

# Agile Italia Magazine

[HTTPS://AGILEITALIA.AGILEFORITALY.COM](https://agileitalia.agileforitaly.com)



RIVISTA BIMESTRALE FREE

2023 - 5

fondata da AgileForItaly nel 2019

## REQUISITI

.....  
Raccontami una story  
di **Giorgio Irtino**

.....  
I requisiti: fraintesi, obsoleti e  
cambiati

.....  
di **Francesco Nardoni**

.....  
Navigare la trasformazione: i requisiti  
"bussola" per realizzarla

.....  
di **Marco Pecoraro**

.....  
Piovono requisiti da ogni parte?  
Cerca la radice del problema

.....  
di **Susanna Ferrario**

.....  
Le 3 R dei Requisiti

.....  
di **Rosa Linda Romano e Marco  
Arzilli**

.....  
Requisiti di **Gianluca Colonnese**

.....  
Si salvi chi può!! Dobbiamo  
partecipare ad una gara...

.....  
di **Luca Cianci**

.....  
Agile Transformation  
con **Marco Calzolari**

.....  
Sovrastrutture e Norming  
di **Tiziano Interlandi**

.....  
Industrializzazione Agile per  
Hardware di **Massimo Terzo**





## SUMMARY

- 10 Una questione di dialogo di **AgileForItaly**
- 14 Raccontami una story di **Giorgio Irtino**
- 17 I requisiti: fraintesi, obsoleti e cambiati di **Francesco Nardoni**
- 23 Navigare la trasformazione: i requisiti "bussola" per realizzarla di **Marco Pecoraro**
- 26 Piovono requisiti da ogni parte? Cerca la radice del problema di **Susanna Ferrario**
- 30 Le 3 R dei Requisiti di **Rosa Linda Romano e Marco Arzilli**
- 36 Si salvi chi può!! Dobbiamo partecipare ad una gara... di **Luca Cianci**
- 42 Requisiti di **Gianluca Colonnese**
- 46 Agile Transformation con **Marco Calzolari**
- 50 Sovrastrutture e Norming di **Tiziano Interlandi**
- 52 Industrializzazione Agile per Hardware di **Massimo Terzo**



Unsplash thanks to

la maggior parte delle foto all'interno di AgileItalia sono un regalo di Unsplash e dei suoi contributori. A questo link i fotografi che ci hanno aiutato

[https://unsplash.com/it/collezioni/HxVZmTSWE9g/2023\\_5](https://unsplash.com/it/collezioni/HxVZmTSWE9g/2023_5)

## #CHISIAMO

Vi domanderete il perché di questa cosa. Pensiamo che in Italia sia necessaria una divulgazione massiccia su Agile.

Come paese brilliamo di inventiva e capacità tecnica, ci manca però la struttura manageriale per portare questo paese ad un livello successivo.

Agile è un abilitatore. Abilita le persone a creare prodotti migliori, dominare la complessità ed introdurre una disciplina positiva, fatta di raggiungimento di obiettivi.

Ci sono tanti entusiasti in Italia. Agile in Italia sta esplodendo. Vogliamo aggregare e filtrare per tempo questa onda dando a tutti accesso alle informazioni ma anche una corretta e strutturata informativa di qualità.

Spesso siamo caustici, ma non confondetela con arroganza. Abbiamo i nostri modi, un po' da badile e trattore.

Ma ci piace così, genuino.

Cerchiamo di aggregare la conoscenza su Agile con tutto ciò che gira intorno: mondi, universi di informazioni, tecniche e pratiche difficili da sintetizzare e vogliamo far emergere le realtà pratiche che sappiamo essere molte ma ancora da scoprire.

### Sito

<https://agileitalia.agileforitaly.com>

### Fondatori ed Editori

Tiziano Interlandi, Pierpaolo Cimirro e Davide Casari

### Mail

[agileitaliamagazine@gmail.com](mailto:agileitaliamagazine@gmail.com)

Questo non sarebbe possibile senza ...



## TIZIANO INTERLANDI

Co-Founder di AgileForItaly, AgileItalia e AgileExperienceConference.

Agile Coach e Manager

in Cerved Group.

Batterista e Fono.

<https://www.linkedin.com/in/tizianointerlandi/>



## PIERPAOLO CIMIRRO

Co-Founder di AgileForItaly, AgileItalia e AgileExperienceConference.

Data Product Owner in Cerved Group.

<https://www.linkedin.com/in/pierpaolocimirro/>



## DAVIDE CASARI

Co-Founder di AgileForItaly, AgileItalia e AgileExperienceConference.

Product Owner in Cerved Group.

Co-Founder di Vhagin

<https://www.linkedin.com/in/davide-casari-311563123/>



# I requisiti: la base fondamentale per un progetto di successo



a cura di Valentina LANZUISE: Analista funzionale e PMO in Sintegra



I requisiti rappresentano la base fondamentale per il successo di qualsiasi progetto, sia esso di natura tecnologica o meno. La definizione dei requisiti di un prodotto, in maniera chiara e completa, è fondamentale per garantire che questo venga realizzato correttamente e soddisfi le aspettative degli utenti.

#### **Che cosa sono i requisiti?**

I requisiti possono essere definiti come le specifiche delle funzionalità, delle caratteristiche e

delle prestazioni di un prodotto che devono essere realizzate per soddisfare le esigenze degli utenti finali o dei clienti. Questi possono riguardare diversi aspetti, come ad esempio le funzionalità richieste, le prestazioni desiderate, oppure le interfacce con altri sistemi e molto altro ancora.

I requisiti possono essere espliciti, ovvero definiti in modo chiaro e concreto, o impliciti, ovvero dedotti da aspettative generali o buone pratiche nel settore.

segue ▷



◀ segue

In ambito informatico, e nello specifico nell'ambito del software, i requisiti rappresentano le specifiche delle funzionalità e dei comportamenti che questo deve rispettare.

Essi si possono dividere principalmente in due categorie:

- **requisiti funzionali:** descrivono cosa il software deve fare. Per esempio: "Il sistema deve gestire gli ordini dei prodotti, deve permettere di monitorare lo stato degli ordini in corso e di inviare delle notifiche quando questi vengono completati";

- **requisiti non funzionali:** definiscono i criteri di qualità o le limitazioni del sistema, come le prestazioni, la sicurezza o la scalabilità. Per esempio: "Il sistema deve adottare misure di sicurezza per proteggere i dati degli utenti, come crittografare dati sensibili."

### L'analisi dei requisiti e la sua importanza

Per un prodotto di successo l'analisi dei requisiti svolge un ruolo fondamentale. Essa è il processo attraverso il quale vengono identificati, raccolti, analizzati e documentati i requisiti di un prodotto. Questa fase è di vitale importanza perché un'analisi dei requisiti incompleta o inadeguata può portare a gravi problemi nella fase di sviluppo e nell'intero ciclo di vita del progetto. L'analisi dei requisiti richiede una comunicazione efficace tra gli stakeholder del progetto, come i clienti, gli utenti finali, i responsabili del progetto e gli sviluppatori, al fine di comprendere le necessità e le aspettative di tutte le parti coinvolte.

Definire in maniera corretta i requisiti è fondamentale per il successo del progetto. Requisiti chiari e completi consentono agli sviluppatori di comprendere appieno ciò che devono realizzare e aiutano a evitare ambiguità o fraintendimenti che potrebbero portare a errori costosi o a prodotti che non soddisfano le aspettative degli utenti.

Inoltre, definire bene i requisiti fornisce una base solida per il processo di sviluppo, consentendo una pianificazione accurata, una stima dei costi e dei tempi realistica e anche una valutazione dell'avanzamento del progetto.

### Best practice per la raccolta dei requisiti

Per raccogliere e documentare i requisiti in modo efficace esistono diverse tecniche e pratiche. Di seguito si riassumono alcune best practice, che scindono dal contesto e dalle metodologie progettuali adottate:

- **Coinvolgere tutti gli stakeholder:** coinvolgere tutte le parti interessate fin dalle fasi iniziali del progetto permette di ottenere una visione completa delle esigenze e delle aspettative.

- **Documentare i requisiti:** una buona documentazione dei requisiti è essenziale per assicurarsi che siano compresi da tutti i membri del team, così come utilizzare una terminologia chiara, per evitare ambiguità e fraintendimenti.

- **Comunicare in maniera aperta e costante:** il dialogo costante con gli stakeholder è importante per garantire che i requisiti siano allineati con le loro esigenze in continua evoluzione. Mantenere un canale di comunicazione aperto e incoraggiare il feedback regolare può contribuire a individuare eventuali cambiamenti o aggiornamenti necessari.

- **Utilizzare un approccio iterativo:** i requisiti possono evolvere nel corso del progetto; quindi, è consigliabile adottare un approccio iterativo per consentire di rifinire e migliorare i requisiti man mano che si acquisisce una maggiore comprensione del progetto e una migliore consapevolezza delle esigenze degli utenti.

- **Verificare e convalidare i requisiti:** Verificare che i requisiti siano completi, coerenti e non ambigui.

Validare inoltre i requisiti con gli stakeholder è un passo fondamentale per garantire che questi rispondano alle loro aspettative e necessità, ma anche per definire esattamente il perimetro dello sviluppo consentendo stime e una pianificazione esatte.

### Conclusioni

*In conclusione, i requisiti sono la base fondamentale per il successo di qualsiasi progetto, in particolare nel contesto software. L'analisi dei requisiti svolge un ruolo cruciale nel garantire che i requisiti siano chiari, completi e ben compresi da tutte le parti coinvolte. Per portare a termine efficacemente un progetto è fondamentale assicurarsi di coinvolgere tutti gli interlocutori e mantenere una comunicazione aperta e chiara durante tutto il processo.*

*Il consiglio finale è quello di considerare l'analisi dei requisiti come una fase critica del progetto e di dedicarle il tempo e le risorse necessarie. Coinvolgere gli utenti finali e gli stakeholder fin dall'inizio, utilizzare tecniche di analisi dei requisiti comprovate e adottare un approccio iterativo per rifinire e migliorare i requisiti nel corso del progetto possono fare la differenza tra un progetto di successo e uno che fallisce. Investire nella definizione accurata dei requisiti è un investimento che ripagherà ampiamente nel corso del progetto e oltre.*

---

**Sintegra** nasce a Milano nel 2010 sfruttando una miscela unica di imprenditorialità, processi industriali, relazioni con Università e partnership commerciali con leader di mercato per sviluppare prodotti e tecnologie innovative nell'ambito della trasformazione digitale veicolate sul mercato tramite imprese e team dedicati.

Negli anni successivi la forte specializzazione richiesta dai clienti ha portato Sintegra a diventare «Enterprises Factory», un player di riferimento per tutto ciò che concerne l'integrazione ed i servizi informatici in ambito, principalmente, finance.

Divenuta ormai Gruppo nel 2019, Sintegra offre servizi ad alto valore aggiunto in ambiti eterogenei, al di là del mero sviluppo software: trasformazione dell'informazione, CRM, AI/ML, Compliance, AML, Risk Management, LOM, crediti deteriorati, automazione, dematerializzazione.

---



SINTEGRA Group

MILANO: C.so Sempione, 62    RENDE (CS): Via Pedro Alvares Cabral, 6  
Numero Verde 800 689874

# IN QUESTO NUMERO

Editori & Autori



Tiziano Interlandi



Pierpaolo Cimirro



Davide Casari



Autori



Marco Pecoraro



Francesco Nardoni



Susanna Ferrario



Rosa Linda Romano



Marco Arzilli



Giorgio Irtino



Massimo Terzo



Luca Cianci



Gianluca Colonnese



Marco Calzolari



 sessionize.com



SINTEGRA

YEARS ANNIVERSARY 10



**Numeri**



**Articoli**



**Autori**

---

Vuoi contribuire anche tu come autore?

---

Se vuoi contribuire come autore leggi attentamente il disclaimer su  
<https://agileitalia.agileforitaly.com>

 [sessionize.com](https://sessionize.com)



<https://sessionize.com/call-for-agile-for-italy/>

mail



[agileitaliamagazine@gmail.com](mailto:agileitaliamagazine@gmail.com)





<https://www.linkedin.com/showcase/agileitalia>

## Una questione di dialogo



AgileForItaly  
Founders

**A**lla fine è sempre un problema di come comunichiamo: visione, obiettivi, specifiche tutti concetti filtrati dalla nostra forma mentis, preconetti, pareri sociali ecc. Spesso si parlano vere e proprie lingue diverse (grazie Pierluigi Pugliese per gli splendidi interventi su questo tema). Il mondo del business ha una sua lingua, il mondo dello sviluppo un'altra ben diversa. La formalizzazione è quella pratica che in qualche modo cerca di tradurre nelle due direzioni i concetti. Siamo ancora qui a parlare di questi temi quindi qualcosa non funziona ancora. Usiamo molte pratiche e spesso pratiche per agire su pratiche che non riusciamo a far funzionare. Il problema non è quindi fare una buona User Story perché ce lo ha detto il coach. Il succo del discorso è agire sulle basi della comunicazione per avere informazioni, comprenderle e condividerle verso una direzione unica che coinvolga tutti i fruitori del valore generato. A volte è come se Agile sia il fine e non il mezzo, allontanandoci dal senso pratico alla sua base. Quindi meglio parlar meno di come si scrive una storia e verificare se stiamo facendo la cosa giusta. I Team navigati e che hanno compreso il mindset agile sono come quelle coppie o amici a cui basta uno sguardo per capirsi. Dall'altra parte abbiamo Sandra e Raimondo che

passano le giornate a battibeccare per poi a fine giornata dire che è sempre la stessa minestra. Esiste un modo? No, alcuni team ce la fanno altri no. Certo la stabilità è importante. Se un team o i referenti di business cambiano ogni settimana è difficile trovare calma per poter affrontare questi temi. Se sei costretto a cambiare casa ogni settimana difficilmente ti interesserà approfondire il colore del battiscopa. Quindi pratiche sì, pratiche va bene, ma anche pratiche attenzione.

TUTTE LE SETTIMANE

# AGILE FOR ITALY LEAN BEER

## PODCAST



SCAN ME

SU SPOTIFY, SPREAKER, APPLE PODCAST, GOOGLE PODCAST



DAVIDE CASARI

TIZIANO INTERLANDI

PIERPAOLO CIMIRRO





# ***Diario di una change agent***

*Il nuovo libro di Erika Melis*



Quali sono i segreti di una trasformazione snella in azienda? Che ruolo hanno le persone in questo percorso? La metodologia Lean è applicabile nel mondo dei servizi?

Questo diario, scritto dall'intraprendente protagonista Bianca Martini, porta il lettore dritto nel cuore della trasformazione «snella».

*compralo su Amazon*

# L'industria come bottega creativa



**Claudio Saurin**  
**LA FABBRICA AGILE**  
**Lo sviluppo di prodotto**  
**nella quarta rivoluzione industriale**  
pagine 256 - formato 14,5 x 22,5 cm  
€ 28,00 - ISBN 979-12-80359-18-6

La Fabbrica Agile rappresenta una modalità di sviluppo dei prodotti fisici basata sull'Agilità che integra Lean, Waterfall e altre pratiche, eliminando le barriere fra uffici e officina per renderli un sistema artigianale elastico, creativo e umano in grado di trasformare velocemente le idee in prodotti di valore e di successo.

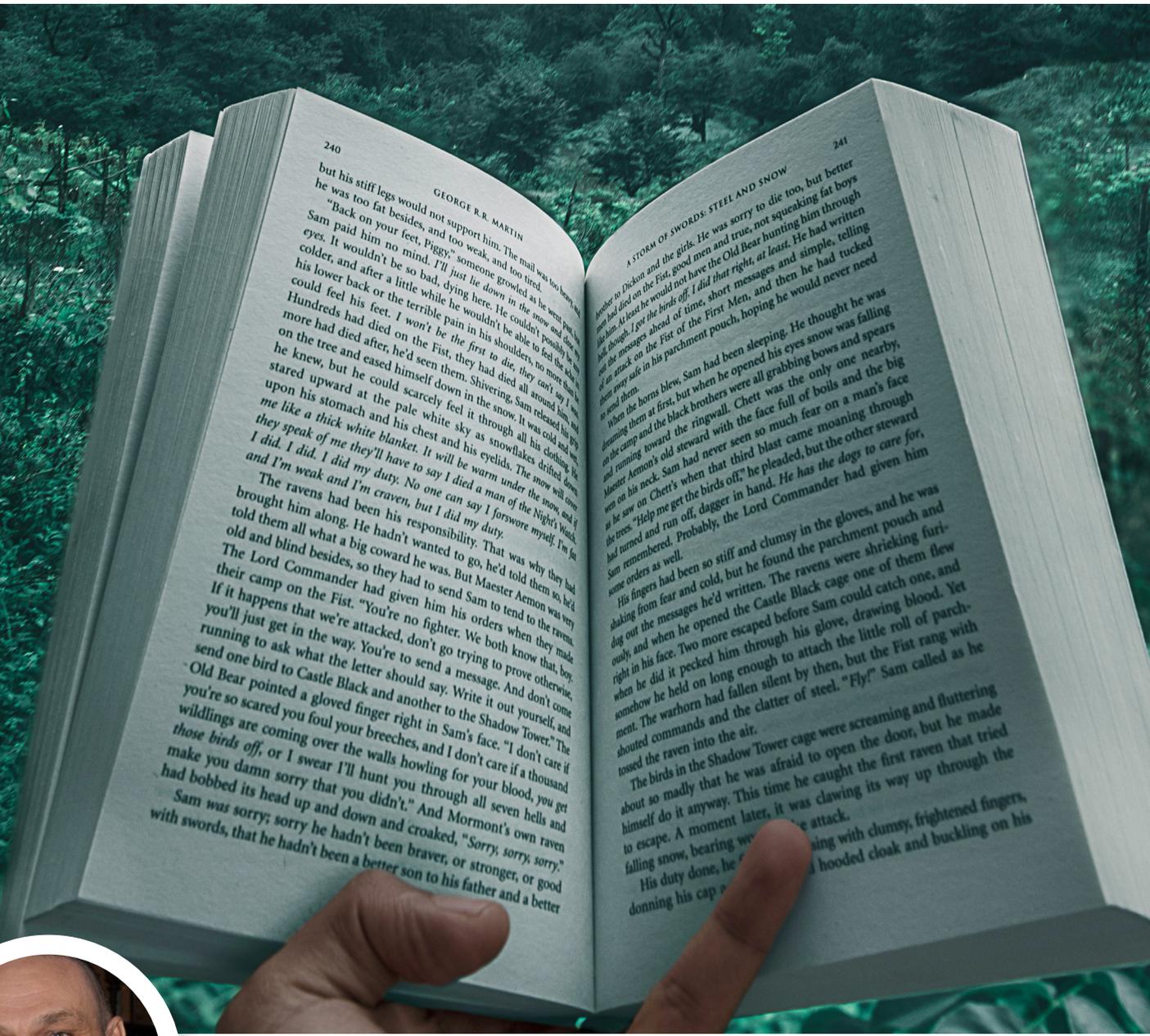
Opportunamente adattato allo sviluppo di prodotti fisici, l'Agile consente finalmente di dare una risposta organica alla complessità delle variabili in gioco – dai bisogni del cliente fino alla costruibilità e all'impatto ambientale – mettendo al centro le persone.



**AYROS**

[www.ayroreditore.it](http://www.ayroreditore.it)

# Raccontami una Story



GIORGIO IRTINO

Esperto di metodologie e tecnologie  
per la sostenibilità

A chi è genitore sarà sicuramente capitato di dover raccontare ai propri figli la classica storia che comincia con “c’era una volta” e finisce (quasi sempre) con “e vissero felici e contenti” ! Storie che bene o male hanno un modello narrativo definito e vengono utilizzate per trasmettere un messaggio preciso, un insegnamento, una morale. Ma in un ambito strettamente professionale chi si sognerebbe mai di chiedere al cliente di raccontare una storia ?

Eppure uno dei punti fondamentali dell’approccio Agile si basa proprio su questa pratica: rilevare i requisiti del cliente traducendoli in storie che alimentano il backlog da cui prendono origine le funzionalità da rilasciare. Può sembrare un esercizio infantile e spesso, forse proprio per questo motivo, viene sottovalutato con il risultato di continuare a definire i requisiti in modalità tradizionale, pensando poi di riuscire a gestirli in un contesto Agile con cui non sono coerenti !

