi fattori problematici non saranno certamente nuovi. Ma l'obiettivo di questa parte è fornire indicazioni sulla gestione di alcuni fattori che, anche se spesso citati, raramente vengono analizzati in dettaglio. Essendo l'argomento complesso, non sarà possibile affrontare tutti i fattori con lo stesso livello di

2.2 Rischi e problemi di un progetto

analisi. La scelta degli elementi d'affrontare deriva comunque da osservazioni che ho potuto compiere durante il periodo di stage in azienda e da ricerche che ho realizzato. Nelle successivi paragrafi verranno analizzati quindi solo un numero ristretto di fattori ma che ritengo siano di maggior impatto per le performance di un progetto IT.

2.2.1 Aspettative degli stakeholders

Nei progetti informatici il tema della gestione delle aspettative è molto rilevante. Purtroppo si sta assistendo ad un fenomeno di banalizzazione della tecnologia, dove la percezione di chi non ha competenze informatiche porta a pensare che tutto possa essere risolto con un semplice tocco del mouse e in pochissimo tempo. A rafforzare questa percezione vi è anche la natura intrinseca dell'informatica, che è fortemente immateriale: complessi progetti si risolvono magari nella scrittura di un software che fisicamente può essere memorizzato su un DVD e questo non aiuta certo i clienti o gli utenti poco avvezzi alla tecnologia a percepire la complessità del software utilizzato. Il problema della creazione di aspettative poco realistiche risiede nel fatto che le persone giudicheranno il progetto basandosi su elementi inizialmente erranei. Se da una parte modificare le aspettative al rialzo è un elemento vissuto positivamente dalle persone, dall'altra l'abbassamento delle aspettative durante il progetto è un'azione molto difficile e dolorosa, poiché può portare alla disaffezione delle persone e a una loro minore volontà di collaborare; inoltre, può causare la formazione di giudizi più severi che non rispecchiano l'impegno di chi sta collaborando e gestendo il progetto. La situazione ottimale prevedrebbe che tutti siano in grado di apprezzare e capire qual è la complessità che risiede nei progetti e quindi formulare aspettative realistiche. Un approccio pragmatico deve invece considerare cosa realmente è possibile fare e non cosa sarebbe piacevole avere. È infatti non realistico pensare di essere in grado di fornire a tutti, quelle competenze che rendono apprezzabile la complessità dei progetti informatici. Dato che questo tipologia di problema è lontana

2.2 Rischi e problemi di un progetto

dall'essere risolvibile pienamente, si propongono quindi delle possibili soluzioni. Infatti, un aspetto importante a riguardo è rendere il più possibile oggettivo il lavoro svolto e quello ancora da svolgere.

2.2.2 Strategie sbagliate

Una strategia è la modalità di applicazione di un piano d'azione a lungo termine usata per impostare e, successivamente, coordinare le azioni tese a raggiungere uno scopo predeterminato. Quindi pianificare una strategia vuol dire utilizzare un metodo per raggiungere gli obiettivi desiderati. La pianificazione serve a chi è coinvolto nel progetto per organizzarsi, per coordinarsi con gli altri e per poter convergere verso lo stesso obiettivo. Per far sì che la pianificazione sia di supporto all'operato, è necessario che sia realistica, ovvero che riesca a rappresentare, seppur in modo semplificato, i fenomeni di interesse del progetto e che quindi venga considerata utile per svolgere le attività. In molti progetti IT si è riscontrato come la pianificazione sia stata completamente lontana rispetto alla reale esecuzione e gestione del progetto. Quando la pianificazione diventa un semplice "codice formale di buona condotta" del progetto, ovvero una documentazione che bisogna esibire al fine di dimostrare la propria professionalità; in quel momento diventa addirittura un impedimento al successo del progetto, in quanto richiede del tempo per essere sviluppata senza essere di alcun supporto alla gestione.

Un problema risiede nell'attività stessa di pianificazione, che consiste nella modellazione, e quindi semplificazione, di eventi futuri. Purtroppo quando ci si approccia alla attività di pianificazione si nota come venga naturale sostituire tempistiche, costi e utilizzo di risorse, non con quanto realmente potrebbe accadere nel nostro caso specifico, ma con valori medi, idealizzati, ritenuti normali.

Spessissimo accade che i rischi vengano completamente sottostimati e quindi si idealizzino delle attività che vengono svolte e completate senza intoppi, senza

2.2 Rischi e problemi di un progetto

interruzioni, senza errori. Una buona pianificazione deve invece tenere conto di questi fattori, in modo da essere il più possibile rappresentativa della realtà così come potrà accadere e non come ci piacerebbe che accadesse. Un problema insito nell'essere umano, e che si correla con la capacità di attribuire tempistiche reali alle attività, è la difficoltà di rappresentare mentalmente tali attività su un orizzonte temporale lungo. Un esempio deriva da un esperimento che ha dimostrato come il modo con cui ci poniamo le domande sul tempo necessario per svolgere delle attività, ha impatti fortissimi sulla realistica della pianificazione. A fronte, infatti, della domanda posta ad alcuni project manager di quanti mesi avessero bisogno per svolgere delle attività, la risposta era di circa un mese. La stessa attività tradotta in giorni portava invece a riconsiderare la risposta precedente e ad affermare che probabilmente sarebbero stati necessari due mesi. Questo perché l'essere umano è "progettato" nella sua percezione del tempo secondo il ciclo giornaliero. Il giorno è quindi la nostra unità fisiologica di pianificazione, il mese è solo una convenzione.

2.2.3 Comportamenti non corretti

L'intervento umano ricopre sempre un ruolo fondamentale in tutte le attività della produzione del software. In un progetto complicato e lungo si suppone che un team di sviluppo sia formato da professionisti con esperienze e abilità differenti fra loro. Perciò, ogni persona dovrebbe essere cosciente delle proprie capacità e cooperare insieme agli altri verso l'obiettivo comune. Talvolta però, si verificano comportamenti che pregiudicano le performance del progetto e ne mettono in seria difficoltà la buona riuscita. Questi comportamenti possono essere caratterizzati da atti non idonei all'interno di un team, poichè qualcuno per prevalere su un altro può arrivare persino a scontrarsi con i membri dello stesso gruppo. Tutto ciò spesso accade solamente per gratificazione personale, in quanto si vuole attirare l'attenzione su di sé e risultare più competente rispetto ad un altro, ottenendo così, magari, una posizione migliore all'interno del team.

2.3 Possibili soluzioni

Tuttavia, non tutte le situazioni sono simili e per ognuna ci può essere una diversa e specifica soluzione. Colui che può trovare tale soluzione e apportare modifiche e miglioramenti al progetto è il *project manager*.

2.3 Possibili soluzioni

In questo paragrafo, si tenta di offrire dei possibili rimedi operativi ai problemi esposti nei paragrafi precedenti. Ovviamente questa parte non vuole e non può certamente essere considerata come una risposta esaustiva e corretta per tutti i problemi affrontati, ma solo dare una visione di come si possa porre rimedio in alcuni ambiti.

Infatti gestire efficacemente tutti i fattori di ostacolo al raggiungimento dei risultati di progetto desiderati è forse un obiettivo impossibile. I progetti di sistemi informativi sono tipicamente caratterizzati da forte dinamicità e ciò rende difficoltoso individuare, in modo efficace, tutte le variabili che possono influire sulle performance di progetto.

Riprendendo i problemi sopra esposti si possono dare le seguenti correzioni operative :

prendendo in considerazione i problemi che possono insorgere a causa di aspettative distorte degli stakeholders, si può ritenere buona norma fornire delle informazioni sin dalle prime fasi del progetto, che siano di facile comprensione ai nostri interlocutori. Un primo passo potrebbe essere la stesura di un piano di progetto chiaro e comprensibile per chiunque, nel quale sia apprezzabile l'impegno e la complessità richiesta in termini di risorse e di tempi. Un altro aspetto che potrebbe apportare utili facilitazioni è la traduzione di elementi tecnici. Bisogna ricordare che al progetto possono partecipare anche persone che non hanno competenze informatiche adeguate per capire appieno tutte le problematiche. Al fine comunque di creare aspettative realistiche, risulta fondamentale la fase iniziale del progetto, la cosiddetta *Conception Phase*, quando le idee e le proposte vengono raccolte e discusse. In questa fase spesso si tende a sopravvalutare

2.3 Possibili soluzioni

le caratteristiche del progetto, portando alla formazione di aspettative distorte e non realizzabili e quindi ad una gestione maggiormente difficoltosa. In questo ambito è importante anche l'esperienza dei project manager e dei consulenti, che riescono, attraverso domande specifiche e interviste, a focalizzarsi sui fabbisogni espressi dal cliente, astraendosi quindi dalle modalità implementative.

Invece per quanto concerne le problematiche derivate da una pianificazione di strategia scorretta, solo l'abitudine a svolgere questo tipo di attività può cambiare questa modalità di relazionare le attività al tempo. Quando si pianifica, quindi, bisogna sempre essere consapevoli dei nostri limiti. Questo aspetto è ancora più rilevante nei progetti caratterizzati da profili con forti competenze tecnologiche, come spesso accade nei progetti IT. Come infatti evidenziato da Jerry Madden, project e program manager della NASA, nel suo studio su "I Fattori Critici di Successo", le persone con competenze tecniche sono tendenzialmente ottimistiche e tendono a sottostimare la presenza e l'impatto delle difficoltà. L'essere ottimisti è senz'altro una caratteristica positiva, ma ciò non deve portare all'essere non realistici, poiché si correrebbe il rischio di trovarsi quasi sempre in ritardo sui progetti, non solo per cause riconducibili a fattori esterni, ma anche per la stessa sottostima delle reali difficoltà che si possono incontrare. In merito a ciò, un esempio trovato in letteratura spiegava che in Microsoft, nei dipartimenti di sviluppo, molti manager applicano di default un raddoppio delle tempistiche dichiarate dai programmatori. Questa può risultare una cosa discutibile dal punto di vista della gestione, ma soluzione concreta al problema.

Sicuramente le soluzioni e risposte date in questa parte non sono sufficienti, o quanto meno, non sono sempre adottabili in tutte le circostanze. È comunque possibile identificare e focalizzare i propri sforzi su quei fattori che, tipicamente con numerosità limitata, sono in grado di influire pesantemente sulle sorti del progetto. Da quanto emerso, molti fattori sono correlati a capacità e sensibilità organizzative e di project management, più che a competenze puramente

2.3 Possibili soluzioni

tecnologiche. Ciò non significa che per un project manager dell'IT le competenze tecnologiche siano messe in secondo piano, ma rimangono fondamentali in quanto permettono di comprenderne più da vicino i problemi.

Competenze tecnologiche, organizzative e di project management devono essere tutte presenti e bilanciate in modo da poter affrontare in modo concreto ed efficace le difficoltà connesse alla gestione di progetti informatici. Seppur esistono studi che offrono una graduatoria dell'importanza dei diversi Fattori Critici di Successo, generalmente non affidarsi a tali soluzioni, ma personalizzare l'approccio sulla specifica realtà in base alla propria esperienza nei progetti. Valori medi sono di poca utilità, perciò è compito del project manager, del team di progetto e degli utenti collaborare per verificare quali siano i punti di forza e di debolezza del progetto in modo da fare leva sui primi e limitare i secondi.

Gestire progetti informatici è quindi certamente un'attività molto complessa ma allo stesso tempo stimolante da comprendere e realizzare. Ogni progetto possiede alcune caratteristiche peculiari che lo rendono diverso, ma non esistono ricette per il successo percorribile in tutte le situazioni.

Capitolo 3

Application lifecycle management

In questo capitolo verrà fornita una panoramica sugli aspetti riguardanti il concetto e l'adozione da parte delle aziende di uno strumento capace di controllare e gestire le attività di sviluppo e progettazione del software. Questo particolare sistema è chiamato Application Lifecycle Management. Durante questa spiegazione verrà introdotto le caratteristiche e le funzionalità che sono alla base di questi soluzioni software per gestione della vita del software. Gli aspetti che limitano l'utilizzo e i possibili difetti saranno parte conclusiva di questo capitolo.

L'ambito in cui si sviluppa questo capitolo è una conseguenza dell'analisi dei progetti software già discussa nel precedente capitolo. L'analisi presentata, pur essendo frutto di una personale esperienza di stage effettuata in una grande azienda di consulenza, prevede una serie di problematiche riscontrabili durante la progettazione di un qualsiasi prodotto o servizio IT offerto al cliente.

A seguito della grande complessità che risiede nei progetti, una serie di nuove sfide fanno la comparsa durante lo sviluppo e l'organizzazione dei prodotti, generando nei responsabili e project manager nuove preoccupazioni e perplessità. Laddove sia possibile ingegnerizzare la dinamicità dei processi e la gestione delle informazioni, nuove soluzioni sono proposte.

Un collegamento importante riguarda l'importanza che ancora oggi riveste il Software engineering. Infatti, pur essendo la figura ingegnere del software

CAPITOLO 3. APPLICATION LIFECYCLE MANAGEMENT

una professione piuttosto recente, può giocare un ruolo molto importante nelle organizzazioni IT, dove riesce a fornire concetti fondamentali, metodologie e strumenti per una produzione affidabile e di qualità. Non di meno lo è la disciplina dell'engineering, che non è solo focalizzata sullo sviluppo e creazioni di semplici programmi software, ma in realtà introduce una serie di aspetti complessi riguardanti il più delle volte il costoso *lifecycle* del software e dei prodotti derivati. Una buona definizione del concetto di software engineering che è spesso citata in letteratura è la seguente :

The practical application of scientific knowledge in the design and construction of computer programs and the associated documentation required to develop, operate, and maintain them.

Un'altra definizione di software engineering data da IEEE Computer Society è la seguente:

1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1).

Dalle definizioni riportate, si può stabilire che l'ingegneria del software fornisce una serie di strumenti capaci di sviluppare e gestire il ciclo di vita del software. I *processi software* si possono definire informalmente come un insieme di attività, metodi e pratiche che sono utilizzate per la produzione di software. IEEE [11] a tal proposito definisce un processo come "a sequence of steps performed for a given purpose" e più in dettaglio il processo di sviluppo software come "the process by which user needs are translated into a software product (. . .)". Da notare che è stato definito come "software process" per indicare, in una visione più generale, non solo il riferimento all'organizzazione complessiva, ma più in generale qualsiasi processo o sotto-processo utilizzato nell'attività di produzione o supporto del prodotto o servizio software.

3.1 Concetti di Software Process Engineering

Dal punto di vista del business, i processi che le organizzazioni software applicano per lo sviluppo dei prodotti e servizi, svolgono un ruolo decisivo nell'implementazione delle strategie e i relativi piani e obiettivi. Infatti le organizzazioni che sono in grado di controllare i loro processi, hanno un maggior grado di previsione rispetto alle caratteristiche dei loro prodotti e servizi, nonché un minor consumo di risorse e di tempo. Ciò permette di migliorare l'efficacia e efficienza del software e di conseguenza la redditività del loro business.

3.1 Concetti di Software Process Engineering

In questa parte verranno brevemente introdotti alcuni concetti relativi ai processi di software engineering, in modo tale da facilitare la comprensione dei paragrafi successivi.

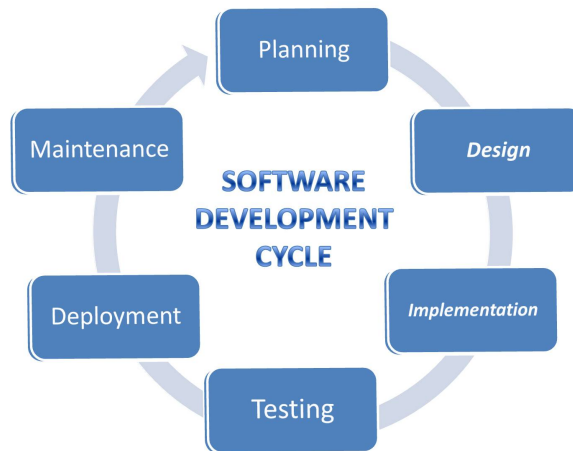


Figura 3.1: Ciclo di vita dello sviluppo software

Software development cycle Con l'espressione *software development cycle* si riferisce al modo in cui una metodologia di sviluppo o un modello di processo scompongono l'attività di realizzazione di prodotti software in sottoattività fra loro coordinate, il cui risultato finale è il prodotto stesso e tutta la documentazione ad esso associata.

