

Il Framework FCO, un Metodo per l'Agile Transformation in tre step: Formazione, Coaching e Outcome

A. Ingrosso*, A. Di Girolamo*, S. Bonasegale**, M. Semprini**, A. Pozzoli**, C. Fioretti**, L. Taurino**, M. Spanò**

* Agile Coach indipendenti

** Progetti e Soluzioni SpA

Pubblicato: Aprile 2021

Introduzione- Il termine Agile è oggi frequentemente utilizzato ogni qual volta ci si trova di fronte alla necessità o all'obbligo di dover cambiare rapidamente i nostri comportamenti. L'agilità è richiesta quando non è possibile rispondere con la tradizionale frase "abbiamo sempre fatto così".

Sono ogni giorno di più le aziende che in Italia e nel mondo provano ad avviare progetti di adozione di metodologie Agile. Sono utilizzati approcci e metodi differenti, esistono numerosi framework, ma non esiste una formula certa per garantire il successo di queste iniziative.

L'articolo si apre con una panoramica sulle trasformazioni Agile, sulle esperienze fatte sino ad oggi e sui diversi approcci di trasformazione Agile tentati per cambiare il modo di lavorare delle organizzazioni in favore dei benefici che è possibile ottenere con un Agile Mindset e un Agile Framework.

L'articolo nasce da un'esperienza reale di Agile Transformation attuata attraverso un progetto pilota per lo sviluppo di un prodotto software denominato Omnibus. L'azienda proprietaria del prodotto, Progetti e Soluzioni SpA, dopo aver intrapreso lo sviluppo con metodo waterfall, rivelatosi in seguito inadeguato e deludente, ha deciso di modificare completamente direzione scegliendo un approccio Agile. Nel ricercare un metodo di Agile Transformation che potesse funzionare bene nel contesto aziendale è emerso un nuovo framework denominato FCO basato su tre step fondamentali: formazione, coaching e outcome. Il framework FCO include un approccio che esalta i diversi benefici e punti di attenzione evidenziati in diversi articoli disponibili in letteratura e successivamente citati.

Il framework FCO ha sostenuto la road map di trasformazione Agile del progetto pilota Omnibus, consentendo la creazione di valore misurabile per i clienti e gli utenti del prodotto, e non ultimo creando valore all'interno del team che molto rapidamente è divenuto un team ad alte prestazioni.

Visto il successo dell'iniziativa fortemente basato sull'impegno, la motivazione e la determinazione delle persone coinvolte, nell'ultima parte dell'articolo ciascuna delle persone coinvolte ha condiviso una propria riflessione di come Agile abbia cambiato

drasticamente il paradigma del lavoro collaborativo e abbia facilmente riportato al centro il cliente.

I. IL QUADRO DI RIFERIMENTO DELLE TRASFORMAZIONI AGILE

Con il termine inglese Agile (si pronuncia "Agiail") intendiamo quell'insieme di valori, principi, metodologie, pratiche e tecniche che, come descritto da Jeff Sutherland e Ken Schwaber: "aiutano persone, team e organizzazioni a generare valore attraverso soluzioni adattive per problemi complessi" [1].

Ad originare tale rivoluzione nel modo di pensare e di affrontare i problemi complessi c'è l'ormai celebre Agile Manifesto for Software Development [2], i cui 4 valori e 12 principi hanno ispirato milioni di persone nel mondo e la nascita di oltre 80 metodologie e framework; l'Agile Manifesto può essere considerato come la base della terza ondata Agile che si presenta sotto il nome di Business Agility [3].

La Business Agility vede il coinvolgimento dell'intera organizzazione; Fabio Lisca [3] la descrive come la capacità di un'organizzazione di cambiare rapidamente strategia, struttura, processi, ruoli, competenze e tecnologie per cogliere le opportunità e consegnare maggior valore al cliente e a tutti gli stakeholder.

Una persona non avvezza all'approccio Agile, potrebbe pensare che la Business Agility altro non è che una metodologia, un insieme di processi integrati da implementare in azienda, in sintesi basta "fare" Agile per essere Agile. Questo modo di implementare Agile porta a minimi vantaggi e in alcuni casi al fallimento dell'iniziativa.