Una storia in stile Agile suona così:

**Come <<utente>> voglio <<obiettivo>> in modo che <<beneficio>>.**

Chi si è sforzato di utilizzarla, certamente con una certa difficoltà iniziale, sa però che la sintassi delle user-stories una volta metabolizzata diventa estremamente efficace per semplificare il requisito e per esprimerlo in modo sintetico e chiaro al fine di consentire al team di sviluppo di tradurlo velocemente in una funzione rilasciabile che soddisfa le aspettative del cliente. In fin dei conti a tutti piace il lieto fine, soprattutto quando si tratta di gestire progetti che potrebbero trasformarsi in un incubo e costare lacrime e sangue.

Tuttavia non è solo una questione di chiarezza e semplicità: le user-stories facilitano anche l’interpretazio-

ne dei requisiti da punti di vista diversi e l’analisi dei rischi connessi. Mi spiego meglio con alcuni esempi, rimanendo sempre nella metafora familiare con cui ho aperto questo articolo. Un bambino piange perché la mamma non lo lascia andare a scuola con la maglietta di Batman.

Il requisito espresso dal bambino sarebbe il seguente:

**come <<bambino>> voglio <<mettermi la maglietta di Batman>> in modo che <<possa giocare ai supereroi con i miei compagni>>**

il requisito espresso dalla mamma invece sarebbe il seguente:

**come <<mamma>> voglio <<che mio figlio non metta la maglietta di Batman>> in modo che <<la possa lavare in quanto sporca>>**

Capita spesso che i requisiti espressi da utenti diversi possano essere contrastanti e prenderne atto è il presupposto per trovare una soluzione che accontenti tutti. Come questa:

**come <<mamma>> voglio <<che mio figlio vada a scuola con la maglietta pulita di Superman>> in modo che <<possa giocare ai supereroi con i suoi compagni>>**

Altro esempio: quale tra queste due storie è più critica ?

**come <<bambino>> non voglio <<fare i compiti>> in modo che <<possa andare a giocare con gli amici>>  
come <<bambino>> voglio <<attraversare la strada da solo>> in modo da <<dimostrare a tutti che sono grande>>**

Se il bambino non vuole fare i compiti, al massimo potrà succedere che prenda una nota dalla maestra. Ma se il bambino vuole attraversare la strada da solo, potrebbe finire sotto un'automobile ! Sarà meglio quindi concentrare prioritariamente l'attenzione sul secondo requisito.

Oltre alle storie, anche il backlog ha la sua importanza proprio perché ci deve dare la priorità di implementazione, calibrata sull'importanza/criticità del requisito, sul suo grado di definizione, sull'urgenza manifestata dal cliente. E qui viene fuori la vera bravura del product-owner, l'interfaccia che ha la responsabilità di costruire le storie, di stabilire un ordine di lavorazione e di affinare ed aggiornare costantemente il backlog interagendo con il cliente da un lato e con il team di sviluppo dall'altro, in base ai rispettivi feedback.

A seconda di come si approccia alla gestione dei requisiti, il product-owner può avere un profilo da anaconda oppure da colibrì: cosa significa ? L'anaconda è in grado di divorare una preda di notevoli dimensioni digerendola poi lentamente nell'arco di diversi giorni senza quindi aver ulteriore necessità di cibarsi; il colibrì invece magia pochissimo ma molto frequentemente, se non si ciba almeno ogni tre ore rischia la morte.

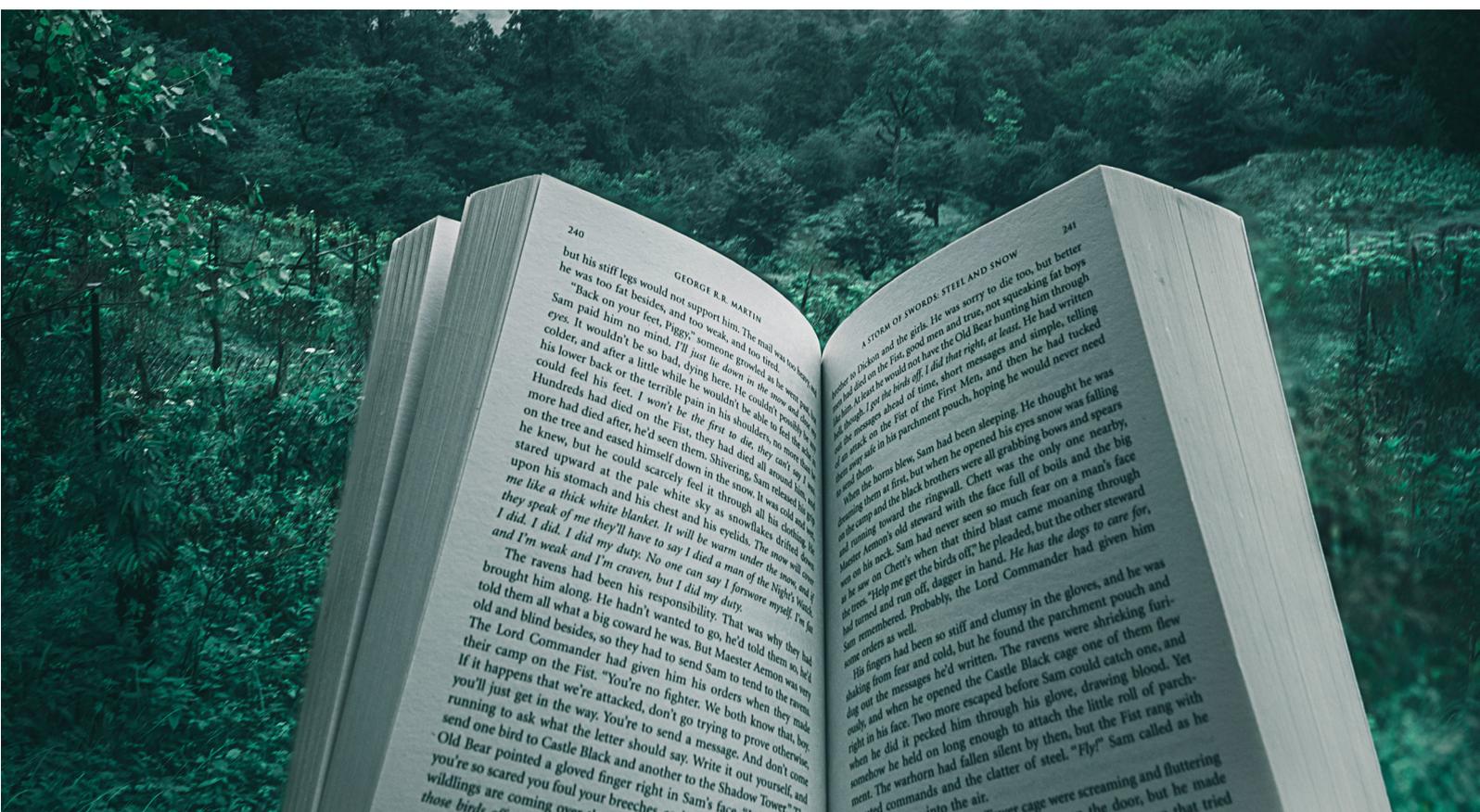
Dunque il product-owner anaconda è quello che tende a definire tutti i requisiti, a impilarli nel backlog per poi evaderli senza più andare a metterli in discussione introducendo modifiche o integrazioni (o almeno cercando di limitarle il più possibile). Il product-owner colibrì è invece quello che stratifica il backlog differenziando i macro requisiti (epics) dalle funzioni che compongono ciascun requisito (featu-

res) e dalle singole storie che suddividono la funzione in moduli autonomamente rilasciabili.

Così mentre il team di sviluppo lancia uno sprint, il product-owner colibrì lavora incessantemente alla trasformazione delle epics in features e delle features in stories. In questo modo il team di sviluppo non deve fare altro che prendere in carico le storie da implementare in ogni sprint, quantificarle per impegno (story-points) e scomporle ulteriormente in task elementari. Se il lavoro a monte è stato fatto bene, la strada è tutta in discesa (magari con qualche dosso qua e là) !

Ma tutto questo non è ancora sufficiente a garantire il lieto fine: manca ancora una componente indispensabile che è la DoD (Definition of Done), ovvero la formalizzazione dei criteri in base ai quali si possa deliberare che una funzione da rilasciare soddisfa pienamente le user-stories che l'anno originata. Allo stesso tempo, i criteri di accettazione aiutano il team di sviluppo a valutare la propria produttività e quindi a stabilire la quantità di storie che possono essere lavorate in uno sprint.

Un lavoro tanto certosino quanto strategico quello di definire le user-stories, il backlog e la DoD, considerato che sono tutte attività in cui non è ancora stata scritta neanche una linea di codice, ma l'esperienza insegna che, se fatte bene possono veramente consentire al team di sviluppo di rispettare con facilità tempi, costi e qualità producendo un risultato che soddisfa pienamente le aspettative del cliente. E sappiamo bene che questo è il lieto fine che tutti vorremmo !



# I requisiti: fraintesi, obsoleti e cambiati



FRANCESCO NARDONI

Agile Coach & Trainer | Guitar Player

Prima di iniziare a parlare di requisiti, vorrei raccontarvi una storia: siamo agli inizi degli anni 2000, c'è un'azienda di sviluppo software nella quale lavorava un team che implementava prodotti, prevalentemente per aziende del settore Telecomunicazioni. Il processo di produzione era quello classico, il cliente forniva un documento di requisiti di prodotto, spesso formulati tramite un lungo file word, venivano "spediti" all' EI-Team (il gruppo di lavoro si chiamava così), e, tramite il suo responsabile, dopo un'attenta analisi, produceva una pianificazione del lavoro, che, molto spesso, doveva rispettare delle scadenze già stabilite.

L' EI-Team era composto per la maggior parte di developers che, una volta terminato il processo di sviluppo, installavano l'applicativo in ambiente di collaudo per tutti i test di integrazione e di accettazione del cliente. Successivamente, veniva realizzato il "pacchetto" per l'installazione in produzione, e veniva inviato al cliente, che si occupava di rendere fruibile agli utenti il prodotto realizzato.

Sembra una storia molto tranquilla, quasi una favola, quindi aggiungo qualche dettaglio in più.

Dentro l'azienda cliente del settore Telecomunicazioni, c'era un'unità che si occupava di interpretare le esigenze dei loro utenti, che iniziava a stilare la lista dei requisiti di nuove funzionalità, o modifica di funzionalità esistenti. Successivamente questo documento veniva ulteriormente elaborato da un'altra unità aziendale (l'area Demand), per aggiungere dettagli e complessità alle richieste iniziali. Per finire, il documento veniva passato all'IT aziendale che, dopo aver elaborato i requisiti e verificata la fattibilità, si rivolgeva ad un fornitore esterno (l'EI-Team), per l'implementazione del software.

Come vedete, il documento di requisiti, che è il vero protagonista della nostra storia, faceva molti passaggi prima di arrivare alle persone che dovevano tradurre quel documento in software. Cosa succedeva quando l'EI-Team aveva un dubbio sul documento di requisiti? Si seguiva, in senso opposto, il percorso fatto dal documento: chiedevano all'IT del cliente, che chiedeva al Demand, che chiedeva all'unità che aveva inizialmente espresso le esigenze. Ogni volta, per ogni dubbio, per ogni richiesta, per ogni domanda, l'informazione percorreva questa strada tortuosa, a volte in maniera rapida, spesso in maniera molto lenta.

Al termine dell'implementazione del prodotto, le persone che avevano espresso l'esigenza spesso commentavano le nuove implementazioni dicendo: **"ma non era questo quello che intendevo"**, oppure **"ma io non lo volevo così"**, o spesso **"ma adesso il mio utente ha un'esigenza diversa"**. E quindi partivano controversie e rilavorazioni per correggere e migliorare il prodotto, con sprechi di tempo e di soldi. Molto spesso gli utenti (che nella maggior parte erano esterni), invece

di attendere un prodotto migliorato, si rivolgevano altrove.

Questa storia non ha un lieto fine, la frustrazione che vivevano giornalmente, ma soprattutto dopo la fine di un progetto, era talmente alta che la maggior parte dei componenti dell'EI-Team dopo qualche tempo cambiò lavoro, e l'azienda di sviluppo software, dopo qualche anno, fallì.

Starete pensando: "il processo di raccolta dei requisiti può far fallire un'azienda che produce software per altre aziende?" Sicuramente non è stato il principale fattore, ma ha contribuito nella realizzazione di prodotti a scarso valore.

## Requisiti ed esigenze

Torniamo ai nostri requisiti, leggendo la definizione su Wikipedia, troviamo questa definizione:

*"Nello sviluppo del prodotto, un requisito è una singola esigenza fisica o funzionale documentata che un particolare progetto, prodotto o processo mira a soddisfare"*

E' sicuramente importante soddisfare tutti i bisogni dei nostri utenti/clienti, nel più breve tempo possibile, e quindi è molto importante avere ben chiare le esigenze espresse che daranno vita ad un nuovo prodotto, o alla modifica di uno esistente.

Come trasformare una serie di esigenze, in una guida che possa supportare l'implementazione di un prodotto, è fondamentale per la creazione di prodotti di valore.

## Approccio collaborativo

La storia che ho raccontato, si riferisce ad un contesto lavorativo dei primi anni 2000, ma non è molto diversa da situazioni che ho visto in aziende piccole e grandi negli ultimi anni. Non è cambiato molto, il famoso documento di requisiti, viene creato e passato di mano in mano, con una distanza a volte troppo ampia, tra chi pensa ad un'esigenza, e chi è chiamato a trasformarla in prodotto.

Partendo da un'esigenza iniziale, i requisiti invece dovrebbero essere scritti in maniera collaborativa, non ci dovrebbe essere una netta separazione tra "chi pensa" e "chi fa" come se fossimo in catena di montaggio, per una serie di motivi che influiranno negativamente sulla realizzazione del prodotto:

- Frintendimenti
- Focus sul prodotto
- Mancata percezione dell'obiettivo del prodotto
- De-responsabilizzazione da parte di chi implementa il prodotto
- Basso valore del prodotto creato



Pertendo da un'esigenza, il documento di requisiti dovrebbe essere realizzato da chi ha intercettato l'esigenza degli utenti (di solito le aree di business in un'azienda), e dai team che si occuperanno di risolvere quell'esigenza implementando una soluzione.

Nell'approccio Agile, esiste il concetto dei Three Amigos: Business Analyst, Developers e Tester che analizzano insieme ed integrano i requisiti, mettendo a disposizione le loro differenti prospettive, superando le lacune tra interpretazioni e reali aspettative degli utenti. Le sessioni dei Three Amigos vengono effettuate di solito nella fase di Inception e durante le iterazioni di lavoro.

Prendendo spunto dai Three Amigos, un approccio collaborativo nella realizzazione dei requisiti deve tendere a ridurre i passaggi degli stessi, e dovrebbe riunire le persone che hanno identificato l'esigenza, con le persone che implementeranno la soluzione, per creare delle aspettative comuni sul prodotto.

Quali sono i vantaggi di questo approccio?

- Le esigenze sono chiare a tutti, senza fraintendimenti
- I requisiti ambigui e mancati, se presenti, vengono identificati in anticipo e convalidati con le parti interessate
- Eventuali malintesi, presupposti ed interpretazio-

ni errate, verranno eliminati prima dell'avvio di qualsiasi lavoro di sviluppo sui requisiti

### Scope variabile

Un altro aspetto da tenere in considerazione è che, a volte, i requisiti possono cambiare perché cambiano le esigenze degli utenti. Non sarebbe quindi produttivo occupare un lungo periodo di tempo a scrivere, dettagliare ed approfondire tutti i requisiti. Al termine di quel periodo, paradossalmente, le esigenze dei nostri utenti potrebbero essere cambiate. Ma soprattutto, non sarebbe neanche produttivo, iniziare ad implementare un prodotto basato su dei requisiti che potrebbero essere già vecchi. E allora, cosa facciamo?

Partiamo dalle esigenze dei nostri utenti, iniziamo, con l'approccio collaborativo descritto sopra a creare dei macro-requisiti, che dovranno esser messi in priorità. Si parte approfondendo i requisiti ad alta priorità, e si inizia con la loro implementazione in maniera iterativa ed incrementale. Allo stesso modo, oltre ad implementare il prodotto, si approfondiranno i requisiti rimanenti.

Non necessariamente tutti i requisiti ipotizzati ad inizio progetto dovranno esser realizzati, sicuramente saranno realizzati quelli che hanno maggior valore per l'utente finale.



## Feedback continui

Il terzo aspetto, anche questo fondamentale, è la raccolta dei feedback dagli utenti, ed in generale dagli stakeholders di prodotto. Un approccio iterativo ed incrementale, permette di avere tanti momenti di feedback nei quali mostrare l'incremento di prodotto realizzato.

In questi casi la dimostrazione potrebbe esser fatta anche agli utenti finali, che avranno la possibilità di utilizzare l'incremento prodotto (non sarà "completo" ma dovrà essere utile per l'utente).

In queste occasioni i feedback degli utenti e degli stakeholder saranno fondamentali per apportare modifiche ai requisiti (inserendone di nuovi o togliendo quelli non più utili), ed in generale saranno utili a capire se la strada intrapresa nello sviluppo del prodotto, è quella giusta.

## Conclusioni

In vent'anni non è cambiato molto, oggi, con grande stupore, vedo comportamenti che vedevo all'inizio degli anni 2000, che contribuiscono spesso alla creazione di prodotti che non utilizzerà nessuno, o prodotti che risolveranno pochi problemi.

Si tende spesso a considerare le implementazioni software come prodotti da costruire in una catena di montaggio, si tende ancora a formulare documenti di requisiti che verranno passati alla "factory" che spesso utilizza quel documento in maniera meccanica, implementando in maniera precisa (non sempre) quello che il documento richiede.

I silos aziendali di cui abbiamo parlato in un recente articolo di AgileItalia, influiscono negativamente anche su questo processo.

Fortunatamente l'approccio Agile, che stiamo contribuendo a diffondere, ci aiuta anche nella fase di "requisizione", facendola diventare una componente collaborativa che non è relegata solo nelle fasi iniziali del progetto, ma è "spalmata" lungo tutto il periodo di implementazione.

Il successo di un prodotto, la felicità di un gruppo di lavoro, le revenue aziendali, passano anche attraverso la creazione e la gestione sensata di un documento di requisiti, o, per meglio dire, di un Product Backlog.

# PRODUCT MANAGEMENT DAY 2023

27 ottobre 2023

Nana Bianca, Firenze

La terza edizione della più grande conferenza italiana sul Product Management

2 Keynote Speaker + 15 speaker / 2 track



Workshop

**Evidence-Guided Product Management Using the GIST Framework**

con Itamar Gilad

Product management coach,  
speaker and author, Ex-Google PM

per la prima volta in Italia

Workshop

Tavole rotonde

Networking

Uno spazio di incontro e confronto tra professionisti e aziende. Un evento per approfondire i grandi temi legati al prodotto e ascoltare i consigli e le esperienze degli speaker.

[productmanagementday.com](https://productmanagementday.com)



Prendi posto alla conferenza italiana interamente dedicata al prodotto.

Stacca il tuo biglietto

# THE FUTURE OF MANAGEMENT

MIT Sloan Management Review / Guerini Next



## CROMOSOMA INNOVAZIONE e DOPPIA ACCELERAZIONE

I PRIMI DUE VOLUMI DELLA COLLANA MIT/GUERINI NEXT  
A CURA DI ALBERTO MATTIELLO

Strategie e visioni dal MIT per capire come  
la tecnologia riscrive la genetica delle organizzazioni

gueriniNEXT



MARCO  
PECORARO

Scrum Master in Maire

# NAVIGARE LA TRASFORMAZIONE: I REQUISITI “BUSSOLA” PER REALIZZARLA

Il 70% dei progetti di innovazione e trasformazione si è arenato o non ha raggiunto tutti gli obiettivi che si era prefissato, come certificato da numerosi centri di ricerca in tutto il mondo.

È un dato innegabile.

Tuttavia, **un'azienda che aspiri ad essere generatrice di innovazione e di valore attraverso il cambiamento non può accettare che solo il 30% delle sue iniziative abbia esito positivo.**

È evidente per chiunque operi in quest'ambito che il successo o il fallimento di un progetto spesso dipende dalla definizione iniziale degli obiettivi stessi. In questo articolo, esploreremo come l'approccio AGILE ed i suoi principi possano essere applicati non solo nella fase esecutiva di progetto ma anche in quella di definizione dei requisiti (la cosiddetta "Product Discovery").