3.1 Concetti di Software Process Engineering

Come riportato in figura 3.1 lo sviluppo di software è composta genericamente dalle seguenti attività:

- *Planning* : è forse la fase più importante nella creazione del software. In questa parte si estraggono i requisiti o le richieste del cliente. Il cliente tipicamente ha un'idea astratta del prodotto finale che vuole. Spesso i requisiti del cliente risultano essere incompleti, ambigui o addirittura contrastanti ed è proprio il compito del software engineer stabilire in maniera chiara e comprensibile anche al cliente quali sono i requisiti ideali. Il risultato degli incontri con il cliente sono generalmente una serie di documenti di analisi funzionale o di processi che devono essere convalidati dal cliente stesso. Una volta generati i requisiti, la successiva fase comprende la pianificazione dello sviluppo del software con i relativi tempi e costi calcolati anche sulla base delle informazioni provenienti dai programmatori. Certamente come spiegato nel capitolo precedente 2.2.2, in questa fase è necessario essere chiari e realistica rispetto agli obiettivi del cliente, senza quindi indurre in aspettative distorte.
- *Design* : una volta che il management ha approvato il piano di sviluppo e il budget è stato determinato per il progetto, in questa fase viene creato l'architettura del programma. Nella fase di Design, gli sviluppatori definiscono la struttura del software, scegliendo quale approccio tecnologico utilizzare e come definire i moduli funzionare.
- *Implementation* : è la fase di sviluppo del software stesso. in questa fase i programmatori sviluppano il prodotto software in base alle richieste del cliente. E' importante chiarire subito tutti le possibili incertezze e incomprensioni pratiche nelle primissime fasi iniziali di questo processo.
- *Testing* : la fase di testing risulta anch'essa fondamentale per il ciclo del software. Infatti questa fase vengono riscontrati i possibili errori nel funzionamento del programma. Inoltre vengono controllata la correttezza delle

3.1 Concetti di Software Process Engineering

funzionalità in base alle richieste del cliente. Ogni possibile errore o bug nel programma deve essere segnalata ai programmatori, che procederanno alla loro correzione e alla risoluzione dei problemi.

- *Deployment* : è la fase in cui il prodotto o programma vengono rilasciati al cliente. Dopo aver passato la fase di testing si procede a messa in funzione del software dal cliente o si procede alla vendita nel canale di distribuzione. In questa attività può essere inserita anche la parte che riguarda il software training e il supporto al prodotto venduto. Talvolta questo tipo di attività non viene realizzata, così facendo il cliente può perdere molto tempo nel cercare di capire tutte le funzionalità del prodotto. Infatti le persone sono piuttosto contrarie nel cambiare il proprio software usato da anni con uno nuovo e una buona attività di training aiuterebbe a superare quest'ostacolo.
- *Maintenance* : questa fase può comprendere una serie di attività atte a mantenere e migliorare il software. Dove si sono verificati errori o bug all'interno del programma una volta venduto, è necessario correggere gli eventuali problemi. Potrebbe essere anche necessario aggiungere nuove funzionalità nel prodotto che ne migliorano l'utilizzo o presentare versioni più aggiornate ai clienti che hanno già acquistato l'applicativo.

Software process models Esistono numerosi modelli per coordinare le attività di sviluppo. Ognuno di essi offre una serie di vantaggi e svantaggi a seconda dell'organizzazione del team e del tipo di software che si vuole produrre. Generalmente si utilizzano varie combinazioni di questi modelli; la figura 3.3 riporta alcuni schemi di modelli più utilizzati tra cui:

- Modello a cascata o "waterfall model" è un modello di ciclo di vita del software secondo cui la realizzazione di un prodotto software consta di una sequenza di fasi strutturata. Ciascuna di queste fasi produce un ben

3.1 Concetti di Software Process Engineering

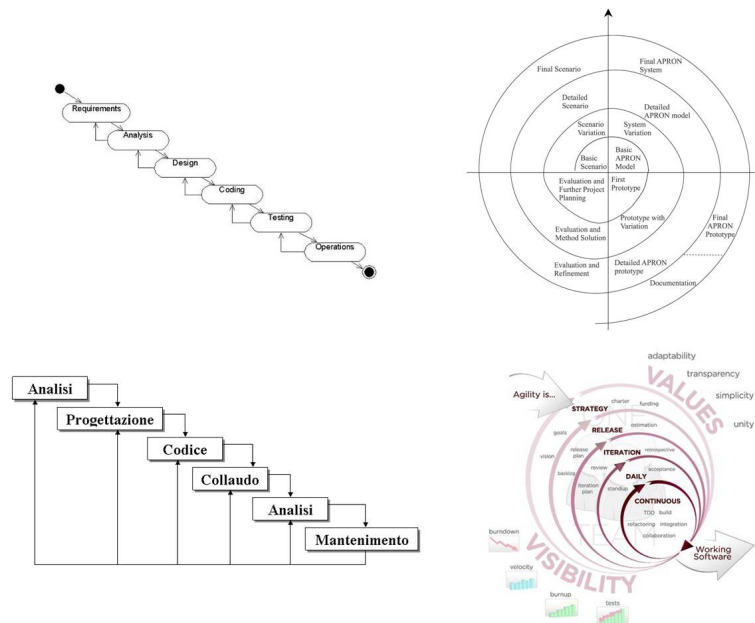


Figura 3.2: Modelli in software engineering

preciso output che viene utilizzato come input per la fase successiva; da qui la metafora della cascata.

- Modello evolutivo è uno dei modelli del ciclo di vita del software che cerca di superare i limiti principali del modello a cascata. Si basa sulla prototipizzazione che consiste nell'uso di specifici strumenti software per la realizzazione rapida di una versione semplificata del sistema informativo, con la quale sperimentare le sue funzionalità. La verifica del prototipo può portare a una modifica dei requisiti e una eventuale revisione del progetto.
- Modello a spirale è un modello del ciclo di vita del software che consente di rappresentare i diversi cicli di vita. Si basa, appunto, sul concetto di rischio, ovvero un insieme di circostanze avverse che possono pregiudicare il processo di sviluppo e la qualità del software. Il modello a spirale si concentra sull'identificazione e l'eliminazione dei problemi ad alto rischio tralasciando quelli banali. La caratteristica principale del modello è quella di essere ciclico e non lineare. Ogni ciclo di spirale si compone di quattro

3.1 Concetti di Software Process Engineering

fasi, il raggio rappresenta il costo accumulato e la dimensione angolare il progresso nel processo.

- Modello a fontana è un modello di sviluppo altamente iterativo, che ben si sposa con lo sviluppo del software con metodologie orientate agli oggetti, è particolarmente adatto a grandi progetti sviluppati da un gran numero di persone, specialmente se questi progetti riguardano prodotti "mission critical", che non possono fallire. Il modello a fontana riconosce che, nonostante alcune attività non possano iniziare prima di altre, c'è una considerevole sovrapposizione tra le attività durante il ciclo di sviluppo.
- Metodologia agile è rappresentata da un particolare metodo per lo sviluppo del software che coinvolge quanto più possibile il committente, ottenendo in tal modo una elevata reattività alle sue richieste. La gran parte dei metodi agili tentano di ridurre il rischio di fallimento sviluppando il software in finestre di tempo limitate chiamate iterazioni che, in genere, durano qualche settimana. Ogni iterazione è un piccolo progetto a sé stante e deve contenere tutto ciò che è necessario per rilasciare un piccolo incremento nelle funzionalità del software: pianificazione (planning), analisi dei requisiti, progetto, implementazione, test e documentazione.

Generalmente software process è una attività molto complessa che comprende un numero molto elevato di passi, risorse, artefatti, vincoli, etc. e l'uso di modelli per il software process riflette i vari gradi di questa attività così complessa.

Aspetti di cambiamento Il cambiamento nelle organizzazioni è spesso visto come un costo aggiuntivo alla scarsa qualità del servizio o un'esposizione e eccessiva al rischio per l'impresa. Quando tali cambiamenti si riescono a gestire significa che si può raggiungere una maggiore capacità produttiva, un migliore utilizzo degli asset ed, in modo particolare, ottenere un livello di flessibilità

3.1 Concetti di Software Process Engineering

operativa ed organizzativa, necessario per competere su qualsiasi opportunità di business. Infatti, nel contesto economico attuale, dove i servizi erogati ed erogabili diventano un asset aziendale tangibile e rivendibile, come un qualsiasi altro bene, le organizzazioni IT costituiscono un elemento critico in grado di condizionare pesantemente le performance aziendali. Considerando lo sviluppo di prodotti software sempre più legati agli aspetti di business, le considerazioni economiche e le strategie di mercato influenzano notevolmente la progettazione di nuove soluzioni che in tal modo permettono di creare un maggior valore e una riduzione dei rischi di fallimento.

Tradizionalmente nello sviluppo di progetti software si tenta di sottolineare in maniera troppo eccessiva le differenti fasi del progetto e così facendo di separare le funzionalità nell'organizzazioni con ruoli distinti e specifici. Questo influenza alcuni aspetti e fa emergere una serie di problematiche tra le differenti percezioni degli stakeholder nello stesso progetto. In particolari situazioni queste difficoltà possono essere anche amplificate con l'incremento delle dimensioni delle organizzazioni (team, reparti di sviluppo), che spesso si intrecciano con la gestione di un crescente numero di prodotti o servizi necessari per le comunicazioni o la gestione delle informazioni. Inoltre nello sviluppo di software, tutti questi aspetti devono essere sincronizzati o "allineati" con gli obiettivi strategici e di business per fornire in modo continuativo un supporto ai bisogni e ai rapidi cambiamenti dei processi e metodi all'interno dell'azienda.

Come analizzato nel precedente capitolo, in molte organizzazioni è possibile riscontrare una serie di difficoltà comuni in molti progetti, molti dei quali è possibile ingegnerizzare e gestire attraverso l'utilizzo di tool specifici:

- mancanza del riuso di artefatti che non comprendono solamente il codice sorgente dell'applicazione, ma anche i requisiti o il design del progetto
- motivi poco chiari riguardanti informazioni passate e decisioni non documentate come ad esempio requisiti o architetture specifiche del progetto

3.2 Concetto di ALM

- conseguenze non previste riguardanti i cambiamenti durante il mantenimento o miglioramento del software
- mancanza di integrazione tra product management e project management, in particolare quando la produzione del software evolve negli anni e sono presenti aspetti di manutenzioni e miglioramenti su più aree dell'applicazione
- incapacità di ricostruire gli stati passati di alcuni artefatti, in particolare il lifecycle di artefatti che non sono codice sorgente

Molte soluzioni sono state studiate e proposte negli anni per sopperire, almeno in parte, a questi aspetti. Tuttavia, le aziende cercano modi sempre più sistematici e più efficienti per supportare lo sviluppo e la gestione dei prodotti in un ambiente sempre più globale. Una soluzione recente offerta dalle grandi software house per rispondere alle sfide di mercato risiede nel concetto di Application Lifecycle Management (ALM). ALM infatti promette di risolvere un'ampia gamma delle sfide sopra citate e di essere in grado di supervisionare l'insieme dei processi e dei cambiamenti necessari, al fine di raggiungere gli obiettivi di business dell'azienda. Nonostante questo termine sia abbastanza recente e non abbia ancora una definizione comune, i maggiori vendor di questi prodotti evidenziando le loro capacità di gestire e integrare tool suite che coprono l'intero lifecycle del software. Infatti secondo la letteratura, una migliore gestione del ciclo di vita del software fornisce benefici sia in termini di qualità del prodotto, sia per quanto riguarda la riduzione del time to market.

3.2 Concetto di ALM

Come già esposto in precedenza, ALM è l'abbreviazione di Application Lifecycle Management. Il concetto di ALM è ripreso in molti documenti scientifici, ma con molte varianti in base al contesto analizzato. In altri casi, invece, si sostiene

3.2 Concetto di ALM

che la definizione di ALM non derivi da un termine scientifico ma da un concetto industriale concepito per definire questa nuova tipologia di prodotti.

Infatti, ricercando la definizione di ALM in motori di ricerca si può capire che questo termine non ha una definizione univoca, ma presenta sempre una serie di varianti. In numerosi articoli vengono spesso riportati le definizioni di vari software project, test managers, o di chi riveste un ruolo decisionale in progetti IT. Secondo queste ricerche si può intuire che il concetto di ALM non è stato pienamente compreso, e ogni definizione fornita è frutto di esperienze dettate più dal ruolo assunto nel progetto che da un'idea generale delle funzionalità offerte da queste soluzioni ALM. Ciò dimostra che la confusione spesso è alta, e in modo particolare per chi dovrebbe gestire e organizzare progetti e risorse all'interno dello sviluppo.

Spesso il significato si differenzia anche in base allo scopo di commercializzazione e di marketing svolto dalle aziende, ossia quali funzionalità del prodotto vogliono enfatizzare.

Ad esempio, il vendor Borland, con la sua soluzione ALM, non definisce chiaramente il concetto; spiega solamente che è una integrazione di tool che vengono utilizzati durante lo sviluppo dell'intero ciclo di vita dell'applicazione, senza esplicitamente definire cosa effettivamente è un *application life cycle*. Tuttavia, il focus del loro marketing si concentra solo sulla creazione delle applicazioni e non copre in modo decisivo l'intero ciclo di vita. Un altro esempio di questa definizione è dato dalla nota società di consulenza Capgemini che definisce ALM come :

"ALM is a standardised approach for the management of applications during their whole life cycle. ALM has the goal of maximizing the functional and technical life cycle of application by offering the current required functionalities" (Capgemini, 2008a).

Una definizione simile è fornita da una fonte più indipendente; Forrester Research definisce ALM come:

3.2 Concetto di ALM

"The coordination of development life-cycle activities, including requirements, design, development and testing, through: 1) enforcement of processes that span these activities; 2) management of relationships between development artefacts used or produced by these activities; and 3) reporting on progress of the development effort as a whole".

Ma un aspetto importante introdotto appunto con questo nuovo concetto risiede soprattutto in un collegamento tra la parte di gestione del business e la parte più operativa del software engineering. Sul sito di wikipedia viene appunto inserito nella definizione : "ALM is the *marriage of business management to software engineering made possible by tools that facilitate and integrate requirements management, architecture, coding, testing, tracking, and release management*". Questo aspetto secondo una mia personale opinione rende possibile una reale distinzione tra i diversi prodotti del passato e ALM. Come riportato da [29], le soluzioni ALM affondano le proprie radici in un concetto passato : Configuration Management. Configuration Management, o più precisamente dal punto di vista della gestione del software, Software Configuration Management (SCM) è una disciplina che fornisce gli strumenti e i processi per controllare la produzione del software. Infatti, come ALM, un configuration management ha lo scopo di controllare e gestire le attività, sia documentali sia implementative, che portano alla produzione di software. Questa attività si appoggia quasi sempre su un database dove sono memorizzati gli oggetti sottoposti a controllo di configurazione, ovvero i *configuration items*.

Secondo "Estublier (2000)" le funzionalità di un SCM si possono dividere in tre classi principali: repository per components, supporto per attività di ingegneria e controllo e supporto dei processi. Ma tipicamente, pur essendo SCM basato su di un concetto di filebased, cioè con granularità della gestione il file stesso, non si riesce bene a gestire il ciclo di vita degli artefatti (documenti, codice sorgente, test data, ecc). Questa e altre lacune vengono colmate con introduzione

3.3 Caratteristiche ALM

delle soluzioni ALM che ne fanno una delle funzionalità principali.

3.3 Caratteristiche ALM



Figura 3.3: Application lifecycle management

In molti documenti si afferma che sebbene si pensi che sia solo un prodotto commerciale, ALM fornisce molto altro ancora. Secondo l'autore dell'articolo [8] per capire i veri scopi di un ALM è importante spiegare i seguenti punti:

- ALM è da considerarsi una disciplina oltre che una categoria di prodotto. Sebbene molti vendor forniscano soluzioni ALM, è possibile realizzare tali soluzioni anche senza l'utilizzo di tool di supporto. Infatti, sempre secondo l'autore, esistono tre pilastri su cui si reggono i prodotti ALM : tracciabilità, processi automatici e reporting e analisi. Queste attività possono anche essere svolte manualmente con la stessa efficienza ed efficacia dei tool integrati.
- ALM non gestisce una specifica attività di lifecycle. Con questo punto si vuole sottolineare che una caratteristica fondamentale del concetto di

3.3 Caratteristiche ALM

ALM risiede, non tanto, nella gestione dei cicli di vita di un prodotto, ma, piuttosto, nel loro coordinamento e nella loro sincronizzazione. Infatti lo sviluppo può andare in contro a fallimento anche se le attività di analisi funzionale, la costruzione dei modelli o l'attività degli sviluppatori sono state eseguite in modo corretto. Lo scopo dell'ALM è infatti quello di assicurare un coordinamento tra le attività, e il mantenimento degli obiettivi di business nel rilascio dell'applicazione.

- Una soluzione ALM può essere integrata con altri tool di supporto, e non semplicemente con una collezione di questi elementi. Infatti, tool che possono risultare efficaci nella produzione di software, possono essere integrati per migliorare le performance dell'azienda nel rilasciare i prodotti. Tali tool, come ad esempio IDE, strumenti di build management e test management possono essere collegati e tenuti insieme attraverso, appunto, una soluzione ALM.

3.3.1 Principali elementi

Secondo [5] i principali elementi che si possono riscontrare in molte soluzioni ALM sono caratterizzati dalle seguenti proprietà:

- Tracciabilità : con il termine tracciabilità si intende la possibilità di ricostruire la relazione fra i diversi documenti prodotti nel corso di un progetto di sviluppo software, inclusa la stessa implementazione del sistema in uno o più linguaggi di programmazione. In particolare, la possibilità di ricostruire le relazioni degli elementi di un progetto con le specifiche dei requisiti iniziali viene detta tracciabilità dei requisiti. La correlazione tra i diversi cicli di vita degli artefatti, come requisiti, modelli, codice sorgente, script di build e test cases, aiuta a comprendere meglio se i requisiti di business sono stati rispettati. La tracciabilità è un aspetto di qualità di un progetto software fondamentale per una vasta gamma di attività, come l'analisi

3.3 Caratteristiche ALM

degli impatti di un cambiamento di requisiti, la verifica della correttezza di un'implementazione, il testing e il regression testing.

