

Agile Project Management

L'approccio "Agile" al Project Management" e la relazione con il Portfolio Management.

Il fenomeno del momento

C'è un gran fermento intorno allo sviluppo "Agile", ma in realtà cosa è?

Il Leone e la gazzella

*Ogni Giorno, in Africa, si svegliano un leone ed una gazzella.
Il leone sa che deve correre più veloce della gazzella,
altrimenti morirà di fame. La gazzella sa che deve correre più
veloce del leone, altrimenti sarà mangiata.
Che tu sia leone o gazzella, appena sveglio, comincia a correre!*



Agile significa leggero, veloce, capace di cambiare direzione velocemente, ... e la gazzella sa di esserlo. Ma sanno di esserlo anche leone e ghepardo. *"Il ghepardo non attacca la gazzella con rabbia. La tocca per non farla scappare. Non la morde ma aspetta. L'accompagna al suolo e con molta attenzione cerca il collo. La Gazzella non odia il ghepardo."* Da **"Earth" - Impariamo dagli animali.**

Con le metodologie "Agile" si ottengono dei risultati fin dalla fase iniziale del progetto, anche se non è ancora chiaro cosa effettivamente bisogna produrre. Lo scoprirà successivamente il team di progetto in collaborazione con il referente dell'utente. L'approccio "Agile" consente di sviluppare software di alta qualità in breve tempo ed in armonia con l'utente.

Quest'approccio certamente accattivante per il management, può trasformarsi in un enorme rischio per il team di progetto, se non riesce a dominare le pretese dell'utente.

Le metodologie "Agile" si applicano esclusivamente ai progetti di sviluppo software e difficilmente sono trasferibili in altri settori. Però in questo settore l'approccio "Agile" è veramente competitivo se si combinano l'esperienza dello sviluppatore e la collaborazione fattiva dell'utente. E' fondamentale la lealtà dell'utente, perché lo sviluppatore ha poco margine per delimitare l'ambito del progetto nel dialogo con l'utente.

Che cosa è "Agile" ?

Il movimento "Agile", nato negli anni 80, riassunto in un "Manifesto" nel 2001, consiste in un insieme di **principi tecnici** e **prassi** che consentono di sviluppare un progetto software per approssimazioni successive. Lo sviluppatore avvia la realizzazione del prodotto sulla base di specifiche molto generiche, producendo e mostrando qualcosa all'utente, il quale con altri particolari chiarisce meglio la sua idea iniziale fino a rappresentare la vera esigenza. Lo sviluppatore lo asseconda e gli mostra la versione successiva del prodotto. L'utente si entusiasma e avanza altre richieste di particolari del prodotto. Questo idillio continua finché l'utente non si contraddice o finché lo sviluppatore non invoca il bisogno di concludere il progetto.

Insieme sviluppatore e utente sono molto più produttivi di un team tradizionale, a condizione che riescano a negoziare la quantità di variazioni in funzione del tempo disponibile. La qualità del software sarà superiore perché, di solito, questi progetti vengono affidati a sviluppatori esperti.

Questi progetti sono idilliaci, se affrontati con il giusto entusiasmo e finché sviluppatore e utente non entrano in contrasto. Se litigano è la fine del progetto!

Perché utente e progettista dovrebbero litigare?

Il conflitto nasce spontaneo quando l'ambizione dell'utente non corrisponde alla disponibilità di tempo e di budget dello sviluppatore. Mentre l'utente è focalizzato esclusivamente sulle funzionalità dell'applicazione, lo sviluppatore deve considerare anche l'ambiente che sta creando ed il successivo supporto. Ogni ulteriore compromesso può comportare un vincolo per la manutenzione successiva.

Ad un certo punto, le prospettive possono essere diametralmente opposte: L'utente vede l'applicazione dal punto di vista esterno, mentre lo sviluppatore si preoccupa più dei vincoli interni al codice, al disegno del database, alle procedure di salvataggio e ripristino, alle sincronizzazioni, alla integrità dei dati, etc. Spesso l'utente ignora completamente questi fattori ambientali, preoccupandosi esclusivamente dell'automazione di un processo.

Ad esempio, l'utente dovrebbe essere il principale destinatario della documentazione, dovrebbe pretenderla e contribuire a crearla durante il progetto. La documentazione riguarda il manuale utente, il codice sorgente, la configurazione del database, le immagini e le procedure di installazione. Non dovrebbero mancare le procedure di salvataggio e ripristino di applicazione e dati.