L'approccio Agile è molto di più del semplice "fare", è un modo di pensare, di essere: Agile è un **Mindset** e non è possibile realizzare la Business Agility senza lavorare sulla cultura organizzativa, sulla cultura dei singoli, sulla resistenza al cambiamento, sulla comunicazione, e in generale sulle soft skill. Solo attraverso questo cambiamento nel "software" aziendale è possibile "fare" e far funzionare Agile, diversamente continueremo a lavorare come abbiamo sempre fatto, all'interno di una scatola con l'etichetta Agile, ma non "Agile Inside".

Quando parliamo del solo fare Agile ci stiamo riferendo alla sua adozione, volta quindi al cambiamento dei processi affinché siano allineati ai valori e principi Agile, come per esempio utilizzare Scrum per lo sviluppo del software o modificare i processi in modo da essere compliance con quanto descritto nella Guida Scrum; il resto dell'organizzazione, le funzioni aziendali, i processi di sviluppo restano di fatto invariati e non interessati dal cambiamento; questo, nella migliore delle ipotesi, porta alla creazione in azienda di due anime che viaggiano a velocità diverse, sono guidate da motivazioni diverse e parlano lingue diverse; nella peggiore delle ipotesi, la pura e semplice "installazione" di nuovi strumenti operativi senza interventi sulla cultura e sulle relazioni organizzative, porterà a un sostanziale fallimento dell'iniziativa.

Al contrario quando parliamo di Agile Mindset (fare + essere) ci stiamo riferendo alla trasformazione dell'intera organizzazione: cambia la cultura, cambia la burocrazia, cambia la reason why. [4][5]

In sostanza, se l'adozione di Agile cambia il modo di fare le cose, la trasformazione Agile cambia il modo di essere: non ci riferiamo soltanto al modo di essere "azienda" ma al modo di "essere" persone che interagiscono, che collaborano, che generano valore.

Vista la complessità e l'imprevedibile impatto che può avere sull'organizzazione, la trasformazione Agile di un'azienda è condotta – ovviamente – con un approccio Agile e la strategia che vede tutti d'accordo è quella di procedere attraverso lo sviluppo di un progetto pilota [3].

Nel 2016 Gandomani e Nafchi [6] hanno condotto una ricerca che ha visto il coinvolgimento di 49 professionisti Agile in 13 differenti Paesi, con l'obiettivo di esplorare quali siano le origini e le ragioni che rendono altamente sfidante una trasformazione Agile.

La natura di Agile è centrata sulle persone, questo pone diverse sfide e ostacoli lungo il percorso di trasformazione, tra cui:

- Mancanza di conoscenza;
- Questioni culturali;
- Resistenza al cambiamento;
- Mentalità sbagliata;
- Mancanza di collaborazione efficace.

Gandomani e Nafchi aggiungono a queste sfide anche quelle legate alla percezione che le persone hanno del processo di cambiamento:

- Alcune persone sono preoccupate all'idea di cambiare;
- Altre sono entusiasmata ma non hanno realmente compreso di cosa si tratta;
- Altre non credono più in generale nei benefici del cambiamento, ne hanno visti troppi;
- Ci sono gli indifferenti;
- Infine, c'è chi ha aspettative non realistiche.

La trasformazione e adozione di Agile nello sviluppo di progetti e prodotti, implica il passaggio dall'uso di metodi tradizionali di tipo predittivo a metodi empirici di tipo adattivo.

Tutto ciò richiede un elevato sforzo e può essere necessario molto tempo [7][8]: l'Agile Transition Process (ATP) è una vera e propria rivoluzione nelle aziende e pertanto necessita di un'elevata collaborazione e coinvolgimento da parte dei principali stakeholder (top e middle management, utenti, clienti e sviluppatori), il che significa, tra le altre cose, che tutti i soggetti coinvolti devono poter maturare una serie di competenze trasversali indispensabili: la capacità di lavorare in modo collaborativo, la capacità di comunicare assertivamente, la capacità di scambiare feedback continui, la capacità di abbandonare approcci competitivi e infine la capacità di adottare comportamenti ispirati da una logica WIN WIN.