- Version control: è la gestione di versioni multiple di un insieme di informazioni. Durante il ciclo di vita dell'applicazione, nuove versioni del software vengono rilasciate ed è necessario tenere traccia di tutte le informazioni relative alle evoluzioni, in particolare al codice sorgente, alla documentazione, e ai requisiti del progetto.
- Misurazione : Il recupero di informazioni sui prodotti, i processi e le risorse, nonché le loro relazioni, è la base per effettuare una valutazione oggettiva e trasparente. Il ruolo della misurazione è un aspetto essenziale per la gestione dei progetti software, che deve essere gestito durante l'intero ciclo di vita dell'applicazione. Con un processo di valutazione è possibile intervenire e migliorare le prestazioni dei processi in funzione degli aspetti di business.
- Workflow support : workflow consiste in una sequenza o flusso di attività, operazioni, risorse, regole e informazioni che permettono di raggiungere il risultato desiderato. In ALM si promuove un approccio che favorisce l'interoperabilità dei tool e dei processi per stabilire un workflow tra l'intero ciclo di vita del software.
- Automatizzazione dei processi alto-livello: ALM fornisce una serie di automatizzazioni che rendono più efficienti alcune procedure per lo scambio di documenti e la memorizzazione di tutti i documenti associati. In molte organizzazioni si è soliti lavorare attraverso lo scambio cartaceo di documenti riguardanti le funzionalità, come ad esempio l'analisi funzionale o il design e il build o il testing. Da questo punto di vista ALM migliora l'efficienza, riducendo i tempi morti e quindi, migliorando il time to market del prodotto.

3.3 Caratteristiche ALM

- **Visibilità** : molti dei manager hanno una visibilità limitata durante lo sviluppo del progetto. La visibilità che hanno è data di solito da impressioni o testimonianze soggettive dei collaboratori, piuttosto che da dati oggettivi e concreti. Inoltre, in molte aziende si riscontra il problema che la reportistica delle informazioni sui progetti è la stessa che si svolgeva molti anni prima, senza aver portato nessun cambiamento effettivo. Molte attività vengono svolte ancora manualmente senza una precisa tracciatura dei documenti.

L'identificazione di questi elementi hanno un forte impatto nell'integrazione di soluzioni ALM all'interno di una organizzazione. Si possono individuare due principali categorie su cui è possibile focalizzare questi elementi : la prima corrisponde a tutte quelle attività per migliorare e supportare le operazioni ingegneristiche che coprono tutto il ciclo di vita dell'applicazione, che comprendono l'utilizzo dei tool e i processi di produzione. Una seconda categoria comprende invece una serie di attività che promuovono un collegamento tra le percezioni manageriali con la parte tecnica del progetto. Forse questo tipo di attività risulta davvero l'aspetto innovativo introdotto in queste soluzioni ALM negli ultimi anni.

3.3.2 Ruoli e processi coinvolti

La gestione del ciclo di vita di un sistema software è un'attività complessa che richiede l'interazione di un numero molteplice di partecipanti. I ruoli coinvolti in un progetto software sono mostrati in figura 3.4.

Come definito nei paragrafi precedenti non è semplice stabilire una definizione rigorosa di Application Lifecycle Management. ALM include l'intero tempo speso per la produzione del software, durante il quale le organizzazioni investono un notevole quantitativo di denaro. Secondo lo studio effettuato da [3], si possono distinguere generalmente tre differenti aree in cui il ruolo di una soluzione ALM ha particolare influenza.

3.3 Caratteristiche ALM



Figura 3.4: Ruoli e processi coinvolti

La figura 3.5 mostra su una linea temporale le principali aree in cui si svolgono le attività dell'intero ciclo di vita del software.

Come sostenuto nell'articolo, nel ciclo di vita di una applicazione sono riportati gli eventi più significativi che ne hanno segnato lo sviluppo e l'utilizzo.

Lo scopo della governance è di cercare di fornire applicazioni capaci di soddisfare i bisogni del business. Come riportato in figura 3.6, l'attività di governance è attiva per tutta la durata del ciclo di vita del software. Il primo step in ALM governance comprende lo sviluppo dei business case, i quali sono un'attività che precede l'inizio dello sviluppo vero e proprio dell'applicazione. Una volta che i business case sono approvati, lo sviluppo dell'applicazione può iniziare. Quando l'applicazione è stata completata e rilasciata, questa diventa parte del portfolio delle applicazioni dell'organizzazione.

Una volta che i business case sono stati approvati, lo sviluppo del software può avere inizio. Come mostrato in figura 3.7 se si espande la linea dello sviluppo, questa può essere probabilmente rappresentata una serie di sviluppi iterativi raffigurati con processi SDLC (Software Development Life Cycle) o da processi di Maintenance. Ogni iterazione potrebbe contenere una sequenza di documenti

3.3 Caratteristiche ALM

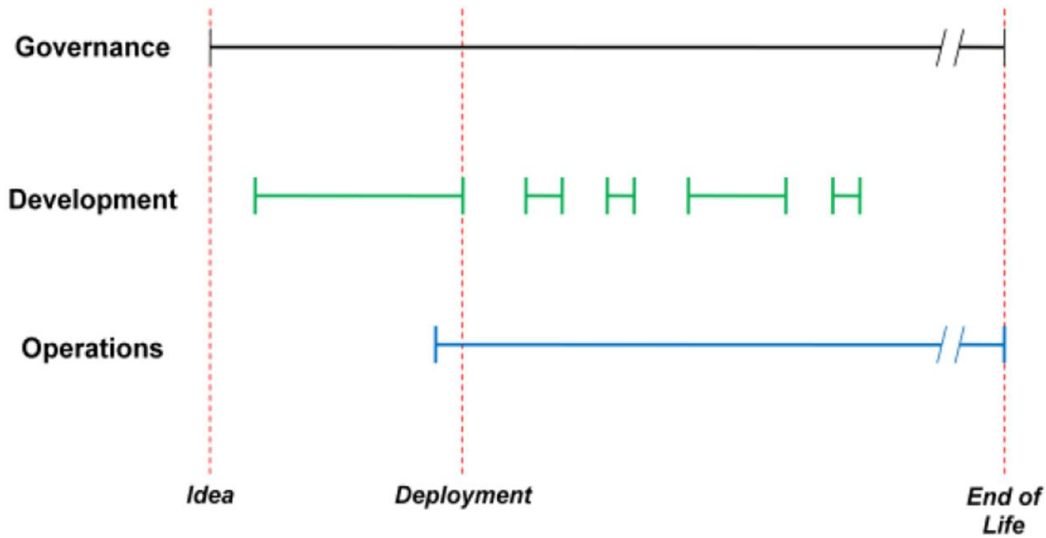


Figura 3.5: Tre aree in cui agisce ALM

come definizione di requisiti, design o sviluppi o testing. Tuttavia questo tipo di sviluppo iterativo non è sempre appropriato e per alcuni progetti è necessario utilizzare metodi più tradizionali.

Una volta che la prima fase di sviluppo è stata completata, si può procedere al rilascio dell'applicazione. Molti dei progetti però non si concludono con la fine dello sviluppo, ma proseguono con un periodo di aggiornamenti e supporto all'applicazione, come mostrato in figura 3.7. Se le fasi di aggiornamento non sono soddisfacenti nel migliorare le funzionalità del prodotto, si può intervenire procedendo con un'altra attività di sviluppo SDLC e con il suo successivo rilascio. Infatti, per molte applicazioni, gli investimenti che si effettuano per aggiornare una nuova versione sono spesso maggiori rispetto allo sviluppo di un nuovo software partendo dalle fasi iniziali.

In questo scenario si capisce l'importanza delle soluzioni ALM. In contesti molto complicati e soggetti a variazioni come lo sono lo sviluppo, l'aggiornamento o il supporto di un'applicazione, tenere traccia di tutte le informazioni, di tutti documenti, aiuta certamente a comprendere meglio quali sono i costi e i benefici di ogni mossa che si vuole apportare. Calibrare le risorse all'interno di un

3.3 Caratteristiche ALM

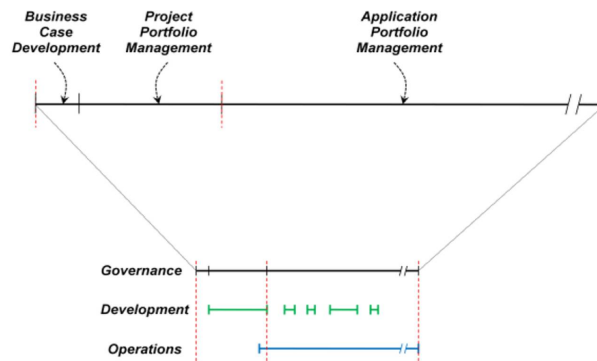


Figura 3.6: Governance in ALM

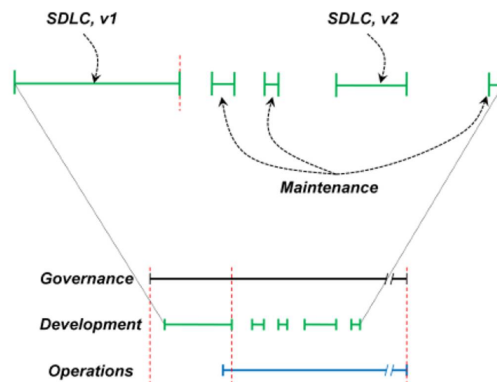


Figura 3.7: Development in ALM

progetto e calcolare le tempistiche di intervento possono certamente migliorare l'efficienza e l'efficacia dell'organizzazione, e quindi aumentare le performance generali.

Un altro aspetto che è necessario prendere in considerazione è, certamente, il monitoraggio e quindi la gestione dell'applicazione. Come mostrato in figura 3.8, una volta che la prima versione dell'applicazione viene rilasciata, il prodotto diventa un asset per l'azienda e quindi ha bisogno di un certo controllo e supervisione. Questa fase è collegata con la linea di sviluppo in maniera molto profonda. Infatti, è proprio lo studio dell'utilizzo dell'applicazione che permette di verificare che le funzionalità siano pienamente corrette e quando, invece,

3.4 Possibili approcci alla tecnologia

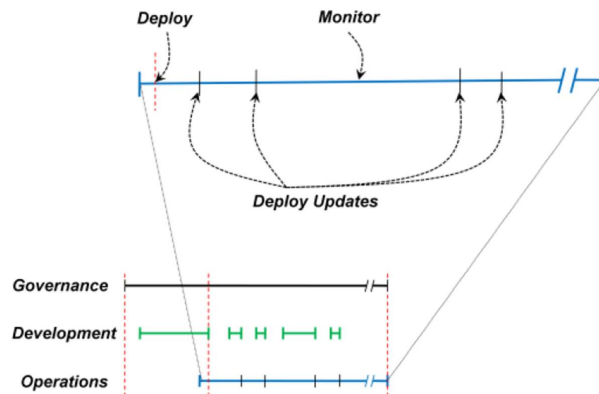


Figura 3.8: Operations in ALM

sia giusto apportare un aggiornamento. Questa fase di monitoring occorre per tutta la vita dell'applicazione e dunque, è un'attività che necessita di gestione e catalogare molte informazioni durante questo periodo.

3.4 Possibili approcci alla tecnologia

Secondo studi [8] effettuati sui principali vendors, si può notare che l'approccio alla vendita delle soluzioni ALM si differenzia in tre aspetti tecnologici:

- Single vendor platform : prevede che i vari vendor che utilizzano questa tipologia, mettano in risalto la interoperabilità delle loro soluzioni ALM e si aspettano che altri produttori forniscano i tool necessari per l'integrazione per la loro piattaforma.
- Multi-vendor platform : il framework è sviluppato attraverso l'utilizzo di una comunità open-source, dove esperti del settore possono aiutare a fornire supporto e guidare le richieste influenzando la direzione del framework o partecipare direttamente allo sviluppo.
- Single repository : questo approccio prevede che il vendor utilizzi un solo repository generale per tutte le informazioni. Questo prevede la costruzio-

3.5 Soluzioni ALM in commercio

ne di un insieme di tool o applicazioni per supportare la tracciabilità e il reporting tra tutte le discipline.

Ogni soluzione adottata dalle aziende possiede aspetti positivi e negativi, in base anche alle scelte strategiche ed economiche per la commercializzazione. Secondo [10], per esempio, l'utilizzo di un solo repository dove vengono concentrate tutte le informazioni raccolte, permetterebbe una più semplice produzione di metriche per il controllo dello sviluppo durante le fasi del ciclo di vita. D'altra parte questo tipo di approccio limita fortemente l'integrazione di tool esterni, che rimangono per certi versi "bloccati" nelle loro funzionalità, nelle differenti fasi del progetto. Infatti, capita che in particolari casi, sia necessario sostituire alcuni tool di sviluppo.

Multi-vendor platform possono essere viste come una buona soluzione che combinano sistemi specifici indipendenti; ma le limitazioni tecnologiche rendono piuttosto difficile implementare tali soluzioni. Nonostante ciò, è soprattutto la poca volontà da parte delle diverse figure che compongono il progetto, ad impedire la concretizzazione di progetti di tale tipologia. Un esempio di questo tipo è la piattaforma multi-vendor inclusa in Eclipse denominata ALF project. Questo progetto offre un buon supporto per estendere le funzionalità di Eclipse attraverso l'utilizzo di un plug-in ALF (Application Lifecycle Framework). Pur essendo una buona soluzione come framework, il progetto ALF è stato chiuso a causa del poco supporto da parte dei vendor ed è stato interrotto nel 2008.

3.5 Soluzioni ALM in commercio

Negli ultimi anni molte soluzioni ALM sono state introdotte sul mercato. Questa vasta gamma ha permesso certamente una maggiore concorrenza tra i vari vendor ma ha anche ampliato le difficoltà dei clienti nella scelta del prodotto. Spesso, quando un'azienda deve selezionare una suite ALM, la scelta è guidata da diversi aspetti che riguardano i diversi ruoli svolti nei progetti (sviluppatori,

3.5 Soluzioni ALM in commercio

tester, manager, etc.). I "rappresentanti" di ogni settore valutano molti prodotti e fanno la loro scelta in base al loro campo di utilizzo. Ad esempio, chi effettua i test sul prodotto valuterà in maniera più soddisfacente i prodotti che hanno strumenti migliori nell'analisi automatica, mentre gli sviluppatori sceglieranno prodotti con build test superiore ad altri.

Di seguito vengono riportate le più comuni soluzioni che si possono trovare in commercio :

- VersionOne - la versione supporta una vasta serie di metodologie includendo DSDM [32], Agile UP [31] e Lean/Kanban [30]. La suite è composta da : Agile Team per i piccoli team che utilizzato la metodologia agile; Agile Enterprise una versione completa di ALM per team di grosse dimensioni e di esperienza; Agile Ideas, un portale customizzabile. Forniscono inoltre una serie di web service per l'integrazione con componenti di terze parti.
- Aldon - Aldon Lifecycle Manager. Questo soluzione utilizza un sistema multiplatforma SCCM[33], con la quale è possibile coordinare sviluppi su piattaforme complesse. Aldon supporta inoltre prodotti di terze parti per aumentare la versatilità delle proprie funzioni. Altre caratteristiche importanti comprendono la configurazione dei processi di lifecycle nelle sue diverse parti, una buona architettura IDE, e una interfaccia roles-based per i diversi ruoli nel progetto.
- Serena - Serena Dimensions 10. Serena è specializzata in soluzioni ALM per grandi realtà aziendali. La sua soluzione usa un repository per i requisiti, la configurazione, i change e i rilasci. E' inoltre disponibile una serie di modelli per lo sviluppo di applicazioni in ambito global distribution. Inoltre, questo tipo di soluzione permette anche di supportare una lista di processi automatici e funzionalità per il rilascio. Serena utilizza come framework di base Eclipse Application Lifecycle anzichè della sua piattaforma proprietaria.

3.5 Soluzioni ALM in commercio

- Polarion - Polarian Requirements. Questa soluzione è basata sull'utilizzo del Web, Wiki, Google Docs e MS Office, ma anche di framework open standard. Le sue principali caratteristiche comprendono traceability matrix, supporto CMMI[34], e workflow guidato dall'User Interface.
- IBM - ClearCase, ClearQuest. La divisione Rational Software di IBM fornisce una forte selezione delle applicazioni ALM che integrano o completano, in vari modi, i processi di sviluppo. ClearQuest è un potente tool automatizzato che può essere utilizzato per molti modelli di workflow. ClearCase è il tool di source code revision di IBM che può essere utilizzato con un gran numero di piattaforme e gestire numerosi file. La sua caratteristica principale è il MultiVersion File System(MVFS), utilizzato per supportare le viste dinamiche della mappatura del software.
- Microsoft - Visual Studio Team System. Microsoft Visual Studio Team System (VSTS) un repository integrato con i processi di sviluppo. Invece di avere un change, versioning e configuration basati su code change, VSTS si basa sugli elementi di lavoro, riducendo l'overhead dei programmatori nella gestione dei cambiamenti. Inoltre, VSTS fornisce un'interfaccia web per un migliore supporto allo scambio e alla distribuzione tra i membri del team. Ovviamente questa soluzione si integra con le versioni di .Net e con i documenti Office del sistema.
- Rally - Rally Enterprise. Rally fu la prima azienda a creare un SaaS (Software as a Service) data warehouse per lo sviluppo del software. Rally ha due versioni: la community edition, limitata nell'utilizzo fino a 10 utenti e la versione enterprise, ovviamente la versione commerciale, la quale fornisce utilizza la metodologia Agile processes con un sistema multi-tier per la definizione dei requisiti. Inoltre questa piattaforma è integrata con i più popolari software opensource e tool commerciali.