Normalmente, quando si adotta un approccio "Agile" all'inizio va tutto bene. Dopo alcune iterazioni (chiamate Sprint) emergono piccole frizioni. L'utente avanza più pretese, lo sviluppatore tende a resistere, creandosi due fazioni contrapposte. Qualcuno assume la leadership (effetto branco). Quando nessuno dei due riesce ad imporsi si genera una latente sfiducia reciproca ed il progetto inizia a perdere colpi, fino alla sgradevole situazione in cui lo sviluppatore non può né tirarsi indietro, né consegnare ciò che ha prodotto, perché non ha il gradimento dell'utente oltre a non essere un prodotto finito. E' il preludio del conflitto, che presto arriverà. E' solo questione di tempo.

Ciò accade quando si adotta il processo "Agile" senza invocare precise prassi per il livello di qualità, l'ambito del progetto ed il numero di variazioni all'ambito.

In simili situazioni, tutti rimpiangono e invocano le regole dello sviluppo tradizionale a cascata (raccolta e analisi requisiti, disegno, codifica, test, delivery, supporto, manutenzione) e spesso è l'unica via d'uscita, anche se a posteriori.

Che cosa è l'approccio SCRUM ?

SCRUM è uno degli approcci "Agile" allo sviluppo software. Più che una metodologia è una filosofia. Anziché richiedere la descrizione dettagliata dei requisiti e come dovrebbe essere il prodotto finale, viene lasciata la massima libertà al team di sviluppo ed all'utente, il quale periodicamente integra i requisiti alla soluzione provvisoria ipotizzata dallo sviluppatore.

Con l'approccio SCRUM l'utente si esprime in termini di risultati desiderati e non di specifiche o dati d'impianto, senza vincoli specifici: definizione attività, criteri di collaudo, uscita di fase come prevedono le tradizionali metodologie di project management dette "a cascata".

Comunque, trattandosi di progetti relativamente piccoli, neanche il team segue la logica della definizione e schedulazione delle attività, ma realizza e modifica il prodotto sulla base di ogni richiesta accettata.

Il progetto procede in modo duale:

1. il team realizza ciò che è stato accettato e messo backlog e lo mostra all'utente.
2. L'utente visiona ciò che è stato realizzato e avanza nuove richieste che vanno a ripopolare il *backlog* per gli *sprint* successivi.

Ciclicamente si procede con questa alternanza finché il prodotto non viene ritenuto soddisfacente.

Come scegliere un approccio "Agile"

Non c'è alcun dubbio che in certi scenari l'approccio "Agile" sia vincente, però i criteri per investire in un approccio "Agile" dovrebbero essere:

- ✓ **Realizzare prodotti di alta qualità** – L'attenzione di un esperto dovrebbe garantire la massima qualità, motivando il gruppo e negoziando scelte adeguate a budget e tempo.
- ✓ **Time to Market** – per definizione una soluzione con approccio "Agile" dovrebbe rispettare o bruciare i tempi.
- ✓ **Maggiore produttività** – la semplicità dell'approccio richiede decisioni veloci e cicli di revisioni molto stretti per soddisfare le aspettative dell'utente.
- ✓ **Minima manutenzione** – la soluzione dovrebbe includere tutte le esigenze dell'utente, in modo da ridurre la manutenzione ai soli cambiamenti di legge.

- ✓ **Allineamento architetturale** – L’approccio “Agile” consente di sviluppare soluzioni perfettamente allineate alle tecnologie disponibili, sfruttandone le capacità a garanzia del ritorno degli investimenti.
- ✓ **Riduzione della complessità** – disegnare e sviluppare solo ciò che serve, con la possibilità di potenziare l’applicazione con nuove funzioni.
- ✓ **Aumento della prestazione** – i team di progetto devono imparare ad interagire con l’utente, migliorando continuamente le prassi, in modo da evitare discussioni inutili e pretese impossibili. Le prestazioni sono strettamente legate ai prodotti di alta qualità.
- ✓ **Aumento della cultura aziendale** – focus alla soluzione piuttosto che alle singole attività.