Molti studi hanno messo in evidenza che non esiste un unico modello ATP, uno standard, o una best-practice da cui fare copia e incolla [9][10][11][12][13][14][15][16]: ogni azienda rappresenta un caso unico di persone, relazioni, storie, contesti, norme e cultura organizzativa, che costituiscono un tessuto ad alta complessità e che – in quanto tale – non può essere gestito con approcci standardizzati, né tanto meno è possibile prevedere quale sia il miglior processo di transizione e la durata dello stesso [17][18][11].

Gandomani e Nafchi [19] nel loro studio empirico relativo al fenomeno delle trasformazioni Agile, hanno identificato due ambiti ben distinti: le **caratteristiche strutturali** e le **attività chiave**.

Le caratteristiche strutturali sono considerate essenziali nella trasformazione, la rendono meno costosa e impegnativa, e più efficace.

La trasformazione Agile dovrebbe considerare le seguenti caratteristiche strutturali e portare l'organizzazione a riflettere sui seguenti punti:

- **Basata sul valore**, inteso come valore per il business, risponde alla domanda: *perché stiamo adottando Agile?* È necessario definire quale sarà il valore atteso e come verrà misurato prima di iniziare la trasformazione.
- **Iterativa**, ovvero procedere per iterazioni adottando una pratica per volta secondo la logica "utilizzare Agile per implementare Agile"; Agile agisce primariamente sui comportamenti delle persone ma questi cambiamenti richiedono tempo, impegno e dedizione; potrebbe essere utile, in questa sede, riportare una frase utilizzata in Toyota: "Building people before building cars".
- **Continuativa**, nel senso che le iterazioni devono ripetersi in modo continuo senza interruzioni, ricordando che l'agilità non ha uno "stato finale" da raggiungere ma è miglioramento continuo: la transizione Agile è un viaggio, non una destinazione.
- **Graduale**, perché alcuni comportamenti richiedono maggior tempo per poter essere attuati; molte aziende non sono immediatamente pronte a cambiare, e non è necessario partire con un approccio di tipo Big Bang: procedere in modo graduale riduce i rischi, le sfide ed è un aiuto per tutti quelli che non sono pronti alla transizione Agile o sono ad essa avversi.

La trasformazione Agile include le seguenti attività chiave che si ripetono ad ogni iterazione:

- **Selezione delle pratiche:** guidati dal principio di creazione del valore, le pratiche Agile da adottare possono essere ordinate per priorità e selezionate ad ogni iterazione.
- **Adattamento:** l'adozione di Agile deve essere sostenibile ed è necessario ricordare che l'adattamento non è un'attività predittiva poiché ha a che fare con il cambiamento del mindset delle persone. Costringere le persone a cambiare è il peggior modo di adottare Agile, se non il modo di **non** adottarlo. Aumentare significativamente il successo della trasformazione è possibile: è stato dimostrato che ciò è favorito dalla istituzionalizzazione dell'adozione di Agile insieme al supporto di coach e mentor.
- **Valutazione:** questa attività può aiutare a identificare le sfide che il team sta affrontando, monitorare i processi e i progressi della transizione. Può riguardare diversi aspetti tra i quali: il ritmo produttivo, la durata delle iterazioni, la dimensione del team, il livello di collaborazione, la formazione, la leadership, l'autogestione del team, i benefici per il business, ecc. Agile richiede l'uso di nuove metriche orientate alla misurazione dei benefici creati in termini di business, il che significa che le metriche tradizionali non funzionano.
- **Retrospectiva:** i risultati ottenuti dalla valutazione devono essere rivisti e discussi dal team durante specifici incontri di retrospectiva; la retrospectiva si svolge ad intervalli regolari e ha come obiettivo quello di identificare le aree di miglioramento su cui intervenire a partire dall'iterazione successiva.
- **Adeguamento:** ha come obiettivo quello di adeguarsi e adattarsi alle pratiche Agile correnti prima di prenderne una nuova.

Che sia un prodotto da sviluppare da zero o un cambiamento in corsa, solo le persone possono decidere se avere o meno un approccio Agile nello sviluppo del prodotto.