3.6 Sviluppi futuri : ALM e Software as a Service

Il concetto di hosting software venne introdotto nei primi anni del 2000 e si pensava che questo concetto non sarebbe mai stato usato nelle grandi organizzazioni, ma solo in ambito high-tech o a livello commerciale. Tuttavia, i molteplici benefici introdotti con SaaS, Software as a Service, attirano l'attenzione di numerose compagnie ad adottare questa nuova funzionalità di business.

Secondo il modello di distribuzione del software, Software as a service (SaaS), il produttore di software sviluppa, opera direttamente o tramite terze parti e gestisce un'applicazione web che mette a disposizione dei propri clienti via internet. Ciò permette ai clienti di non pagare per il possesso del software bensì per l'utilizzo che ne fanno.

Con la continua espansione di questo modello, sempre più organizzazioni IT si stanno affacciando a soluzioni Application Lifecycle Management "on-demand". I benefici che si vogliono ricercare in questa soluzione riguardano principalmente la grossa quantità di denaro che le aziende potrebbero risparmiare a fronte tipicamente di una sottoscrizione mensile per l'utilizzo del software. Infatti, ogni compagnia vuole rendere più efficienti le proprie risorse, riducendo le spese per gli investimenti e massimizzando gli output. L'utilizzo di soluzioni ALM in cloud tenta di ridurre le spese iniziali di un progetto IT riducendo sia i costi relativi a nuove installazioni di apparati hardware sia i componenti software. Quando un'azienda pianifica l'adozione di una soluzione software per la gestione delle proprie attività di sviluppo, dovrebbe tenere in considerazione le spese relative all'acquisizione di nuovo hardware, la sua implementazione, il supporto alla manutenzione e tutti i relativi servizi correlati. Con l'adozione di una soluzione SaaS ALM è possibile pianificare in maniera predeterminata i costi di gestione di queste soluzioni, attraverso appunto una spesa mensile costante. In molti casi, questo tipo di soluzione on cloud, renderebbe possibile implementazione di funzionalità solo quando sono realmente necessari e in tempi molto più veloci, con risparmi ulteriori.

Capitolo 4

Attività svolte in azienda

In questo capitolo verrà affrontato l'argomento riguardante il lavoro svolto durante il periodo di stage in azienda. La società in questione è una grande realtà della consulenza in ambito ICT, strategy, servizi tecnologici e outsourcing, che mi ha dato l'opportunità di lavorare sotto la supervisione di un tutor ad attività inerenti il settore CRM. Durante questo periodo di stage ho potuto lavorare e osservare progetti e soluzioni di grandi aziende internazionali e clienti importanti, nonché imparare nuovi metodi di lavoro e osservare problematiche nuove che difficilmente si possono riscontrare in ambito universitario.

Le attività che ho svolto insieme ad altri colleghi del team durante lo stage, sono collegate all'attività di sviluppo di alcuni scenari standard di SAP CRM, in particolare nell'ambito del *sales management*. Lo scopo primario di questa attività riguarda la necessità di fornire un sistema CRM funzionante che abbia la possibilità di essere testato con nuovi aggiornamenti ed essere a disposizione di altri team per valutazioni o per studio. Questo sistema costituirà in futuro una forma di demo tester utilizzata da tutti quei clienti che desiderano osservare un sistema funzionante con applicazioni o aggiornamenti particolari. Infatti, le società che sono "gold partner" di SAP e principali vendor di questa software house, hanno la possibilità di ricevere in anteprima molte aggiornamenti o feature innovative, e la creazione di un ambiente con queste caratteristiche

4.1 Processo di vendita

installate e configurate può risultare molto conveniente per incentivare il cliente ad acquistarle. Un altro aspetto importante di questa alleanza commerciale tra le due aziende e che riguarda da vicino anche la mia attività lavorativa, è quello di permettere ai professionisti del settore o ai consulenti SAP, di conoscere e imparare meglio le nuove funzionalità proposte nei sistemi, cosicché, si possa essere maggiormente preparati e pronti alle esigenze dei clienti.

Nei paragrafi successivi verranno introdotte alcuni concetti per comprendere meglio le attività di sviluppo dei processi svolti in questo capitolo. A tale scopo, si verrà presentato il concetto di "processi di vendita" e le sue principali caratteristiche. In seguito saranno introdotte alcune descrizioni dei tool e dei sistemi utilizzati.

4.1 Processo di vendita

Con il termine *processo di vendita* si intende il complesso delle operazioni coordinate attraverso le quali nascono, si sviluppano e si concludono le vendite dell'impresa. Tali attività possono essere molto diverse a seconda del tipo di prodotto in vendita, del posizionamento dell'azienda o dalle caratteristiche del cliente a cui ci si rivolge. Generalmente le fasi principali sono caratterizzate da tutte quelle attività che sono necessarie per cercare, acquisire, e mantenere i clienti.

La ricerca di un possibile acquirente parte dal lavoro svolto dalle attività di marketing durante le quali si tenta di analizzare le interazioni dell'impresa sul mercato e di pianificare tutte le azioni al fine di catturare l'attenzione dei clienti per proporre la vendita dei prodotti.

Una volta che il possibile acquirente è entrato in contatto con l'azienda si possono sempre creare collegamenti con le varie fasi del processo di vendita. Un prima fase prevede la creazione di una *Lead* che rappresenta una semplificazione della scheda cliente dell'azienda, proprio per rendere più veloce tutto il processo di gestione delle informazioni e per tener traccia di tutti i contatti stabiliti. Questa forma di gestione dei dati è molto utile per poter tenere sempre ben

4.1 Processo di vendita

distinte le attività da fare sui potenziali clienti, rispetto a quelle sui clienti reali con i quali si sono già svolte operazioni di vendita. Infatti, molte di queste Lead dovranno essere scartate a causa della differenza tra la volontà dell'acquirente e le proposte di prodotti e servizi dell'azienda.

Non appena è verificata da parte dell'azienda un reale interesse di acquisto del cliente, si può procedere alla creazione di un'*opportunità* di vendita reale. Questa fase si realizza attraverso la conversione di una Lead, sia quando l'interesse viene manifestato dal potenziale cliente, oppure, direttamente, quando l'interesse arriva da un cliente già acquisito. In entrambi i casi è opportuno registrare i prodotti o servizi richiesti per poter meglio comprendere e pianificare le possibili offerte dedicate a tali clienti. Solitamente se il cliente acconsente è possibile pianificare delle attività successive, come ad esempio programmare una chiamata da parte del call center o la visita di un rappresentante delle vendite per esporre meglio le informazioni o le offerte sulle tipologie di prodotti richiesti.

Se tale fase ha avuto successo, solitamente si può procedere verso un contratto di vendita. Tuttavia, una fase intermedia, che spesso si introduce prima della vendita effettiva, risulta essere la cosiddetta la "quotation". La quotation è la fase in cui si definisce l'offerta di vendita o la stesura di un preventivo da associare all'opportunità creata nella fase precedente. Anche in questa fase è opportuno segnare quali sono le esigenze del cliente o i servizi che sono collegati al prodotto che il cliente ha intenzione di acquistare.

La fase che concretizza la vendita del prodotto o servizio è rappresentata dalla creazione dell'ordine di vendita a partire dal preventivo precedentemente creato. Tuttavia, tale ordine può essere creato anche in seguito a una richiesta diretta del cliente fatta attraverso i canali di vendita messi a disposizione dall'azienda.

Infatti un aspetto fondamentale che oggi influenza notevolmente le decisioni dei clienti è rappresentato dalla disponibilità e dall'integrazione tra i vari canali di contatto. Infatti, la sfida odierna è concentrata sull'integrazione dei

4.1 Processo di vendita

vari canali distributivi, allo scopo di fornire al cliente un'interazione con l'organizzazione in modo consistente attraverso tutti i canali. I canali maggiormente utilizzati dalle aziende sono:

- Call Center e Contact center, per la ricezione di ordini o assistenza ai cliente. Generalmente questo canale é usato dai clienti solo per uso telefonico, ma non si escludono altre modi per comunicare, come ad esempio attraverso e-mail, fax, o chat.
- Internet, attraverso portali web sia in maniera B2B (Business to Business), dedicato ai fornitori o rivenditori dell'azienda, sia con tipologia B2C (Business to Consumer), dedicato al mercato retail dei consumatori.
- Rappresentanti o promotori, che si dedicano alle visite direttamente dai cliente presenti sul territorio. Ogni rappresentante é responsabile di una determinata area e di determinati clienti. Molto spesso questo tipo di canale di vendita é la conseguenza di un interazione precedente del cliente su altri canali, come ad esempio la richiesta di una visita avvenuta con attraverso il call center.
- Rivenditori o reti di concessionari, che offrono un sistema per la vendita indiretta al cliente. Inoltre, tale canale fornisce un supporto di assistenza per tutte le esigenze del cliente.

Un'ultima fase, che completa il processo di vendita, comprende l'assistenza al cliente nel periodo post-vendita del prodotto o servizio. La fase di "service" é alla base nel sistema di reclami e garanzia del prodotto nonché di notevole importanza per l'immagine complessiva dell'azienda.

Le informazioni registrate durante tutte le fasi del processo di vendita possono essere utili all'azienda per essere studiati analiticamente e portare a una serie di benefici, tra i quali :

1. accertare e controllare l'immagine dell'impresa sul mercato.

4.2 Sistema SAP CRM e Solution manager

2. verificare gli andamenti nel tempo degli ordinativi, del fatturato, della clientela, dei reclami.
3. confrontare gli andamenti suddetti sul territorio con riferimento a determinate zone di vendita.
4. controllare il funzionamento dell'organizzazione commerciale.
5. verificare e controllare la soddisfazione dell'acquirente.

4.2 Sistema SAP CRM e Solution manager

In questo paragrafo verranno presentati gli strumenti e i sistemi su cui ho svolto la mia attività lavorativa; tra questi il sistema SAP CRM, su cui si sono progettati e sviluppati gli scenari business e SAP Solution Manager, per il coordinamento e la gestione delle attività.

SAP CRM In sintesi, il termine CRM è usato per descrivere il software e le tecnologie che interessano le aree aziendali nel processo di acquisizione e di gestione dei clienti, in particolare nelle aree di vendita, del marketing e del servizio ai clienti. Il cliente diventa per certi versi il centro di interesse principale da parte dell'azienda, e attorno al quale si sviluppa il business.

SAP implementa nel suo CRM moltissime funzionalità e applicazioni per la gestione delle relazioni con i clienti.

SAP CRM è parte della SAP Business Suite, che comprende numerosi moduli e sistemi per la gestione e analisi di tutti gli aspetti business di una azienda. Infatti il sistema SAP CRM può essere pienamente connesso e integrato con gli altri sistemi: SAP ERP come sistema per le funzioni di back-end; sistema Business Intelligent (BI) come supporto per statistiche e analisi, utili per la gestione delle campagne marketing; infine SAP Supply Chain Management (SCM) per la gestione e pianificazione del ciclo di produzione.

4.2 Sistema SAP CRM e Solution manager

Marketing	Marketing Resource Management		Segmentation & List Management		Campaign Management		Real-Time Offer Management		Lead Management		Loyalty Management
Sales	Sales Planning & Forecasting	Sales Performance Management	Territory Management	Accounts & Contacts	Opportunity Management	Quotation & Order Management	Pricing & Contracts	Incentive & Commission Management	Time & Travel		
Service	Service Sales & Marketing	Service Contracts & Agreements	Installations & Maintenance	Customer Service & Support	Field Service Management	Returns & Depot Repair	Warranty & Claims Management	Service Logistics & Finance	Service Collaboration, Analytics, Optimization		

Figura 4.1: SAP CRM - Schema funzionale

La figura 4.1 mostra le principali funzionalità divise per settore di appartenenza. Marketing, Sales e Service sono le principali aree in cui sono ripartite i vari moduli della soluzione CRM di SAP. Per ogni settore che si desidera implementare e configurare sono presenti diversi canali di interazione con il cliente: interaction center, internet e rappresentanti vendita.

Il sistema SAP CRM comprende una serie di argomenti, alcuni dei quali non riguardano prettamente il sistema SAP, ma si rivolgevano più in generale ad aspetti gestionali delle attività dei sistemi CRM.

Un tema principale che sta alla base di questi sistemi CRM è proprio la figura fondamentale del cliente, denominato da SAP come "Business Partner" (BP). In realtà la figura del Business Partner è gestita da SAP nella maniera più generale possibile: un BP è un termine generico che si riferisce a qualsiasi soggetto, persona, organizzazione o gruppo, che è coinvolta in interazioni con l'azienda stessa. Per identificare e gestire tutti i BP del sistema è necessario che per ogni figura sia definita una "business partner role". Una BP role è un insieme di regole assegnate ai BP e usate per classificare le loro funzionalità. Ovviamente molte sono le tipologie standard di BP che sono già inserite nel sistema SAP; le figure più importanti possono essere classificate ad esempio come:

- Sold-to-party, è colui che genera l'ordine di acquisto della merce o servizio;

4.2 Sistema SAP CRM e Solution manager

- Payer, una persona o un'organizzazione che paga l'ordine;
- Ship-to-party, una persona o un'organizzazione che riceve la merce;
- Bill-to-party, la persona o l'organizzazione che riceve la fattura della consegna;
- Employee, identifica una persona dipendente dell'azienda;
- Contact person, una persona designata per dare informazioni o essere rappresentante per una organizzazione.

Inoltre nella gestione dei BP all'interno del sistema, è possibile gestire e controllare le relazioni tra i vari BP. Infatti una *"business partner relationship"* definisce la relazione business che si viene a creare tra due partner in base alla natura della loro relazione. La definizione di una relazione può essere di due tipi: one-way, quando la relazione si estende da un BP ad un altro, ma non viceversa; two-way, quando invece la relazione si estende da entrambi le parti. Ad esempio in SAP CRM è possibile avere una relazione one-way "has contact person" tra un'organizzazione e una persona, ma si utilizza "is contact person for" per definire la relazione opposta. Invece una semplice relazione two-way tra due BP è data dalla tipologia di relazione "partner of", che può identificare una relazione commerciale tra due aziende.

Oltre ai Business Partner sono molti gli aspetti gestionali che compongono un sistema SAP CRM. Infatti, le modalità di configurazione del sistema partono dai concetti dei vari modelli di organizzazione aziendale fino ad arrivare alla customizzazione di nuove *"business transaction"*. Il termine di business transaction è un concetto molto importante anche se viene usato nel sistema SAP in modo molto generale; infatti il suo significato, secondo SAP, è quello di definire e gestire i processi business secondo le attività richieste dell'azienda, fornendo delle specifiche strutture e funzionalità, che possono essere usate in vari transazioni. Ogni business transaction rappresenta quindi una sorta di documento, ma

4.2 Sistema SAP CRM e Solution manager

più in generale una transazione, nella quale viene coinvolto un BP. La struttura di una transaction dipende fortemente dalla categoria di appartenenza, che stabilisce quali parti risultano obbligatorie da completare, o quali tipi di BP vengono coinvolti. L'uso di queste particolari funzioni permette di semplificare notevolmente le operazioni necessari per il completamento delle transazioni business. Un esempio di ottimizzazione è rappresentato dalla possibilità di auto-completamento offerta dal sistema quando si inizia una transazione con un cliente. Un esempio pratico può essere offerto nel caso in cui un operatore del call center procede alla creazione di un documento per un ordine, nel quale alcuni campi sono da completare. Se il sistema è stato configurato, tali campi, come il nome, l'indirizzo o il numero telefonico del cliente, potrebbe essere completati automaticamente con le informazioni provenienti dal Master Data, riducendo i tempi e gli errori nella compilazione.

Tutti questi aspetti e molti altri risultano molto interessanti dal punto di vista della gestione dei processi e delle transazioni, anche se a primo impatto non risulta di semplice comprensione. Infatti, ogni transazione, che si viene a generare, coinvolge moltissimi aspetti legati alla struttura, sia del sistema che dei dati presenti sul database. E' necessario prestare molta attenzione ad ogni piccolo cambiamento nella configurazione del sistema, per non ritrovarsi a gestire errori o informazioni non corrette, derivate magari dall'inconsistenza dei dati.

SAP Solution Manager L'opportunità di aver collaborato all'interno di un settore dell'azienda, in cui si sviluppano e organizzano soluzioni *SAP CRM*, mi ha portato ad osservare che molte delle attività venivano eseguite attraverso l'utilizzo della soluzione ALM di SAP chiamata *Solution Manager*.