Come partire con il piede giusto

Gli aspetti più importanti nell’adozione di una metodologia, compreso l’approccio “Agile” sono:

- ✓ Le **persone** - occorre la giusta esperienza per comprendere il linguaggio dell’utente.
- ✓ I **processi** - occorrono processi semplici e chiari definiti a livello aziendale, in modo che in caso di contrasti sia chiaro come individuare il *decision maker* finale.
- ✓ Le **tecnologie** - esistono troppi tool open che aiutano ad automatizzare i processi. Le nuove tecnologie sono importanti, ma non bisogna farsi condizionare più del necessario.

Anche con l’approccio “Agile” è fondamentale creare la cultura condivisa e procedere ordinatamente:

1. **Formare le persone** - l’approccio “Agile” richiede capacità comunicative e abilità nel negoziare le migliori soluzioni. Bisogna saper ascoltare, avere esperienza nel settore d’industria e saper gestire le persone. Il PMI lancia un’apposita certificazione sulle competenze “Agile”.
2. **Stabilire delle regole** - anche se il processo iterativo tra team di progetto e utente è molto semplice ed informale, occorre stabilire a priori delle regole, per evitare la situazione di panico quando emergono divergenze di opinione o di interesse. Le regole non son statiche, perciò dovranno essere oggetto di miglioramento continuo con l’apporto dei suggerimenti di tutti e di qualcuno che decide di emendarle a nome dell’azienda.
3. **Best practice** - si suppone che i progetti “Agile” vengano affidati a persone di esperienza, ma spesso i progetti sono delle innovazioni uniche. Bisogna in ogni caso avere dei modelli di riferimento per non essere completamente in balia dell’inventiva del team di progetto. Ciò che viene proposto dal team di progetto deve essere considerato fattibile dall’organizzazione ed apprezzato per le piccole innovazioni, ma non potrà mai essere uno stravolgimento - saremmo nella ricerca.
4. **Integrazione continua** - Lo sviluppo software, a volte banalizzato, può riservare delle brutte sorprese se le soluzioni proposte non si integrano con il contesto tecnologico disponibile o in prospettiva. Non si può ignorare l’architettura o la piattaforma sulla quale poggia l’applicazione da realizzare. In sostanza, la creatività deve fare prima i conti con il contesto, altrimenti è un boomerang.
5. **Procedere per passi successivi** - disponendo di principi, prassi e strumenti è opportuno stabilire la giusta sequenza dei passi da compiere. Occorre partire da poche prassi per volta (una o due) e verificare l’effetto prima di procedere oltre. Per adottare più prassi contemporaneamente bisogna aver maturato una enorme capacità decisionale di fronte a scelte che possono produrre anche effetti contrari a quelli desiderati. Ricordarsi sempre che i problemi li creano (e li risolvono) le persone e non le tecnologie.
6. **End to End** - anche se lo sviluppo software si concentra sul codice da produrre, non bisogna trascurare che il progetto, in definitiva, è la soluzione di un problema o la realizzazione pratica di un’idea. Il percorso dal problema alla soluzione deve essere lineare e consequenziale. Alterare l’ordine dei processi può causare inutili rifacimenti e perdite di tempo e denaro.

Quando utilizzare l’approccio “Agile”

Lo sviluppo “Agile” è fortemente indicato quando esiste un senso di appartenenza e tutti sono motivati a realizzare un prodotto di qualità. Non è adatto agli ambienti troppo burocratizzati o conflittuali.

L’adozione dell’approccio “Agile” richiede consenso e ambizione a dimostrarne la validità. Bisogna esserne convinti, per avere successo. Il modo più facile per creare consenso è la formazione **di base** perché coinvolge i singoli individui offrendo loro l’opportunità di esprimere le proprie potenzialità. Con la formazione si possono trasmettere principi e prassi ed, in fine, far conoscere gli strumenti.

Nel mondo "Agile" si inizia a parlare di "**Agile Processes**", "**Agile Portfolio Management**" ed altri accoppiamenti che fanno pensare all'agilità.

Ecco una riflessione necessaria su un Portfolio Agile: un insieme di processi per scegliere quale lavoro eseguire, quali risorse dedicarvi e come gestire il lavoro. La gestione del Portfolio consente di reagire velocemente alle nuove esigenze del business. Si dispone del cosiddetto "**pivot**" - l'uomo che in una partita di basket decide il cambio di direzione.