II. LA CREAZIONE DEL FRAMEWORK FCO PER RISPONDERE ALL'AGILE TRANSITION DI PROGETTI E SOLUZIONI SPA SUL PROGETTO OMNIBUS

Il progetto OMNIBUS è stato seguito da due Agile Coach, Anna Di Girolamo e Alessandro Ingrosso, che insieme hanno sviluppato l'idea di una doppia conduzione delle attività di formazione e di coaching; data l'innumerabile quantità di documenti e testimonianze che raccontavano come la formazione del mindset fosse indispensabile per poter utilizzare produttivamente (al massimo del suo potenziale) il framework, Anna ed Alessandro decidono di strutturare degli interventi che consentano ai team di lavorare tanto sull'*essere* quanto sul *fare*.

Le competenze dei due Coach sono diverse ma complementari: Anna Di Girolamo proviene dalla comunicazione, dal marketing e dalle soft skill (e si occupa prevalentemente del mindset, inteso

come *essere Agile*), Alessandro Ingrosso è un ingegnere, ha una formazione tecnica e una solida esperienza in ambito IT (e si occupa prevalentemente degli strumenti, intesi come *fare Agile*). Gli studi e le esperienze riportati in letteratura (e sopra citati) hanno ispirato il modello di transizione Agile studiato per il progetto pilota Omnibus.

Il risultato è stato la creazione di un framework di trasformazione Agile iterativo, adattivo ed empirico costituito da tre fasi principali: **Formazione, Coaching e Outcome** (figura 1).

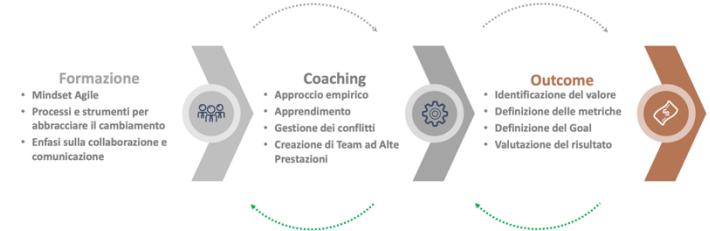


Figura 1 - Framework di trasformazione Agile iterativo, adattivo ed empirico.

Le fasi sono principalmente sequenziali; alla fine di ciascuna fase è previsto un momento di valutazione o feedback che, se necessario, consente di tornare alla fase precedente per colmare gli scostamenti rilevati.

La durata di attraversamento delle fasi è timeboxed ed equivalente alla durata di un'iterazione.

Le fasi di formazione e coaching interagiscono tra di loro in modo iterativo.

Il metodo Agile utilizzato per lo sviluppo del prodotto è stato il framework Scrum [1], il framework di trasformazione Agile è stato integrato al framework Scrum grazie al supporto degli Agile Coach.

Di seguito la descrizione delle tre fasi del framework di trasformazione Agile.

1. Fase di formazione

La formazione viene svolta in modo informale, evitando quindi l'approccio frontale e favorendo una comunicazione orizzontale che permetta di percepire l'aula come un laboratorio in cui condividere esperienze; questa fase ha come obiettivo quello di aiutare le persone ad approcciare il mindset Agile e fornire i primi strumenti per abbracciare il cambiamento. Particolare attenzione è posta all'acquisizione di una logica collaborativa e alle competenze trasversali necessarie per la sua maturazione.

2. Fase di coaching

Terminata la formazione, prende concretamente il via il progetto pilota e viene avviata la fase di coaching: il team viene affiancato dagli Agile Coach durante tutti gli eventi dello Sprint: Planning, Daily, Review e Retrospective. L'obiettivo di questa fase è quello di rendere il team autonomo nell'adozione di Agile creando un vero e proprio banco di prova dove attuare quanto appreso in formazione e adattarlo continuamente. In questa fase il team sperimenta l'approccio empirico, apprende come gestire al meglio i conflitti di diversa natura (tecnici, di processo e relazionali), sviluppa la capacità di collaborazione e una

propria intelligenza di team che consente alle persone di massimizzare le prestazioni come squadra oltre che come individui.

Compito degli Agile Coach è quello di sostenere il cambiamento e aiutare le persone nel loro umano percorso di adozione del nuovo mindset.