SAP Solution Manager è la soluzione ALM che SAP ha integrato nella Suite dei sistemi, per la creazione, la gestione, l'organizzazione e il monitoraggio di tutte le attività legate allo sviluppo di applicazioni in ambiente SAP. Solution Manager supporta quindi sia il lavoro svolto dal cliente, che inserisce nel proprio sistema una soluzione SAP, sia chi implementa e sviluppa l'applicazioni,

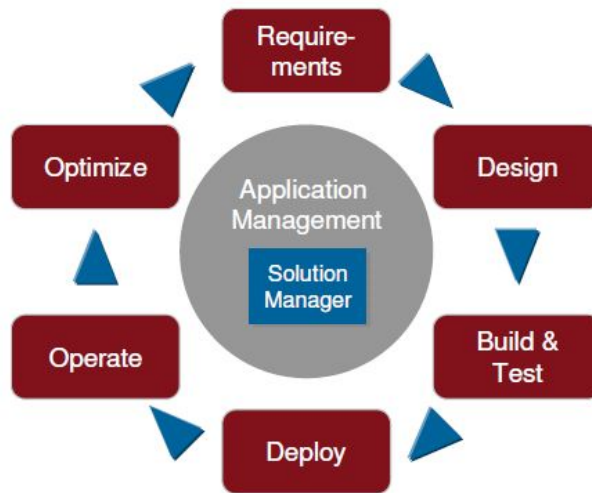


Figura 4.2: Solution Manager platform

solitamente rappresentato da un Partner SAP. Questo tipo di prodotto è stato introdotto nell'ambiente SAP agli inizi dell'anno 2001, e nel corso degli anni ha continuato ad ampliare le proprie funzionalità, grazie anche al supporto e alle osservazioni dei clienti.

Secondo "best practice"[35] di SAP, un progetto che vuole avere un buon grado di successo deve partire da una giusta metodologia di sviluppo. Solution Manager mette a disposizione una metodologia tale da garantire la massima efficienza e il raggiungimento degli obiettivi nei tempi e nei modi stabiliti. Tale metodologia, chiamata ASAP (Accelerated SAP)[36], è una soluzione globale per lo sviluppo dei progetti implementata da SAP e supportata in Solution Manager. Tale metodo di lavoro è stato creato per fornire un modello standard per ottimizzare l'intero processo per ogni tipologia di progetto: l'approccio *step by step* e i modelli proposti hanno l'obiettivo di aumentare la semplicità e l'efficacia dell'implementazione, garantendo in questo modo una notevole riduzione dei tempi e dei costi.

Tutta la documentazione e lo sviluppo di ogni singola applicazione all'in-

4.2 Sistema SAP CRM e Solution manager

terno del sistema viene organizzata all'interno di precise cartelle nel Solution Manager. Infatti, una volta che la preparazione del progetto è stato stabilita, si può procedere a organizzare i business blueprint[35], che descrivono i processi business e i requisiti di sistema, nonché tutte le attività e la documentazione relativa alla configurazione, agli sviluppi e al . Tutti questi elementi di trovano all'interno di una stessa base dati comune.

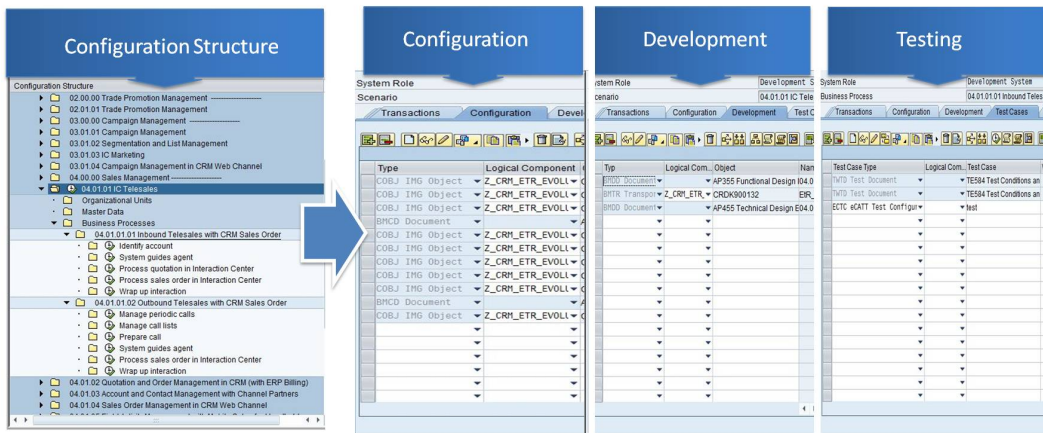


Figura 4.3: Solution Manager folder

Come mostrato in figura 4.3, una volta che la fase di preparazione del progetto è stata portata a termine, si può procedere alle successive fasi creando le relative strutture dei business process. Durante questa fase è possibile selezionare scenari business dal *repository standard*, comprendenti già alcuni documenti e schemi, o crearne alcuni customer. Nella configuration structure, infatti, sono riportate le varie cartelle e sottocartelle degli scenari di business che si desidera inserire nel progetto e i vari step di ogni processo.

Una volta conclusa, si può procedere alla fase per la gestione dei documenti, all'esecuzione della configurazione, allo sviluppo e al testing, direttamente dall'applicazione Solution Manager usando la struttura proposta da SAP, come mostrato in figura 4.3. Oltre che inserire documentazione riguardante il progetto, è importante notare che è possibile catalogare, nelle apposite cartelle, anche altri tipi di oggetti :

4.2 Sistema SAP CRM e Solution manager

- sezione configuration : IMG object, funzioni usate per configurare, Business Configuration Set, per la gestione e il rilascio delle modifiche, applicazioni Dynpro e contenuti NetWeaver;
- sezione development : programmi ABAP o Java, function modules, transaction;
- sezione testing : test document, casi di test automatici attraverso tool CATT;

Da ogni oggetto registrato e salvato nel Solution Manager, i membri del team possono accedere direttamente allo sviluppo dell'applicazione o customizzare alcune parti della configurazione. Tutte queste operazioni vengono registrate nel sistema, cosicché i responsabili del progetto possano controllare ogni progresso effettuato a ogni livello di dettaglio.

Un aspetto importante introdotto nelle ultime versioni di questa applicazione ALM di SAP, risulta essere la possibilità da parte del Solution Manager di creare processi strutturati nella fase di testing attraverso l'utilizzo di tool di Test Workbench. Questi tool permettono la pianificazione dei test plan, e, inoltre, forniscono una serie di vantaggi per quanto riguarda la gestione nella correzione degli errori. Infatti, i processi offerti in questa fase sono integrati con il Service desk. Questo permette ai tester di riportare all'interno del sistema gli errori ottenuti come "ticket", che verranno gestiti in seguito come messaggi di problema. Questo ambiente di testing all'interno del Solution Manager favorisce anche l'adozione di un singolo punto di contatto e la conservazione per tutto il materiale di testing e dei risultati ottenuti.

Oggi, SAP Solution manager mette quindi a disposizione una vasta serie di funzionalità e tool per il supporto da parte del cliente di tutto il ciclo di vita delle proprie applicazioni in ambiente SAP. In effetti questa ampia gamma di funzioni viene spesso definita come un "Swiss Army Knife", da molti clienti che hanno utilizzato questa soluzione per le proprie attività. Tuttavia, si ritiene che

4.3 Attività svolte

alcuni moduli rimangono insufficienti e difficili da utilizzare appieno sfruttando la loro capacità, e la stragrande maggioranza dei clienti utilizza solamente un limitato numero di tali funzionalità. Un altro fattore da considerare nell'utilizzo di questo tipo di soluzione per la gestione delle proprie attività risiede nel fatto che, pur considerando che Solution Manager è fornito gratuitamente all'interno del sistema, bisogna tener presente i costi aggiuntivi legati all'installazione di nuovo hardware, al training degli operatori e ovviamente all'implementazione.

4.3 Attività svolte

Il lavoro svolto durante il periodo di stage è rappresentato da una serie di attività atte a comprendere meglio lo sviluppo di soluzioni in ambito SAP CRM, in particolare nella gestione di alcuni "scenari business" di utilizzo. In questo ambito ho utilizzato alcune applicazioni per il coordinamento e la gestione, tra cui, in particolare, la soluzione SAP Solution Manager.

SAP definisce scenari business come un insieme di processi business che definiscono una serie di operazioni eseguite in modo completo e autosufficiente. Inoltre, i business scenari possono consistere in numerose varianti, ognuna delle quali descrive l'intero flusso dall'inizio alla fine; ogni flusso è rappresentato infine da una sequenza ordinata di processi business.

Le attività svolte sotto la supervisione di un tutor e in collaborazione con un team, sono rappresentate dallo sviluppo di alcuni scenari standard di SAP CRM, in particolare in ambito *Telesales* e *Campaign Management*.

Tuttavia, il lavoro svolto in questo progetto interno, cioè nel quale non si prevedeva la consegna ad un cliente, è da inquadrare in uno scenario ben più ampio e complesso. Le mie attività, infatti, sono solo una piccola parte e coinvolgono solo alcuni elementi dell'intero sistema.

Tuttavia, gli scenari presentati in questo elaborato di tesi, sono stati creati sulla base delle ricerche che ho effettuato, prendendo spunto dalle metodologie e dai manuali presenti in SAP e in altri sistemi CRM. Aspetto importante é

4.3 Attività svolte

stato il supporto da parte dei colleghi del team che mi hanno aiutato, attraverso la loro esperienza in altri progetti, a completare e definire meglio alcuni aspetti decisionali nei flussi.

Un aspetto decisionale preso per tutti gli scenari presentati è quello di offrire un sistema di semplice utilizzo ma, allo stesso tempo, che riesca a coprire le funzionalità dei casi più generali. SAP fornisce già alcuni processi standard che coinvolgono un buon numero di scenari e attività, ma non sempre si adattano alle situazioni più comuni.

Quindi, pur trattandosi di scenari poco complessi e caratterizzati da un singolo livello di profondità, le problematiche, affrontate durante la stesura dei documenti di analisi e la relativa configurazione degli scenari, rilevano comunque un buon grado di difficoltà e complessità generale. Da osservare, infine, che in un'attività reale di realizzazione e configurazione di uno scenario come quelli esposti, implicherebbe uno sforzo lavorativo notevole, calcolato in molte ore lavorative e diluito su un periodo temporale di parecchi mesi.

Per ogni scenario descritto, sono stati svolti tre tipologie di documenti, utilizzando la "naming convention" derivata da manuale interno all'azienda:

<xx.xx.xx>_<File Name>_< code type>

dove il primo numero indica la cartella dello schema blueprint in Solution Manager; tale numero è calcolato contando sia la posizione che la profondità della cartella in cui si trova il file. Il secondo elemento indica il nome del file che generalmente ne dà una descrizione sintetica. Invece il terzo elemento è scomposto nella seguente forma :

<Letters Workstream> <Digit1 Stage> <Digit2-3 incrementation>

Letter Workstream è formato da due lettere che identificano:

- BP per il Business Process Design
- AP per Application Design

4.3 Attività svolte

- TE per il Test Document
- TR per il Training e Performance Support
- DP per il Deploy

Il primo digit comprende invece gli stadi per lo sviluppo e il secondo e terzo un numero incrementale dei file.

I tre documenti creati sono stati nominati con il seguente codice:

- BP310 : Business Process Design, utilizzato per documentare le principali caratteristiche del processo e il flusso che lo descrive.
- AP322 : Configuration Design, nel quale vengono descritte, passo per passo, le attività di configurazione del sistema.
- TE584 : Test Document, nel quale vengono segnati tutti gli step da effettuare per effettuare il test funzionale.

Di seguito verranno introdotti tre scenari business : Telesales inbound e outbound e lo scenario Campaign Management.

4.3.1 Scenario Telesales

Per *Telesales* si intende la vendita o tentata vendita di un prodotto particolare o un servizio da parte di un venditore (Call Center), che effettua un approccio iniziale attraverso l'utilizzo di dispositivi telefonici. Questo scenario telesales é stato suddiviso successivamente in due sotto-scenari , secondo gli standard di SAP :

- Inbound Telesales : si riferisce ai processi relativi alla ricezione di ordini o alla richiesta di informazioni o ai problemi da parte dei clienti.
- Outbound Telesales : si riferisce alle attività relative all'esecuzione di chiamate di "call list", precedentemente elaborate.

4.3 Attività svolte

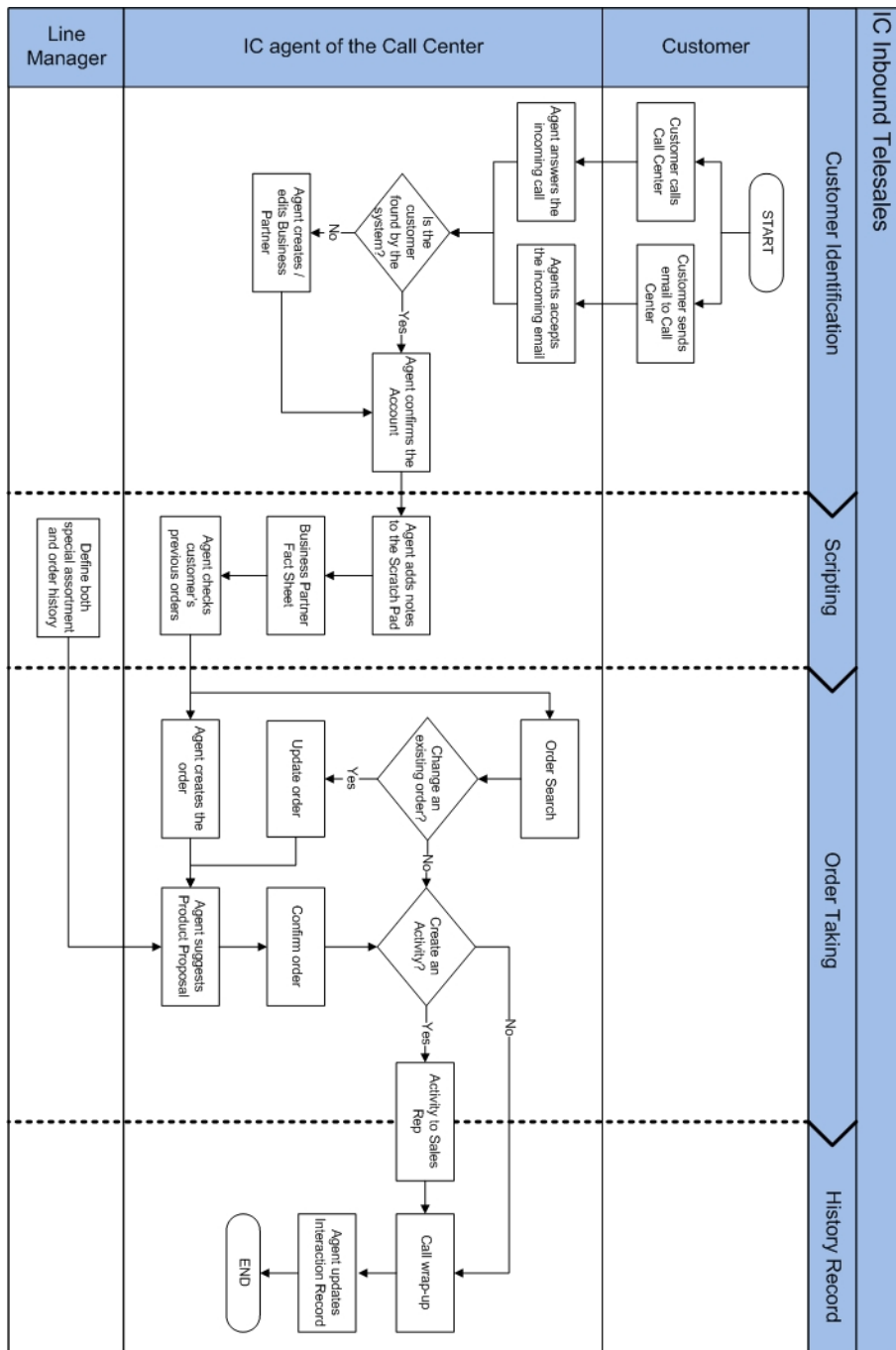


Figura 4.4: Inbound flow diagram

4.3 Attività svolte

Inbound Telesales Lo schema presentato in figura 4.4 descrive il flusso dello scenario telesales inbound. Tale scenario è stato creato per mettere in comunicazione un clienti e l'azienda con l'idea di creare uno scenario tipico nel quale un possibile cliente richiede informazioni sui prodotti o sui servizi prima di procedere alla creazione di un ordine. Per la creazione di questo scenario mi sono basato sulla documentazione e sui forum specifici messi a disposizione da SAP attraverso il portale, fornendomi una base sulla quale inserire successivamente altre attività e processi.

Infatti, una delle scelte fatte prevedeva che, oltre alla possibilità di richiedere informazioni ulteriori sulla stato delle transazioni precedenti, il cliente poteva autorizzare l'operatore del call center a effettuare un attività "follow-up" di visita da parte di un rappresentante o promoter.

La caratteristica base di questo scenario inbound é che il cliente può quindi chiamare o inviare un email, e successivamente l'agente del call center potrà eseguire gli step addizionali secondo le richieste del customer.

Il flusso riportato in figura 4.4 può essere suddiviso in sei sottoprocessi :

1. Un customer contatta il Call Center
2. L'agente del Call Center identifica e conferma il cliente chiamante
3. L'agente del Call Center consulta le informazioni del cliente
4. L'agente del Call Center esegue le richieste del cliente
5. L'agente del Call Center propone un prodotto aggiuntivo o alternativo
6. L'agente del Call Center conclude la chiamata del cliente

Nella descrizione successiva, alcuni sotto processi saranno ulteriormente divisi per descrivere meglio i differenti flussi del processo.