Immaginiamo che mentre tutti sono dediti al proprio progetto, arriva la notizia di una nuova acquisizione o di una grossa opportunità da cogliere con le stesse risorse disponibili.

In quanto tempo siamo capaci di reagire?

Possiamo rispondere in due modi:

1. Rubare risorse ai progetti in corso, allungando i loro tempi.
2. Cancellare qualche progetto, rinunciando definitivamente a quel business.

L'approccio migliore sarebbe avere un portfolio di progetti ottimizzati in modo da partire e finire il prima possibile. **Ottimizzando l'utilizzo delle risorse i progetti tendono a finire in tempi più brevi.**

La capacità di avviare e ultimare più progetti in tempi brevi consente di scegliere di avviare i progetti più in linea con la strategia e gli obiettivi di business, a discapito dei progetti con priorità più bassa.

Completando più progetti in un determinato periodo, si dispone di più risorse per i nuovi progetti.

Questo ragionamento, molto elementare, risulta evidente in presenza di una gestione appropriata del portfolio progetti fino a tradurlo in cifre (tempo o denaro risparmiato). Provo a spiegarlo con un esempio.

A volte le aziende hanno troppi progetti in corso, più di quelli che potrebbero permettersi in base al numero di risorse disponibili.

Per esempio, invece di avere due progetti che richiedono tre e cinque mesi, ci sono quattro progetti che utilizzano lo stesso gruppo di risorse. In questo modo i progetti si contendono le risorse, con il risultato che il progetto da tre mesi ne impiega sei, mentre il progetto da cinque mesi ne impiega dieci. Si genera un circolo vizioso, perché, pur di avviare più progetti si fanno partire anche in assenza delle risorse necessarie.

In pratica gli Sponsor non aspettano che le risorse siano realmente disponibili, ma pretendono che si avvii comunque il loro progetto, il quale anche se entra in esecuzione, non avanza, a meno che non rubi delle risorse ad altri progetti allungandone i tempi.

Questo scenario non è lontano dalla realtà di molte software house. Si risponde positivamente ad ogni esigenza di business, come se ci fossero risorse infinite. Ma tutti sanno che ciò non è vero.

Purtroppo i benefici di un progetto si hanno quando termina e non quando inizia. "*La farina si vede al ritorno dal mulino e non all'andata*". Questo principio è ancora più lampante **con l'approccio "Agile", dove la gestione del portfolio progetti deve basarsi sulla fine dei progetti e non sull'inizio.**

Bisogna determinare il numero di risorse necessarie per ultimare il lavoro il più presto possibile, senza gonfiare il numero di progetti in portfolio. Senza la disponibilità delle risorse necessarie, la presenza di più progetti in esecuzione è una pia illusione che, alla fine, risulterà dannosa, perché si accumuleranno ritardi e allungamenti di più progetti per effetto della contesa delle stesse risorse.

In un portfolio Agile, i progetti devono partire e finire nel più breve tempo possibile. Anche se non hai troppi progetti in un determinato momento, a fine anno scoprirai di averne chiusi molti di più. Inoltre, poiché i progetti partono e finiscono più velocemente, in ogni ciclo, avrai sempre dei progetti che terminano, consentendoti di disporre di risorse per avviare nuovi progetti e in caso di necessità hai più flessibilità nello scegliere i progetti più in linea con la strategia aziendale "*concetto del pivot*".

Il gruppo TenStep sviluppa questa filosofia nei modelli [PortfolioStep](#) e [LifeCycleStep](#).

Se stai pensando di ottimizzare il tuo ambiente di sviluppo con un approccio "**Agile**" stai per fare una scelta giusta, ma non trascurare la **Gestione del Portfolio Progetti** per trarne il massimo vantaggio.

Noi possiamo aiutarti, 1) concedendoti l'utilizzo del modello,2) fornendoti i nostri servizi di formazione e consulenza. Contattaci se sei interessato.

Se vuoi saperne di più, contatta direttamente TenStep Italia.

Vito Madaio, PMP

Info@tenstep.it

348-3974474



Vuoi certificarti PMP o CAPM,
utilizza
PMP-Prep Online

Omaggio del Gruppo TenStep

Nuovo eBook

[Download Modello AGILE](#)

Lascia un feedback in [TenStep Italia News](#).