L'approccio tradizionale a cascata, noto come "waterfall", prevede la definizione di obiettivi rigidi e ben dettagliati. Tuttavia questo metodo comporta il rischio, soprattutto in contesti di incertezza e mutevoli come quelli che caratterizzano questi tempi, di non considerare le nuove sfide che potrebbero emergere durante la fase di esecuzione del progetto, né le opportunità che le nuove tecnologie offrono. Di conseguenza, spesso capita di non andare nella direzione che massimizza il valore per l'intera organizzazione. Punto decisivo è proprio la corretta definizione e comprensione degli obiettivi di progetto.

Troppo spesso, questa fase prende il via con una decisione presa dal management che, in qualità di sponsor dell'iniziativa, assegna ad un proprio collaboratore il compito di definire i requisiti in autonomia. Questa risorsa, sebbene intitolata a svolgere l'attività, non è sempre esperta di tecnologia o di mappatura di processi aziendali e, con grande fatica, rischia di definire i requisiti entrando in un dettaglio spesso non comprensibile da chi non è un addetto ai lavori come lui e senza cogliere tutte le opportunità e gli impatti del cambiamento che sta descrivendo.

***E se ci fosse un'alternativa? E se iniziassimo a trattare anche la definizione dei requisiti come obiettivo di uno specifico progetto che coinvolge tutti gli attori coinvolti dal processo di trasformazione e da chi dovrà pensare come realizzarlo (es. coordinatori sviluppi digitali)?***

L'approccio Agile offre una modalità di lavoro più flessibile e adattabile alla definizione degli obiettivi proprio per i quattro valori fondamentali che la caratterizzano:

- Individui e interazioni più che processi e stru-

menti

- Software funzionante più che documentazione esaustiva
- Collaborazione con il cliente più che negoziazione dei contratti
- Rispondere al cambiamento più che seguire un piano prestabilito

Questi valori possono essere applicati non solo nella fase di esecuzione di un progetto ma anche nella sua fase di definizione.

Come farlo?

Di seguito parte di un framework che stiamo applicando in MAIRE, gruppo che opera a livello globale con soluzioni tecnologiche sostenibili e servizi integrati di ingegneria e costruzione per l'industria di trasformazione delle risorse naturali e per la transizione energetica:

1. Intervista allo sponsor di progetto per comprendere le figure aziendali impattate dall'iniziativa di trasformazione e le ragioni che lo hanno spinto a proporla.
2. Identificazione dei "Subject Matter Expert", ossia dell'insieme di risorse chiave con tutte le competenze necessarie ad affrontare il processo di trasformazione, sia perché attori della sua realizzazione, sia perché figure di business che di quel processo fanno parte.
3. Intervista a tutti gli attori identificati in modo da raccogliere le esigenze da ciascun punto di vista, chiedendo evidenza sia dei punti di difficoltà riscontrati nelle attività di tutti i giorni, sia delle proprie idee di miglioramento. Non è opportuno andare in un estremo dettaglio mentre è decisivo per una efficace riuscita di questa fase che tutti gli attori coinvolti si sentano "liberi" di esprimere le idee che hanno in base alla propria percezione del pezzo di processo che si sta andando ad aggredire, senza essere forzati a prendere una posizione che non gli appartiene. Compito fondamentale dello Scrum Master o dell'Agile Coach è fare in modo che queste condizioni si creino.
4. Clusterizzazione dei punti emersi in "desiderata" che costituiranno una prima bozza di product backlog da rivedere con lo sponsor dell'iniziativa.
5. Review della bozza di product backlog con lo sponsor ed eventuale pianificazione di incontri di chiarimento per meglio comprendere i punti emersi con chi li ha espressi.
6. Revisione con tutto il team coinvolto dei "desiderata" di progetto che, una volta concordati, si tramuteranno nel vero e proprio product backlog da soddisfare in fase esecutiva.



Il vantaggio del continuare a lavorare in questo modo anche successivamente durante la fase esecutiva è che, proprio perché nei progetti di trasformazione le condizioni possono variare rapidamente, potrebbe essere necessario correggere la direzione verso cui si sta andando. E, di conseguenza, **anche gli obiettivi potrebbero cambiare.**

Agile, per propria natura, è il framework ideale per garantire al team di adattare e correggere in modo trasparente questi obiettivi condivisi anche durante la fase esecutiva, per rispondere meglio alle sfide emergenti e massimizzare i risultati, cioè il valore complessivo per tutti gli stakeholder impattati dall'iniziativa.

In sintesi, la fase di definizione dei requisiti di un progetto di trasformazione è un progetto.

Alistair Cockburn, uno dei firmatari del manifesto Agile, la definisce *“come una mappa che guida il percorso del progetto di trasformazione, fornendo una visione chiara delle mete da raggiungere”*.

Trattarla con questa dignità operativa di “bussola” che guidi i nostri processi di trasformazione è la chiave per ridurre poi il rischio di non raggiungere i propri obiettivi durante la fase esecutiva.

# Piovono requisiti da ogni parte? Cerca la radice del problema.



## SUSANNA FERRARIO

Product Coach & Senior Product Manager Consultant | Owner @Product Studio

Ho iniziato a progettare prodotti digitali nel 2000 in una delle allora pochissime Internet company italiane. La passione per il prodotto non è mai venuta meno.

Nel 2011 ho scoperto le metodologie Agili e non le ho più abbandonate. Amore a prima vista.

Nella mia carriera ho ricoperto diversi ruoli - Product Owner, Senior Product Manager, Head of Product - all'interno di aziende B2C e B2B in Italia e all'estero.

Oggi mi dedico alla formazione e al coaching in ambito product management e porto avanti collaborazioni con start-up e scale-up aiutandole a validare idee di business e creare prodotti che i clienti amano.



## Non dirmi cosa, spiegami il perché

“Start with why”<sup>1</sup> di Simon Sinek è uno dei miei libri preferiti. Ho trovato una miniera di spunti al suo interno, non solo per la vita professionale ma anche per quella personale.

Nel suo libro d’esordio del 2011 l’autore americano introduce il concetto di Golden Circle: le aziende per attrarre il loro pubblico devono partire comunicando il proprio perché, i propri valori. Le leve strategiche utilizzate per perseguire la propria visione, la descrizione dei prodotti e dei servizi offerti sono aspetti secondari per fidelizzare la propria clientela.

Ecco, riprendo questo concetto proprio per la gestione dei requisiti: anche qui è necessario **partire dal perché**.

Se non sappiamo a quale obiettivo di business stiamo puntando, se non sappiamo esattamente quale impatto vogliamo creare nell’utente finale parlare di requisiti di prodotto è l’ennesimo modo per non centrare la questione e finire per creare “waste”.

Lo scrive una che per molti anni all’inizio della propria carriera nel 2000 ha costruito prodotti digitali e tante funzionalità in questo modo distorto.

Oggi è sotto gli occhi di tutti: siamo sommersi da funzionalità non richieste, prodotti che fanno il loro dovere a metà (meh!), servizi scadenti e talvolta da uno sconcertante risparmio sulla qualità. Non contribuiamo oltre a questo spreco di risorse!

Capire il vero motivo per cui ci viene richiesta una funzionalità è ciò che fa di una persona di prodotto un vero Product Manager o Product Owner.

Una volta che quell’aspetto è chiaro tutto ciò che ne consegue verrà - quasi - da sé.

Una nota però: non basta chiedere il perché una sola volta, a volte non bastano neanche i famosi 5 whys... come spiega Michael Cooper<sup>2</sup> - executive coach e

consulente di management per svariate Fortune 1000 - potrebbero volercene molti di più di perché per arrivare al vero nocciolo della questione. Mettete da parte il problema di superficie e scavate; saprete di esserci arrivati quando nel flusso delle risposte qualcosa vi risuonerà più a fondo. Quella risposta emotiva è il segnale che siete giunti alla radice del problema.

## Fare framing ovvero inquadrare il problema

Sappiamo di avere capito il contesto ed il perché nel momento in cui siamo in grado di spiegare il problema in maniera chiara e lineare... (hai detto niente!). Per farlo dobbiamo avere trascorso nello spazio del problema abbastanza tempo, aver vagliato diverse possibilità ed essere arrivati ad un framing dello stesso. Siamo riusciti ad incorniciarlo.

Il contesto e la cornice di riferimento indirizzano il modo in cui processiamo gli stimoli. E’ questo il framing o effetto framing, quel fenomeno in base al quale lo stesso contenuto - presentato con una differente prospettiva - viene interpretato in maniera diversa dall’interlocutore.

Nel caso della presentazione di un problema le soluzioni che andremo a cercare sono condizionate dall’effetto framing; cambiano a seconda di come il contenuto viene presentato.

Se cambi la cornice della tua prospettiva, cambia tutto.

Se cambi la cornice della tua prospettiva, arrivi a soluzioni profondamente diverse.

Allo stesso modo un problema ben inquadrato è più facile da risolvere.

La dichiarazione del problema ci aiuta a stabilire dei confini, a capire di cosa dobbiamo occuparci e che cosa invece è out of scope.

<sup>1</sup> <https://simonsinek.com/books/start-with-why/>

<sup>2</sup> <https://coachingforleaders.com/podcast/160/>

## Problema per chi?

Una parte essenziale nel framing del problema è individuare la tipologia di utenti che sperimenta il pain point, capire come sono impattati e perché per loro è importante risolverlo.

Quando cominciamo a fare queste domande a chi arriva da noi con un requisito la qualità della conversazione cambia immediatamente.

Il soggetto di cui stiamo parlando è uno stakeholder interno o un utente? Di chi parliamo? Di quante persone parliamo? Siamo sicuri che si tratti di un vero problema? Che evidenze abbiamo a riguardo?

Non si tratta di “mettersi di mezzo come spesso fa la funzione di prodotto”, bensì di rendere esplicito l’impatto di ciò che ipotizziamo di realizzare.

Ci troviamo continuamente come PM a fare scelte rispetto a cosa dire sì e a cosa dire no.

Chi sperimenta il problema è parte integrante della definizione dell’impatto. Stiamo fornendo una soluzione alla maggioranza dei nostri utenti, ad un segmento, ad una nicchia o a una singola persona? La domanda non è peregrina.

Essere consapevoli di questa dimensione ci aiuta a mettere le richieste in prospettiva ed è un primo passo per prioritizzare i problemi da risolvere.

## Requisiti a caccia di problemi

Perché è così importante decodificare le richieste? Semplice! La maggior parte delle richieste che ci arrivano sono soluzioni, non problemi.

Sono risposte preconfezionate, desiderata, problemi malformati che contengono una marea di assunzioni. Se ci fissiamo su quei particolari invece di investigare e comprendere il quadro generale abbiamo la ricetta per un disastro annunciato o - per lo meno - per l’ennesima funzionalità che non sposta un numero e men che meno un comportamento dell’utente.

Noi siamo bravissimi a trovare soluzioni, un po’ meno a comprendere i problemi...

Siamo “by design” progettati per buttarci quanto prima nello spazio della soluzione, a trovare il prima possibile una via d’uscita e a non pensarci più.

La parte di definizione del problema è ambigua, tutt’altro che confortevole, spesso incasinata (“messy” come la definisce Teresa Torres). Eppure è solo attraversando questa fase di incertezza e apprendimento che potremo arrivare ad esprimere la migliore soluzione in un prodotto finito.

Pensate è talmente comune ricevere richieste di cui non è chiaro il problema che Stephen Anderson - relatore, educatore e design leader oggi in Mural - si è divertito a classificare le cosiddette “bad request” ovvero tutte le richieste malformate.

Qualche esempio? Riflettete su un problema recente che vi hanno chiesto di risolvere e confrontatelo con questi esempi. Vediamo quanto questo approccio è comune...

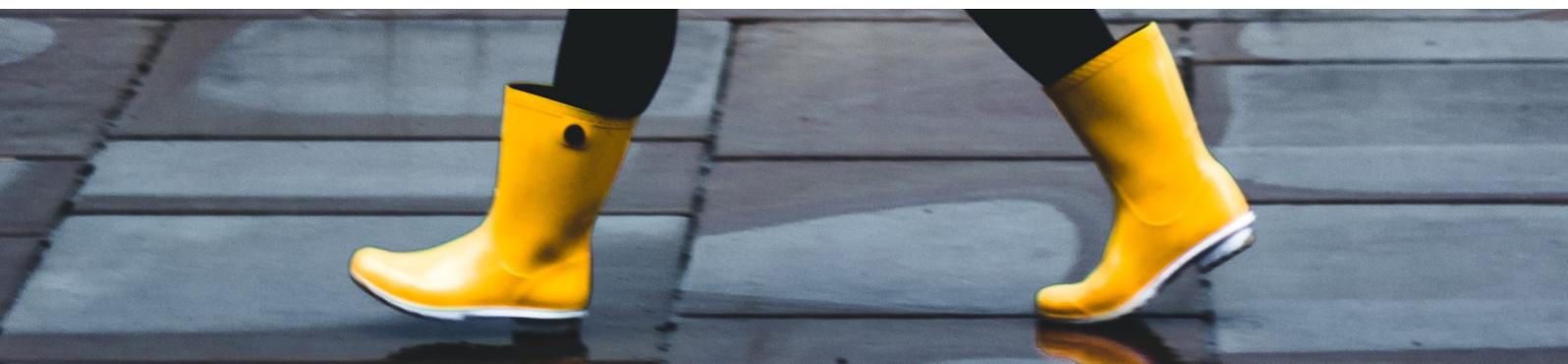
“Ci serve un CRM per gestire i nostri clienti professionali”. A che scopo? Cosa voglio ottenere che facciano i miei clienti professionali? Tecnicamente si parla di “Anchoring” quando il problema viene inquadrato nel contesto di una singola soluzione specifica che scarta immediatamente tutte le altre alternative.

Oppure il problema viene descritto in termini di acquisto di una particolare tecnologia quando la questione potrebbe essere tutt’altro che tecnica: “abbiamo bisogno di un’installazione Sharepoint”. Per chi? Perché? Si parla di “solutioneering” in questo caso. Potreste anche imbattervi in una richiesta “mash-up” altrimenti detta di tutto un po’... “Vogliamo creare un nuovo servizio che sia YouTube + Vinted + Discord messi insieme” (apperò!).

Potrebbero chiedervi di trattare un sintomo invece di andare alla radice del problema - “mettiamo un chatbot per far fronte alle richieste degli utenti” - o trovarvi di fronte al grande classico “l’ha chiesto il CEO”... che con mia grande sorpresa viene annoverata in una categoria a sé stante, l’ego-riferita ego-stroking ovvero un problema inesistente importante solo per l’HIP-PO di turno.

Insomma, avete capito... chi più ne ha più ne metta e questa lista evolve nel tempo. Se siete curiosi di questo progetto open source vi lascio nelle note i riferimenti<sup>3</sup>; siamo “solo” a 25 categorie di cattive richieste quindi state all’occhio!

<sup>3</sup> <http://bit.ly/badproblems>



## Ritorno dal futuro

Un'altra possibilità per carpire informazioni preziose dal nostro interlocutore carico di requisiti è farlo ragionare sulla situazione finale: ipotizziamo di avere già realizzato completamente la feature che ci viene richiesta... in questo scenario come cambia il comportamento del nostro utente? Che cosa fa diversamente? Ha smesso di fare qualcosa? Ha iniziato a fare qualcosa di nuovo? Cosa riesce ad ottenere che oggi non ha? E - non dimentichiamo - qual è il vantaggio per l'azienda? E' scalabile?

Proiettarci nello scenario futuro ci consente di esplorare insieme ai richiedenti le caratteristiche che dovrebbe avere un rilascio di successo. A quel punto proviamo a capire come misurare questo esito positivo, quali KPI si muoveranno e in che modo (parlare di metriche di successo ci offre un po' la prova del 9 del ragionamento che va dagli output all'outcome).

Infine andiamo a ritroso: sappiamo dove dobbiamo arrivare; ora proviamo a capire il percorso di realizzazione. Cosa deve accadere affinché quello scenario si realizzi? Quali sono gli step che devono essere compiuti prima? Quali dopo? C'è una o più sequenze corrette?

Sono tutte domande che ci aiuteranno a far emergere il sommerso, il non detto, le aspettative. E se il nostro interlocutore non è in grado di rispondere abbiamo due possibilità: rimandare la conversazione al momento in cui avrà raccolto queste informazioni o accompagnarlo nel percorso di scoperta assicurandoci che sia il vero stakeholder (non un proxy!) dotato di potere decisionale.

## Non innamoriamoci perdutamente delle nostre idee

Bene, ho il problema e la situazione finale a cui voglio arrivare; non mi resta che trovare la soluzione. Tutto il percorso che ho compiuto fin qui mi consente ora di vagliare diverse idee.

Non esiste "la soluzione perfetta", neanche "la soluzione unica". Esistono tante alternative che possono essere realizzate - siano essi prodotti o funzionalità - per raggiungere il risultato. Ecco perché non ci interessa il requisito in sé ma il motivo che lo ha scatenato.

Facciamo allora brainstorming con il team, proviamo a capire quali iniziative consentono di avvicinarsi all'outcome magari con il minor effort possibile. Come dico sempre alle persone a cui mi capita di fare formazione e coaching dobbiamo tenere a mente che le soluzioni che proponiamo - i requisiti che ci sono

stati portati inizialmente - sono "solo" ipotesi, non certezze.

Non sappiamo a priori se funzioneranno; magari sposteranno le metriche che abbiamo individuato come chiave ma nella maggior parte dei casi questo non accadrà (ed è ahimè normale che sia così).

Dobbiamo validare queste idee, provare a testarle, raccogliere i feedback e capire se è necessario fare modifiche radicali alle ipotesi iniziali.

Non innamoriamoci troppo presto di requisiti e soluzioni; ci troveremo spesso nella condizione di doverle lasciare andare e riprendere la via verso nuove strade più promettenti.

## Smontare e rimontare

Il percorso lungo il quale accompagniamo i nostri interlocutori - coloro che sono arrivati da noi con la richiesta iniziale - è di decostruzione e ricostruzione. Dal particolare del requisito ragioniamo insieme per individuare il cuore del problema; una volta inquadrato il problema abbiamo la possibilità di spostarci nello spazio della soluzione con più chiarezza e contesto.

Facciamo brainstorming per generare il maggior numero di soluzioni possibili; ci interessa la quantità più che la qualità all'inizio e la possibilità di sfruttare le dinamiche di gruppo per costruire sulle idee degli altri. In una seconda fase andremo a filtrare quali sono le proposte più promettenti attraverso un qualche strumento di prioritizzazione (ad esempio mediante la matrice value / effort o tecniche come ICE o Moscow).

E' un percorso di "educazione" all'approccio di prodotto nel quale coinvolgiamo i nostri stakeholder. Ci saranno delle inevitabili frizioni all'inizio ma se abbiamo lavorato bene insieme e li abbiamo portati a bordo nelle varie fasi del ragionamento finiranno per percepirne il valore molto presto. Dal requisito, al problema, alla soluzione. Buon viaggio di scoperta!

*"Non risolvere il problema che ti viene chiesto. È quasi sempre il problema sbagliato.*

*Quasi sempre, quando qualcuno viene da te con un problema, ti sta davvero dicendo i sintomi e la prima e più difficile parte del design è capire cosa è veramente necessario per arrivare alla radice del problema e risolvere il problema corretto."*

*Don Norman, co-fondatore e direttore di Nielsen Norman Group*

# Le 3 R dei Requisiti



ROSA LINDA ROMANO

System Analyst presso EUforLEGAL,  
Educatrice e Facilitatrice di gruppo,  
Scrum Product Owner Certified



MARCO ARZILLI

Function Point Advisor & Agile  
Project Manager (IFPUG CFPS - PMI  
PMP - Scrum PSPO & PSM)

**Il termine “Requisiti” contiene una “R”, ma sono ben 3 le “R” di primaria importanza: Raccontare, Riscrivere, Relazionarsi. Ho voluto incontrare Marco Arzilli, Senior Management Consultant ed esperto di metodologie Agili, per commentare questi aspetti e provare a scoprire i segreti per una buona gestione dei requisiti.**

**Marco, partiamo dall’inizio. Se dico “Requisiti”, a te cosa viene in mente?**

Innanzitutto, la relazione con il cliente/utente e l’importanza di coinvolgere tutte le persone che possono esprimere requisiti e che hanno interesse al progetto. Mi viene in mente che bisogna scegliere una modalità da utilizzare per rappresentare i requisiti e per validarne la corretta rappresentazione. Infine, mi viene in mente che i requisiti sono una cosa piuttosto fluida, e normalmente è un’utopia immaginare che siano sempre ben strutturati, definiti e stabili nel corso del progetto.