1. Un customer contatta il Call Center.
 - Primo scenario: incoming call

4.3 Attività svolte

- (a) Un customer chiama il Call Center : il cliente contatta il Call Center facendo una nuova chiamata. Questo step potrà essere simulato attraverso un tool chiamato Contact Center Simulator.
 - (b) Agente risponde alla chiamata : quando un customer effettua una chiamata all'agente, un segnale di allerta è mostrato sullo schermo dell'operatore del Call Center. Se il numero telefonico del cliente è registrato all'interno del sistema, i nomi dei contact person e il sold-to-party sono visualizzati sullo schermo. L'agente può decidere se accettare la chiamata o rifiutarla attraverso un pulsante apposito.
- Secondo scenario: incoming email
 - (a) Customer invia un'email al Call Center : una seconda possibilità è offrire al cliente di mettersi in contatto con l'agente del Call Center. Questo passaggio potrebbe essere simulato attraverso il Contact Center Simulator.
 - (b) Agente accetta l'email : Come per la parte delle telefonate, quando una nuova email è ricevuta dall'agente, un messaggio di allerta viene visualizzato. Come nel caso precedente, se il sistema riconosce l'email del cliente, i nomi dei contact person e il sold-to-party sono visualizzati sullo schermo. In seguito l'agente può decidere se accettare o rifiutare l'email

2. L'agente del Call Center identifica e conferma il cliente chiamante

- Primo scenario: il customer è automaticamente riconosciuto dal sistema

Il chiamante è automaticamente identificato dal sistema e i suoi dettagli, come ad esempio ordini passati e richieste recenti, vengono visualizzati in automatico, aiutando l'agente a capire meglio le volontà del cliente.

4.3 Attività svolte

- Secondo scenario: il cliente deve essere identificato manualmente

Se, per esempio, il cliente chiama il Call Center con un nuovo numero o utilizza un numero personale, o se i dettagli del chiamante non sono ancora mantenuti nel sistema, il customer non può essere riconosciuto in maniera automatica. In questo caso, l'agente deve ricercare il corretto account del cliente, inserendo nella ricerca una o più parole chiavi.

 - (a) L'agente crea o modifica il Business Partner

Se una chiamata in ingresso è fatta da un customer sconosciuto, l'agente ha la possibilità di registrare il nuovo cliente prima di proseguire nella telefonata. Se il cliente non è ancora presente nel sistema, l'agente può creare un nuovo Business Partner, il quale può essere riempito con i dati necessari e più importanti. Invece, se il cliente è già presente nel sistema, ma per esempio, il contact person che ha chiamato l'agente ha un nuovo numero telefonico, l'agente deve essere capace di aggiornare l'account del cliente, inserendo le informazioni aggiornate.
- Primo e secondo scenario
 - (a) L'agente conferma l'account.

Dopo un breve controllo con il chiamante, l'agente può confermare l'account riconosciuto dal sistema. Appena l'account è confermato, segnali di alert sono mostrati nella pagina Interaction Center, e Account Fact Sheet è automaticamente caricato. In questo sistema sono previsti due tipi di segnali di alert :

 - Gold customer: quando il customer è un cliente molto importante. Tale caratteristica è stata inserita precedentemente nel master data del sistema.
 - Numero di ordini di vendita aperti: quando il customer effettua un alto numero di ordini.

4.3 Attività svolte

(b) L'agente aggiunge delle note al Scratch Pad

Durante l'intera conversazione, l'agente è capace di annotare, in qualsiasi momento, informazioni importanti riguardanti il cliente, grazie a un tool dedicato, chiamato Scratch Pad. Queste informazioni possono successivamente essere importate nelle transazioni business come ad esempio gli ordini o le attività. Questa nuova modalità fa parte del nuovo aggiornamento introdotto con SAP Enhancement Package.

3. L'agente del Call Center consulta le informazioni del cliente

(a) Business Partner Fact Sheet

Come scritto in precedenza, Business Partner Fact Sheet è automaticamente visualizzato appena l'account è confermato. L'Account Fact Sheet fornisce le seguenti informazioni riguardanti il customer :

- Dettagli account
- Ordini storici
- Google Mash-Up (derivante dall'aggiornamento di SAP Enhancement Package 1)
- Interaction History
- Activity Overview

(b) L'agente controlla gli ordini precedenti del customer.

Attraverso il Fact Sheet, l'agente è capace di consultare gli ordini passati del customer allo scopo di analizzare il suo comportamento e le sue preferenze.

4. L'agente del Call Center processa le richieste del cliente

- Primo scenario: un nuovo ordine è richiesto

(a) Agente crea l'ordine

4.3 Attività svolte

Attraverso la schermata del Sales Order, l'agente è capace di creare un nuovo ordine per conto del cliente chiamante. Per migliorare la comunicazione, la sezione Notes può essere utilizzata per inserire informazioni, importate dallo scratch pad.

- Secondo scenario: il topic della chiamata è un ordine precedente

(a) Ricerca degli ordini passati

Attraverso la schermata Sales Order, l'agente ha la possibilità di ricercare gli ordini esistenti che sono stati precedentemente creati dal customer chiamante. Per far ciò sono presenti molte modalità di ricerca.

(b) Aggiornamento ordine

L'agente può rielaborare un ordine esistente allo scopo di aiutare il customer chiamante nelle sue necessità. Per migliorare la comunicazione con il cliente, la sezione Notes può essere riempita con i dati provenienti dallo scratch pad.

5. L'agente del Call Center propone un prodotto aggiuntivo o alternativo

(a) Definire la lista di prodotti speciali

Line Manager è il ruolo nel sistema responsabile del mantenimento del Product Proposal. Questa figura può cambiare la lista dei prodotti che dovranno essere proposti ai clienti durante le telefonate.

(b) L'agenti suggerisce i Product Proposal

Sia nella creazione di un nuovo ordine che nell'aggiornamento dello stato di una transazione, l'agente ha la possibilità di proporre una serie di prodotti al cliente, secondo appunto la lista dei Product Proposal.

Molte sono le liste di Product Proposals che sono disponibili nella creazione o aggiornamento di un ordine :

4.3 Attività svolte

- Listings : sono configurate allo scopo di suggerire una lista di prodotti top. Questa lista è gestita dal Line Manager
- Ordini passati: è possibile proporre prodotti in base all'interesse del customer, prendendo spunto dalle informazioni degli ordini passati.
- Cross Selling : vengono proposti prodotti che hanno una relazione con gli articoli inclusi nell'ordine attuale

6. L'agente del Call Center conclude la chiamata del cliente

- Primo scenario: un'attività di Follow-Up è richiesta
 - (a) Attività del Sales representatives

Accedendo alla schermata Interaction Record, l'agente è capace di creare una nuova Activity come ad esempio Follow-Ups per la corrente interazione. Per esempio, un agente potrebbe creare un'attività di follow-up per una chiamata che verrà eseguita nell'arco delle 48 ore. Un'attività può essere assegnata a un Sales Rep che ne diventa il responsabile dell'esecuzione. Per migliorare la comunicazione con il cliente, è possibile inserire nella sezione - Notes informazioni provenienti dallo scratch pad.
- Secondo scenario: l'attività di Follow-Up Activity non è richiesta

Le necessità del cliente sono state soddisfatte dalla chiamata, quindi non è necessario pianificare un'ulteriore attività.
- Primo e secondo scenario
 - (a) Call wrap-up

L'IC agent si appresta a concludere la conversazione con il cliente, modificando i dati e salvando le informazioni più importanti. Prima di concludere la chiamata, l'agente chiede al customer ulteriori informazioni riguardanti possibili giudizi rispetto al servizio o alla soddisfazione dell'operato dell'agente. La chiamata può

4.3 Attività svolte

essere appena sia dal cliente sia dall'operatore. L'agente può successivamente lavorare a richieste ulteriori del cliente e prepararsi per nuove chiamate.

(b) Agent aggiorna l'Interaction Record

Prima di procedere alla chiusura dell'interazione e essere pronto per una nuova interazione con un altro customer, l'agente ha la possibilità di modificare l'interaction record aggiungendo note o nuovi dettagli o apportare cambiamenti allo status dell'interazione.

Outbound Telesales Lo schema in figura 4.5 descrive, invece, il flusso dello scenario telesales outbound che verrà poi eseguito sempre attraverso l'utilizzo della SAP Interaction Center WebClient. In questo scenario vengono presentate le principali attività di un Call Center, impegnato nella proposizione e nella vendita di prodotti o servizi allo scopo di avvicinare la domanda dei clienti agli scopi dell'azienda. In questa parte verranno descritte le Planned Call List, che rappresentano una lista di clienti che devono essere chiamati. Questa lista è formata e creata in maniera regolare o dalla figura dell'IC Manager o derivata da una campagna marketing che offre particolari prodotti o servizi per un determinato target di customer.

Nello scenario presentato, vengono coinvolti due principali ruoli :

- IC Manager : il cui ruolo prevede la pianificazione e la creazione delle Planned Call. A seconda delle necessità, il cliente potrebbe avere un giorno prestabilito per la consegna delle merci e, per garantire un buon approvvigionamento delle merci, plan call telesales elencano le specifiche date e orari per le telefonate da fare al cliente. Una scelta alternativa poteva essere fatta, escludendo la figura del IC Manager dallo scenario outbound. Infatti, quando in un organizzazione sono presenti pochi operatori di call center, con una buona preparazione o magari con pochi clienti, è possibile

4.3 Attività svolte

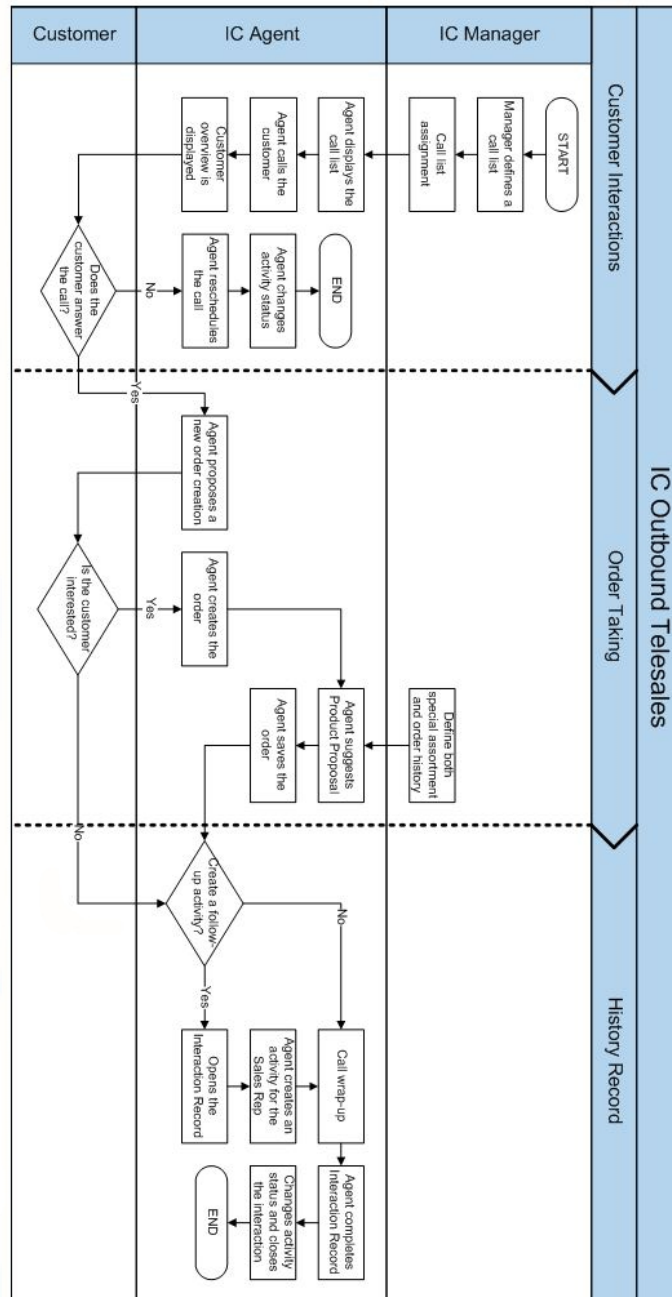


Figura 4.5: Outbound flow diagram

4.3 Attività svolte

sostituire il ruolo manageriale all'interno del flusso, incaricando gli stessi operatori a eseguire le varie operazioni di assegnamento e pianifica delle chiamate. Inoltre in uno scenario senza IC Manager e con pochi customer, il sistema potrebbe essere configurato per poter assegnare in automatico le chiamate dei clienti sempre allo stesso agente, in maniera tale da offrire un miglior servizio e aumentare il grado di soddisfazione del cliente stesso. Tuttavia nello scenario presentato è stato preferito introdurre comunque la figura del manager, ritenendo che questo tipo di soluzione sia più generale e utilizzata. Infatti, nei call center sono solitamente presenti numerosi operatori, spesso non specializzati e con alto tasso di turnover, che devono gestire un gran numero di clienti. In questi casi il ruolo manageriale diventa molto importante e essenziale per la buona gestione di questo settore aziendale.

- IC Agent : è la figura che ha il ruolo di eseguire le chiamate e raggiungere l'obiettivo di creare un nuovo ordine.

Una volta che una call list è stata creata, l'agente a cui è stata assegnata procede all'esecuzione delle varie chiamate presenti nella lista. In questo scenario l'agente deve anche sapere di gestire lo status delle chiamate effettuate, poiché se il cliente non risponde sarà necessario rischedulare la chiamata. Quando il customer risponde alla chiamata, l'agente propone un nuovo ordine, avendo prima controllato gli ordini storici del cliente stesso. L'agente infine valuterà se sia necessario schedulare un visita al cliente da parte del sales representative. Differenti status/results possono essere assegnati prima che la chiamata sia conclusa, in relazione all'approvazione degli ordini da parte del cliente o in base alle attività di visita schedulate.

Lo schema di flusso rappresentato in figura 4.5 può essere suddiviso in cinque sotto-processi identificati nel seguente modo :

1. IC Manager definisce e assegna una call list

4.3 Attività svolte

Una Call List è una lista automatica preparata dall'IC Manager, la quale permette una gestione delle chiamate dei clienti, programmando le risorse in modo più efficiente. Il processo Outbound Telesales comincia appunto con la preparazione delle call list e il loro assegnamento e, in seguito, la loro esecuzione

In generale, la procedura di Outbound consiste in due step principali:

- (a) Generazione: Le Call List possono essere formate attraverso l'utilizzo di script che schedulano le chiamate in maniera settimanale o giornaliera in base alle richieste dei clienti. Un'altra modalità per la creazione delle liste consiste nella creazione manuale da parte dell'IC Manager, secondo specifici gruppi di clienti. Inoltre, l'IC Manager può definire specifiche regole per la generazione delle Call List e modificare il tempo o le date di alcune chiamate. L'IC Manager risulta anche responsabile dell'assegnamento delle liste ai vari operatori del Call Center.
- (b) Assegnamento: è la fase che prevede la consegna delle call list ai vari operatori. Questa fase può essere svolta manualmente o sulla base di una specifica tabella di regole. La lista può inoltre, essere assegnata ad un singolo agente o all'intero reparto di un'organizzazione o, ancora, essere divisa in più liste ed essere assegnata a diversi Call Center.

2. L'agente del Call Center prepara la chiamata

Quando una call è stata assegnata, l'agente preposto avrà le relative chiamate all'interno della sua work list. Le work list permettono all'utente di monitorare gli assegnamenti dei compiti e l'organizzare il lavoro nel modo migliore. Inoltre, l'agente può direttamente gestire le sue attività scegliendo l'ordine delle chiamate e rischedulare quelle mancate. IC agent prosegue la sua attività scegliendo una chiamata dalla sua lista e controllando le relative informazioni del cliente da chiamare. Le informazioni

4.3 Attività svolte

disponibili sono collegate con ordini passati o ordini con uno stato ancora aperto. Una volta avviata la connessione con il cliente, esistono molte attività per soddisfare le necessità dei clienti. Tutte le attività vengono eseguite attraverso un tool che gestisce l'ordine di esecuzione e l'entrata in processi logistici e finanziari.

In questa fase sono possibili due sotto processi

- Customer risponde alla chiamata
- Customer non risponde alla chiamata

3. L'agente del Call Center rischedula la chiamata.

Questo processo occorre quando il customer non risponde alla chiamata e l'interazione viene chiusa. A questo punto IC agent cambia Activity status in No Answer. Dopo di che l'agente rischedulerà la chiamata verso questo customer durante la giornata, o in relazione alle richieste del cliente. L'operatore conclude la chiamata

4. L'agente del Call Center propone e crea un nuovo ordine.

Questo processo occorre quando il customer risponde alla chiamata. A questo punto IC agent propone un nuovo ordine al cliente. In base alla risposta del cliente si possono avere due possibili sotto processi :

- Il cliente non è interessato alla creazione di un nuovo ordine.

L'agente segnerà nelle apposite notazioni il "non interesse" da parte del cliente ad acquistare i prodotti o servizi proposti. Queste note verranno successivamente prese in esame nelle prossime chiamate verso questo cliente, in maniera tale da proporre prodotti alternativi o a costi diversi.

- Il cliente è interessato alla creazione di un nuovo ordine.

L'agente provvederà a creare un ordine e proporre alcuni prodotti. La ricerca dei prodotti è supportata in differenti modalità : una ricerca

4.3 Attività svolte

standard dei prodotti, attraverso un catalogo o in base alla ricerca di prodotti che si basano su campagne marketing. Durante la chiamata l'agente può proporre anche dei prodotti, i quali sono definiti attraverso liste disponibili attraverso la ricerca negli ordini passati o mediante elenchi di prodotti in offerta. Anche in questo caso, per migliorare le successive comunicazioni con il cliente, è possibile inserire delle "Notes" che riportano le informazioni più importanti

Dopo l'interazione con il customer, IC agent dovrà salvare l'ordine appena eseguito.