3. Fase di outcome

Quest'ultima fase consente di verificare la rotta in termini di **business value**, (valore prodotto per il business) ed è particolarmente importante: la formazione e il coaching non servono a nulla se non sono orientati a risolvere i bisogni dei clienti, utilizzatori del prodotto. Valgono i principi della customer centricity e del value driven development; ad inizio di ogni Sprint il team ha identificato il valore da produrre in un incremento di prodotto, ha definito le metriche e le misurazioni da effettuare per stabilire il raggiungimento del **goal**. Questa fase, generalmente coincidente con l'evento Scrum chiamato Sprint Review, valuta il risultato ottenuto dal team in termini di creazione di valore. È di fondamentale importanza riuscire a coinvolgere stakeholder esterni al team, meglio se clienti effettivi e user test, in grado di confermare la soddisfazione dei bisogni e allo stesso tempo di fornire indicazioni utili all'evoluzione del prodotto per gli Sprint successivi.

Il framework di trasformazione Agile attuato per il progetto Omnibus lavora su un doppio binario: da una parte le **pratiche** e gli **strumenti** con le loro complessità tecniche, dall'altra le **persone** e le **soft skills** necessarie per governare quelle complessità e raggiungere gli **obiettivi di business**.

III. AGILE TRANSFORMATION: LA ROAD MAP

Il progetto pilota Omnibus ha seguito una Road Map (figura 2) costituita da cinque diversi momenti per facilitare l'introduzione, la transizione e l'adozione di Agile.

L'obiettivo di una transizione e adozione Agile è quello di rendere il team autonomo nel mantenere lo slancio Agile in termini mindset, pratiche e in generale attuare processi di miglioramento continuo.

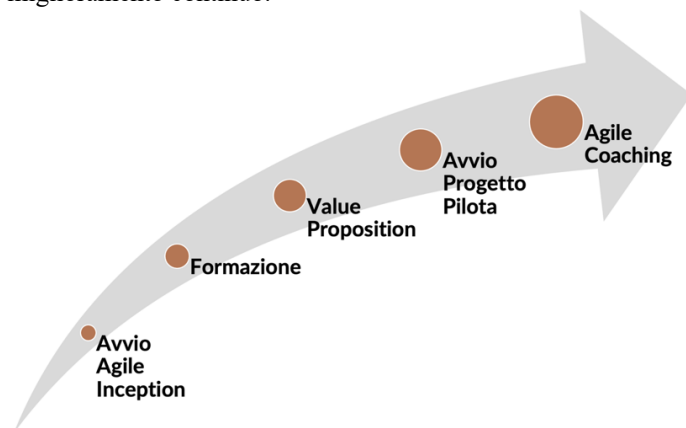


Figura 2 – Road Map di progetto

Il progetto pilota ha visto sin da subito un elevato coinvolgimento dei vertici aziendali, sponsor dell'iniziativa. In

molte ricerche è riportato che il coinvolgimento dei vertici aziendali nella trasformazione Agile è uno degli elementi che aumentano significativamente le probabilità di successo dei progetti pilota Agile. Istituzionalizzare il progetto pilota di trasformazione Agile aiuta a creare e mantenere un alto livello di impegno e coinvolgimento di tutti gli attori coinvolti.

La pandemia di Covid19 ha costretto per la prima volta tutti gli stakeholder alla collaborazione da remoto. Il progetto pilota ha avuto una durata complessiva di tre mesi (giugno – settembre 2020) e tutte le attività si sono svolte continuamente a distanza.

Vista l'assenza di precedenti storici, l'impossibilità della presenza fisica ha sollevato diverse preoccupazioni, tuttavia il framework di trasformazione Agile proposto ha ispirato fiducia in tutti gli stakeholder coinvolti. La fiducia è stata inoltre data a tutte le persone coinvolte, adeguatamente sensibilizzate ai rischi e alle opportunità del lavoro da remoto.

In seconda battuta il framework ha facilitato le frequenti interazioni tra gli stakeholder, riducendo il rischio di incomprensioni nelle comunicazioni, abilitando risposte immediate ai cambiamenti e favorendo un elevato livello di collaborazione.

Il lavoro da remoto è stato un vincolo esterno che ha contribuito ad aumentare ulteriormente il valore dei risultati raggiunti dal progetto pilota. Tipicamente gli ostacoli introdotti dal lavoro remoto in termini di strumenti e metodi di comunicazione aumentano la probabilità di fallimento delle trasformazioni Agile. In questo caso l'alta motivazione delle persone coinvolte, la loro concentrazione e l'impegno hanno portato a superare la barriera della distanza fisica, facendo emergere nuove dinamiche di collaborazione.