Ho imparato un’altra cosa: avere la percezione che rappresentare bene i requisiti in una certa fase di un progetto dia garanzia che quei requisiti rimarranno immutati nel tempo è una pia illusione, quindi bisogna farci i conti e capire se c’è un modo per affrontare questa potenziale criticità. Dico potenziale, perché dipende da come ognuno organizza il lavoro.

**Tu come lo organizzi, di consueto, il lavoro?**

Il lavoro si può organizzare in diversi modi. Il classico dei modi per organizzare un progetto di sviluppo, o, in genere, qualunque progetto, è l’approccio waterfall, che prevede di svolgere le fasi in sequenza: dedicare tempo ed energie a raccogliere tutti i requisiti degli utenti, formalizzarli e stabilizzarli, e poi, solo dopo aver finito questa fase, avviare la fase di analisi di dettaglio coinvolgendo il gruppo di progetto e di sviluppo, solo successivamente iniziare la progettazione, al termine della quale avviare la realizzazione del prodotto, il test e così via.

Questo processo deriva dai processi industriali, ingegneristici, e funziona bene quando siamo in contesti molto strutturati, ripetitivi, standardizzati. Se stiamo realizzando per l’ennesima volta un edificio in un condominio, questo approccio è perfetto: si parte dalla progettazione della pianta dell’edificio, e poi si svolgono tutte le fasi in sequenza. Nel mondo del software si è compreso che il contesto non è mai

così ben chiaro e predefinito, sappiamo bene quale sia il maggior problema derivante dall’approccio waterfall: i requisiti, che inizialmente sembrano chiari e stabilizzati, e su cui si basa tutta l’impalcatura del progetto, ad un certo punto risultano non soddisfatti, perché si scopre che dell’utente non si sono comprese fino in fondo tutte le esigenze. Ma, con l’approccio waterfall, la scoperta arriva quando è ormai troppo tardi, perché l’utente, “messo da parte” dopo aver raccolto i requisiti, non viene quasi mai coinvolto a validare in corso d’opera quanto realizzato ma viene nuovamente chiamato in causa solo al termine di tutto lo sviluppo, e questo molto spesso è causa di spiacevoli sorprese...

Per fortuna, sappiamo anche che, per cercare di ridurre al minimo questo rischio, esistono altri approcci, e il più diffuso negli ultimi anni è quello Agile, da cui sono derivate anche altre forme di discipline “ibride” che si ispirano ai principi e ai valori del Manifesto Agile. Questi approcci non solo suggeriscono una modalità diversa per trattare e rappresentare i requisiti, riducendo il rischio che abbiamo appena descritto, ma inducono anche a ripensare il modo di fare sviluppo software e non solo. Infatti, dal mondo software da cui è nato, l’approccio agile si sta diffondendo in altri settori, ormai si parla di marketing agile, HR agile, proprio perché i principi e i valori del manifesto sono validissimi anche in questi ambiti.

I principi agili prevedono la collaborazione tra le persone, la valorizzazione delle persone e la capacità di venire incontro alla flessibilità e al cambiamento necessari in un contesto come quello che viviamo noi, che non è esattamente strutturato e ripetitivo, ma piuttosto è destrutturato e variabile, mutevole nel tempo. Un approccio proattivo permette di far fronte a questo tipo di modelli e di situazioni, e in questi contesti le persone e l’approccio collaborativo giocano certamente un ruolo chiave. In particolar modo, diventa quasi rivoluzionario rispetto al modello waterfall la stretta collaborazione tra chi realizza il prodotto e chi ne definisce le specifiche, quindi cade la barriera tra cliente e gruppi di sviluppo e si forma un gruppo unico composto da tutte queste figure che lavorano insieme per un obiettivo comune.

Si tratta di una rivoluzione assoluta, ma, dobbiamo ammetterlo, non sempre così facile da applicare.

**È certamente assodato che l'approccio agile permette di risolvere le problematiche a cui hai accennato, ma allora perché, secondo te, non si riesce a lavorare in maniera "agile" anche ove converrebbe? Quali sono i limiti, se ce ne sono?**

Limiti ce ne sono, e, dal mio punto di vista, sono limiti culturali, organizzativi e anche relazionali. Partiamo dai limiti culturali e organizzativi: si è sempre lavorato in un certo modo, ed è difficile scardinare le barriere consolidate. Proprio il rapporto cliente-fornitore è il punto più difficile da rivoluzionare. Il cliente ha sempre visto il fornitore come "altro" rispetto a sé, dal canto suo il fornitore vede il cliente come "colui che definisce i requisiti e poi ci lascia lavorare". L'approccio agile ci suggerisce che l'elemento di successo è rappresentato proprio dal lavorare insieme, quindi ci vuole un cliente che capisca il vantaggio di rimanere in contatto e collaborare col gruppo di sviluppo. Anzi, se ci riferiamo ad uno dei framework più diffusi che è Scrum, dovrebbe anche nominare un Product Owner (PO) ovvero la figura chiave, se non la più importante, del progetto.

Il PO sa quello che vuole il cliente/utente, e se ne fa portavoce, anche se non sa come va realizzato, ma questo è ciò di cui deve preoccuparsi il team di sviluppo. Il PO ha il compito anche di indicare le priorità dei requisiti, perché è responsabile del prodotto da realizzare e sa, o dovrebbe sapere, cosa è più importante e urgente per ottenere un prodotto valido.

Non è così facile trovare clienti che abbiano questa attitudine e, soprattutto, il tempo necessario per svolgere questo lavoro.

È più facile applicare con successo gli approcci agili all'interno della stessa azienda. Si fa spesso riferimento, ad esempio, a Spotify, diventato addirittura un modello di riferimento, ma credo che ciò sia stato possibile principalmente perché è stata l'azienda stessa a finanziare l'intero progetto, e non si è venuto a creare il classico rapporto cliente-fornitore.

Di norma i contratti "vecchio stile" sono stringenti, prevedono il rispetto di clausole che vincolano i rapporti cliente-fornitore e non rendono applicabile l'approccio agile. Pensiamo ad esempio alle strutture pubbliche dove i contratti limitano fortemente l'applicazione della metodologia agile in quanto si rifanno esplicitamente al vecchio modello waterfall. Dal punto di vista relazionale, invece, l'approccio agile è fortemente basato sulla fiducia e sul rispetto tra tutte le persone del team.

Faccio un esempio concreto: se una persona del gruppo di sviluppo ti comunica la stima per realizzare un'attività, devi fidarti che stia fornendo una stima corretta, e non abbia intenzione di fregarti. Se si innesca questo timore o, viceversa, se il gruppo di sviluppo pensa che il PO abbia idee insensate e siano più valide le proprie, si creano delle distonie non funzionali alla riuscita del progetto.

Si può e si deve lavorare su queste situazioni, e infatti esistono figure di riferimento come agile coach e facilitatori che hanno proprio il compito di lavorare sulle persone e con le persone.

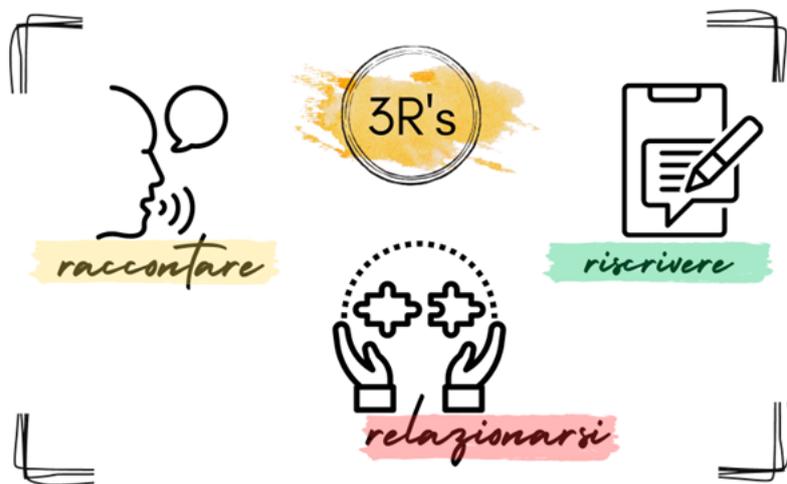
**Bisogna, allora, avere l'intuizione di affidarsi a qualcuno che supporti il team o l'intera organizzazione verso il cambiamento...**

È ancora una volta un problema di cultura, tutti i processi di change management hanno bisogno di agenti del cambiamento, hanno bisogno di volontà di cambiare e di persone che siano una guida. Purtroppo in giro si vedono ancora esempi di organizzazioni che si dichiarano "agili" ma non riescono ad andare oltre ad una lavagna piena di post-it.

**Tornando alla gestione dei requisiti, in base a quel che dicevi mi sembra che si parta proprio dalla gestione delle persone. Da tempo ragiono su questo aspetto: capita di avere all'interno dell'organizzazione o, genericamente, del team molta preparazione, un'elevata competenza tecnica e di dominio, ma magari è carente la capacità di gestire le persone, il team, la capacità di ingaggiare le persone e attivare i talenti, di lavorare non sull'intelligenza dei singoli ma sull'intelligenza collettiva. In che misura influisce questo aspetto, secondo te, sulla gestione dei requisiti e, in generale, sulla buona riuscita del progetto?**

Nella realtà dei fatti, spesso, non solo si deve colmare la distanza che c'è tra il cliente e il fornitore, ma anche tra i componenti del team di lavoro. Sono frequenti le situazioni in cui le persone hanno la tendenza ad assumere l'etichetta... io sono Software Architect... io sono Java Developer... affinché si possa aumentare la qualità delle soluzioni individuate bisogna lavorare di squadra, togliersi le etichette e - con un po' di umiltà - mettersi in gioco.

Bisogna riconoscere che il confronto porta sempre a un vantaggio, il team è tale se lavora come team e non semplicemente come gruppo, e la differenza tra le due modalità si nota sempre. Anche nelle competizioni sportive, quando è un gruppo di giocatori che gioca, ognuno si attiene rigidamente al suo ruolo, alla sua "etichetta", mentre quando gioca una squadra ognuno dà il massimo per raggiungere l'obiettivo: si vedono difensori che attaccano, attaccanti che difendono... la logica in campo è molto diversa e i risultati spesso sono sorprendenti. Culturalmente si dovrebbe ragionare in termini di obiettivi di team, responsabilità di team, successi di team. In questo contesto, è lo Scrum Master che gioca un ruolo fondamentale. Purtroppo a volte viene visto come una sorta di assistente, che fissa i tempi e le cerimonie del team, quando in realtà è una figura fondamentale perché da vero facilitatore - quasi da psicologo, a volte - deve aiutare il team ad assumere l'approccio giusto.



**Parliamo delle tre “R” dei requisiti: Raccontare, Riscrivere, Relazionarsi. La R di Raccontare appartiene all’utente, che deve essere in grado di spiegarti cosa cerca, cosa deve poter fare. La R di Riscrivere appartiene al team di lavoro, che riscrive, appunto, la storia raccontata all’utente formalizzando il requisito. L’ultima R, secondo me, è la più importante e costituisce il ponte tra le prime due: la relazione. Più la relazione è forte e basata sulla fiducia più diminuisce il gap tra ciò che l’utente racconta e ciò che il team riscrive e realizza. Questo può avvenire perché le due parti iniziano pian piano a parlare la stessa lingua e si lasciano condurre l’uno nel mondo dell’altro. Che ne pensi?**

È una buona immagine per rappresentare il processo di condivisione dei requisiti. Tocca le leve fondamentali che a mio avviso vanno curate con particolare attenzione. L’utente all’inizio si basa esclusivamente sulla propria esperienza, dà le cose per scontate, segue il suo personalissimo modo di ragionare (diverso da quello di altri utenti), per questo il team di sviluppo deve imparare a condurlo ad un nuovo modo di pensare, e questo avviene ponendo le domande giuste.

Anche avere un modo canonico di rappresentare i requisiti è di grande aiuto. Agile suggerisce l’utilizzo delle User Story, che solitamente si rappresentano nella forma “Io come utente voglio che il sistema faccia questa cosa per ottenere questo risultato”.

Inizialmente l’utente va aiutato a esprimersi in questo modo e a fornire tutti i dettagli su cui il team potrà lavorare. Siccome la lingua italiana è complessa e assai soggetta a interpretazioni, un altro elemento di aiuto è realizzare prima possibile qualcosa di tangibile che l’utente possa visionare e validare, anche attraverso prototipi e mockup. Appena si entra nel processo, i requisiti iniziano a venir fuori anche dal team di sviluppo, che inizia ad aiutare l’utente a interrogarsi su particolari scenari. Mi riconduco a Mike Cohn che suggerisce la regola delle tre C: Card, Conversation, Confirmation.

Una user story dovrebbe essere caratterizzata da un titolo e una descrizione sintetica che consenta di individuare il requisito, da un momento in cui il gruppo di lavoro e l’utente “conversano” per descriverlo al

meglio e dalla formalizzazione condivisa dei relativi criteri di accettazione.

**Con requisiti così poco definiti, soprattutto nelle fasi iniziali, l’utente può continuamente cambiare idea, o raccontare di volta in volta le cose in modo diverso, o aggiungere informazioni che vanno a modificare radicalmente ciò che è già stato realizzato. Ci sono dei suggerimenti per mitigare i rischi derivanti da ciò?**

Qualunque “agilista” ti risponderebbe che non puoi evitare questa situazione, devi solo seguire il cambiamento, ed è insito nei progetti che l’utente possa cambiare idea e aggiungere, togliere o modificare i requisiti. Ed è proprio per questo motivo che non si segue una pianificazione immutabile, bensì si suddivide il progetto in SPRINT di breve durata in modo da pianificare solo il lavoro a breve, con l’obiettivo di realizzare continuamente qualcosa che sia tangibile e verificabile. Tuttavia, è vero che rimettere mano alle stesse cose per troppe volte può essere frustrante, ed è chiaro che bisogna evitare che si esageri in questa direzione.

Costruire una relazione di fiducia col cliente consente di lavorare con spirito collaborativo e capire che si rema nella stessa direzione, tenendo conto anche dei costi che ci sono dietro al progetto.

In tale contesto, gioca ancora una volta un ruolo fondamentale il Product Owner che ha proprio il compito di assegnare le priorità, e nel far ciò deve tenere presente che, anche se non si è di fronte a una pianificazione rigida e fissa, i costi lo sono, e al termine del progetto quel che si è prodotto deve avere valore per tutti gli stakeholder. Diversamente, si rischia di avere poche funzionalità realizzate alla perfezione, ma non aver raggiunto il minimo indispensabile per considerare il prodotto completo.

Anche questa è un’attitudine, ma proprio perché si collabora e ci si aiuta, può intervenire lo Scrum Master a rendere maggiormente esplicite queste situazioni laddove si verificano.

**Un' ultima domanda: secondo la tua esperienza, quali sono le cose che hai trovato più difficili o che avresti fatto diversamente e che ti permettono, oggi, di formulare consigli concreti per i team agili per una buona gestione dei requisiti?**

Sicuramente non aver paura di conoscere l'utente e andare preparati agli incontri di approfondimento dei requisiti, perché nessuno vuole perdere tempo. Se, ad esempio, si organizza una sessione di refinement dei requisiti, prepararsi può voler dire studiare il dominio dell'utente, stilare una serie di domande o una checklist di punti da verificare, portare esempi di mockup che descrivono soluzioni realizzate per requisiti simili.

Infine, consiglio vivamente di cercare di mettersi dalla parte dell'utente e non restare nella posizione di chi ha la soluzione tecnica in tasca. Non ascoltare l'utente, non capire i suoi bisogni e pensare che la nostra soluzione sia già definitiva e corretta non crea la relazione di cui abbiamo parlato e di cui ricono-

sciamo l'importanza. Aggiungo anche l'importanza di individuare una modalità per formalizzare i requisiti, quindi utilizzare le user story o qualsiasi modalità ben strutturata, da perfezionare e adattare nel corso del progetto, se necessario.

Gli errori sono difficili da classificare e generalizzare perché dipendono molto dai contesti, tuttavia mi sento di dire che il maggior rischio di insuccesso deriva dal rimanere chiusi nelle proprie convinzioni. Lavorare e affinare le competenze tecniche in un team, infatti, è molto più semplice che lavorare sulle soft skills, perché le competenze alla fine si allineano e tecnicamente si cresce durante il progetto, mentre l'attitudine a lavorare in team si costruisce con più fatica. Infatti, la scelta delle persone deve essere oculata, e magari bisogna prevedere un lavoro preliminare di formazione e change management. Agile non fa per tutti, ma se si vuole lavorare in maniera agile bisogna porre tanta attenzione su questi aspetti. D'altra parte, cinque solisti non fanno un bel concerto...



# ITALIAN AGILE DAYS 2023

La conferenza italiana dedicata all'agilità, 20<sup>a</sup> edizione  
17-18 Novembre 2023

## iad23

Italian Agile Days è la principale conferenza italiana rivolta alle persone interessate alla cultura e alle pratiche agili.

Una storia che inizia nel 2004 e che fa di iad, giunta alla sua ventesima edizione, una delle più longeve e attive conferenze agili a livello mondiale.

Seguici sui nostri canali social per rimanere aggiornato sull'evento.

### DOVE

Politecnico di Milano - Milano

### QUANDO

Venerdì e sabato

17-18 Novembre 2023



# Si salvi chi può!! Dobbiamo partecipare ad una gara...



LUCA CIANCI

Scrum master e facilitatore del metodo LEGO SERIOUS PLAY  
Mi piace aiutare le persone a mettersi in gioco per trovare insieme nuovi modi di fare le cose

Se parliamo di requisiti e specifiche, saremo tutti d'accordo nel dire che solitamente le gare con i relativi documenti di capitolato rientrano in quella categoria di attività che chiameremo qui semplicemente "fumose".

Documenti diversi che raccontano aspetti simili di una stessa vicenda spesso andando in contraddizione tra loro; ecco allora che possiamo fare le richieste di chiarimento per provare ad avere risposte a specifiche e requisiti che spesso non sono chiari nemmeno nella testa di chi la gara la sta richiedendo perché sappiamo bene che molte volte si lavora ad un'idea, si insegue un desiderio e si riusano altri documenti preparati in precedenza per altri scopi.

Il dramma delle gare è che, in ogni caso, dovrai rispondere.

Per fare questo dovrai necessariamente farti un'idea e aver capito qualcosa riguardo all'argomento in questione poi in base al tuo livello di comprensione dovrai uscire con un'offerta adeguata.

Da qualche anno all'interno della mia azienda quando abbiamo a che fare con le gare più complesse, propongo **un workshop di gruppo che utilizziamo per comprendere come team i contenuti di una gara**, condividere le possibili strategie, riflettere e poi decidere sul da farsi.

Il presupposto è appunto quello di lavorare come team e non individualmente come persone coinvolte alla gara che nei ritagli di tempo producono documentazione.

**Lo scopo principale è quello di confrontarsi sui requisiti e le specifiche oggetto delle gare perché pro-**

prio da questi partiremo per comprendere e confezionare la nostra offerta.

Dato che ognuno di noi è diverso, se leggiamo in quattro un capitolato, troveremo requisiti più o meno utili in base alla nostra percezione del tema, della fattibilità, delle attitudini e altro ancora. In questo senso quindi il nostro workshop parte dal presupposto che le persone al tavolo conoscano la gara e abbiano letto i documenti di capitolato che tipicamente vengono anticipati rispetto alla data dell'incontro.

