

Project Management: L'Avvento delle Metodologie Agile

*Le metodologie Agile riscuotono un grande successo presso le aziende più evolute.
Il PMI propone una nuova certificazione **Agile PMI-ACPsm**. Significa che i tempi sono maturi?
Forse sì anche se non mancano gli scettici. **Tu da che parte stai?***

Cosa è una metodologia Agile

Il Project Management è “**l’arte di far accadere le cose**” e per farle accadere occorre disporre di una valida metodologia, non ci si può affidare al caso!

Una metodologia è un insieme di principi che guidano l’attuazione dei processi a prescindere dagli strumenti utilizzati. Si tratta di cultura aziendale condivisa, sviluppata per perseguire il miglioramento continuo. **L’Agile Project Management** è un esempio di metodologia che persegue il miglioramento continuo, la soddisfazione del cliente e quindi la produttività.

L’Agile Project Management è una **filosofia di vita** da non confondere con gli importanti tool che lo attuano. *Gartner Group* sostiene che oltre l’80% delle aziende sta sperimentando qualche tool Agile, indice che l’interesse per l’Agile è altissimo, mentre la cultura in termini metodologici è ancora bassa. Le principali categorie di metodologie di sviluppo software sono:

1. **Waterfall** - Il progetto è suddiviso per fasi sequenziali (analisi requisiti, disegno applicazione, codifica applicativo, test funzionale, rilascio in produzione).
2. **Hybrid Waterfall** - Il progetto suddiviso per fasi è caratterizzato dalla possibilità di sovrapporre alcune.
3. **Incremental** - Il progetto si basa sulla consegna ciclica di poche funzionalità per volta.
4. **Iterative** - Il progetto si basa su tanti brevi cicli, ognuno approvato dall’utente.
5. **Agile** - E’ l’insieme di metodologie incrementali ed iterative che forniscono valore all’utente.

Le **prime due** categorie “**a cascata**”, vedono l’utente coinvolto all’inizio (durante la raccolta dei requisiti) e alla fine (durante il collaudo di accettazione):

- L’utente racconta le sue esigenze agli sviluppatori che le analizzano per disegnare l’applicazione.
- L’utente approva analisi e disegno ed esce di scena, in attesa della fase di collaudo.
- Gli sviluppatori realizzano l’applicazione sottoponendola a collaudo a fine ciclo di vita del progetto.

Le **ultime tre** categorie, dette “**iterative**”, procedono con piccoli cicli (**sprint**) senza concetto di fase.

I requisiti vengono raccolti, discussi, valutati e stimati all’inizio di ogni ciclo con l’utente sempre coinvolto:

- L’utente è parte integrante del team di progetto.
- La licitazione e l’analisi dei requisiti è distribuita su tutto il progetto, con approvazioni di piccoli incrementi successivi in base ai risultati ottenuti.
- Il team di progetto e l’utente collaborano intensamente sotto la guida di un esperto di architetture risolvendo un problema per volta e incontrandosi frequentemente e velocemente per condividere le decisioni maturate con il contributo di tutti.

In sostanza, **l’Agile Project Management** si affianca alle metodologie tradizionali nell’area dello sviluppo software e in alcuni casi le sostituisce egregiamente.

Negli anni ’80 abbiamo assistito all’avvento dello sviluppo software a oggetti ai danni dei classici linguaggi di programmazione Cobol, Pl/1, Fortran sostituiti da C+, Java, HTML, etc. Le applicazioni distribuite hanno generato un nuovo paradigma in termini di sicurezza, integrità e prestazioni, rendendo indispensabile l’automatizzazione della gestione del ciclo di vita dello sviluppo.

Visti i risultati deludenti, è stato gioco forza abbandonare i mega progetti di sviluppo pluriennali con contratti miliardari e preferire progetti più piccoli in grado di dare risultati immediati. La crisi finanziaria globale consiglia l’approccio: *pochi, maledetti e subito!*

I principi del Manifesto Agile

L’esplosione delle applicazioni distribuite con la diffusione del Personal Computer ed Internet hanno determinato un certo disagio negli esperti di sviluppo software, trasformatosi in un movimento che **17**

visionari hanno sintetizzato nel famoso "**Manifesto per lo Sviluppo Software**" (<http://www.tenstep.it/Agile/Agile-Manifesto.html>), nel febbraio del 2001.

