

Con il fine di strutturare meglio le informazioni contenute all'interno dell'intervista, questa è stata divisa in capitoli. Ad ogni riga della tabella corrisponde un capitolo.

Nella colonna sinistra è riportata l'intervista, in quella di destra sono riportati il titolo del capitolo, che ne riassume il contenuto, e gli insights ricavati dal capitolo.

I colori mettono in relazione l'insight ricavato con il pezzo di intervista a cui fa riferimento. Per ogni capitolo, i colori fanno riferimento solo al capitolo specifico e sono indipendenti dagli altri.

<p>R quale è stato il tuo ruolo nel progetto e come hai contribuito al successo del progetto?</p> <p>I Allora il mio ruolo nel progetto è stato PM project manager e Analista Funzionale, quello che in Coolshop chiamiamo Paladin Senior Nella dalla parte di Gara e Discovery, Fino a circa la Release, quattro del progetto. dalla Release Due in avanti, coadiuvato in parte da un'altra figura Paladin Junior Su alcune funzionalità, anche su FD che non erano propriamente Salesforce.</p>	<p>RUOLO</p>
<p>R Quindi ti chiedo subito come secondo te, se secondo te il passaggio di consegna è avvenuto con successo oppure no?</p> <p>I No. Perché sono una brutta persona, ossia non ho consegnato correttamente il progetto, Principalmente, secondo me, per mancanza di documentazione di cosa sia davvero importante consegnare quando si consegna un progetto.</p> <p>R secondo te cosa mancava?</p> <p>I Eh, mancava diciamo un po' del famoso non detto che, per quanto una persona spera che non ci sia, c'è. Secondo me l'altra parte principale è che tendenzialmente non si deve solo fare Knowledge Transfer del prodotto ma anche del cliente. Che diventa nettamente più difficile rispetto al fare semplicemente KT del prodotto. Inoltre, ci sono tutta una serie di cose che si fa tendenzialmente quando fai queste cose, fai. Un quadrante di quattro settori. know/unknown. Molte cose finiscono nel unknown/unknown Perché tendenzialmente sono piccolezze, cioè nel senso sono cose che magari ti sei fatto scappare in call con il cliente, accordi arrivati in call magari in momenti un pochino concitati, in cui però devi chiudere qualcosa, quindi finisci con l'avere Un accordo in call molto veloce di scambio. Tutta l'altra parte molto difficoltosa è quando devi discutere un pochino del processo di gara nel senso che molte cose vengono... Diciamo secondo me abbiamo dato molta poca attenzione a quello che era il</p>	<p>PASSAGGIO DI CONSEGNA</p> <p>Non sapere cosa comunicare</p> <p>Importante trasferire informazioni sul cliente</p> <p>Standardizzazione del ruolo del Paladin</p>

<p>documento iniziale di offerta che invece doveva essere quello che avrebbe guidato nettamente di più il KT almeno sul progetto, quindi noi ci siamo ritrovati un pochino a galoppare dicendo boh più o meno il cliente lo ho conosciuto perché comunque ci ho lavorato un po' in parallelo con te, più o meno il progetto l'ho capito, Andiamo avanti. KT fatto. E poi sono mancate un po di robe PMmose. Che però, secondo me sono su due livelli di difficoltà principali. Dettati un pochino dal fatto che A) magari l'altro Paladin era anche fra l'altro nuovo in azienda, quindi c'era anche un po di difficoltà propria nel nessuno che gli aveva spiegato tendenzialmente qual era ciò che faceva il paladin all'interno di un progetto, B) sono stato tacitato di essere un paladin speciale, quindi di fare le cose in maniera un po' diversa da come la fanno gli altri e quindi mancava tutta una parte, ad esempio c'è stato un breve disguido dettato qualche tempo dopo riguardante il budget, il fatto che ci fosse un cuscinetto di budget venduto che però dovevamo cercare di non utilizzare perché la stima nostra era stata più bassa, quindi si sarebbe dovuto stare dentro la stima e bla bla bla nonostante economicamente. Il progetto avesse un discreto amount di Man Days come cuscinetto venduti. In pratica i Man Days venduti al cliente erano in over rispetto a quanto stimato con tutti i cuscinetti nostri, Quindi, con i vari stima tecnica, poi jedi, poi funzionali, poi i test, poi Post Go Live. On top a questi c'erano altri giorni venduti Quindi On top a tutto questo diciamo c'è stata un po Di roba Addizionale da fare, da vedere.</p>	
<p>R diciamo che a un certo punto, verso il periodo di agosto, Il budget inizia proprio a scendere verso il basso in modo inesorabile.</p> <p>I Sì, perché il progetto doveva terminare da iniziali calcoli nostri. Secondo la nostra stima iniziale, il progetto doveva essere terminato. Il Progetto deve essere consegnato. almeno doveva essere terminate da noi internamente ad agosto Secondo il più tradizionale Approccio waterfall di planning.</p> <p>R e non era pronto, perché?</p> <p>I Per inesattezza delle stime, per inefficienza accumulata nell'arco del progetto Quindi inefficienza da parte nostra e ambo due i team [Coolshop e cliente] e per una non corretta allocazione dello staff, quindi lo staffing è stato cambiato rispetto allo staff che era stato preventivato inizialmente.</p> <p>R</p>	<p>BUDGET</p> <p>Budget bruciato perché Progetto andato lungo nei tempi, a causa di stime errate, inefficienze</p> <p>Cattiva pianificazione allocazione risorse</p> <p>Difficile coniugare bisogni di azienda e di progetto</p> <p>Cambi di requisiti non comprati bruciano budget</p> <p>Team gonfiato rispetto quanto preventivato brucia budget</p>