Il team ha selezionato strumenti di collaborazione remota che avessero una curva di apprendimento bassa, per semplicità d'uso o perché già in uso. I principali strumenti di collaborazione utilizzati sono stati:

- Zoom (<https://zoom.us>), piattaforma di videoconferenza utilizzata per le attività di formazione; è stata scelta per la possibilità di creare stanze private per facilitare il lavoro di gruppo.
- Whereby (<https://whereby.com>), piattaforma di videoconferenza già in uso in azienda utilizzata per riunioni e attività di collaborazione; è stata utilizzata durante tutti gli eventi Scrum. In particolare, lo ST è stato dotato di una stanza virtuale costantemente a disposizione per poter lavorare insieme, come se fosse un vero e proprio ufficio "fisico".
- Mural (<https://www.mural.co/>), piattaforma di collaborazione visuale basata sulla metafora della parete e dei post-it; è stata utilizzata non solo per le esercitazioni nei momenti formativi, ma anche per tutte le attività legate alla fase di definizione e aggiornamento della Vision di business del prodotto.
- Slack (<https://slack.com/intl/it-it/>), strumento di collaborazione utilizzato per inviare messaggi istantanei

tra i membri del team; è stato utilizzato quotidianamente per scambiare messaggi immediati tra i diversi componenti dello Scrum Team.

- Trello (<https://trello.com/it>), strumento di gestione visuale del lavoro in team in stile Kanban; è stata creata una Scrum Board dove i developers hanno riprodotto lo Sprint Backlog di ciascun Sprint. La Scrum Board è stata suddivisa in cinque colonne: User Story, To do, Doing, Review e Done.
- Google Sheets (<https://www.google.it/intl/it/sheets/about/>), strumento gratuito per la creazione di fogli di calcolo incluso nella suite Google Documenti; è stato utilizzato per rappresentare i contenuti del Product Backlog, al suo interno sono state riportate le User Story. Ciascuna User Story era costituita da diversi attributi tra cui: valore, priorità, stima, criteri di accettazione, definition of done, ecc.

1. Avvio Agile Inception

Il progetto di transizione e adattamento Agile chiamato **Agile Inception** è iniziato con un evento di due ore durante il quale i vertici aziendali hanno sottolineato l'importanza strategica del cambiamento di Mindset, le nuove sfide dell'organizzazione e l'impegno nel fornire supporto alla trasformazione Agile. Durante l'evento è stata presentata la Road Map, i membri del team e gli Agile Coach. Aver istituzionalizzato l'avvio della trasformazione ha motivato tutti gli stakeholder coinvolti, che si sono dimostrati curiosi e impazienti di iniziare a lavorare.

2. Formazione

La formazione è stato il secondo passo della Road Map, fondamentale per impostare e preparare in modo adeguato e uniforme tutte le persone coinvolte nel progetto di trasformazione. Mantenere la formazione isolata e informale aiuta a creare un ambiente dove le persone, non essendo pressate dal lavoro da svolgere, possono riflettere, discutere, confrontarsi e iniziare a conoscersi in un contesto il più possibile neutro.

I partecipanti alla formazione sono stati selezionati in modo da garantire eterogeneità di ruoli e competenze. Oltre ai membri del team di progetto (Scrum Team) sono stati coinvolti anche stakeholder esterni al progetto ma interni all'azienda. La partecipazione è stata estesa anche ai non membri dello Scrum Team con l'obiettivo di raccontare e condividere la filosofia Agile anche in aree dell'azienda non direttamente coinvolte nella trasformazione (ma che in futuro potrebbero decidere di intraprendere un progetto di trasformazione Agile) con l'obiettivo primario di creare una comune sensibilità e gettare le basi per la costruzione di un linguaggio condiviso.

Hanno partecipato alla formazione: personale operativo, sviluppatori di altre aree, figure di business e il CEO. I partecipanti direttamente coinvolti nel progetto pilota erano sei: il Product Owner, lo Scrum Master e i Developer (Scrum Team [1]). Lo Scrum Team ha collaborato e continua ancora oggi a

collaborare attivamente con un alto livello coinvolgimento durante ogni singolo Sprint.