Dopo un decennio la promessa è stata più che mantenuta. I firmatari del manifesto sono rimasti attivi, contribuendo allo sviluppo dei principi enunciati nel Manifesto con nuovi tool, libri e attività di formazione in tutto il mondo come riassunto in "**Autori del Manifesto Agile**" (<http://www.tenstep.it/Agile/Autori-Agile-Manifesto.html>).

Il Manifesto Agile fu una dichiarazione di guerra agli sprechi ed un inno alla produttività, alla collaborazione con l'utente per la soddisfazione di entrambi: sviluppatori e utenti. Secondo il Manifesto Agile sono più importanti:

- *Individui e interazioni* piuttosto che processi e strumenti.
- *Software funzionante* piuttosto che la documentazione esaustiva.
- *Collaborare col cliente* piuttosto che negoziare contratti.
- *Rispondere al cambiamento* piuttosto che seguire rigidi piani.

I principi del Manifesto Agile sono:

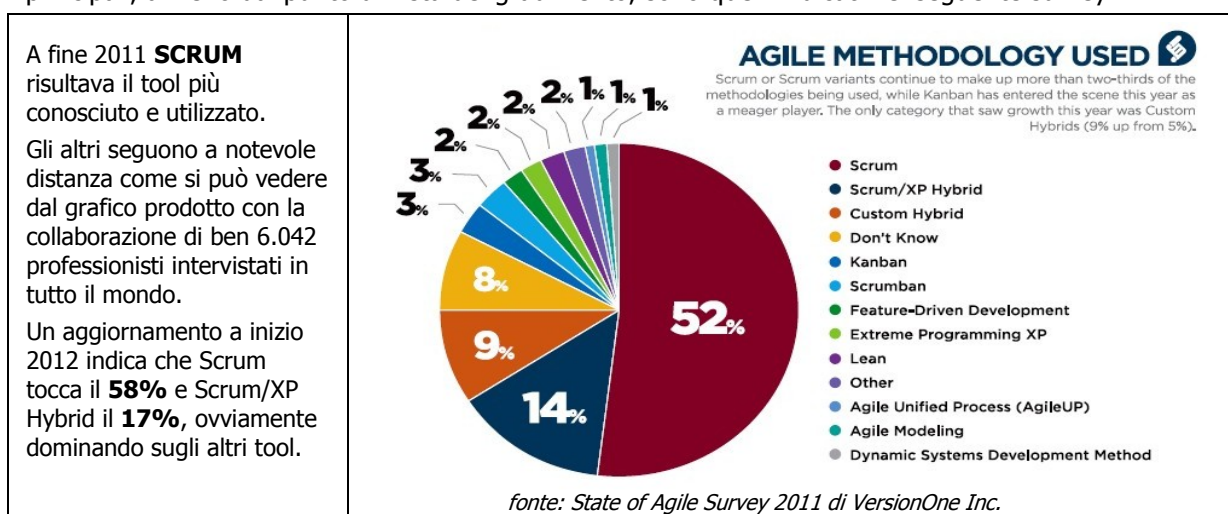
1. Soddisfare il cliente rilasciando software di valore fin dalle prime interazioni.
2. Permettere al cliente di continuare a chiedere modifiche all'applicazione.
3. Consegnare software funzionante con più frequenza.
4. Collaborare con l'utente durante tutto il progetto.
5. Affidare i progetti a persone motivate.
6. Preferire la comunicazione faccia a faccia.
7. Misurare gli avanzamenti con la quantità del software rilasciato.
8. Promuovere lo sviluppo sostenibile.
9. Esaltare l'agilità con buone soluzioni software.
10. Semplificare, massimizzando il lavoro rimanente.
11. Aiutare il team a dare risultati migliori, autogovernandosi
12. Essere più efficaci, cambiando comportamento.

Tutti i tool sviluppati successivamente si sono ispirati a questi principi.

I Tool Agile

L'industria dello sviluppo software ovviamente per prima cosa produce tool per lo se stessa. Ne esistono di molto semplici a molto sofisticati o esclusivi per determinati ambienti proprietari.

I principali, almeno dal punto di vista del gradimento, sono quelli indicati nel seguente survey.



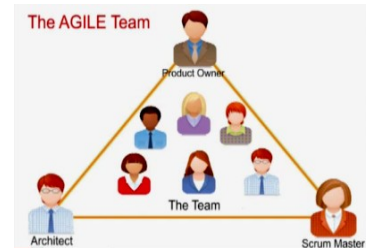
Cosa è SCRUM

Scrum, è uno schema di project management "**general-purpose**" dove i progetti procedono con una serie di iterazioni della durata da 2 a 4 settimane, dette **sprint**. Il team, formato da 5 a 9 risorse, in alcuni casi da centinaia di persone o anche da una sola persona, non prevede le figure tradizionali (analista, programmatore, etc.), ma una serie di nuovi ruoli, ben definiti, che collaborano nel cercare di

soddisfare tutte le richieste dell'utente. Tutte le decisioni vengono prese a livello basso in un clima di auto governo.