Cioè intendi proprio i membri del team?

I

Esatto Quindi abbiamo iniziato, dove eventualmente all'interno di questo progetto ci sarebbero stati Un massimo di un PM e un analista funzionale, Quindi due figure PM non erano neanche staffate eventualmente full time da un punto di vista funzionale siamo arrivati a un certo punto ad occupare tre slot di figure e quindi questo brucia nettamente più budget del preventivato. Poi ovviamente, per come è forecastato il nostro budget, È un'individuazione molto difficile da fare questa Perché per fare un per individuare che uno dei problemi è stato questo, tu dovresti aver fatto un budget per risorse dicendo Allochero, un senior Developer, tre developer, un designer, un PM, a mezzo FTE e un analista funzionale ad un FTE. Cosa che dal nostro file di budget non c'è. Noi abbiamo un aggregato di quanto c'è di venduto per quel progetto, Per quel mese sì.

R

Quindi abbiamo un'allocazione sulle attività e non sulle risorse.

I

Esatto. Dove poi più o meno cerchiamo di farla quadrare sulle risorse, ma in maniera un pochino spannometrica.

R

E quindi il motivo per cui è entrata nella terza risorsa era che?

I

Ci sono discorsi laterali di seniority, Cambiamenti di carriera. Questo fa parte un po' del macro sistema: attenzione che c'è un progetto però c'è sempre Un'azienda e delle persone Dentro quel progetto. E magari cozza un pochino col fatto che ogni tanto un po' a cuor leggero, ci diciamo. Ma si dai, è da fare e vabbè, anche se magari non è molto efficiente dal punto di vista del budget, però è da fare e lo facciamo.

R

Ok, quindi un po' per investimento interno, si è andato a bruciare del budget di progetto

I

Esatto, Quindi un pochino per quello un pochino perché Alcune delle CR che non sono poi state comprate, hanno pesantemente influito sul budget.

R

Come hanno fatto ad influire se poi non sono state comprate?

I

Ad esempio, lo studio delle performance per capire se potevamo usare quote 12 [versione nuova rispetto quella che il prodotto utilizza attualmente] come motore di configurazione. Quindi si è andato a capire se La lentezza nelle performance con questo applicativo sarebbe stato distruttiva e a capire se con Quote 12 avremmo performato meglio, ci è Costata una buona fetta di quel budget extra. Che è andato oltre il nostro stimato.

R

OK, quindi noi abbiamo fatto un investimento per questa CR nel momento in cui non è stata comprata quei soldi da Stati bruciati.

I

Esatto, inoltre abbiamo comunque incertezza perché abbiamo giocato con Tool e applicativi Da integrare, che era la prima volta che andavamo a integrare. Ad esempio optimizely o Keycloak, per tutta la gestione degli utenti, del login, della profilazione. anche lato CR, ad esempio, sulla multi sales org Siamo andati in affanno, sia lato cliente perché ci hanno fatto perdere un po' troppo tempo per definire requisiti, capire come avrebbe funzionato, sia poi lato nostro diciamo ci è Costata forse un'analisi preliminare, a un certo punto, fatta un po a cuor leggero, buttati oltre l'ostacolo, abbiamo finito col Diciamo pagare un pochino questa questa inesperienza. Ah, tanti altri fattori secondo me hanno influito. Hanno influito il roster Di team gonfiato rispetto a quanto doveva essere, perché comunque siamo arrivati a gestire in teoria 2 tecniche al leader allo stesso momento, che gestivano non numero infinito di risorse, Ma un numero abbastanza finito, comunque a livello di sviluppatori. Abbiamo toccato un apice di circa 5 6 sviluppatori E gestivamo, sei sviluppatori con due technical leader.