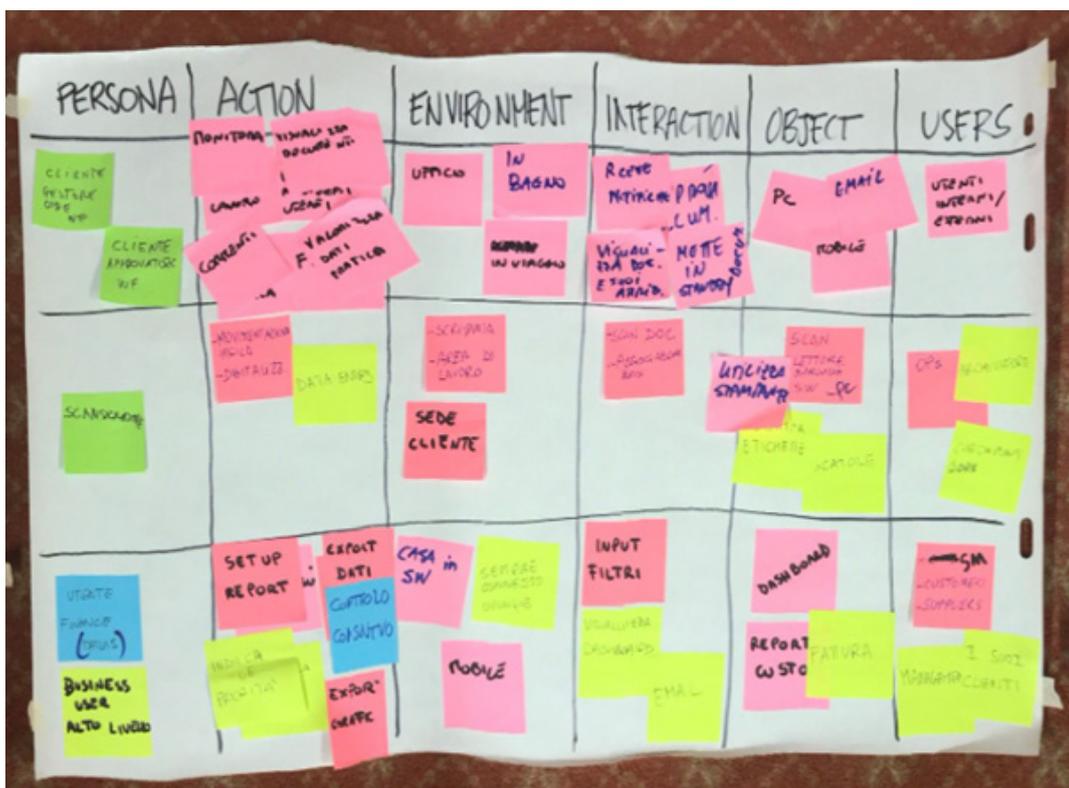
### Gli utilizzatori determinano i requisiti

Nella prima fase del workshop lavoriamo alla veloce definizione delle user personas, gli utilizzatori del sistema e proviamo rapidamente a mappare il flusso di processo legato ad essi.

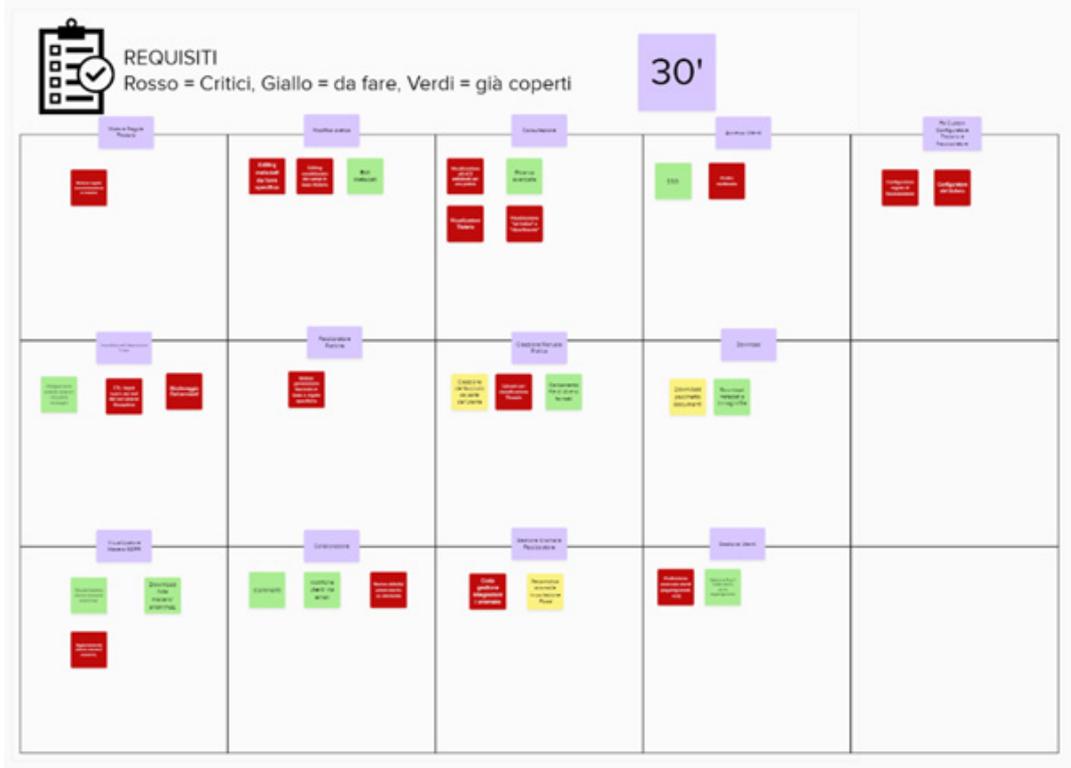
La mappatura delle persone avviene di solito con il metodo AEIOU dove ogni lettera ci ricorda un elemento chiave associato all'utilizzatore:

- **Action** - Cosa fa?
- **Environment** - Dove si trova?
- **Interaction** - In che modo si muove e con cosa interagisce? (strumenti, sistemi, ...)
- **Object** - Cosa utilizza?
- **Users** - Interagisce con altre persone o altri sono inseriti nel suo processo di lavoro?

Non occorre tanto tempo per compilare questo canvas, ma farlo insieme condividendo i diversi punti di vista pone le basi per un processo di raccolta dati e informazioni strutturato. All'inizio le persone ti chiederanno "ma a cosa serve?" tuttavia scopriranno presto che c'è differenza tra un generico utente utilizzatore del sistema e utenti diversi che in momenti diversi hanno necessità e quindi esigenze diverse.







**La pianificazione non si nega a nessuno**

Chiudiamo il nostro workshop con una fase di stima veloce dei diversi requisiti e il loro posizionamento all'interno di un **release plan** che ci permette di comprendere cosa possiamo rilasciare rapidamente e cosa invece diventerà parte dei problemi da gestire.

Con queste stime alla mano e con il piano di rilascio condiviso chiudiamo il nostro workshop lasciando che le persone titolate a farlo confezionino l'offerta in base a quanto emerso con il gruppo di lavoro.

**Lavorare in modo diverso dipende da noi**

Quello che ho raccontato non è fantascienza: il workshop che ho descritto può durare da 4 ore a diversi

giorni in base al grado di complessità della gara. Potete fare tutto subito in un'unica sessione da 4-6 ore accettando il livello di incompletezza fisiologico o spezzettare i diversi temi e trattarli in momenti specifici per avere maggiore focus. Serviranno magari degli approfondimenti specie sui requisiti che marcheremo in rosso ma questo fa parte del processo di esplorazione e condivisione.

La cosa veramente importante è che il team di lavoro sia abbastanza piccolo, 4-6 persone al massimo, e che tutte le persone utili siano coinvolte al tavolo in modo da sfruttare al meglio il tempo e le competenze di tutti. Solitamente un sales, un presales e un paio di sviluppatori ben ferrati sono in grado di portare a casa il risultato ma molto dipende anche da come è strutturata la vostra azienda.



# Le Community italiane che supportano AE Conference

All'interno delle community locali ci  
sono ricchezze ed esperienze che vanno  
sostenute e raccontate



Scrum Agile Milano SAM  
Meetup : Scrum-Agile-Milano-SAM



Agile For Italy - Milano  
Meetup : AgileForItalyMilano  
sito: <https://www.agileforitaly.com>

Agile Marketing Italia  
LinkedIn : Agile Marketing Italia  
<https://www.linkedin.com/groups/13607881/>



**italian  
agile  
movement**

Italian Agile Movement  
sito: <https://agilemovement.it/>

facitalab  
sito: <https://facitalab.it/>



Grusp  
LinkedIn : [linkedin.com/  
company/GrUSPMarketing](https://linkedin.com/company/GrUSPMarketing)  
sito: [grusp.org](https://grusp.org)



La prima conferenza interamente dedicata alle esperienze pratiche e realmente sperimentate dalle aziende.  
La prima conferenza adattata per un'esperienza totalmente online.  
La prima conferenza fruibile quando vuoi.  
La prima conferenza in cui risponderanno solo alle vostre domande.

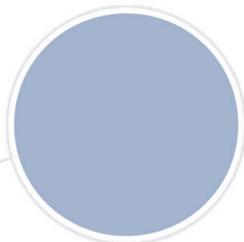
# AGILE EXPERIENCE

## CONFERENCE

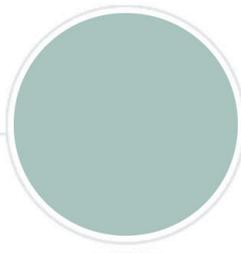
da Dicembre 2021



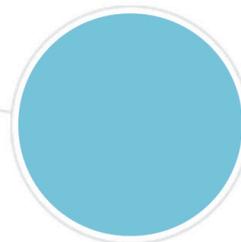
La prima conferenza interamente dedicata alle esperienze pratiche in cui un gruppo di esperti viene intervistato da un gruppo di intervistatori.



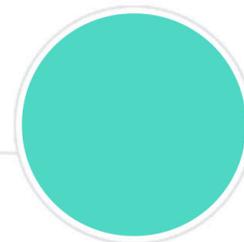
Intro 15 min di  
presentazione  
dell'ospite



Un gruppo  
ristretto di  
persone fa  
domande in  
diretta



Risposte  
concrete di  
esperienze reali



Streaming su  
Youtube

<https://agile-experience.agileforitaly.com/>



# REQUISITI



**GIANLUCA COLONNESE**

Client & Project Manager presso Aubay Italia

Ho pensato tanto a come affrontare questo tema con un articolo ed alla fine ho realizzato che la cosa più utile è parlare della mia esperienza. Come dice un caro amico “L’esperienza di molti è patrimonio per tutti”.

Io sono un project manager che si occupa di coordinare lo sviluppo di software da zero alla consegna finale e, qualora necessario, anche organizzare la maintenance per i successivi anni.

Tutti i progetti nascono per soddisfare bisogni, aspettative e necessità di un soggetto, privato o azienda che sia.

Il progetto da seguire avrà sicuramente come target: benefici e valore al committente, che sono il primo nodo della matassa necessario da sciogliere. Valore e beneficio sono profondamente soggettivi, capire davvero cosa serve al destinatario spesso è un’impresa. Infatti un progetto ha successo non solo per una buona gestione ma, principalmente, se soddisfa le esigenze del cliente. Un primo passo è quello di trovare un modo per rappresentare e descrivere queste esigenze. Questa modalità si chiama requisiti di progetto. Il requisito è una caratteristica misurabile e verificabile necessaria al prodotto affinché possa creare valore.

Spesso si rischia di sottovalutare le esigenze del richiedente soprattutto se sono stati già eseguiti progetti simili, ma per comprendere realmente cosa desidera il richiedente bisogna mettersi il più possibile dalla sua parte e creare il software con un occhio alla sua visione.

La bontà dei requisiti può essere valutata in 2 passi: Identificazione dei requisiti e verifica della loro qualità.

Nel caso dell’identificazione dei requisiti, è necessario evitare di confondere ciò che vorrebbe il cliente rispetto a ciò di cui ha realmente bisogno. I requisiti sono ciò di cui il cliente necessita e non ciò che dice di volere. Se ci si limita a soddisfare i desideri della controparte si rischia che in corso d’opera quello che gli serve davvero non sia ciò che si aspettasse realmente. Da qui prolungamento dei tempi di consegna, costi extra, corpose liste di variazioni quando ancora il software non è stato terminato, che successivamente si chiamerebbero CR, ma durante lo sviluppo del progetto spesso vengono definiti “misunderstanding”. Da non sottovalutare neanche le complicazioni che emergono nel momento in cui si progetta un nuovo software come, prima tra tutte, a mio avviso, è la difficoltà di affrontare il cambiamento, di qualsiasi natura esso sia, tecnologico, organizzativo ecc..., che rende i requisiti di qualsiasi progetto instabili e vola-

tili. Per evitare tutto ciò ci sono diversi modi, quello da me più utilizzato è la tecnica dei “five whys”. Di sicuro non sempre è necessario arrivare alla profondità di 5 perché che concatenano l’esigenza e giustificano quindi il requisito, ma è importante che in questa fase chi si occupa della “requisitazione” indossi la veste di un valido investigatore affinché il tool che si sta progettando sia realmente utile e funzionale.

Tornando ai 2 punti di cui scrivevo sopra, riguardo alla verifica della qualità dei requisiti, è fondamentale partire dalla necessità dei futuri utilizzatori del tool e non da una soluzione presentata all’inizio dal richiedente. Spesso basterà rispondere a questo quesito: “ciò che si sta realizzando come progetto, è utile ad agevolare la vita di qualcuno o risolvere l’esigenza del richiedente?”

In questa fase subentra un altro nodo da sciogliere, il nostro software andrà a risolvere talvolta la richiesta del cliente solo nel caso in cui ci sarà anche la possibilità di integrarlo con i sistemi già esistenti. Per arrivare subito al punto e comprendere cosa serve realmente, la prima cosa è parlare con l’utente ovvero con chi realmente utilizzerà il tool che si sta progettando. Il futuro utilizzatore del software sicuramente può rispondere ai nostri quesiti in maniera tale da farci capire come migliorare la sua vita lavorativa e non come complicargliela.

Ove possibile e, se il progetto lo permette, per affrontare gli incontri con l’utente si utilizzano le “User Story” provenienti dalla metodologia Agile.

Per chi legge questa rivista probabilmente non è necessario approfondire il termine User Story quindi investo giusto due righe per dire che: Le User Story sono le specifiche da condividere con il team di sviluppo. E’ un approccio diverso da quello classico dove Specifiche Tecniche oppure Requisiti Funzionali arrivavano come dictat dall’Analista Funzionale/PM e che trascuravano l’esigenza reale del requisito per l’utente finale. Di fatti il concetto di User Story mette l’utilizzatore del nostro prodotto al centro dell’attenzione durante il processo di sviluppo. Pari importanza viene data alla Storia ovvero a ciò che la nuova funzionalità si prefigge di ottenere in modo da soddisfare i requisiti emersi.

Alla base di ogni progetto c’è sempre un’esigenza da soddisfare e, come dico sempre, “più scendi in profondità all’inizio e meno dovrai ritornare sul tema successivamente!”

# QRP

Developing Professionals

**Fornitori di formazione  
sui metodi Best Practice**

**Developing Professionals  
to facilitate Changes.**



NEIL PERKIN

# AGILE TRANSFORMATION

SOPRAVVIVERE, SVILUPParsi  
E COMPETERE NELL'ERA DIGITALE

edizione italiana a cura di  
**MARCO CALZOLARI**



gueriniNEXT



gueriniNEXT

Le tecnologie digitali hanno avuto un impatto enorme sulla nostra realtà. Il mondo cambia più in fretta, e così le richieste del mercato. Nuovi competitor e nuove tecnologie dall'effetto dirompente compaiono sulla scena a ritmo sempre più serrato. Eppure moltissime imprese restano bloccate. Bloccate da metodologie di lavoro sorpassate, in processi decisionali poco orizzontali e che inibiscono l'innovazione, in una cultura aziendale che premia il conformismo e l'efficienza immediata anziché l'intraprendenza e la volontà di imparare.

Agile Transformation offre una guida ricca e dettagliata a uno dei più importanti modelli di organizzazione aziendale del terzo millennio. Il testo traccia la strada per creare imprese che sappiano continuamente rimodellarsi sulla base delle esigenze dei clienti e dei loro feedback, che lavorino in team orizzontali e flessibili, supportati dalla tecnologia e che sappiano utilizzare le proprie risorse in modo continuamente adattabile al futuro. Un contributo fondamentale per orientarsi in una nuova era per le organizzazioni.

---

**Neil Perkin** è il fondatore dell'agenzia di consulenza digital Only Dead Fish, attraverso cui aiuta le organizzazioni a relazionarsi alle nuove tecnologie, innovarsi e diventare più agili. Ha lavorato con decine di aziende leader a livello mondiale, come Vodafone, Astra Zeneca, Samsung, YouTube e Warner Bros. Dirige per Google il podcast Firestarters ed è keynote speaker per il Google Squared Programme.

---

**Marco Calzolari**, imprenditore e filosofo, è co-fondatore e CEO dell'agenzia di coaching e consulenza Agile Reloaded, con sede a Milano e Londra. Ha esperienza come project manager, direttore generale, keynote speaker e investitore. Attualmente collabora con piccole, medie e grandi aziende italiane e internazio-

# AGILE TRANSFORMATION



**TIZIANO INTERLANDI**

Co-Founder di AgileForItaly, AgileItalia e AgileExperienceConference.

Agile Coach e Manager in Cerved Group.

Batterista e Fónico.



**MARCO CALZOLARI**

CEO @ Agile Reloaded. Entrepreneur, Philosopher, Management Coach, Keynote Speaker

Cogliamo l'occasione dell'uscita del libro Agile Transformation di Neil Perkin per fare alcune domande a Marco Calzolari, che ha curato l'edizione italiana.

Ho avuto il piacere di conoscere Marco nelle varie manifestazioni e conferenze italiane e lo ringrazio personalmente per il tempo che ci ha dedicato.

Il libro (ringraziamo Guerini Next per la possibilità di recensirlo) è un'opera importante e che vuole trattare tutti gli aspetti ad oggi conosciuti all'interno delle trasformazioni agili. Un libro che consigliamo e che potete trovare nei link in fondo all'articolo.

Quale migliore occasione per fare qualche domanda a Marco.

## L'intervista

Parliamo spesso di transizioni e di come queste siano esposte alla mutabilità del Management ma soprattutto necessitano di energia continua, quasi a seguire le leggi della termodinamica, paragone che spesso facciamo per far comprendere la non stazionarietà dei contesti complessi.

Leggendo la tua prefazione ed in generale alcuni punti chiave del libro abbiamo pensato di farti tre domande:

**Quando, come e perché è corretto chiamare un coach esterno per transizioni locali o generali. I principi del Lean suggeriscono che è necessario far crescere leader interni perché comprendono e vivono il cambiamento come missione (nel libro si utilizza il termine "missionari" appunto per indicare persone che credono e vivono quello che fanno).**

**Sappiamo anche che il mondo non è bianco o nero. Secondo te in quali casi è necessario far riferimento a forze esterne? In che dosi e in che modalità?**

I motivi per cui può essere opportuno il supporto di consulenti o coach esterni durante un percorso di cambiamento organizzativo e metodologico sono diversi. Non voglio riprendere il famigerato concetto del "punto di vista esterno", soprattutto perché dipende dal "peso specifico" del punto di osservazione e ancora di più dalle azioni che ne seguono. L'opinione di qualsiasi "terzo" ci permette di crescere soltanto se la consideriamo autorevole e se incarna i nostri valori. Purtroppo ho assistito a molte crisi di rigetto di dettagliate indicazioni di consulenze poco congruenti con la realtà aziendali e la cultura delle persone.

Indubbiamente, un valore che può portare un coach esterno è la sua precedente esperienza in progetti di cambiamento presso altre aziende, che permette di accelerare il processo e ridurre i rischi di incorrere in errori più o meno comuni. Un coach con una buona esperienza può riconoscere gli antipattern, collaborando attivamente con il personale interno, e stimolare nuovi accordi sul processo di adozione di nuove pratiche, di sperimentazione, di riconfigurazione di ruoli e responsabilità, per risolvere i problemi o diminuirne l'impatto promuovendo nello stesso tempo

i risultati e gli apprendimenti.

Un'azienda non è mai completamente autosufficiente, ha bisogno di introdurre regolarmente nuovi elementi per evolvere, mentre porta avanti la sua strategia di business. Tuttavia, come anche tu suggerisci nella domanda, il contributo di figure di questo tipo va integrato e pensato con lo scopo di rendere "residenti" le competenze dell'azienda, distribuendole tra le persone che governano realmente i processi che andranno modificati. Non solo: la competenza di poterli cambiare ulteriormente è altresì da sviluppare: immaginare un contributo esterno come se fosse l'installazione di un nuovo software o l'erogazione di una formazione, senza davvero considerare un investimento in attività e focus di personale interno, è una perdita di denaro e di tempo.

Un altro fattore da considerare nel momento in cui ci si rivolge a un supporto esterno è la scalabilità, o meglio la dimensione dell'eventuale riorganizzazione da svolgere. Una trasformazione aziendale non ha una vera "fine", è una sequenza a volte non lineare di iniziative di cambiamento in progressione: da un piccolo progetto pilota a una intera area con numerosi team coinvolti, da una porzione di processo a tutto un value stream di prodotto o servizio, da un'introduzione di tecniche di lavoro come Scrum alla definizione del processo di alimentazione e aggiornamento di più backlog. Le competenze in gioco e l'esperienza nel dominio – sia metodologico sia di mercato – sono tante, e un coach da solo potrebbe non riuscire ad abbracciare la dimensione del cambiamento, per quanto supportato dal personale interno: vanno create da subito squadre che comprendano consulenti, personale di coordinamento (leggi: manager) e personale coinvolto nel lavoro. Nel nostro approccio di consulenza e di coaching ci impegniamo ad accompagnare l'azienda da un livello "start" a un livello "evolution", in cui il bisogno rappresentato dai nostri coach è soddisfatto da quello che abbiamo messo in atto con il personale interno: competenze agili, di facilitazione, di sviluppo, di process e organization design, nonché community a supporto delle future iniziative di riadattamento o cambiamento dei processi di lavoro. Ad un certo punto, il supporto esterno diventa più orientato alla definizione degli obiettivi di business delle potenziali iniziative di trasformazione, più che alle specifiche metodologie da applicare.