Un team di progetto Agile si compone di:

- Product Owner
- Scrum Master
- Team di sviluppo
- Scrum Architect.



L'approccio Scrum prevede regole precise per le riunioni iterative e la gestione del *backlog* del lavoro, sviluppando un senso di cameratismo che favorisce creatività, collaborazione e produttività.

Il backlog si alimenta gradualmente, man mano che l'utente propone altre *stories*, dando origine al *product backlog*, dal quale viene estratto lo *Sprint backlog* di ogni iterazione.

Lo *Scrum Master* e lo *Scrum Architect* intervengono rispettivamente per verificare l'utilizzo appropriato dei processi Scrum e l'aderenza alle architetture disponibili o da acquisire.

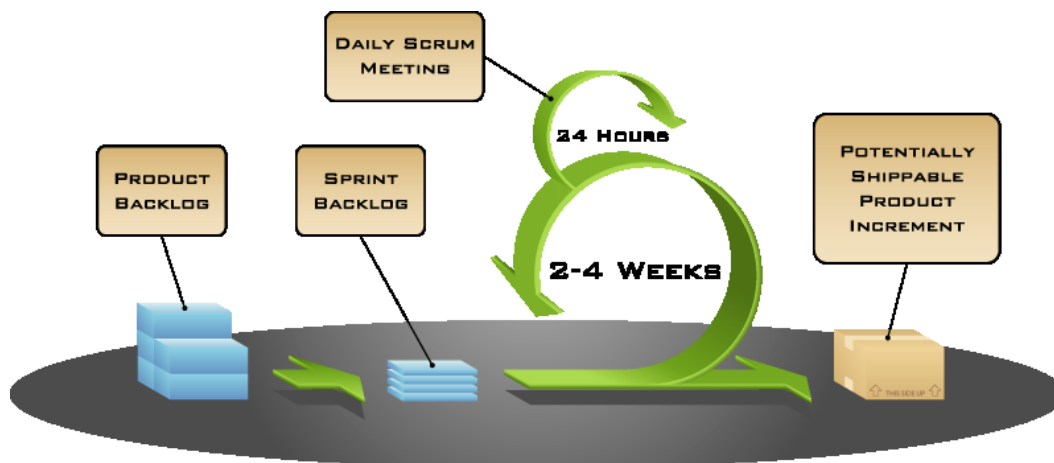
Il team si incontra ogni giorno in una breve riunione detta "*Daily Scrum*" dove ogni partecipante risponde alle seguenti domande:

1. Cosa hai fatto ieri?
2. Cosa farai oggi?
3. Quali ostacoli vedi nel tuo lavoro?



Alla fine di ogni *sprint*, il team tiene una demo, detta *sprint review meeting*, per esporre ciò che ha realizzato durante lo *sprint*. Dopo la demo in un'altra riunione, detta *sprint retrospective*, si esamina cosa cambiare per lavorare meglio.

Mike Cohn rappresenta Scrum con il seguente grafico di notevole eleganza:



COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

A sinistra vediamo la *product backlog* con le relative priorità stabilite dal *product owner* con tutto ciò che si sa del prodotto alla data, al centro lo *sprint* tipo di 2-4 settimane. All'inizio di ogni *sprint*, il team decide lo *sprint backlog*, la lista di task con le rispettive stime da completare nello *sprint*. Alla fine di ogni *sprint*, il team produce un rilascio parziale, mentre ogni giorno inizia con il *Daily Scrum Meeting*.

Un team Agile si muove velocemente nel contesto con questa nuova terminologia, rilasciando continuamente incrementi di applicazione. Il team si auto governa demandando le decisioni a chi è effettivamente commesso e non a chi è solo coinvolto, motivo per cui durante il *Daily Scrum*, alcune figure possono assistere alle riunioni ma non possono parlare o proporre soluzioni.

Chi è lo ScrumMaster

Lo **ScrumMaster** è l'allenatore, il coach del team come in una squadra di rugby. Si preoccupa di far utilizzare le prassi Scrum traendone il meglio in termini di produttività e di coesione del team. Viene chiamato

anche *process owner* in contrapposizione al *product owner* con il quale negozia il contenuto del *product backlog*, specie per le *stories* che entrano negli *sprint* successivi.