R

Vero, però è anche vero che per come è impostata la metodologia Con sei sviluppatori Oscar diventa un collo di bottiglia troppo forte

I

quindi, vuol dire che abbiamo un problema, oppure dobbiamo dire che con questo genere di team, noi dobbiamo andare a sovrastimare perché sappiamo che era certo punto Noi non potremmo permetterci di usare un solo Technical Leader Quindi diciamo che ambedue le parti [Paladin e Hero] lo staffing poi applicato per varietà di motivi Non ha rispettato lo staffing inizialmente pensato per il Progetto Quindi questo, se vogliamo andare a vedere La pecca del problema principale che c'è stato sul budget.

<p>R</p> <p>mi ha un po' toccato il discorso di Quote 12 perché evidentemente è stato fatto un investimento un po' sbagliato nel senso. Ma col senno di poi, c'erano degli indicatori che potevano far dire OK, effettivamente fare quell'analisi potevamo capire che era stato inutile oppure no?</p> <p>I</p> <p>In parte perché comunque loro continuavano a professare che con le performance Che si stavano vedendo Non avrebbero potuto dire che Si andava Live. Per loro è stato messo come stopper. Poi cliente, dopo avere un attimino capito quali erano le Conseguenze, cose da fare per poter Migrare a una soluzione diversa, ahimè ha soppesato dicendo Costi benefici di avere questa cosa per me non è buono. Quindi partendo inizialmente nel dire Eh no, con queste performance non si va live perché se noi diamo uno strumento nuovo dobbiamo anche migliorare le performance e, dimostrando noi che con questo motore di configurazione noi avevamo già fatto il possibile, L'unico altro modo era cambiare Motore. Il problema è che questo genere di stima, Richiede un approfondimento non indifferente. Quindi lì poi si azionano anche diciamo delle strategie commerciali Che probabilmente ci varranno altre cose in futuro. Dire però guarda che tu ci avevi spergiurato che Non si andava live, per cui noi abbiamo fatto Questo investimento per fare questa analisi è tu poi non c'hai comprata.</p> <p>R</p> <p>Però non non si possono adottare delle misure per mitigare il rischio, ad esempio dividere il costo dell'analisi.</p> <p>I</p> <p>Sì, Cosa che il CEO, hanno poi deciso di non applicare. Poi eventualmente il problema che noi abbiamo in questo momento è che se vogliamo andare a fare le pulci, noi avremmo dovuto andare a sistemare il file di budget O rimuovendo quei giorni dal dal carico o mettendo quei giorni come Investimento Coolshop Per andare a sanare il budget del progetto dopo che è stato detto che a livello commerciale non voglio esporre quei giorni al cliente. Perché più o meno quelle è un po' il livello grigio che si fa molta difficoltà e che bisogna pensare a un livello Progettuale e a un livello commerciale. Ovviamente le due cose ogni tanto sono un po' diverse, lì è un po' l'azienda dare la sua visione di cose è progettuale, cose commerciale. Cos'è un effort di progetto e Cos'è un effort commerciale</p>	
<p>R</p> <p>sempre per quanto riguarda un po' la dispersione di budget,</p>	<p>TURNOVER</p>

noi per come è fatta la metodologia e anche all'interno di questo progetto abbiamo avuto Un grande ricambio delle risorse Lato dev, soprattutto, pensi che questo abbia portato delle inefficienze oppure per come è impostato il progetto

I

Sì, perché comunque Il lavorare su progetti CUSTOM e lavorare su progetti di così ampio respiro, Sono una grossa barriera all'ostacolo Del puro agile, perché fanno sì che sia molto difficile Entrare e uscire, perché giustamente per noi entrare vuol dire Studiare tutto. Il Progetto non è stato pensato e ideato inizialmente per essere diciamo a micro servizi e quindi avere la possibilità di accendere una persona su Una piccola parte, fargli fare poca conoscenza di quel pezzo, fargli fare il ruolo di cui abbiamo bisogno e lasciarlo perdere.