Come ultimo aspetto che determina il "peso" di un supporto di coaching, aggiungo un ulteriore criterio da considerare: l'aderenza ai valori dell'organizzazione e delle persone. Non c'è cambiamento che si possa realizzare senza una adeguata consapevolezza del bisogno di affrontarlo: una delle prime attività di un coach è quella di integrare le diverse necessità esistenti in azienda, valorizzando gli asset e rendendo quanto più possibile espliciti i risultati che ci si può aspettare nel breve e nel medio periodo. Nel lungo periodo ci si può immaginare senz'altro uno "stato futuro", ma va disegnato insieme.

**Una seconda domanda abbastanza agganciata alla prima. Come anche citato nel libro a volte viviamo fasi in cui alcuni concetti diventano buzzwords e spesso vengono svuotate di significato. Questo sicuramente nuoce al concetto che è la cultura ed il mindset a mantenere risultati o attivarli. In questo contesto il Management spesso è sommerso di concetti che poi faticano a mettere a terra. Un tempo si parlava di Spotify Model, oggi più di Business Agility. Come possiamo agire sul Management in maniera concreta ed efficace?**

Guarda, lo stesso concetto di “management” è forse la buzzword più vuota. In molte aziende management può indicare un gruppo di persone con poteri e autonomia decisionale specifica, mentre in altre è un aggregato di cultura aziendale espressa sia in comportamenti sia in strutture di governance “separate” da chi svolge il lavoro. Tuttavia, un concetto non diventa buzzword senza contenere qualcosa di importante, che intercetta un bisogno o un aspetto su cui vale la pena investire. Se con “agire sul Management” intendi attivare un volano di cambiamento e innovazione che tenga conto nello stesso tempo del business e delle persone, la risposta consiste nel fare chiarezza sulle aspettative, partendo proprio da modelli e definizioni preconfezionate per arrivare alle proprie, e rendendo espliciti i criteri di successo. Prendiamo ad esempio lo Spotify Model: una volta compreso cosa sia stato per chi lo ha “inventato” e per il loro specifico modello di business, cosa possiamo farcene? Ogni modello alla fine è un insieme di pratiche che lavorano in modo sinergico. Di quali pratiche possiamo beneficiare, e quale sforzo ci richiede la validazione di questa ipotesi, prima di “cambiare tutto” e sperare che funzioni? (Spoiler: non funzionerà come previsto). Allo stesso modo, prendiamo il concetto di Business Agility, definiamola in termini di aspettative e potenzialità per la nostra azienda, con la sua specifica configurazione: business e agilità possono avere caratteristiche molto diverse, ad esempio se eroghiamo servizi tramite piattaforme software proprietarie, oppure se vendiamo prodotti tramite canali di terze parti, oppure se il nostro modello di business è un mix di diversi sotto-modelli che si intersecano. Una delle sfide più stimolanti in un processo di “agilizzazione” è proprio questa: “definire” cosa ci si può aspettare e come misurarlo, coinvolgendo le persone per visualizzarlo e poi tradurlo in realtà.

**Sto vivendo un momento di cambiamento professionale ed organizzativo che sta portando alla luce alcuni concetti prima semplici da raccontare e vivere, ma oggi critici e con pochi esempi di letteratura. Mi riferisco al mondo delle Operations. Poco citato oggi diventa attore di un cambiamento epocale in cui la tecnologia guida (a volte male) e crea nuove forme organizzative (a volte mirate al mero taglio dei costi). Sicuramente un ragionamento Lean sui costi oggi è d'obbligo (siamo costretti a ripensare ai nostri sprechi), ma penso anche che le OPS saranno la chiave per le future aziende capaci di modellare**

**la tecnologia e che il contenuto umano sarà sempre più di valore in un mondo che vorrà farne a meno. Cosa ne pensi su questo tema?**

Ti ringrazio per questa domanda che ti coinvolge anche sul piano professionale, e che riguarda un aspetto che io stesso mi stupisco di quanto sia arrivato “tardi” nel perimetro dell’agilità, almeno in Italia. L’argomento è complesso per una risposta breve, per cui provo a farlo con due spunti soltanto. Il primo pensiero è organizzativo: stiamo affrontando le conseguenze di una strategia di separazione delle competenze che negli anni è sembrata promettente in termini di efficienza e scalabilità: chi si occupa di sviluppo non si occupa di operations, e viceversa. Spesso non sono nemmeno dei silo della stessa azienda, bensì aziende diverse legate da contratti di fornitura, non sempre trasparenti. Ragionare per value stream ha reso più evidente i limiti di questa configurazione: i team hanno imparato a sviluppare in modo iterativo e incrementale, ma il prodotto o servizio, una volta sviluppato, non esaurisce il ciclo di vita con la consegna delle sue funzionalità al cliente: ha bisogno di tutta una serie di interventi, manutenzioni, evoluzioni e misure che aumentano la complessità: molti più fattori da considerare in fase di progettazione, di definizione delle priorità, di scelte architettoniche. Questo comporta una ulteriore integrazione di competenze e un coinvolgimento di persone la cui expertise è cresciuta in modo verticale, con una limitata visione d’insieme, ma dal mio punto di vista è un investimento necessario, e può essere affrontato anch’esso in modo iterativo e incrementale. Il secondo pensiero è legato alla domanda precedente: forse il tema delle operations “agili” sta già diventando una buzzword? Come ben dici, è sempre stato un “territorio” del Lean, per cui dovremmo già sapere come ridurre i costi mantenendo la qualità del servizio al cliente. L’agilità ha fatto propri molti dei principi Lean, per cui è paradossale che oggi si parli di difficoltà di integrazione di sviluppo e operations, a mio avviso. La sfida in realtà è proprio quella di riconsiderare questi due termini alla luce della costante evoluzione tecnologica: il “digital” non comporta una nuova configurazione di macchinari dentro una fabbrica, bensì un ripensamento dell’azienda stessa in termini di design organizzativo. Data la vastità del dominio tecnologico, non può che essere affrontata dalle persone che sanno “di cosa stiamo parlando”, ovvero che possano mettere in gioco competenze, interessi e soluzioni specifiche a sfide sempre nuove.

In conclusione, anche per rispondere alla tua provocazione sul contenuto umano di cui il mondo vorrà fare a meno, sono convinto al contrario che il contributo delle persone possa essere addirittura valorizzato, in particolare nel mercato dei servizi digitali. La responsabilità però di trovare lo spazio per trasformare questo contributo dal “fare le cose” al “come far succedere le cose” ricade sia sull’azienda, sia sulle singole persone e la loro professionalità.

# Il valore etico della Qualità

La passione per la QUALITÀ' del GRUPPO GALGANO è insita nel nostro DNA dal 1962.

Con le nostre attività di Consulenza di Direzione e Formazione Manageriale, lavoriamo a fianco di aziende di ogni settore e dimensione grazie ad un team di consulenti realizzatori fortemente orientati ai risultati.

La nostra missione è far crescere le persone, trasferendo know-how innovativo, che aiuti a sviluppare la competitività delle aziende nostre clienti

“Non si può fare Qualità senza coinvolgere la mente e il cuore delle persone”



**GRUPPO GALGANO**

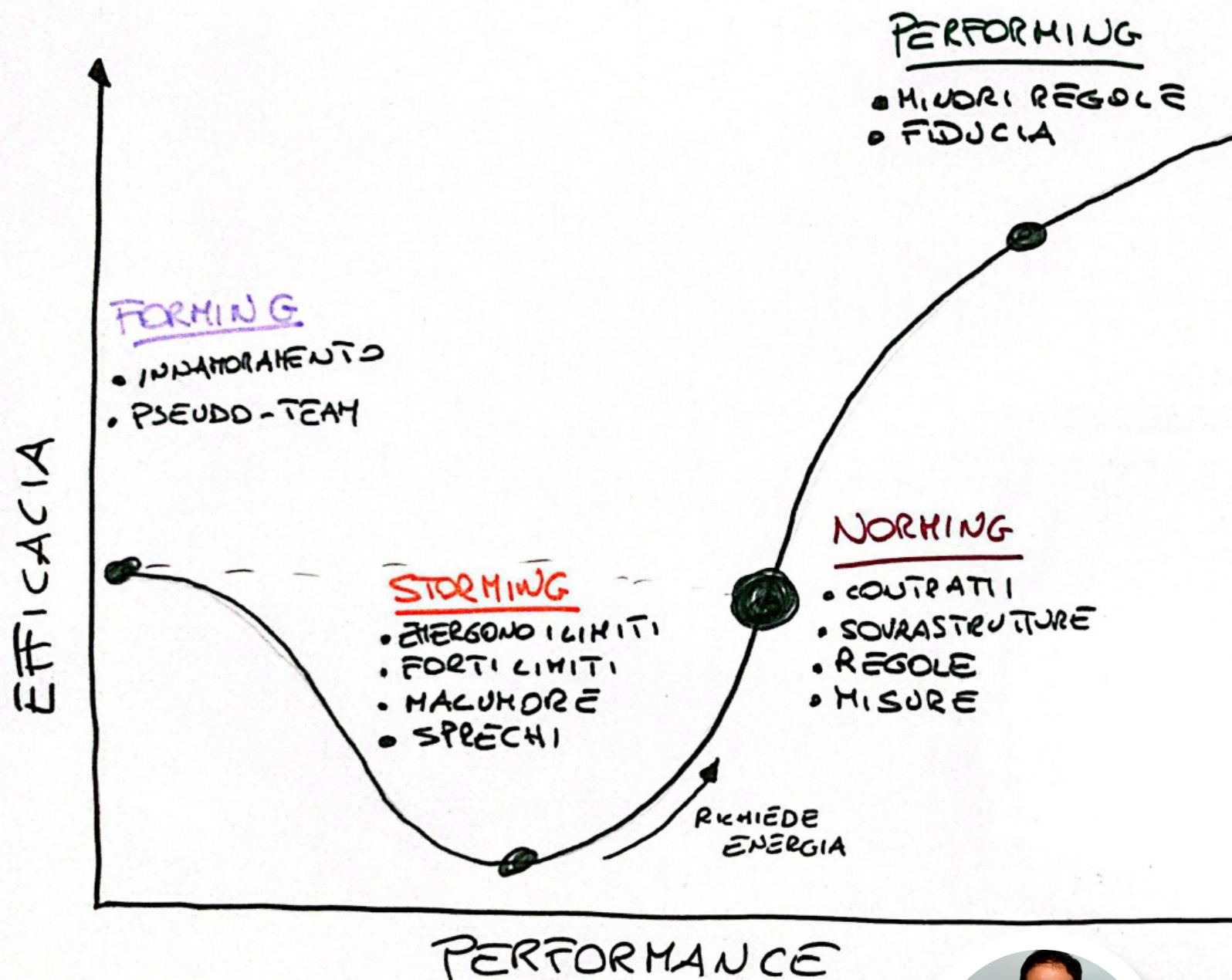
consulenti di direzione

*La Leadership attraverso l'Eccellenza*

Via della Moscova 10, 20121 Milano

Tel 02.39.605.1 - [infoga@galganogroup.it](mailto:infoga@galganogroup.it) - [www.galganogroup.com](http://www.galganogroup.com)

# Sovrastrutture e Norming



TIZIANO INTERLANDI

Co-Founder di AgileForItaly, AgileItalia e  
AgileExperienceConference.

Agile Coach e Manager in Cerved Group.

Batterista e Fonico.



Cerco di andare in continuità con le domande/risposte poste a Marco Calzolari nell'articolo precedente.

Leggendo il libro Agile Transformation, lettura sicuramente completa soprattutto con l'aggiunta della chiave italiana, mi sono domandato nuovamente perché parliamo da così tanto tempo di Agile e sembra mai esserci una fine. Per carità, a Milano si dice: "Fa e disfa l'è tut un laurà" (fare e disfare è sempre lavorare), però effettivamente dobbiamo chiederci come mai è così complesso lavorare insieme.

Agile affronta di fatto le relazioni umane, la complessità che le accompagna, cercando sempre di riportare la discussione sulle cause radice dei problemi.

Spesso, appena rientriamo nella nostra zona di comfort, tendiamo a darci regole per il fatto che le regole precedenti non hanno funzionato. Da questo momento in poi le chiameremo sovrastrutture.

Quante volte all'uscita di una retrospettiva abbiamo aggiunto sovrastrutture alle sovrastrutture dello Sprint precedente. Quante volte normare è stata la soluzione.

Ed è il momento di descrivere nuovamente il Ciclo di Tuckman. In Y abbiamo la performance del team, in X non il tempo, ma una serie di fasi discrete.

Tutti i Team devono far fronte a queste fasi, ma non è detto che le affronteranno tutte:

- **Forming:** in questa fase (che si ripete ogniqualvolta un membro del team si aggiunge o si sottrae) il gruppo si forma ed il più delle volte si parla di "fase dell'innamoramento". Noi tutti mostriamo i nostri lati migliori e tendiamo ad evitare lo scontro.
- **Storming:** la fase in cui si litiga, una fase che è meglio affrontare il prima possibile. Nello storming le cose non vanno bene, ci si arrota su problematiche semplici ma che vengono vissute come insuperabili. La performance del team è ai minimi.
- **Norming:** la fase che approfondiremo in questo articolo. In questa fase cerchiamo delle regole e sovrastrutture per superare la fase di storming. E' la fase più delicata che segna la performance del Team. Se guardate tra Forming e Norming abbiamo un falso piano. Anzi il norming è in salita come ad indicare che necessitiamo di uno sforzo maggiore per superare questa fase ed andare verso il performing.

- **Performing:** il team performa al suo meglio.
- **Adjourning:** il team conclude la sua vita per conclusione progetto o prodotto. Le performance sono più basse, il team sapendo del proprio scioglimento in un certo senso "si rilassa".

Facciamo un passo avanti e concentriamoci sul norming. Spesso confondiamo la creazione ed applicazione di regole con la soluzione dei problemi. Facciamo un esempio. La Definition Of Ready.

Già in molte occasioni ho espresso le mie perplessità rispetto a questa pratica in quanto è una sovrastruttura appesantente e che nasconde i problemi sotto al tappeto.

La Definition Of Ready è quella pratica che consiglia di avere check in ingresso ai requisiti (spesso sotto forma di storia) per garantire che il contenuto sia sufficientemente chiaro al Team e quindi azionabile.

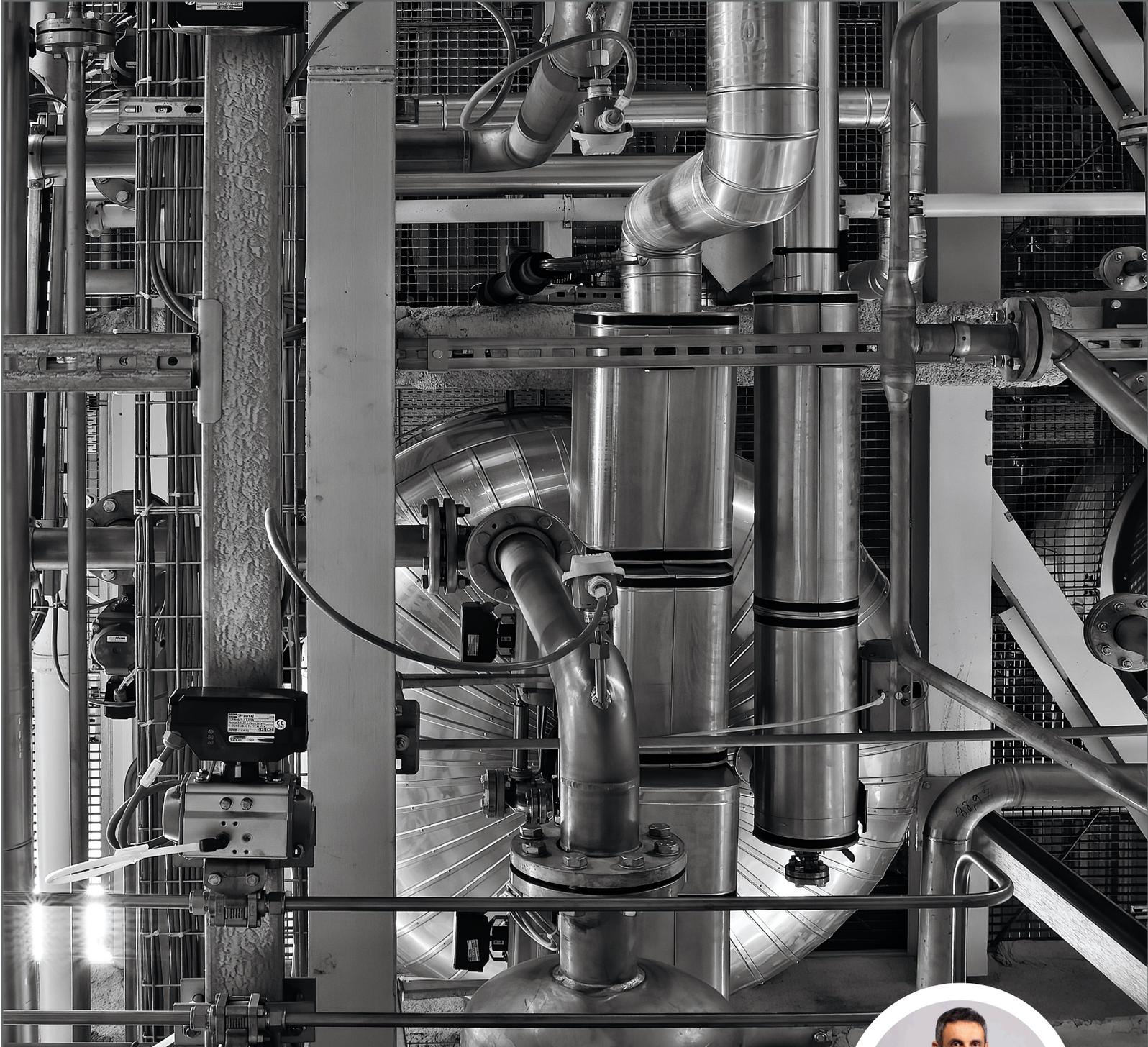
Detta così sembra molto innocua, ma non ho ancora visto una volta che non abbia provocato effetti a cascata, magari l'aggiunta di ulteriori sovrastrutture.

Quindi, il Team impone verso l'esterno (a volta anche verso se stessi) quella che innocentemente sembra un check list ma che non è altro che una maniera "difensiva" di applicare Agile. Già ne abbiamo parlato, Agile è un abilitatore e deve innescare collaborazione e non una serie di regole. Esiste una maniera migliore? sicuramente tendere alla collaborazione e non è uno schema Excel o Jira che risolverà i nostri problemi.

Spesso si parla di comportamenti e pratiche "ecologiche" nel senso siano facilmente praticabili. Nel momento in cui vogliamo cambiare i nostri comportamenti le regole possono in un primo momento aiutarci, ma a patto che non trasformino il tutto in qualcosa di impraticabile.

Quindi... prima di aggiungere "on top" una pratica o una serie di regole pensiamoci bene. Cerchiamo sempre di rimanere "semplici" e di preferire la risoluzione dei problemi alla loro radice.

Concludendo, pensate all'evoluzione della guida Scrum. Gli ideatori, insieme ai feedback della community mondiale, hanno rimosso regole mantenendo la focalizzazione sui valori e principi. Scrum è volutamente incompleto perché il confine tra gabbia e spazio aperto spesso è molto sottile.