Questo ruolo viene ricoperto, di solito, da un project manager di esperienza con lo scopo di proteggere il team dalle pretese del *product owner* o dalla compiacenza.

Il team potrebbe accontentarsi di facili successi e smettere di cercare il miglioramento continuo. Lo ScrumMaster previene questi rischi, frenando la pressione del *product owner* quando il team è troppo carico e richiedendo più lavoro quando il team è più scarico. Questa figura, non ha potere sul *team*, ma ne ha tanto sui processi, per cui può determinare come lavorare meglio per raggiungere gli obiettivi del progetto. Questi scenari sono descritti in **Fondamentali di Scrum** (<http://www.tenstep.it/Agile/Download/WP/Fondamentali-di-Scrum.pdf>) e sintetizzati in **Introduzione a Scrum** (<http://www.tenstep.it/Agile/Introduzione-SCRUM.pdf>) di Mike Cohen.

Approccio Sashimi

Il “*Sashimi*” è un piatto di pesce crudo tagliato a fette sottili ed infilzato per dare l’idea del pesce intero, una specie di spiedino.

Il termine “**Sashimi**” in giapponese si riferisce ad un “*corpo infilzato*” con riferimento alla pratica di infilare la coda e le pinne nella polpa del pesce per renderlo riconoscibile durante la preparazione dei loro piatti tipici. Un altro riferimento, sempre giapponese, è quello di infilzare il cervello del pesce appena pescato con la lenza, in modo da ucciderlo e metterlo subito sotto ghiaccio, per prevenire la formazione di acido lattico. Il **sashimi** è una raffinatezza giapponese.



I primi ad utilizzare il termine “*sashimi*” nell’industria furono gli ingegneri della Fujitsu negli anni ‘80 per indicare “**una serie di task correlati scelti per una iterazione**”. Agile ha mutuato questo concetto di velocità decisionale delle iterazioni di un progetto di sviluppo Agile.

Aniché disegnare tutta l’architettura ai vari livelli, ci si concentra sulla minima quantità di codice necessaria per legare le parti indispensabili a realizzare le funzionalità. Quest’approccio minimale consente a sviluppatori e utenti di valutare immediatamente la qualità del software prodotto, concentrandosi subito sull’iterazione successiva per migliorarlo. Le iterazioni successive aggiungono altre funzionalità architetturali in base alle esigenze dell’utente. L’**approccio Sashimi**, di tipo incrementale, fa affidamento su un valido sistema di interfacce che garantiscano il legame tra le varie iterazioni. In realtà, in ambiente SCRUM, con il termine **Sashimi** si intende anche il report di ciò che è stato realizzato (**done**), ossia il resoconto delle iterazioni realizzate, di quelle restanti e la stima a finire.

Dal report Sashimi nasce anche il concetto di “**velocità di iterazione**”: la produttività del team di progetto per singola iterazione o *sprint*, sulla base delle iterazioni già realizzati.

Certificazione Agile PMI-ACPsm

Agile è una filosofia di vita che vale la pena sperimentare e contribuire a migliorare come tutti i fenomeni evolutivi. Il **PMI** ha istituito la certificazione “**Agile PMI-ACPsm**” (<http://www.tenstep.it/Agile>) che garantisce la conoscenza dei principi Agile e la metodologia Agile basata su un rigoroso programma di esame su 43 aree di conoscenza. Infatti questa certificazione PMI, aperta a tutti i ruoli, prevede la conoscenza dei principali metodi: Scrum, XP, Lean, Kanban ed è molto più rigorosa delle certificazioni dei singoli tool. I prerequisiti sono: **1.500 ore** di esperienza nello sviluppo software con un approccio Agile più **2.000 ore** di esperienza generale sul project management (o il possesso della certificazione PMP) più **21 contact hours** di formazione attraverso un **REP** (*Registered Education Provider*) del PMI.

Vito Madaio, PMP

TenStep Italia



Vito Madaio - esperto di project management di lunga data - Già Service Manager alla Camera dei deputati per Cap Gemini; Direttore Sistemi Informativi del Gruppo Skandia e Architetto di Sistemi di IBM Italia. Attualmente, da circa 9 anni Responsabile di **TenStep Italia** e **REP** del PMI, si occupa principalmente di Certificazioni PMP, CAPM e Agile PMI-ACPsm, oltre a diffondere l’adozione della Metodologia di Project Management TenStep.