R

E perché pensi che questo non sia stato fatto?

I

Perché non è nello stile del progetto e non è come è stato pensato il progetto, cioè, tecnicamente parlando, quello è un monolite. Non è mai stato pensato per essere fatto a micro servizi. E quindi questo implica che c'è molta difficoltà sia nel coordinare il lavoro e questo ci può trovare un po' di inefficienza Sul technical LEADER. Quindi, ahimè può seguire poche figure insieme e ci porta molto in efficienza nel portare a bordo una persona nuova. Perché o la persona lavora su un pezzo completamente nuovo, Oppure deve imparare molto prima di poter mettere le mani su quella parte di codice. Mentre magari in progetti pensati in maniera diversa e molto più facile andare a dividere In parti abbastanza semplici quello che bisogna fare. L'altro discorso è che Tutto dipende anche dal livello di seniority delle risorse Portate a bordo perché ovviamente più un dev è senior più ci si aspetta che lui sia in grado di lavorare su del codice scritto già da altri, che deve magari solo modificare o a cui deve aggiungere dei pezzi. Invece noi abbiamo avuto, facciamo un po' i conti, una seniority in media molto basso sul progetto perché comunque stiamo parlando di persone che sono nello sviluppo da esclusione fatta per il TL, Tre anni, due anni. Qualcosa di questo tipo. Che è uno dei motivi per cui non è possibile fare Agile.

R

Però, cioè mi ha dato solo vantaggi per il fatto di farlo in micro servizi, ma allora non ho capito il motivo per cui è stato deciso a farlo in modo monolitico.

I

Perché più o meno è lo stile Coolshop. farlo in maniera

Turnover porta inefficienza

Il prodotto è assimilabile ad un monolite

Gli sviluppatori necessitano di grande aiuto da parte del TL

Seniority dei Developers influenzano utilizzo del TL

<p>monolitica offre determinati vantaggi di Diciamo, velocità, affidabilità Del codice, perché è Un'architettura un pochino più semplice da. orchestrare, organizzare e anche da mantenere nel tempo. Perché a micro-servizi serve un modo di pensare iniziale molto diverso.</p>	
<p>R ti parlo di comunicazione. Volevo chiederti come reputi che sei stata condotta la comunicazione all'interno del TEAM? Sia riguardo cosa c'era effettivamente da fare per ogni risorsa, il grado di priorità dei vari task è anche il momento in cui cambiamo il grado di priorità dei task.</p> <p>I Allora, Il mio feedback che la comunicazione sia stata. Abbastanza buona. Trovo che l'avere introdotto le cerimonie un po più classiche dell'Agile, Stand up meeting, La Sprint Review, lo sprint planning anche in parte ci abbia aiutato molto nel riuscire a ad avere un pochino Di controllo su quello che avveniva Abbiamo forse fatto una scarsa comunicazione iniziale nei confronti del team di quello che era il piano, Chiamiamolo Waterfalliano tirato fuori. Sono 5 sprint grandi linee. Il progetto farà questo, questo, questo e quello, a grandi linee Noi avevamo in mente chi avrebbe fatto ciascun blocco e quindi tutto questo è un pochino mancato. Anche per tra Virgolette, dar modo alle persone di Prepararsi a quello che avrebbero dovuto affrontare più Avanti. L'altra cosa che secondo me ci manca un pochino è Una comunicazione più efficace su quello che quando andiamo a scendere un pochino più nel dettaglio del Task. Quindi dobbiamo essere più bravi nell'andare a Descrivere in maniera più corretta Quello che dobbiamo andare a fare quello che esigiamo sia il risultato finale. In questo siamo stati un pochino. Ci siamo un pochino fregati da soli per una serie di circostanze varie ed eventuali, nel senso che come analista funzionale, ahimè, dovevo capire prima che non potevo fare ambo due le figure Perché non avrei avuto abbastanza tempo materiale per seguire ambedue le cose e quindi siamo in realtà andati un po' in affanno sulla parte iniziale di progetto con propriamente la parte davvero funzionale, gli FD, L'analisi funzionale delle cose, Che finiva con l'essere un pochino trasmesso in maniera un po orale perché era stata fatta in fase di gara, in fase di Discovery era stata fatta, però non era mai stata davvero formalizzata e quindi quello è un pochino ciò che ci è mancato e che ci ha un po destabilizzato e magari non reso completamente efficienti.</p>	<p>COMUNICAZIONE</p> <p>Comunicazione valutata positivamente, ha aiutato il team a capire cosa stesse accadendo al momento</p> <p>Migliorare la visione di insieme del team</p> <p>Migliorare descrizione dei task di basso livello</p> <p>Carico di lavoro troppo elevato sul Paladin non ha permesso una buona raccolta dei requisiti</p>
<p>I L'altra cosa che ci è mancato a un certo punto un pochino di Qualcuno che tenesse un pochino le tirasse un pochino le somme. Quindi a un certo punto ci siamo trovati a dire ma in</p>	<p>ORGANIZZAZIONE DELLE RELEASE</p> <p>Release poco strutturate</p>