## Industrializzazione Agile per Hardware



MASSIMO TERZO

Agile Strategic Consultant  
Agile Coach | Speaker  
Gentiluomo di Fortuna  
Sono rientrato in Italia dopo  
aver lavorato a Londra,  
Parigi e Durham; sono  
appassionato di Agile, in  
particolare di Agile for  
Hardware, Agile at Scale e  
Lean Startup

Il processo di creazione di un prodotto fisico è una cosa affascinante, complicata ed estremamente variabile. Non è un monolite che si sviluppa nella stessa maniera dall'inizio alla conclusione ma richiede anzi un continuo cambio di focus e di comportamento. Per queste ragioni, lo sviluppo di un prodotto fisico deve essere adattabile, e da qui l'importanza di utilizzare un approccio Agile nel senso primo della parola: un approccio che evolve con l'accrescere della maturità del prodotto. Vi sono tre fasi principali nello sviluppo di un prodotto hardware:

1. Una fase di esplorazione (exploration)
2. Una fase di sviluppo prodotto
3. Una fase di industrializzazione

La prima di queste tre fasi, l'**esplorazione**, è molto importante per i prodotti innovativi e serve a capire se ha senso sviluppare un prodotto. La scuola del Lean Startup ci insegna che questa fase è fondamentale per validare il Problem-Solution Fit, ovvero se la soluzione risponde a un vero problema e il Product-Market Fit, ovvero se il prodotto che svilupperemo ha un mercato. La fase di exploration serve a capire se esiste una risposta affermativa a queste domande; nel caso si volesse procedere anche se non sussistono queste due condizioni si rischia di incorrere nel **più grosso degli sprechi, che non è quello di creare la soluzione sbagliata per un problema, bensì di creare la soluzione perfetta che non risolve alcun problema**. In questo caso, infatti, utilizzeremo tempo ed energie per creare un'eccellenza tecnica che non serve al cliente. Questa fase di exploration risulta più estesa e importante per progetti estremamente innovativi, per i quali c'è un'alta incertezza: la prima fase sarà dunque fondamentale identificare le ipotesi su prodotto, mercato, cliente e tecnologia da validare per colmare i vuoti di conoscenza al fine di prendere delle decisioni chiave (**Key Decision Point, KDP**). Per progetti meno innovativi, come l'adattamento di un prodotto esistente a un nuovo mercato o lo sviluppo prodotto a partire da una piattaforma conosciuta, questa fase durerà molto meno.

La seconda fase è lo **sviluppo del prodotto**. A questo punto le ipotesi principali sono state validate, soprattutto quelli che Eric Ries chiama i leap of faith, le ipotesi che possono uccidere un prodotto qualora si rivelassero sbagliate, in particolare quelle sul valore del prodotto (value) e sulla capacità di crescere (growth). Si può iniziare a progettare il prodotto, creare prototipi e prototipi, ordinare pezzi, contattare fornitori etc. Anche in questa fase ci saranno delle ipotesi e dei knowledge gap da chiudere, e parte del lavoro di sviluppo sarà di disegnare ed eseguire esperimenti e test. Questa fase è importante in tutti i progetti, e si

basa sul lavoro svolto nella fase precedente di exploration. Alcuni problemi emergono quando si inizia la fase di sviluppo senza una solida base di esplorazione. Spesso il team inizia a lavorare su un progetto una volta ottenuto il design freeze da parte della Direzione ma non sempre siamo in presenza di una validazione di tutte le ipotesi: in molti casi, al contrario, sarà necessario riprendere in mano il disegno e rivedere certi elementi e decisioni che possono essere basati su ipotesi non corrette o validate. In questa fase si svilupperanno dei **Minimum Viable Product (MVP)** con lo scopo di testare in maniera diretta con il cliente o l'utilizzatore certe funzionalità, per avere dei feedback affidabili e per testare l'integrazione di vari componenti. È da notare che il termine MVP è uno dei più utilizzati nel vocabolario di Agile e spesso ricopre aree semantiche molto lontane fra loro. Non discuterò qui le sfumature che può assumere questa espressione, mi limito solo a dire che nel contesto di questo studio, con MVP si indica un embrione di prodotto che racchiude alcune funzionalità da testare. Un MVP può non essere un prodotto omologato e pronto per il mercato (sappiamo che nel caso di prodotti fisici ottenere questo può richiedere molti mesi di lavoro e specifiche certificazioni) ma che permette comunque di dare una risposta al bisogno tecnico o di un cliente.

Infine, quando il prodotto è stato definito e progettato entriamo nell'ultima fase di creazione, ovvero l'**industrializzazione** che rappresenta il trait d'union con la fase di exploitation rappresentata dai processi produttivi. In questa fase il prodotto ormai definito, progettato e testato nelle sue componenti e nella sua totalità, è pronto per essere studiato dal punto di vista dalla fabbricazione. Arriviamo in questa fase con tutti i disegni finali del prodotto, la lista dei componenti e dei relativi fornitori e con uno o più prototipi validati e corrispondenti al prodotto finale. L'industrializzazione deve occuparsi del fine tuning del prodotto, capire se ci sono delle modifiche da apportare per poterlo produrre in maniera più efficace, veloce, o conveniente. Siamo alla fine del ciclo di creazione di un prodotto, e qui il costo del cambiamento può essere molto elevato perché ogni cambiamento può creare un **effetto domino che coinvolge la riprogettazione e il test di molte componenti**. L'importanza della fase di industrializzazione viene spesso sottovalutata e si tende a considerare un progetto "chiuso" quando la fase di sviluppo finisce. Questo può essere deleterio per il progetto visto nella sua globalità ed è dunque importante considerare la fase di industrializzazione una parte integrante della creazione agile di un prodotto.

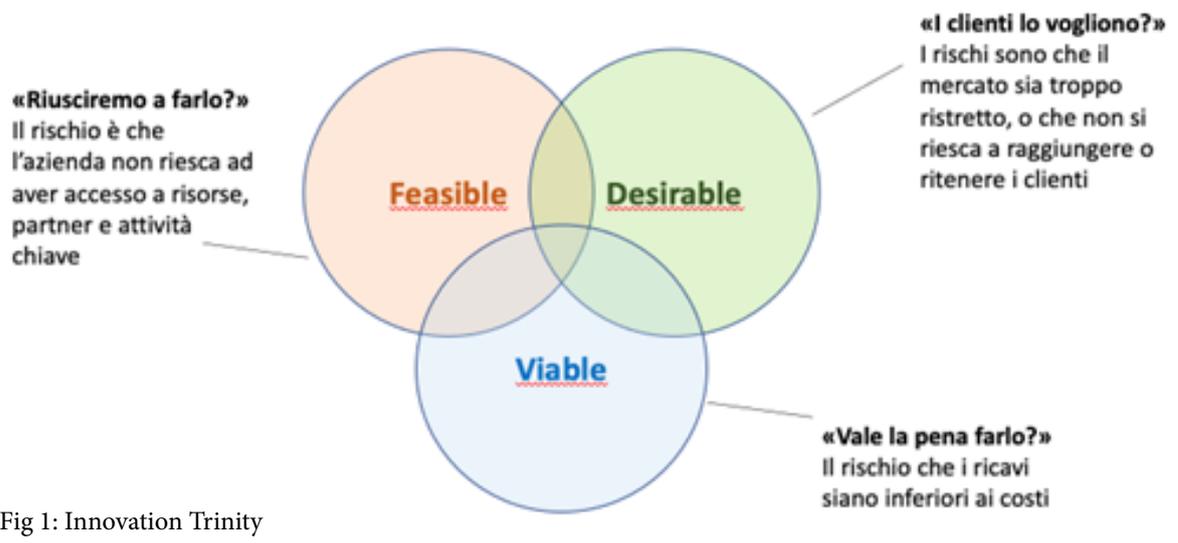


Fig 1: Innovation Trinity

La fase di industrializzazione segna il passaggio da quella che può essere un'ottica prettamente Agile a una Lean<sup>1</sup>. Si tratta di trovare la maniera di definire una sintesi fra efficacia del prodotto ed efficienza del suo processo produttivo. In questa fase avviene l'hand-over dell'ownership di prodotto dal team di ricerca e sviluppo a quello di fabbricazione. Come ogni passaggio di consegne, anche questo è delicato, rischioso, e può creare un'interruzione nel processo di consegna di valore al cliente. Se ci orientiamo alla creazione di prodotto con il modello dell'Innovation Trinity (Figura 1) in questa terza fase ci concentriamo soprattutto sul cerchio della feasibility.

Esaminiamo quali siano gli elementi della fase di industrializzazione: a seconda del prodotto e dell'organizzazione potremmo avere elementi distinti e specifici, in generale mi sento di individuare questi che servono a garantire il fine-tuning del prodotto per renderlo pronto alla produzione. È una sorta di lavoro di lima che serve a garantire un passaggio fluido ai reparti del monozukuri.

- **Tecnologia:** occorre considerare che tipo di tecnologia sarà utilizzata, con i limiti e i benefici che ne derivano. Nella fase di sviluppo del prodotto possono essere utilizzati strumenti come la stampa 3D, materiali sperimentali o altro. Nella fase finale i prototipi vengono solitamente realizzati con i materiali e i componenti che saranno usati in produzione, ma non è escluso l'uso di componenti specifici. Prendiamo, ad esempio, un prodotto fatto in fibra di vetro: molte volte il prodotto è disegnato per lo stile, per avere delle alte performance di aerodinamica o una buona ventilazione ma certi bordi troppo angolati non sono fattibili da alcuni processi di produzione: il risultato è un prodotto fragile o non in linea con l'aspetto estetico atteso.
- **Impianto di produzione:** legato alla scelta della

tecnologia c'è il tema dell'impianto di produzione. Occorre considerare le capacità di produzione dell'impianto, il bilanciamento della linea, il bisogno di manutenzione e i costi di set-up. Le scelte fatte in fase di sviluppo possono avere un forte impatto su queste aree: come si deve organizzare il lavoro di varie stazioni che devono assemblare un sottosistema complesso? Ogni quanto tempo devo cambiare uno stampo? Il team di progetto influenza la maniera con cui il team di produzione lavora.

- **Automazione:** la produzione e l'assemblaggio di certi componenti e sottosistemi può essere automatizzata al fine di ridurre i costi o migliorare la performance di produzione.
- **Formazione degli operatori:** è necessario che il personale di produzione sia formato alla modalità di produzione, ai punti critici del prodotto. A questo fine devono essere prodotti i manuali e le schede di assemblaggio e produzione, installate delle linee pilota e identificate le competenze necessarie. La definizione degli standard è un elemento chiave per permettere una produzione conforme alle richieste del cliente e al disegno del team sviluppo e rappresenta la base di partenza per il miglioramento continuo.
- **Qualità:** è importante identificare quali certificazioni e requisiti sono necessari alla commercializzazione di un prodotto. Mercati differenti richiedono certificazioni e il rispetto di norme differenti. Il team Qualità è inoltre incarica di assicurare che le norme relative alle certificazioni di produzione o altri standard internazionali, di azienda o di gruppo (es. ISO 9001 etc) siano rispettati.

<sup>1</sup> Non sono un gran sostenitore di una divisione manichea fra Agile e Lean, credo che questi due approcci abbiano molto in comune e che spesso sia difficile dissociare uno dall'altro.

Fig 2: Sviluppo prodotto waterfall con team dedicati

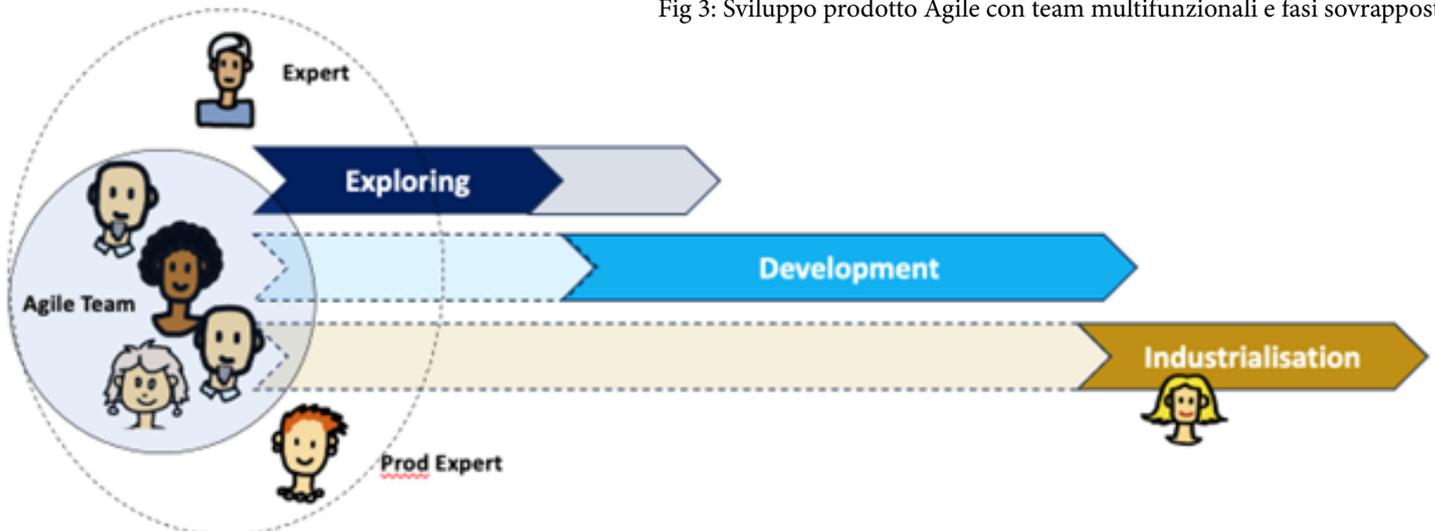


- **Fornitori:** è importante considerare quali parti, moduli e sotto-sistemi saranno presi dall'esterno, quale deve essere la qualità richiesta, quali i lead-time e quali i fornitori alternativi. I fornitori devono essere prima identificati e poi validati. Occorre fare attenzione al fatto che, nel corso della creazione di un prodotto, i materiali utilizzati, e i relativi fornitori cambiano: nelle fasi iniziali, in cui si vuole validare le ipotesi, basterà avere dei componenti semplici che permettano di creare mock-up veloci. Durante la fase di sviluppo si potrà aver bisogno di componenti innovativi, molto costosi, in tempi brevi. Quando il progetto sta per entrare nell'ultima fase di industrializzazione è necessario usare i componenti e i materiali che saranno utilizzati nel prodotto che sarà commercializzato. Tutto ciò implica un'attenta elaborazione della strategia di sourcing.

Per avere una fase di industrializzazione efficace occorre considerare le esigenze della fabbrica dalle prime fasi della creazione del prodotto. Nella fase di exploration, ad esempio, potrà essere utile avere un esperto di produzione che funga da consulente per

il team, identificando quali siano le tecnologie o i vincoli di produzione da considerare. Questo permetterà di identificare delle ipotesi da validare collegate al mondo della fabbricazione. Spesso il personale di fabbrica, soprattutto quando si tratta di esperti con una solida esperienza nel campo, sono risucchiati dall'operatività giornaliera ed è difficile avere disponibile una fetta importante del loro tempo. Il loro contributo al team di sviluppo sarà quasi sicuramente limitato in termini di tempo ma dovrà essere incisivo. Se, idealmente, vorremmo avere una persona della produzione all'interno del team Agile, sarà più probabile averlo come "consulente" a chiamata. Una buona regola è quella di far partecipare gli esperti di produzione alle Review di prodotto: avere una cadenza regolare di questi meeting è di grande aiuto per il team di produzione, sempre coinvolto nel gemba e spesso richiamato per le emergenze. La regolarità di questi eventi aiuta ad assicurare che un membro del team di produzione possa essere presente e fornire il suo prezioso contributo.

Fig 3: Sviluppo prodotto Agile con team multifunzionali e fasi sovrapposte



Quello che può fare ogni team Agile, in ogni caso, è considerare le esigenze della produzione come elementi della creazione di un prodotto:

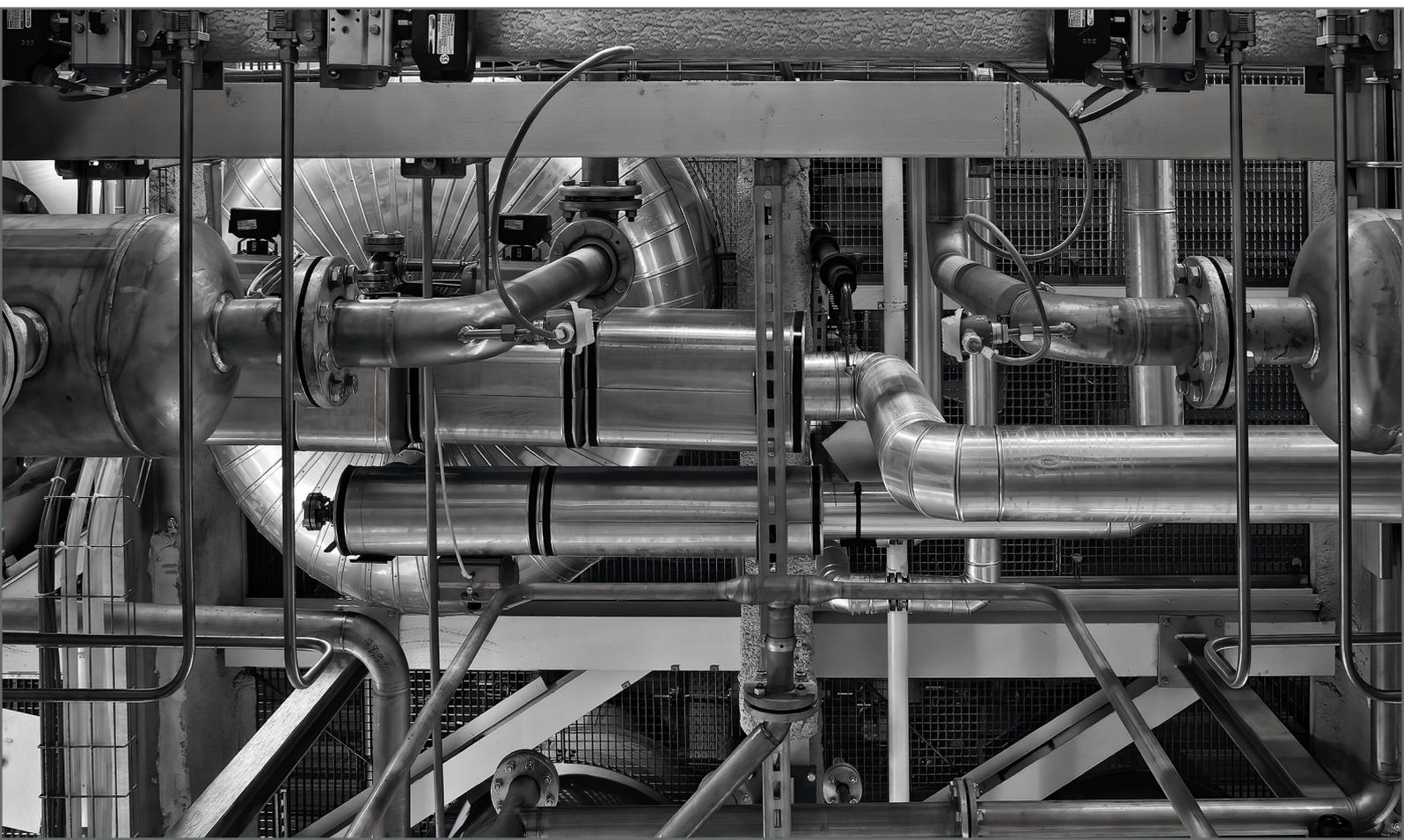
- I vincoli imposti dalle normative o dalle regolamentazioni possono diventare dei veri e propri criteri di accettazione che vincolano certe scelte di design.
- Includere la producibilità fra le ipotesi da testare: è possibile utilizzare il concurrent design per scegliere i componenti e i moduli che possono essere prodotti più facilmente ed economicamente. Nel caso del prodotto che richiede lo stampaggio di plastiche citato in precedenza, è possibile fare dei test per capire come un processo produttivo traduce il progetto generato dal team di sviluppo: con dei test rapidi potremo capire se una certa forma è producibile o meno.
- Definire dall'inizio la strategia di sourcing al fine di pilotare e far evolvere il contributo dei singoli fornitori nel corso della creazione del prodotto. Cercheremo vari fornitori, ognuno dei quali possa supportare una specifica fase della creazione del prodotto. Già dalle prime fasi inizieremo uno scouting delle possibili alternative che affineremo con l'avanzamento del progetto.
- Abbattere i silos fra le funzioni di sviluppo prodotto e gli attori a valle come la produzione, il servizio assistenza e i fornitori che possono offrire interessanti prospettive per la creazione di un prodotto innovativo.
- Testare i processi di produzione su impianti pilota in scale ridotta che permettano di identificare i potenziali problemi, In questa maniera sarà possibile anche capire il miglior approccio per formare

gli operatori. In un progetto in cui sto lavorando, ad esempio, è stato possibile utilizzare una prima serie di prototipi per iniziare a formare il personale e testare alcune fasi di produzione come la verniciatura e l'assemblaggio di componenti.