Il percorso di formazione è stato strutturato in due moduli: Agile Mindset e Framework Scrum; l'ordine di presentazione dei contenuti ha tenuto conto del primo valore dell'Agile Manifesto (*persone e interazioni più che strumenti e processi*) ed ha avuto come obiettivo quello di sensibilizzare i partecipanti ai temi della responsabilità condivisa.

Il corso si è aperto con un'introduzione sulla definizione di complessità, sulla consapevolezza dell'incertezza in un periodo storico in cui l'unica costante è il cambiamento. Si è creata in questo modo la giusta cornice per iniziare a parlare dell'Agile Mindset, partendo dall'Agile Manifesto [2] con i suoi valori e principi. Gradualmente si è affrontato il passaggio dal paradigma pianificazione e controllo (predittivo), al paradigma ispezione e adatta (adattivo o empirico). Questo percorso è stato integrato con gli aspetti legati alla customer-centricity e allo sviluppo guidato dalla creazione di valore per il business (value driven development).

Il secondo modulo ha avuto come obiettivo quello di familiarizzare con il framework Scrum [1]. In particolare, approfondire come il framework aiuti a implementare l'approccio empirico attraverso i concetti di trasparenza, ispezione e adattamento. Il modulo si è articolato su diversi argomenti: i valori di Scrum, i ruoli, gli eventi e gli artefatti.

L'intero percorso di formazione integra aspetti legati alle soft skills, come ad esempio la comunicazione efficace, la gestione dei conflitti, la self-organization, l'intelligenza emotiva e la servant leadership, dando così evidenza di come l'Agile Mindset non sia solo fare ma anche e soprattutto essere.

La fase di formazione è stata gestita dai due Agile Coach insieme, si è svolta totalmente da remoto e ha avuto una durata complessiva di 20 ore, suddivise in 5 sessioni da 4 ore.

3. Value Proposition

Pur esistendo un'idea generale di prodotto da sviluppare, il concetto di Minimum Viable Product (MVP) [21] era del tutto nuovo, così come il paradigma della continua analisi dei bisogni di business. L'approccio tradizionale del definire il 100% dei requisiti del prodotto era l'unico a cui si era sempre fatto riferimento.

Per questa ragione, prima di passare alla fase di sviluppo del prodotto, sono state organizzate 16 ore di workshop in 4 sessioni sulla creazione dell'MVP della **Value Proposition**. Al workshop hanno partecipato tutti i membri dello Scrum Team e il CEO.

Il team è stato guidato dagli Agile Coach nell'esplorazione del valore e nella definizione dell'MVP in modo collaborativo e con il supporto di strumenti di creazione di Vision come ad esempio:

- Business Model Canvas
- Impact Mapping
- Buyer Personas
- User Story

L'uso di questi strumenti in combinazione con tecniche di brainstorming e collaborazione ha consentito al team di identificare un ambito di business circoscritto ad una tipologia di cliente specifica, con la quale poter sperimentare sin da subito i primi incrementi di prodotto. Si è trattato di un MVP concreto che ha consentito di scrivere le prime User Story che sono andate ad alimentare la prima versione del Product Backlog.

Questo approccio ha consentito al Product Owner di identificare i primi potenziali clienti test, da poter coinvolgere attivamente sin da subito, cioè senza dover attendere che l'intero prodotto fosse finito. Questo approccio consente di integrare immediatamente il feedback di clienti e utenti, rispondendo reattivamente ai loro bisogni e realizzando in pochi Sprint una prima release del prodotto che può essere proposta sul mercato e iniziare a produrre ricavi.

Il team, una volta sperimentato l'approccio allo sviluppo dell'MVP, della value proposition e alla creazione delle User Story, è stato sensibilizzato nel continuare a utilizzare questi strumenti, nella consapevolezza che il prodotto - nella logica Agile - è in continua e costantemente evoluzione: non esiste un end-state e la sua innovazione avviene iterativamente attraverso una interazione costante con gli stakeholder interni all'organizzazione e con il contributo dei clienti/utenti esterni. Questa fase è stata gestita solo dall'Agile Coach Alessandro Ingresso.