<p>teoria quella persona aveva fatto quella parte. Ma è davvero così? .Andiamo a vedere.</p> <p>R</p> <p>Ok quindi anche mancavano dei feedback a al contrario.</p> <p>I</p> <p>Sì, diciamo che ogni tanto ricevevamo solo uno solo spostare in Done Rispetto a Aver fatto tutto il resto delle cose.</p> <p>R</p> <p>Eh, però questo non dovrebbe essere tracciato in modo sensato attraverso le epiche? Non abbiamo già un degli strumenti di questo tipo?</p> <p>I</p> <p>Si. Ma quello che a un certo punto a noi è mancato è stato il concetto di release anche perché è vero che certi task sono stati sovrastimati, altri sono stati sottostimati, quindi guadagnavamo tempo e finivamo con il dover fare le cose un po' prima di quando avevamo pensato. E quindi anche l'organizzazione della release Ogni tanto ci ha un po fregato e soprattutto anche qua di nuovo, un po di mancanza di funzionale Nei primi periodi per testare e quindi diciamo che Ahimè, a un certo punto ci siamo inceppati sulla code review che fa parte un po di questo nostro modo ibrido di avere comunque un approccio... L'Agile direbbe che la code review non esiste, quando un task viene finito, viene mergiato e viene testato, punto e basta. E quindi a un certo punto, invece, avendo ancora questa code review, sempre per mitigare i problemi di seniority delle persone che abbiamo all'interno del progetto, Ci siamo un po inceppati. E quindi Eh no, ma quella roba è in code review, eh no ma quella roba è in code review. Non sono ancora riuscito a code revieware. quindi finché non la code reviewo non la puoi testare. La persona nel frattempo viene dislocata da quella cosa e va a fare la prossima cosa Io testo, mi deve correggere qualcosa, deve stoppare quello che sta facendo o magari devo aspettare qualche giorno perché lo finisca in modo che possa correggere. Nel frattempo stiamo andando un po in ritardo Perché magari lui aveva giustamente iniziato a lavorare alla Release dopo Aveva finito per tempo di fare le sue cose, però noi diciamo ok avevamo un bellissimo buffer di 5 giorni da quando la release finiva, fino a quando la rilasciavamo al cliente e in quei 5 giorni avremmo dovuto Finire di fare i test, sistemare quel p' di roba che dovevano essere sistemati a essere pronti al Rilascio però finiamo con il dire Oddio devo fare code review, oddio devo fare code review. Manca un giorno oddio, devo fare con review noi nel frattempo non possiamo tenere la risorsa ferma</p>	<p>Fase di test non strutturata</p> <p>Fase di code review non strutturata</p>
---	--