5. L'agente del Call Center conclude la chiamata

Questo processo può seguire due differenti percorsi, in base alla necessità di creare delle attività successive alla chiamata.

- Se una attività di follow-up è richiesta, IC agent può procedere nella creazione di una attività per la attuale interazione. Per esempio, si potrebbe creare una nuova chiamata nei giorni successivi o stabilire una visita del rappresentante di vendita in un giorno prefissato.
- Se una attività di follow-up non è richiesta, l'agente di prepara a concludere la chiamata con il cliente salvando le informazioni più importanti. Inoltre, l'agente ha la possibilità di chiedere al customer un giudizio riguardante il servizio svolto dall'operatore.

Una volta conclusa l'interazione, l'agente può successivamente lavorare a richieste ulteriori del cliente e prepararsi all'esecuzione di nuove chiamate della call list.

4.3.2 Scenario Campaign Management

In questo paragrafo verrà descritto ad alto livello uno scenario di Campaign Management.

4.3 Attività svolte

Campaign management è uno strumento CRM per gestire tutte le attività legate alla gestione delle campagne, tra cui il design, l'esecuzione, il coordinamento, l'ottimizzazione e il monitoraggio delle campagne marketing. I responsabili del marketing in azienda possono creare campagne personalizzate e mirate attraverso l'utilizzo di tutti i canali di comunicazione disponibili: direct sales, call centers, mail, e-mail, fax, e Internet.

Una volta che la campagna è stata definita, con questo strumento è possibile monitorare e tenere traccia di tutti gli avvenimenti nel corso della durata della campagna stessa, analizzando i risultati ottenuti per le future pianificazioni.

Secondo le best practice[35] di SAP CRM il Campaign Management può essere suddiviso in sei principali processi :

1. Marketing Planning and Budgeting : unisce e coordina tutte le iniziative e le risorse nell'organizzazione per la pianificazione e lo sviluppo di processi marketing. Nella gestione delle attività di marketing è previsto il controllo del budget e la distribuzione dei fondi in base alla segmentazione tra le varie Business Segmentation.
2. Campaign Planning : questo processo business permette di pianificare e definire i criteri usati per la segmentazione dei clienti, l'assegnamento dei prodotti, la responsabilità dei dipendenti e l'allocazione del budget. Inoltre, la Campaign Planning comprende molte altre attività, tra le quali integrated budgeting, pianificazione del prodotto, pianificazione dei costi e della distribuzione, e la pianificazione della campaign response.
3. Campaign Development : in questa fase viene sviluppata la campagna marketing in base alle scelte e alle richieste degli step precedenti.
4. Segmentation : la segmentazione è un processo che è generalmente eseguito dal manager del Master Data. In questa fase si procede all'analisi delle informazioni presenti sul database e alla suddivisione dei clienti e dei prodotti in base alla strategia adottata. In questo modo si creano li-

4.3 Attività svolte

ste di target group di customer a cui vengono associati una o più liste di prodotti.

5. Campaign Execution : l'esecuzione della campagna è subordinata al giudizio da parte del manager responsabile, il quale può decidere di approvare la campagna o modificarla in base alle necessità del business. Inoltre, in questa fase si procede alla selezione di uno o più canali per l'esecuzione
6. Campaign Analysis : In questa fase la campagna marketing viene analizzata e misurata. Inoltre, l'analisi di monitoraggio, disponibile durante e dopo l'esecuzione, permette di raffinare le campagne per ottimizzarne l'efficacia e l'efficienza.

A causa della complessità generale, lo scenario presentato sarà una versione semplificata di una Campaign Management. Infatti, per le due fasi di "Marketing Planning and Budgeting" e "Campaign Planning" vengono fatte delle assunzioni che, in pratica, ne limitano l'esecuzione: una prima scelta è stata quella di utilizzare il Call Center come canale di comunicazione per l'esecuzione delle campagne. Tale scelta deriva dal fatto che lo sviluppo e l'implementazione dello scenario telesales outbound era già stato svolto e con pochissime modifiche era possibile riadattarlo a questo scopo. Tuttavia, altri canali potrebbero essere utilizzati come alternativa o come supporto al call center. Altre scelte sono legate al fatto di non avere un budget predefinito e di non prevedere nessuna assegnazione di fondi alle business unit.

Durante l'esecuzione verranno formate delle liste di clienti che saranno utilizzati dal call center per formare delle call list di customer da chiamare. Infine, allo scopo di analizzare i risultati della campagna, in questo scenario i feedback dei clienti verranno collezionati attraverso l'utilizzo di sondaggi o survey e saranno forniti al cliente attraverso l'IC agent.

Come mostrato in figura 4.6, all'interno della fase di Campaign Development, si possono riscontrare tre sottoprocessi :

4.3 Attività svolte

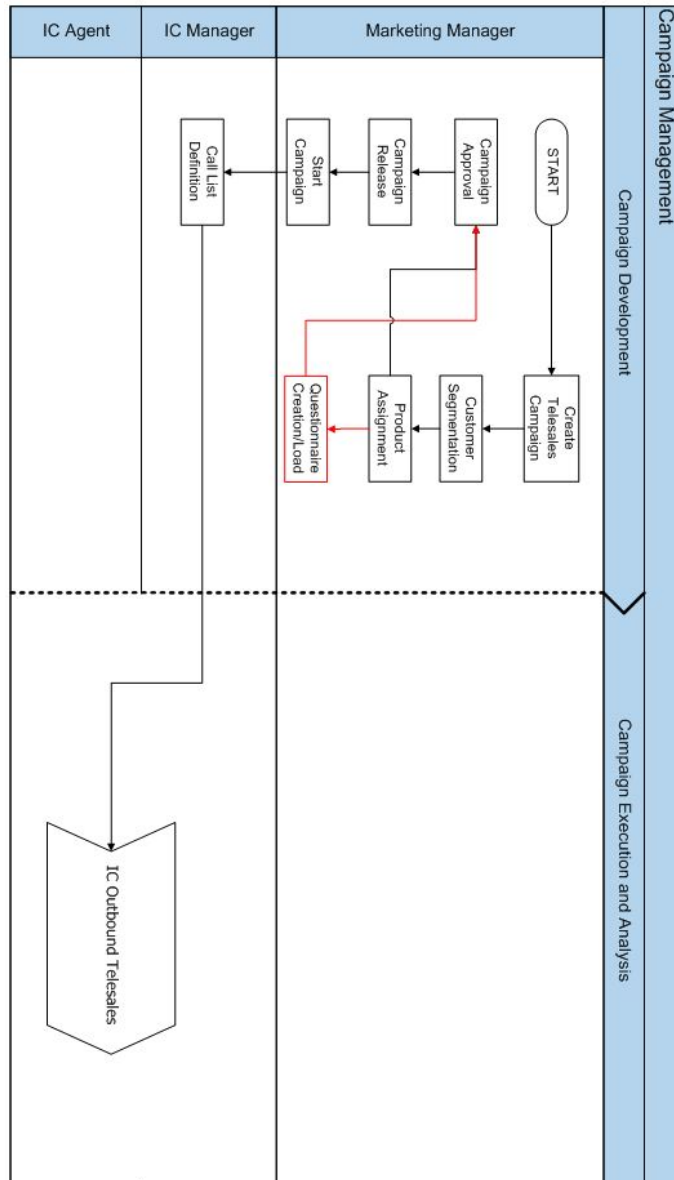


Figura 4.6: Campaign Management flow diagram

4.3 Attività svolte

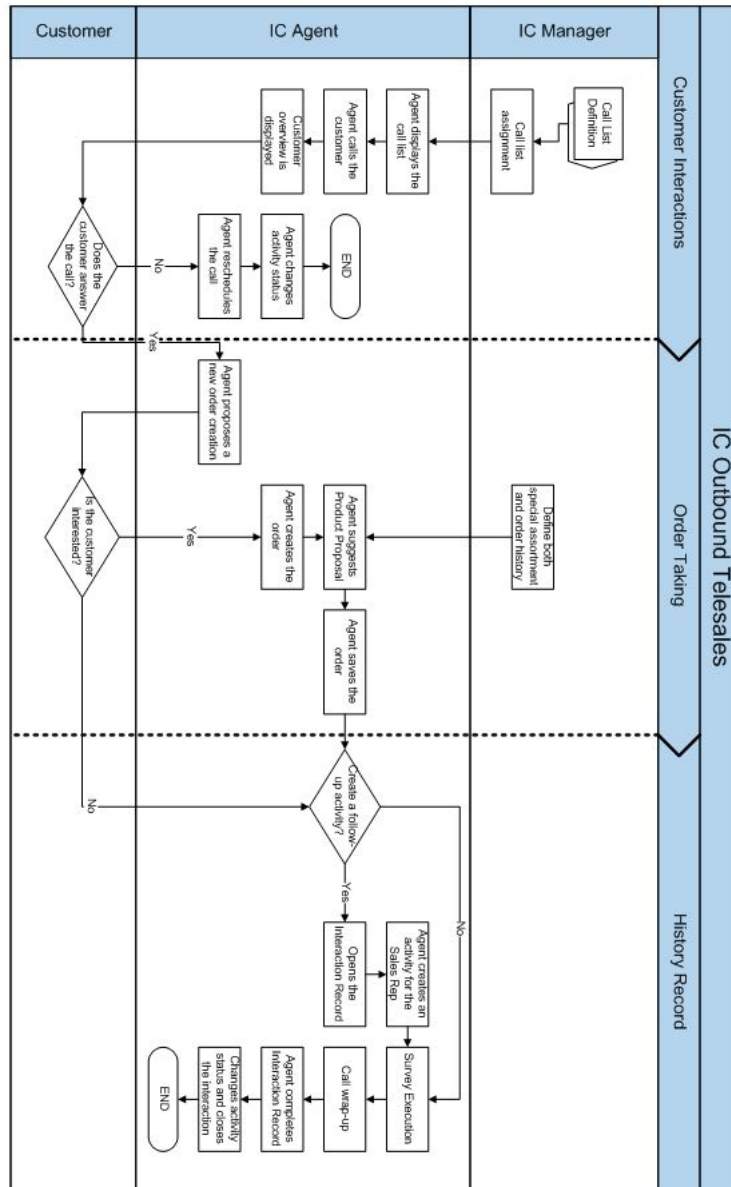


Figura 4.7: IC Outbound Telesales flow diagram

4.3 Attività svolte

1. Il Marketing Manager crea una Telesales Campaign.
2. Il Marketing Manager approva e rilascia la campagna.
3. L'IC Manager assegna e attiva la call list agli IC Agent

All'interno della fase di Campaign Execution and Analysis, si possono riscontrare due sottoprocessi :

1. L'IC Agent esegue l'Outbound Telesales.
 2. L'IC Agent sottopone il sondaggio al customer.
-
1. Il Marketing Manager crea una Telesales Campaign.

Questo sotto processo descrive la creazione della campagna telesales. Generalmente il Marketing Manager inizia questo processo collezionando e assemblando le informazioni riguardanti i customer e i prospecti, insieme ai documenti interni ed esterni. In base al tipo di campagna, il Marketing Manager definisce tutti i dettagli che riguardano la creazione della campagna : ID, descrizione, tipo di prodotti, canale, segmentazione dei customer, tattiche da utilizzare, priorità e ulteriori dettagli.

- Segmentazione dei customer: in questa fase vengono segmentati i customer in una lista di Business Partner che sono usati per creare i Target Groups. Tali gruppi rappresentano un sottoinsieme di tutti i business partner contenuti nel SAP CRM. La suddivisione avviene per mezzo di un tool apposito che utilizza degli attributi assegnati ai BP per formare i gruppi. Tali attributi sono chiamati Marketing Attributes e spesso sono derivati da informazioni forniti dal Master Data : country, Sales organization, etc.
- Assign Products: in questa fase vengono selezionati quali tipi di prodotti verranno usati nella campagna e collegati alla segmentazione dei customer.

4.3 Attività svolte

- Channel Strategy: in questa fase viene selezionato il canale attraverso il quale si effettuerà la campagna. Nel nostro caso la campagna verrà effettuata attraverso il Call Center. Tuttavia per una stessa campagna potrebbero essere selezionati più canali di vendita : Outbound Call, E-mail, Web, ecc.

2. Marketing Manager approva e rilascia la campagna.

Una volta che la strategia della campagna marketing è stata definita, si procede alla sua approvazione. Ogni fase dello sviluppo è controllata al meglio per soddisfare le necessità dei clienti, per migliorare le relazioni con i customer e creare un maggior valore di ritorno. Se non è necessario introdurre correzioni o cambiamenti, la campagna marketing è rilasciata e in seguito può iniziare. Una volta rilasciata, il Marketing Manager può far partire immediatamente la campagna o aspettare la data stabilita

3. L'IC Manager assegna e attiva la call list agli IC Agent

Le call list sono preparate dal l'IC Managers attraverso l'utilizzo di tool specifici e sono usate per gestire le chiamate verso i clienti. Le chiamate vengono gestite in base alle esigenze dei customer e schedulate a seconda della strategia adottata e infine vengono assegnate ai vari operatori del Call Center e attivate nel work flow dei vari agent.

In quest'ultima fase della gestione della campagna si assiste allo svolgimento dei processi coinvolti nello scenario Outbound, già precedentemente discusso. Tuttavia in questa fase della campaign tale scenario è ulteriormente sviluppato. Infatti le sue funzioni si articolano in due principali step:

1. The IC Agent processa l'Outbound Telesales. In questa fase l'agente processa le chiamate presenti nella call list, proponendo nuovi ordine in base alle transazioni storiche dei clienti.

4.3 Attività svolte

2. L'IC Agent sottopone un sondaggio al customer allo scopo di analizzare i risultati della campagna. I feedback vengono raccolti all'interno del sistema in maniera tale da offrire una visione concreta della riuscita della campagna.

4.3.3 Configurazione e testing degli scenari

Una volta completata la parte di sviluppo dei processi si è proceduto alla configurazione e customizzazione del sistema SAP CRM in base ai processi e alle scelte effettuate nei documenti precedenti. La customizzazione è un aspetto molto importante che prevede l'inserimento e la modifica di particolari record all'interno delle tabelle di configurazione. Tali tabelle consentono di poter gestire tutti gli aspetti fondamentali del sistema SAP CRM, permettendo la configurazione di qualsiasi aspetto funzionale o di processo. Questa fase è possibile eseguirla attraverso l'interfaccia grafica di SAP chiamata SAP GUI. Dopo aver effettuato il login e essersi autenticati nel sistema è possibile richiamare particolari funzioni in modo tale da permettere di attuare le varie modifiche. Una funzione, con la quale è possibile configurare buona parte degli aspetti relativi ai processi, è denominata "SPRO". Da questa funzionalità è possibile procedere alla generazione e alla gestione delle nuove business transaction, con la quale configurare il sistema in base alle richieste fornite dai flussi degli scenari presentati precedentemente.

Per il completamento del progetto svolto, la fase successiva è composta dall'attività di testing. Prima di procedere all'esecuzione delle varie operazioni di test e controllare che il sistema sia compatibile con la descrizione del flusso di processo, è necessario configurare e controllare le informazioni del Master Data.

Infatti, il Master Data è un insieme di informazioni che sono considerate fondamentali per le operazioni business. Le informazioni contenute nel Master Data fanno riferimento a tutti i dati e le relazioni tra i vari clienti, dipendenti, prodotti, fornitori, e molti altri soggetti. Questa base dati è solitamente condivisa

4.3 Attività svolte

da molti utenti o organizzazioni, spesso memorizzata su differenti sistemi.

Per gli scenari presentati, la struttura del Master Data è stata formata con le seguenti operazioni:

- la definizione degli utenti e dei ruoli. Nel sistema sono stati inseriti degli utenti per i ruoli di IC Agent, per quanto riguarda gli scenari telesales; IC Line e IC Manager, per la parte manageriale sia per gli scenari telesales sia per la campaign.
- la definizione di alcuni customer. Per tutte le tipologie di scenari è necessario la presenza nel sistema di figure di customer. I clienti necessitano alcune informazioni particolari per l'attuazioni dei processi, tra cui ad esempio il numero telefonico, l'email e l'assegnamento di un sales representative. Inoltre, è stato necessario specificare alcune business relation tra i vari customer per il corretto funzionamento delle transazioni.
- la definizione di una serie di prodotti per la vendita. Per riuscire a completare le varie operazioni nella creazione degli ordini, è stato necessario inserire nel Master Data una serie di prodotti commerciali e in seguito associarli alla liste per la vendita dei vari operatori del call center. I prodotti inseriti sono rappresentati sia da un nome o descrizione, sia da un costo per unità.

Il completamento delle informazioni nel Master Data e l'inserimento corretto delle relazioni tra i vari customer, sales representative e utenti del sistema SAP CRM, è un processo molto importante e delicato da affrontare. Infatti, seppur il sistema risulta essere corretto e funzionante, una mancata relazione o dati sbagliati comprometterebbe la buona riuscita delle operazione durante l'esecuzione.

Conclusioni

L'obiettivo di questo elaborato di tesi è rappresentato dallo sviluppo e dalla realizzazione di processi di vendita in ambito telesales e campaign management in sistemi SAP CRM. Lo strumento utilizzato, per coordinare e gestire tutte le attività del progetto, è stato SAP Solution Manager. Un altro aspetto introdotto, partendo dal lavoro condotto su questo programma, è stato il concetto Application Lifecycle Management e delle sue funzionalità.