- È possibile utilizzare un approccio di tipo **Test Drive Development (TDD)** in maniera da incorporare i requisiti della qualità nella stessa definizione di feature di prodotto. In questa maniera il lavoro del team Qualità viene agevolato se non addirittura anticipato alla fase di sviluppo prodotto.

In molti contesti hardware, è difficilmente implementabile una logica di tipo Devops, in cui il team di sviluppo prodotto si occupa anche della produzione. È possibile, tuttavia, integrare le attività dell'industrializzazione nelle prime fasi di un progetto considerarle come dei deliverable che portano alla maturità del prodotto.

In un contesto di creazione prodotto secondo un approccio Agile le tre fasi che ho menzionato all'inizio sono sovrapposte e c'è un coinvolgimento di tutte le figure che possono portare un contributo nello sviluppo della conoscenza che serve a prendere le giuste decisioni. Il vantaggio rispetto a un approccio waterfall non risiede nella velocità intrinseca di un approccio rispetto ad un altro, bensì nel poter colmare i gap di conoscenza al fine di creare un prodotto efficace, senza funzionalità non richieste dal cliente e con una maggiore consapevolezza di come ci si stia muovendo all'interno dell'Innovation Trinity.





Oggi è il giorno giusto per **promuovere lo sviluppo personale.**

Formazione online per tutta la tua azienda.

Attiva una prova gratuita di 14 giorni!



**goodhabit**  
online training

TUTTE LE SETTIMANE

# AGILE FOR ITALY LEAN BEER

## PODCAST



SCAN ME

SU SPOTIFY, SPREAKER, APPLE PODCAST, GOOGLE PODCAST



DAVIDE CASARI

TIZIANO INTERLANDI

PIERPAOLO CIMIRRO





## Verona Handcrafted Gin

non ti deluderà, è IL distillato fuori dal comune.

Un Gin diverso che emana sensazioni e profumi mai trovati in nessun altro Gin.

Come produttori di gin abbiamo scelto il metodo LONDON:

Vhagin è un prodotto naturale al 100%, la macerazione delle spezie richiede tempi molto lunghi e solo alla fine di questo processo il gin può essere distillato.

Per disciplinare non può essere aggiunta nessuna correzione dopo la distillazione e trae grande beneficio da un po' di riposo in bottiglia.

Nella giungla dei gin in commercio non è sempre facile orientarsi.

Ogni volta che scegli un gin ricordati di cercare sull'etichetta la dicitura London Gin: Primo e facile criterio per distinguere un GIN di qualità.

Vuoi capire meglio di cosa stiamo parlando?

London Gin: il termine London non si riferisce al luogo di origine, ma al processo produttivo.

Per disciplinare il London Gin, secondo il Reg. CE 110/2008, deve essere ottenuto esclusivamente da alcol etilico di origine agricola, il cui aroma è dovuto esclusivamente alla ridistillazione di alcol etilico in una singola seduta (one-shot) ed in alambicchi tradizionali con presenza di materiali esclusivamente vegetali naturali.

Identico ad un Gin distillato, nel London Gin l'unica variante è che non può esserci aggiunta di nessun tipo di elementi artificiali quali aromi e coloranti.

Il London Gin è trasparente, secco, profumato e deve avere titolo alcolometrico volumico minimo di 37,5% vol.

[www.vhagin.com](http://www.vhagin.com) – [hello@vhagin.com](mailto:hello@vhagin.com)



[www.vhagin.com](http://www.vhagin.com) – [hello@vhagin.com](mailto:hello@vhagin.com)

Ci vediamo  
l'11 settembre

Thank  
you!



# COPYRIGHT E DINTORNI

Come possiamo conciliare diritto d'autore, informazione libera e collaborazione? Per noi è stato un vero e proprio dilemma che cerchiamo di colmare con queste poche righe di contesto e agreements.

In questa rivista trovate tutta una serie di contenuti che possiamo dividere in:

- articoli scritti da autori
- recensioni di libri
- pagine di eventi
- pagine di community
- consigli di acquisto di libri
- patrocinio di aziende

Ma partiamo con ordine.

## Scopo di AgileItalia

AgileItalia è una rivista online gratuita finalizzata alla divulgazione di Agile in lingua italiana. La rivista nasce nel 2019 come iniziativa complementare ad AgileForItaly da parte di Tiziano Interlandi, Pierpaolo Cimirro e Davide Casari.

Lo scopo di AgileItalia è la pura divulgazione di teorie, pratiche, framework del mondo Agile, con particolare focus rispetto alle esperienze pratiche. AgileItalia vuole riportare la discussione/argomentazione sulla realtà di tutti i giorni.

## Gli autori

Gli autori che partecipano ad AgileItalia hanno la totale ownership del diritto d'autore ed anche la piena responsabilità delle proprie affermazioni negli articoli. Gli editori (al momento Tiziano In-

terlandi, Pierpaolo Cimirro, Davide Casari) si riservano di accettare, non accettare o correggere gli articoli proposti, rendendoli conformi ai principi di "buona educazione", correttezza nei confronti di altri interlocutori, linguaggio non violento e rispetto del diritto di autore. Gli autori possono o meno accettare le correttive proposte e quindi optare anche per una non pubblicazione. Il diritto d'autore è assolutamente proprietà dell'autore proponente. Gli articoli non verranno ceduti a terzi (non fa parte dello scopo di AgileItalia). Gli articoli verranno pubblicati su file (pdf, epub ed altri), su piattaforma pubhtml ed in formato plaintext su \*.agileforitaly.com in modo da accedere ad una miglior categorizzazione e indicizzazione da parte dei motori di ricerca. Gli editori si riservano la possibilità di ripubblicare gli articoli in raccolte (ad esempio antologie annuali). Gli autori accettano questi punti proponendosi come autori. AgileItalia non viene stampato e distribuito. E' gratuito. Eventuali stampe sono da considerarsi come utilizzo privato e non afferiscono assolutamente all'ambito commerciale. I numeri, gli articoli e gli autori verranno pubblicati sui canali social di AgileItalia (Facebook, LinkedIn, Twitter, Slack, Telegram). AgileItalia utilizza solo dati pubblici. Non verranno indicati informazioni sensibili. Le uniche informazioni saranno quelle pubbliche (che l'autore pubblica su internet o su social) o eventuali altri riferimenti di contatto solo se indicate espressamente dall'autore. La collaborazione tra AgileItalia e autori è sempre da considerarsi a titolo gratuito. Gli autori non possono chiedere il ritiro o la cancellazione di un loro articolo successivamente alla pubblicazione.

## Condivisione dei numeri e degli articoli

E' vietata l'estrapolazione di parti di PDF e la condivisione parziale su social e altre piattaforme diverse da quella di AgileItalia. AgileItalia è una rivista gratuita e portata avanti da volontari, riteniamo corretto che venga sempre riportata la corretta filiera del copyright. Potrete sempre ricondividere articoli e rivista ma attraverso link al sito ufficiale e non tramite ricaricamento di parti di esso e del PDF. Questo ci permette di capire quanti sono i reali fruitori della rivista, permettendoci quindi le giuste azioni marketing e di auto-finanziamento.

## Formato della rivista e tempistica di consegna articoli

Il formato della rivista viene deciso dagli editori. Ogni numero contiene mediamente 60 pagine. L'autore accetta il formato proposto dagli autori. L'autore proponendosi accetta implicitamente il formato avendo approfondito il taglio, lo spirito e le finalità dei numeri precedenti. AgileItalia è bimestrale ed esce preferibilmente il 9 dei seguenti mesi: gennaio, marzo, maggio, luglio, settembre, novembre. La data di consegna degli articoli è entro e non oltre il 27 del mese precedente all'uscita, ovvero: 27 dicembre per il 9 gennaio, 27 febbraio per il 9 marzo, 27 aprile per il 9 maggio, 27 giugno per il 9 luglio, 27 agosto per il 9 settembre, 27 ottobre per il 9 novembre. Queste scadenze sono necessarie per l'impaginazione degli articoli e la preparazione del sito <https://agileitalia.agileforitaly.com>. AgileItalia si impegna a dare agli autori una versione Draft entro 5 giorni prima rispetto alla data di uscita in modo da controllare eventuali refusi. Il Draft verrà inviato agli autori i quali dovranno effettuare un controllo del proprio articolo e fornire un feedback agli editori in modo da creare una situazione win-win da parte di entrambe le parti. In AgileItalia possono esistere interviste curate da un intervistatore: in questo caso verrà indicato l'intervistato mentre la sezione sarà "a cura" dell'intervistatore. Intervistatore ed intervistato hanno precedentemente concordato su quanto scritto. NON sono accettati patrocini di aziende o organizzazioni tali da offuscare o deviare dallo scopo di AgileItalia. Le aziende ed organizzazioni possono essere citate e raccontate ma solo se propedeutiche al racconto dell'autore. La responsabilità di tali citazioni è a carico dell'autore. Riteniamo che sia una questione di correttezza nei confronti di editori/volontari, sponsor che aiutano in maniera trasparente la rivista e lettori. Gli Editori di AgileItalia si riservano di non accettare articoli che rientrano in questa categoria, proponendo eventualmente una variazione all'autore. In AgileItalia NON esistono articoli frutto di accordi commerciali con terzi. NON esiste quindi spazio per articoli proposti in cambio di un pagamento, in qualsiasi sua forma. Pensiamo che sia corretto proporre contenuti genuini, dal basso e scevri di compravendita che possano far passare messaggi scorretti come corretti. In AgileItalia non c'è quindi spazio per aziende o organizzazioni che vogliono raccontare unilateralmente la loro versione di Agile pagando uno spazio pubblicitario. Il senso di AgileItalia è quello della rivista scientifica con il meccanismo di una revisione da parte di un esperto esterno. Qualora si evidenziasse post pubblicazione la presenza di articoli che non sono aderenti a queste specifiche, AgileItalia potrà rimuovere gli articoli o parti interessate per poi ripubblicare.

## Sponsor e temi economici

AgileItalia è gratuito e senza scopo di lucro. AgileItalia accetta sostegni liberali nella forma di servizi e strumenti. Un esempio può essere l'utilizzo di immagini (unsplash), utilizzo di licenze software o altri servizi online. Gli editori di AgileItalia non recepiscono denaro da parte di sponsor relegando questa attività editoriale alla pura divulgazione gratuita. Gli Editori mettono a disposizione il proprio tempo libero e la propria esperienza in maniera gratuita senza percepire compenso da nessuno.

## Utilizzo di immagini

Riuscire a certificare la provenienza di immagini e del loro diritto d'autore è molto complesso. Per tale motivo ed a seguito di alcuni spiacevoli eventi, AgileItalia, al momento, accetta esclusivamente immagini provenienti da Unsplash e con libertà di utilizzo. Altre immagini, ad esempio foto di eventi o board, devono essere anonime e non contenere informazioni sensibili (sia nel testo che nella presenza di persone riconoscibili). L'utilizzo di questa categoria di immagini è responsabilità dell'autore proponente e libera gli editori di AgileItalia da ogni responsabilità. Le immagini personali degli autori sono state consegnate dagli stessi ed accettano implicitamente di utilizzare la propria immagine in questa rivista.

## Riferimenti, peer review e onestà intellettuale

E' molto difficile per gli editori mettere in pratica un controllo così rigido da poter intercettare ogni singolo riferimento negli articoli. E' richiesto all'autore l'onestà intellettuale di indicare eventuali fonti e riferimenti come avviene nella comunità scientifica. La peer review potrà essere praticata dagli editori o da altri attori terzi che prestano questa attività gratuitamente.

## Eventi, libri e altri placement

Nella rivista troverete diverse pagine di eventi, libri ed organizzazioni frutto di semplice volontà di dare visibilità a iniziative di valore o tramite partnership gratuite atte alla crescita delle parti coinvolte.

Ad esclusione delle pagine iniziali del patrocinante Sintegra tutto il resto delle pagine di placement ricade in questa casistica.

Può succedere che ad AgileItalia vengano inviate copie di libri per recensioni, in tal caso ci impegniamo ad essere oggettivi, se un libro o un evento non ci sono piaciuti lo indicheremo. Anche in questo caso non esiste un compenso percepito.

## Perché non usiamo una Licenza Creative Common?

L'applicazione di una licenza sopra altre forme di licenza è difficile. Non possiamo ad esempio applicare la licenza CC su immagini Unsplash, già in possesso di una propria licenza.

Per sopperire a questa complessità la pagina successiva contiene un riassunto completo di immagini ed articoli per pagina/argomento/categoria al fine di congelare la situazione dei diritti di autore al momento della pubblicazione. La marca temporale viene garantita attraverso una pec in nostro possesso.

# la situazione al 11 settembre 2023

Pagina	Tipo	Diritti	Note	URL
1	Logo	Sintegra Srl		
1	Logo	AgileForItaly		
1	Immagine	"		
Florian Klauer"		Unsplash Licence		<a href="https://unsplash.com/it/foto/mk7D-4UCfmg">https://unsplash.com/it/foto/mk7D-4UCfmg</a>
2	Immagine	Kourosh Qaffari		<a href="https://unsplash.com/it/foto/RrhhzitYizg">https://unsplash.com/it/foto/RrhhzitYizg</a>
2	Immagine	Sebastian Herrmann		<a href="https://unsplash.com/it/foto/AY3cdYQJ4xU">https://unsplash.com/it/foto/AY3cdYQJ4xU</a>
2	Immagine	Samuel Sianipar		<a href="https://unsplash.com/it/foto/4TND3hsW3PM">https://unsplash.com/it/foto/4TND3hsW3PM</a>
3	Logo	Sintegra Srl		
3	Logo	Vhagin (Davide Casari)		
3	Logo	Unsplash	ringraziamento	
3	Logo	Sessionize	ringraziamento	
3	Logo	AgileForItaly		
4	Sponsor	Sintegra Srl	Pagine Intere Inviata	
5	Sponsor	Sintegra Srl	Pagine Intere Inviata	
6	Sponsor	Sintegra Srl	Pagine Intere Inviata	
7	Sponsor	Sintegra Srl	Pagine Intere Inviata	
8	Logo	Sintegra Srl		
8	Logo	Vhagin (Davide Casari)		
8	Logo	Unsplash	ringraziamento	
8	Logo	Sessionize	ringraziamento	
8	Logo	AgileForItaly		
9	Logo	Sessionize		
9	Logo	Gmail - Google		
9	Logo	LinkedIn		
10	Logo	AgileForItaly		
10	Articolo	Tiziano Interlandi	Inviato	
11	Patrocinio	AgileForItaly		
12	Patrocinio	Erika Melis	Pagina inserita come ringraziamento all'articolo creato da Erika Melis	
13	Patrocinio	Ayros - Claudio Saurin	Pagina inserita come ringraziamento per l'amicizia e gli articoli di Claudio Saurin	
14	Articolo	Giorgio Irtino	Inviato	
14	Immagine	Kourosh Qaffari		<a href="https://unsplash.com/it/foto/RrhhzitYizg">https://unsplash.com/it/foto/RrhhzitYizg</a>
16	Immagine	Kourosh Qaffari		<a href="https://unsplash.com/it/foto/RrhhzitYizg">https://unsplash.com/it/foto/RrhhzitYizg</a>
17	Articolo	Francesco Nardoni	Inviato	
17	Immagine	Sebastian Herrmann		<a href="https://unsplash.com/it/foto/AY3cdYQJ4xU">https://unsplash.com/it/foto/AY3cdYQJ4xU</a>
19	Immagine	Caleb Jones		<a href="https://unsplash.com/it/foto/3JMyXWQHxU">https://unsplash.com/it/foto/3JMyXWQHxU</a>
20	Immagine	Caleb Jones		<a href="https://unsplash.com/it/foto/3JMyXWQHxU">https://unsplash.com/it/foto/3JMyXWQHxU</a>
21	Patrocinio	Product Management Day	Inviato	
22	Patrocinio	Guerini Next	Pagina inserita come ringraziamento alla collaborazione nel numero	
23	Articolo	Marco Pecoraro	Inviato	
23	Immagine	Aaron Burden		<a href="https://unsplash.com/it/foto/NXt5PrOb_7U">https://unsplash.com/it/foto/NXt5PrOb_7U</a>
25	Immagine	Tamas Tuzes-Katai		<a href="https://unsplash.com/it/foto/rEn-AdBr3Ig">https://unsplash.com/it/foto/rEn-AdBr3Ig</a>
26	Articolo	Susanna Ferrario	Inviato	
26	Immagine	Jack Finnigan		<a href="https://unsplash.com/it/foto/6WYwCwZbraE">https://unsplash.com/it/foto/6WYwCwZbraE</a>
27	Immagine	Jack Finnigan		<a href="https://unsplash.com/it/foto/6WYwCwZbraE">https://unsplash.com/it/foto/6WYwCwZbraE</a>
28	Immagine	Jack Finnigan		<a href="https://unsplash.com/it/foto/6WYwCwZbraE">https://unsplash.com/it/foto/6WYwCwZbraE</a>
30	Articolo	Rosa Linda Romano	Inviato	
30	Immagine	Jason Goodman		<a href="https://unsplash.com/it/foto/vbxyFxlqjM">https://unsplash.com/it/foto/vbxyFxlqjM</a>
33	Immagine	Rosa Linda Romano	Inviata	
34	Immagine	Samuel Sianipar		<a href="https://unsplash.com/it/foto/4TND3hsW3PM">https://unsplash.com/it/foto/4TND3hsW3PM</a>
35	Patrocinio	Italian Agile Movement	evento IAD 2023	
36	Articolo	Luca Cianci	Inviato	
36	Immagine	Felix Mittermeier		<a href="https://unsplash.com/it/foto/nAjil1z3eLk">https://unsplash.com/it/foto/nAjil1z3eLk</a>
37	Immagine	Luca Cianci	Inviata	
38	Immagine	Luca Cianci	Inviata	
39	Immagine	Luca Cianci	Inviata	
40	Evento	Evento creato da AgileItalia	Partnership tra community	
40	Logo	Scrum Agile Milano SAM - Corrado De Sanctis	Partnership tra community	
40	Logo	AgileForItaly	Partnership tra community	
40	Logo	Agile Marketing Italia - Deborah Ghisolfi	Partnership tra community	
40	Logo	Italian Agile Movement	Partnership tra community	
40	Logo	Facilitalab - Monica Margoni	Partnership tra community	
40	Logo	Grusp	Partnership tra community	
41	Evento	Evento creato da AgileItalia	Partnership tra community	
42	Articolo	Gianluca Colonnese	Inviato	
42	Immagine	Ahmed Zayan		<a href="https://unsplash.com/it/foto/28nzltYzCDM">https://unsplash.com/it/foto/28nzltYzCDM</a>
44	Patrocinio	QRP International	Pagina inserita come scambio di articoli tra AgileItalia e QRP International	
45	Patrocinio	guerini Next		
45	Immagine	Guerini Next	Immagine del libro	
46	Articolo	Marco Calzolari	Inviato	
46	Immagine	Viktor Forgacs		<a href="https://unsplash.com/it/foto/7C5N8yLaeDI">https://unsplash.com/it/foto/7C5N8yLaeDI</a>
49	Patrocinio	Gruppo Galgano	Pagina inserita come ringraziamento alla collaborazione nel numero "Lean"	
50	Articolo	Tiziano Interlandi	Inviato	
50	Immagine	Tiziano Interlandi	Inviata	
53	Articolo	Massimo Terzo	Inviato	
53	Immagine	Ricardo Gomez Angel		<a href="https://unsplash.com/it/foto/41X6FwTwPh4">https://unsplash.com/it/foto/41X6FwTwPh4</a>
54	Immagine	Massimo Terzo	Inviata	
55	Immagine	Massimo Terzo	Inviata	
55	Immagine	Massimo Terzo	Inviata	
56	Immagine	Ricardo Gomez Angel		<a href="https://unsplash.com/it/foto/41X6FwTwPh4">https://unsplash.com/it/foto/41X6FwTwPh4</a>
57	Patrocinio	Goodhabits	Pagina di ringraziamento per la collaborazione e gli articoli	
58	Patrocinio	AgileForItaly		
59	Patrocinio	Davide Casari	Pagina inserita come scambio di articoli tra AgileItalia e Davide Casari	
60	Immagine	Norvanic Lee		<a href="https://unsplash.com/photos/GiUJ02Yj_io">https://unsplash.com/photos/GiUJ02Yj_io</a>
61	Immagine	Umberto		<a href="https://unsplash.com/photos/1hJrm1BRVV0">https://unsplash.com/photos/1hJrm1BRVV0</a>