<p>per quattro giorni disallocata, e quindi incominciamo a dire no comincia a fare le robe dello sprint Successivo.</p>	
<p>I</p> <p>E tutto questo porta un pochino di inefficienza; quindi, diciamo che dove dobbiamo migliorare un pochino è l'allocazione Sia secondo me la differenza PM, analista funzionale. Sia un pochino sull'allocazione del TL, che deve ricevere un po' una mano e in questo, ahimè, siamo anche stati un po' fregati dal fatto che secondo me il TL è colui che era anche responsabile di Analizzare le CR. E quindi questo fa sì che a lui venga portato via ulteriore tempo rispetto al poco tempo, che magari già ha perché gli vengono date nuove risorse non trainate sul progetto Risorse che fra l'altro sono anche un pochino più junior, quindi sono da seguire. Deve magari, non avendo un FD, dare lui un pochino più una struttura o un corpo alle story. Si deve occupare di code review. Deve stimare le CR nuove. È quindi in tutto questo abbiamo avuto una cattiva gestione del tempo del technical leader. Idem con patate nel momento in cui noi perché quello che finiamo col fare è identificare un pochino il TL anche come uno sviluppatore, deve fare l'analisi delle nuove cose... La stessa cosa avviene dall'altra parte, cioè identificare la figura del PM con la figura dell'analista funzionale Diventa complesso perché fa sì che lui debba comunque conseguire tutto il l'iter tradizionale di progetto. controllare il budget. Seguire il cliente nelle situazioni, cosa sta succedendo, con non sta succedendo, dettare un pochino la timeline e nello stesso tempo però deve anche fare analisi, deve anche scrivere un FD, deve anche magari testare. Quindi tutto questo ci ha un po' ingolfato Abbiamo pagato questo ingolfamento con del ritardo. uno potrebbe dire Eh, però hai sottostimato le risorse e quindi dovevi aver avanzato budget, che infatti è quello che si evince, se uno va a vedere il file iniziale: noi nei primi mesi Viaggiavamo con un buon Delta vantaggio. Perché? Perché eravamo effettivamente forse sottostaffati rispetto a quello che ci serviva per davvero e continuava a dire vabbè, siamo sottostaffati. Però abbiamo ancora tempo, dai che recuperiamo, dai che recuperiamo. Ad Un certo punto abbiamo detto forse non recuperiamo, Inseriamo qualche risorsa in più. E da lì si sono innescate delle meccaniche un pochino a cascata di problematiche, le due cose chiave principali secondo me sono queste: Dare un pochino in più di aiuto al leader, quindi Andando a formare le risorse prima che il progetto inizi. Se vogliamo formare, se vogliamo fornire risorse un po' più junior. Oppure andare a sgravare qualcosa dalle spalle del TL, Ciò vuol dire che Tra tutte quelle cose lì qualcosa deve uscire, idem PM, non possiamo mantenere su progetti di così ampio respiro Due ruoli sulla stessa identica figura, cioè sulla stessa identica persona</p>	<p>ALLOCAZIONE DELLE RISORSE</p> <p>Avere una salda distinzione tra PM e Analista Funzionale per non ingolfare Paladin</p> <p>Technical Leader ha troppe responsabilità</p> <p>Spiegazione andamento budget</p> <p>Cattiva fase di Controllo</p> <p>Valutata positivamente la Possibilità di avere due Technical Leader, se dall'inizio del progetto</p>

R

in realtà stavo pensando che questo però contraddice un attimo quello che dicevi prima riguardo il TL, perché poi nella realtà dei fatti quello che è successo è che avete iniziato a bruciare tantissimo budget nel momento in cui c'erano due TL, in quanto c'erano poche risorse.

I

Esatto, perché per compensare il fatto che un leader avendo tutto questo smarmellone di cose da fare non riusciva a stare dietro, abbiamo detto ok, inseriamo un secondo TL.

R

In realtà, secondo te doveva iniziare dall'inizio del progetto in modo che avesse più visione sul progetto?

I

Esatto; quindi, OK o dall'inizio noi andiamo a tenere un TL che poi abbiamo avuto secondo me una forte disparità dell'entrata del secondo TL, perché secondo me si è preso la figura più skillata. e si è preso una sola figura quindi capisci che Non è un divido 50 e 50 e quindi riesco a fare le cose. In realtà è scalfisco qualcosina, Ma in realtà il vantaggio marginale apportato al seconda TL, se vado a vedere la figura che ha preso è il lavoro che ha seguito rispetto a quanto è costato, di budget è molto poco, perché ha rimosso poco peso dal primo tech leader E però ci ha portato un Tempo di budget e una complessità maggiore all'interno del circolo di informazioni non indifferente.

R

Siccome già conoscevate il cliente, avevate già in mano il vecchio progetto, mi sfugge come non siate riusciti a capire prima che forse servivano due TL o è inusuale in azienda?

I

È inusuale in azienda, l'esperimento condotto di cercare di avere due TL Non era andato benissimo. Diciamo che secondo me questo esperimento di avere due technical leader è andato nettamente meglio del precedente, per come è stata fatta la suddivisione, perché non sono stati suddivisi per stack tecnologico, non sono stati suddivisi tra front end e back end, ma sono stati suddivisi per task, quindi uno si è occupato di tutta la parte salesforce mentre l'altro si è occupato di tutto il resto. Quindi in quello, almeno fra virgolette, non ci sono stati problemi tecnici di comunicazione perché lavorando su Due Stream separati abbiamo dovuto comunque perdere un po di budget, un po di tempo lavorando su un determinato tasso in tre persone perché dovevamo accordare L'architettura generale tutti e tre, piuttosto che solo in due. Però, perlomeno abbiamo scampato problemi di prendere due strade diverse