L'esperienza di stage e il lavoro svolto, ambedue molto stimolanti, ma impegnativi al stesso tempo, hanno richiesto di affrontare e approfondire numerose problematiche software relative alla progettazione dei processi all'interno dei sistemi SAP CRM. Quella che mi è stata offerta rappresenta una grande opportunità di conoscere e capire meglio alcuni aspetti relativi alla gestione dei sistemi CRM e lo sviluppo di processi di integrazione nelle vendite.

Lavoro svolto Il lavoro svolto comprendeva una serie di attività per la progettazione e la realizzazione di due processi tipici del panorama CRM : telesales e campaign management. Questo lavoro rientra in un progetto ben più ampio e complesso, che prevede lo sviluppo di un sistema SAP CRM, che costituirà in futuro una forma di demo tester per i clienti, con aggiornamenti e funzionalità particolari.

L'attività iniziale ha previsto un'analisi dei processi di vendita e la stesura

Conclusioni

di flussi di processi. Una volta convalidati i documenti, il lavoro è proseguito alla fase successiva che comprendeva la realizzazione pratica dei processi all'interno del sistema SAP CRM. Questa fase del progetto prevedeva la configurazione e il coordinamento di tutti i processi che si sono descritti in precedenza, attraverso una customizzazione delle tabelle di configurazione. Quando il sistema è risultato funzionante, si è proceduto alla successiva fase di testing con l'utilizzo dei dati precedentemente inseriti nel Master Data.

Pur essendo una piccola parte del sistema, questo lavoro permetterà comunque di inserire nuove funzionalità e essere testato per scopi didattici o per essere presentato a nuovi clienti.

Prossimi sviluppi riguarderanno certamente altre funzionalità del sistema SAP CRM, come ad esempio la vendita con modalità B2C e B2C attraverso il CRM Web Channel oppure lo sviluppo di uno scenario di "service" la gestione dei reclami. Alcuni lavori sono in fase finale e altre attività saranno previste a breve per portare il sistema attuale a essere il più completo e aggiornato possibile.

Osservazioni conclusive Una riflessione da compiere riguarda l'adozioni di tali sistemi da parte dell'organizzazioni. Questa tipologia di soluzioni CRM sono all'avanguardia rispetto alla gestione delle attività e, in generale, aiutano fortemente a diminuire i tempi relativi ai processi collegati alle interazioni con i clienti. Inoltre offrono in maniera sistematica una serie di servizi che permettono di aumentare il feeling con il customer e aiutano a garantire una alta fidelizzazione con lo stesso.

Tuttavia, solo le grandi organizzazioni possono davvero trarre profitto dall'implementazioni di tali soluzioni CRM, in quanto i costi di investimento e training sono molto elevati e certamente non sono alla portata di tutte le realtà aziendali.

Considerazioni su ALM In questa parte conclusiva verranno introdotte alcune considerazioni finali sull'utilizzo delle soluzioni di Application Lifecycle

Conclusioni

Management. Partendo dall'uso del *SAP Solution Manager*, software per la gestione e il coordinamento dei progetti in ambiente SAP, ho potuto conoscere le potenzialità delle soluzioni ALM, dove la tracciabilità e la visione generale dei processi aiuta fortemente la qualità e la buona riuscita dei progetti. Nelle aziende dove vengono affrontate le tutte le fasi di sviluppo del ciclo di vita del software è possibile apportare forti miglioramenti introducendo nella propria gestione una soluzione di questo tipo. Questo approccio prevede un notevole miglioramento dei processi, una maggiore qualità e un più veloce rilascio dei prodotti software. Secondo statistiche dei vari vendor, si può ottenere velocemente un ritorno dell'investimento nell'adottare queste soluzioni, anche se il tempo per l'implementazione e il tasso di adozione è generalmente più lungo di quello che si può immaginare.

Infatti, una volta che questa tipologia di software è stata acquistata, è necessario comunque procedere a fasi di ottimizzazione nei processi interni, a seconda per grado di utilizzo della soluzione adottata.

Tuttavia, pur essendo il costo di questi prodotti non proprio conveniente, e non sempre di facile utilizzo, l'implementazione può portare a una serie di miglioramenti che determinano dei ritorni economici quasi immediati :

- i requisiti e le specifiche del progetto avranno un grado di precisione e completezza maggiore. I clienti saranno sicuramente più soddisfatti del lavoro eseguito e di conseguenza una maggiore fidelizzazione determinerà maggiori guadagni a lungo termine.
- anche la qualità del prodotto finito ne risulta migliorata. Il software rilasciato avrà una presenza di errori o bug inferiore, con una riduzione del tempo speso per la correzione.
- La soddisfazione del cliente risentirà della qualità del prodotto e sarà un buon testimonial verso il mercato per i propri prodotti. I prodotti avran-

Conclusioni

no un interesse maggiore da parte di altri clienti e si potrà acquistare sicuramente un maggiore marketshare.

- i team di sviluppo avranno maggiore soddisfazione personale nello sviluppare applicazioni funzionali e corrette. Le migliori vendite dei prodotti, contribuiranno anche a un maggior salario per i dipendenti e responsabili del progetto, i quali saranno sempre più invogliati ad accrescere le proprie conoscenze e capacità per i successivi prodotti.

Tuttavia, la scelta di una soluzione ALM può risultare piuttosto difficoltosa e un investimento sbagliato in questa tipologia di prodotto può portare a conseguenze gravi, pregiudicando il lavoro svolto o diminuendone la produttività.

L'utilizzo e lo studio delle soluzioni ALM mi ha certamente aiutato a comprenderne meglio le varie sfaccettature nella progettazione e sviluppo del software e quali problematiche è possibile risolvere adottando tali prodotti in team di lavoro eterogenei e distanti fra loro.

Glossario e acronimi

Information and Communication Technology (ICT): é l'insieme di studio, progettazione, sviluppo, implementazione, supporto e gestione dei sistemi informativi computerizzati con particolare attenzione alle applicazioni software ed ai componenti hardware che le ospitano. Il fine ultimo dell'ICT é la manipolazione dei dati tramite conversione, immagazzinamento, protezione, trasmissione e recupero sicuro delle informazioni. L'ICT é, inoltre, considerata una risorsa essenziale delle organizzazioni, all'interno delle quali diventa sempre più importante riuscire a gestire in maniera rapida, efficace ed efficiente il volume crescente di informazioni.

Information Technology (IT): vedi Information and Communication Technology (ICT).

Information Technology Infrastructure Library (ITIL): é un insieme di linee guida ispirate dalla pratica (Best Practices) nella gestione dei servizi IT (IT Service Management) e consiste in una serie di pubblicazioni che forniscono indicazioni sull'erogazione di servizi IT di qualità e sui processi e mezzi necessari a supportarli. Senza pratiche standard, le attività sarebbero generate indipendentemente dalle pratiche di gestione IT e questo duplicherebbe gli sforzi all'interno dei progetti ICT con conseguenti errori ed incremento dei costi.

Product Lifecycle Management (PLM): il sistema per la gestione del ciclo di vita del prodotto; il PLM é un approccio strategico alla gestione delle informazioni, dei processi e delle risorse a supporto del ciclo di vita di prodotti e servizi, dalla loro ideazione, allo sviluppo, al lancio sul mercato, al ritiro. Il PLM non é solo una tecnologia informatica, ma piuttosto un approccio integrato, basato su un insieme di tecnologie, su metodologie di organizzazione del lavoro collaborativo e sulla definizione di processi.

SAP: SAP AG é una multinazionale europea per la produzione di software. E' uno dei principali leader mondiali nel settore degli ERP e in generale nelle soluzioni Enterprise. L'acronimo SAP significa "Systeme, Anwendungen, Produkte in der Datenverarbeitung". In inglese "Systems, Applications and Products in data processing".

Advanced Business Application Programming (ABAP): é un linguaggio di programmazione proprietario creato dalla società SAP AG. E' usato per gestire applicazioni commerciali attraverso una programmazione strutturata.

Sistema Operativo (SO)

Service-Oriented Architecture (SOA): architettura software atta a supportare l'uso di servizi Web per garantire l'interoperabilità tra diversi sistemi così da consentire l'utilizzo delle singole applicazioni come componenti del processo di business e soddisfare le richieste degli utenti in modo integrato e trasparente.

Simple Object Access Protocol (SOAP): é un protocollo leggero per lo scambio di messaggi tra componenti software, tipicamente nella forma di componentistica software.

Glossario e acronimi

Total Cost of Ownership (TCO): costo totale di proprietà o costo totale di possesso, è un approccio utilizzato per calcolare tutti i costi del ciclo di vita di un'apparecchiatura informatica IT (l'acquisto, l'installazione, la gestione, la manutenzione e lo smantellamento). L'approccio TCO è basato sulla considerazione che il costo totale di utilizzo di una apparecchiatura IT non dipende solo dai costi di acquisto, ma anche da tutti i costi che intervengono durante l'intera vita di esercizio dello strumento.

Time-to-market: il tempo che intercorre dall'ideazione di un prodotto alla sua effettiva commercializzazione.

Vincolo di integrità : un vincolo di integrità è una proprietà che deve essere soddisfatta dalle istanze di una base di dati. Ogni vincolo può essere visto come un predicato (funzione booleana) che può assumere il valore vero o falso: se il predicato assume il valore vero diciamo che l'istanza soddisfa il vincolo.

Telesales : si intende la vendita o tentata vendita di un prodotto particolare o un servizio da parte di un venditore (Call Center), che effettua un approccio iniziale attraverso l'utilizzo di dispositivi telefonici.

SAP Interaction Center WebClient : è un thin-client di SAP ottimizzato per la funzionalità del Call Center.

Dynamic Systems Development Method (DSDM): è una metodologia agile distribuita gratuitamente dal consorzio DSDM ai propri membri, è fornito in pacchetto insieme ad un framework. Questo consorzio è nato per definire e distribuire uno standard industriale per il framework RAD (Rapid Application Development).

Agile Unified Process (AUP): descrive un approccio allo sviluppo di applicazioni software, semplice, facile da comprendere e che utilizza tecniche e concetti agili.

Kanban : è un termine giapponese che letteralmente significa cartellino. Il sistema kanban decide la quantità e tipologia da produrre in tutti i processi. Il primo beneficio del sistema kanban è che riduce la sovrapproduzione, producendo soltanto la cosa richiesta, quando è chiesto e nella quantità richiesta.

integrated development environment (IDE) : è un software che aiuta i programmatori nello sviluppo del codice.

Software Change and Configuration Management (SCCM) : è l'attività di tracciatura e controllo delle change nella produzione del software.

Return on Investment (ROI): indice di redditività del capitale investito o ritorno degli investimenti.

Business Configuration Set (BC Set) : rappresenta un tool di gestione che permette agli utenti di registrare, salvare e condividere con la configurazione del sistema.

Computer Aided Test Tool (CATT) : utilizzato per la creazione di test case automatici.

extended Computer Aided Test Tool (eCATT) : utilizzato per la creazione e l'esecuzione dei test cases automatici.

Master Data : rappresenta una sorgente di dati che può essere usata più sistemi, applicazioni o processi. I dati rappresentano informazioni riguardanti i Business Partner, prodotti, contratti, o in generale qualsiasi altra informazione che identifica una entità di business.

Customer Relationship Management

A.1 Customer Relationship Management

Il concetto di Customer relationship management (CRM) o Gestione delle Relazioni coi Clienti é legato al concetto di fidelizzazione dei clienti. Avere una strategia in cui l'attenzione verso il cliente é fulcro di tutti i processi, permette all'azienda di capire meglio il ruolo nella catena del processo di gestione dei clienti acquisiti e potenziali. In un'impresa "Market-oriented" il mercato non é piú rappresentato solo dal cliente ma dall'ambiente circostante, con il quale l'impresa deve stabilire relazioni durevoli di breve e lungo periodo, tenendo conto dei valori dell'individuo/cliente, della società e dell'ambiente. Quindi l'attenzione verso il cliente é cruciale e determinante. Per questo motivo il marketing management deve pianificare e implementare apposite strategie per gestire una risorsa così importante. Il CRM si spinge sostanzialmente secondo quattro direzioni differenti e separate:

1. L'acquisizione di nuovi clienti o "clienti potenziali"
2. L'aumento delle relazioni con i clienti piú importanti
3. La fidelizzazione piú longeva possibile dei clienti che hanno maggiori rapporti con l'impresa

A.1 Customer Relationship Management

4. La trasformazione degli attuali clienti in procuratori, ossia consumatori che lodano l'azienda incoraggiando altre persone a rivolgersi alla stessa per i loro acquisti

Esistono tre tipi di CRM:

1. CRM operativo: soluzioni metodologiche e tecnologiche per automatizzare i processi di business che prevedono il contatto diretto con il cliente.
2. CRM analitico: procedure e strumenti per migliorare la conoscenza del cliente attraverso l'estrazione di dati dal CRM operativo, la loro analisi e lo studio revisionale sui comportamenti dei clienti stessi.
3. CRM collaborativo: metodologie e tecnologie integrate con gli strumenti di comunicazione (telefono, fax, e-mail, ecc.) per gestire il contatto con il cliente.

L'errore più comune, quando si parla di Customer Relationship Management, è quello di equiparare tale concetto a quello di un software. Il CRM non è una semplice questione di prodotto commerciale né di sistemi informatici, bensì si avvale in maniera sempre più massiccia, di strumenti informatici o comunque automatizzati, per implementare il management. Il CRM è un concetto strettamente legato alla strategia, alla comunicazione, all'integrazione tra i processi aziendali, alle persone ed alla cultura, che pone il cliente al centro dell'attenzione sia nel caso del business-to-business sia in quello del business-to-consumer. Le applicazioni CRM servono a tenersi in contatto con la clientela, a inserire le loro informazioni nel database e a fornire loro modalità per interagire in modo che tali interazioni possano essere registrate e analizzate. Prima di seguire la strada del CRM ogni azienda deve essere consapevole che:

- bisogna investire prima in strategia, organizzazione e comunicazione, solo dopo nella tecnologia. La scelta del software non ha alcun effetto sulla

A.1 Customer Relationship Management

probabilità di successo. Ciò non implica che i software siano tutti uguali, ma significa solo che nessun software porterà al successo un progetto sbagliato.

- il CRM é adatto sia a quelle aziende che cercano un Return on investment (ROI) veloce sia a quelle che curano il processo di fidelizzazione e l'aumento del Lifetime value (LTV) dei clienti che richiede del tempo.

Bibliografia

- [1] SAP Solution Manager - Sukrittaya Patchotepong and Alexei Koifman
- [2] White Paper - Obtaining ROI from an ALM Tool - Steve Miller
- [3] What is application lifecycle management? - David Chappell
- [4] Open Application Lifecycle Management (ALM) - Borland
- [5] Chapter IV - Software Engineering - Processes and Tools - Gerhard Weiss, Gustav Pomberger Wolfgang Beer, Georg Buchgeher, Bernhard Dorninger, Josef Pichler, Herbert Prahofer, Rudolf Ramler, Fritz Stallinger, Rainer Weinreich
- [6] Application Lifecycle Management and Business Strategy - David Chappell
- [7] Application Lifecycle Management as a Business Process - David Chappell
- [8] The Changing Face Of Application Life-Cycle Management - Carey Schwaber
- [9] Application Lifecycle Management - In Pursuit of Best-of-Breed - TATA Consultancy Services
- [10] Application lifecycle management in embedded systems engineering - Doyle, C. and Lloyd, R.

BIBLIOGRAFIA

- [11] IEEE Standard Glossary of Software Engineering Terminology - IEEE-STD-610 ANSI/IEEE
- SITOGRAFIA
- [12] <http://www.sdn.sap.com/>
- [13] <http://www.01net.it/>
- [14] <http://searchsoftwarequality.techtarget.com/>
- [15] <http://www.articlesbase.com/>
- [16] <http://sixrevisions.com/>
- [17] <http://blog.infostretch.com/>
- [18] <http://www.internet.com/>
- [19] <http://whatis.techtarget.com/>
- [20] <http://technology.ezinemark.com/>
- [21] <http://it.wikipedia.org>
- [22] <http://www.sap.com/>
- [23] <http://itknowledgeexchange.techtarget.com>
- [24] <http://www.solutionmanagerexpert.com/>
- [25] <http://community.forrester.com/>
- [26] <http://www.projectperfect.com.au/>
- [27] <http://www.slideshare.net/wlcaze/sap-solman-imptools-implementation-cycle-presentation>
- [28] <http://www.accurev.com/alm-solutions.html>

BIBLIOGRAFIA

- [29] <http://www.cmcrossroads.com/cm-journal-articles/7530-application-lifecycle-management-a-look-back>
- [30] <http://it.wikipedia.org/wiki/Kanban>
- [31] http://it.wikipedia.org/wiki/Agile_Unified_Process
- [32] http://it.wikipedia.org/wiki/Dynamic_Systems_Development_Method
- [33] http://en.wikipedia.org/wiki/Software_configuration_management
- [34] http://en.wikipedia.org/wiki/Capability_Maturity_Model_Integration
- [35] <http://help.sap.com/>
- [36] <http://wiki.sdn.sap.com/wiki/display/SM/ASAP+Methodology>