<p>sulla stessa cosa. Diciamo, che secondo me l'inserimento di un secondo leader, appunto, non ha sgravato dalle spalle del primo abbastanza Peso per riuscire a avere un polmone nell'arco del progetto Su quella persona.</p>	
<p>R E invece lato clienti, come valuti la comunicazione?</p> <p>I Con il cliente la comunicazione Può essere valutata positivamente, nel senso, anche qua abbiamo un po' cambiato nell'arco del progetto Dove cambiato vuol dire che da subito il progetto ha avuto un approccio di un Weekly meeting di aggiornamento settimanale con, diciamo il team ristretto Che lavorava sul progetto da lato cliente più un meeting mensile A file più larghe lato cliente Anche, magari con stakeholder non di primissimo piano. Quello che forse è stato un pochino problematico anche per un po' di inesperienza del PM lato cliente È forse stato che avremmo dovuto introdurre Un Daily meeting, PM Coolshop, PM cliente, per cercare di affrontare prima Eventuali criticità, ritardi ambo le parti anche per eventuali risoluzioni di open point Che servivano. Quindi secondo me quello è stato uno dei problemi. Secondo me se avessimo instaurato questa comunicazione un pochino fin da subito, saremmo riusciti ad avere un po più di chiarezza anche per il cliente di quello che stava succedendo. Invece ogni tanto siamo arrivati quasi all'escalation, ma semplicemente perché il cliente si sentiva un po disinformato e magari il PM lato cliente non riusciva a informare completamente della situazione i propri responsabili e quindi abbiamo dovuto magari fare quella riunione in più, Quella call in più per riuscire ad informare anche il livello un po' più alto dello snapshot che si aveva del progetto in questo momento.</p> <p>R Chiaro, quindi comunque le risorse che Coolshop ha richiesto al cliente, comunque ti sono sembrate Soddisfacenti.</p> <p>I soddisfacenti, disponibili, ovviamente Bisogna un attimino secondo me, mediare nel senso che ci troviamo davanti a un caso fortunato, secondo me Dove è molto facile per loro prendere decisioni, non abbiamo un cliente che ha N livelli gerarchici attraverso cui passare per poter prendere una decisione, quindi anche quello è stato fortuna in parte nell'avere un cliente fatto così, che quindi già di suo riduce un pochino i tempi di risposta. E che quindi ha Fatto sì che nonostante non avessimo appunto magari questi meeting giornalieri per pressare sulle risposte Siano riusciti comunque</p>	<p>COMUNICAZIONE CON IL CLIENTE</p> <p>Valutazione positiva della comunicazione</p> <p>Inesperienza PM cliente ha causato problemi, che potevano essere risolti con brevi stand up meeting giornalieri</p>

<p>a Far le cose abbastanza per tempo rispetto a quanto ci aspettavamo.</p>	
<p>R Hai percepito utili le retrospective che ci sono stati durante il progetto? Sono state fatte anche con il cliente?</p> <p>I Non sono state fatte con il cliente, Principalmente per inabitudine Altrimenti avremmo dovuto inserire forse qualcosa che ci avrebbe ancor di più rallentato. Magari ingolfato Nel fare tutte le procedure, perché se Avessimo dovuto anche star dietro ai suoi consigli e le sue cose Sarebbe diventato magari un attimino più complesso fare il tutto. Lato team interno secondo me molto perché permettono a AI team di fare anche un pochino di valvola da sfogo, ove necessario, di individuare qualcosina. Ovviamente, ahimè, non siamo stati in grado di porre rimedio a tutte le obiezioni portate Dal team perché alcune sono Strutturali del progetto, quindi, probabilmente richiederebbero un progetto nel progetto per essere Sistemate, penso ad esempio a la mancanza di documentazione. Che è un po' lo spettro che aleggia su tutti i progetti. Secondo me invece siamo stati un pochino in grado di sistemare qualcosina. Ad esempio abbiamo ricevuto La necessità da parte del team di avere degli stand un po più brevi, siamo riusciti ad applicarlo, di avere più visibilità di quello che sarebbero stati i task settimanali e di qual era la situazione del progetto Che abbiamo cercato di inserirlo con il file che veniva condiviso a tra venerdì e lunedì di cosa sarebbe successo la settimana dopo E di un pochino quello che era la situazione del progetto A livello generale, quindi stiamo andando bene, stiamo per finire la release, la prossima release avrà queste cose. Più o meno anche raccogliere i feedback dei ragazzi, dove appunto magari uno pensa, vabbè, ma perché è così importante che tu sappia cosa stia succedendo A livello generale, invece, ne hanno tratto giovamento, anche perché ha permesso loro di incominciare ad aiutarsi l'un l'altro. O perlomeno di dire Ah OK, tu devi fare quella parte lì. Sappi che senza che te lo dica il TL o senza che te lo dica il PM, L'altra parte collegata a questa cosa l'aveva fatta quell'altra persona quindi se c'è qualcosa che non capisci, se c'è qualcosa che non ti torna, chiedi a lui oppure ah, però sappi che tu devi far questo, ah, io l'ho fatto in un'altro progetto e ho incontrato queste difficoltà e ho risolto in questo modo qua. E un pochino anche quello ha aiutato la comunicazione interna Del team, insomma, decisamente molto, molto utili. Ahimè, c'è questo spauracchio in senso, vanno un pochino settate le aspettative sul fatto che Se c'è qualcosa di molto lungo termine da risolvere è molto difficile risolverlo all'interno del progetto. Perché se ci diciamo manca documentazione del progetto, l'unica cosa che possiamo fare</p>	<p>MIGLIORAMENTO DEL TEAM</p> <p>Difficile organizzare retrospective con il cliente perché team ingolfato</p> <p>Retrospective valutate positivamente, team impara da sé stesso</p> <p>Team è diventato più autonomo nel tempo</p> <p>Aspettative vanno settate perché non è possibile implementare tutto ciò che è richiesto</p> <p>Il miglioramento richiede effort che il team non possiede</p>

per porre rimedio e dire va bene, allora almeno scriviamo la nuova documentazione.

cioè poniamoci come obiettivo che manca la documentazione del passato. Va bene, è un problema? Sì. Però perlomeno cerchiamo di non far mancare nel futuro la documentazione di questo. Quindi questo è documentato. Cercheremo. Con rimasugli di budget in tempi morti di progetto, di creare la documentazione del passato, però va un po' settata l'aspettativa, perché altrimenti corri il rischio di avere gente Delusa che dice Eh, ma tanto non cambia mai niente, è inutile che facciamo queste. Riunioni. Perdiamo solo tempo e andiamo a lavorare perché tanto cosa le facciamo a fare? Se io ti dico questo e questo non cambia mai.

R

Certo, e tu dici, Nel momento in cui siamo già in corsa e io sono iper operato di lavoro, È un po difficile poi applicare le cose che escono dal meeting.

I

No, nel senso vanno applicate, però ci saranno comunque alcune cose che richiedono un effort tale. Che non potrai risolverle In un mese o in due mesi o in tre. E soprattutto ti richiederebbero del budget o magari ti chiederebbero degli dei key user che in questo momento tu hai Ingolfati da un'altra parte, perché per scrivere la documentazione del passato, soprattutto su determinati passaggi, abbiamo bisogno del nostro technical leader.

Che però ci siamo già detti è operato In tutta un'altra serie di cose. Quindi diciamo che quello diciamo è molto importante anche come uno setta

L'aspettativa di questi momenti di comunicazione, perché comunque non siamo macchine e c'è un aspetto emozionale che è molto importante Mantenere e coltivare su un progetto che dura 9/12 mesi. Perché Su un progetto che dura un mese, possiamo dirci che possiamo cercare di tenere il malcontento per un mese. E uno tiene il malcontento per 9, 12 mesi.

Quindi Bisogna cercare di portare avanti il team mostrando che Ciò che è fattibile viene fatto, ciò che non è fattibile si spiega perché non è fattibile, si cerca di dire troviamo il modo per farlo diventare fattibile o ricordiamoci che questo è infattibile, Troviamo dei dei mitigatori Se poi non possiamo risolvere completamente quel punto, Facciamo del risk management per riuscire a mitigare il rischio, tanto che ci basta per non farci venire il mal di pancia.

R

E invece per quanto riguarda dei One to One, magari Tra PME technical leader, sono stati effettuati.

I

Quotidianamente, quasi nel senso che sicuramente c'era un

momento dedicato sicuro che era quello del chiamiamo planning della settimana Però tendenzialmente un punto di wrap up viene poi fatto quasi quotidianamente tra le due persone ed è Utile	
---	--