4. Avvio progetto pilota e Agile Coaching

La fase precedente ha consentito di avere un'idea molto chiara dell'MVP da realizzare e un Product Backlog pronto per essere sviluppato.

L'avvio del progetto pilota è coinciso con l'avvio di Scrum.

Lo Scrum Team ha concordato una durata di due settimane per Sprint.

Per l'attività di Agile Coaching si sono concordati quattro Sprint, al termine dei quali si sarebbe fatta una retrospettiva generale valutando il grado di autonomia del team nel continuare la trasformazione Agile, privilegiando un approccio che rendesse dispensabili gli Agile Coach il prima possibile.

Il progetto pilota è quindi partito con lo Sprint #1 e in particolare con l'evento Sprint Planning [1]. Durante questo primo evento lo Scrum Team ha definito lo Sprint Goal, selezionato le User Story (precedentemente stimate e priorizzate), ha eliminato le ambiguità presenti definendo una soluzione che potesse essere realizzata nei tempi dello Sprint. I developer hanno creato lo Sprint Backlog e hanno iniziato a lavorare seguendo le prescrizioni minime suggerite dalla Scrum Guide.

I developers sono stati dedicati full time allo sviluppo del prodotto, mentre Product Owner e Scrum Master sono stati impegnati al 50% circa del loro tempo.

L'attività di coaching non è stata full-time: per ogni Sprint sono state dedicate 16 ore suddivise in 4 incontri. L'Agile coaching si è svolto durante gli eventi più importanti di Scrum:

- Sprint Planning (4 ore)
- Daily (15')

- Sprint Review (2 ore) e Sprint Retrospective (2 ore)

Durante gli eventi di Planning, Review e Retrospective (per un totale di 8 ore), il contributo dell'Agile Coach è stato duplice: da una parte facilitare il team nell'applicazione dell'Agile Mindset e nel padroneggiare gli strumenti e le tecniche Agile; dall'altra supportarlo nella maturazione di comportamenti collaborativi, nel sostenere attività di sperimentazione, aiutarlo nel rendersi gradualmente autonomo e auto-gestito.

Di particolare rilievo è il contributo dato dall'Agile coach alla corretta esecuzione del Daily Scrum. È noto che questo evento è spesso frainteso e non condotto come un momento di pianificazione ma come uno status report delle attività. Per questo motivo la partecipazione a due daily per Sprint è stata fondamentale per indirizzare il team non solo al rispetto del timebox di 15 minuti ma anche per aiutarlo a comprendere quale fosse lo scopo della Daily in termini di pianificazione delle attività quotidiane e di costante verifica del realistico raggiungimento dello Sprint Goal.

Dopo il Daily Scrum, l'Agile coach è sempre restato con lo Scrum Team per fornire il proprio aiuto e supporto sulle criticità del momento. L'obiettivo dei due incontri bisettimanali di 4 ore a valle delle Daily è stato quello di fornire supporto e facilitare lo Scrum Team su diversi ambiti. A volte si è trattato di fornire supporto nel refinement o evoluzione del Product Backlog. Altre volte sono state fatte emergere delle domande tecniche da parte dei developer che hanno contribuito a trovare soluzioni alternative per non compromettere lo Sprint Goal, o approcci al miglioramento della qualità. Altre volte ancora è stato fatto coaching su singoli ruoli come il Product Owner e lo Scrum Master. In buona sostanza le ore di coaching fuori dagli eventi Scrum sono state utilizzate per liberare le persone dalle strutture pre-esistenti e per rimuovere gli anti-pattern di Scrum già noti e studiati da Stefan Wolpers [22].

IV. UN'APPLICAZIONE REALE DEL FRAMEWORK FCO: IL PROGETTO OMNIBUS DI PROGETTI E SOLUZIONI SPA

Come già descritto il framework FCO emerge dalla necessità di ideare un metodo che consentisse di avviare un processo di Agile Transformation per realizzare il progetto pilota Omnibus. In piena sintonia con l'approccio di tipo empirico, il framework FCO si è dimostrato funzionante a posteriori.

Ciascuna persona che ha partecipato al progetto pilota con il proprio bagaglio di esperienze passate e con un punto di vista differente, ha vissuto in modo diverso questo cambiamento. Pur essendo tutti accomunati dall'evidenza del risultato positivo, ciascuna persona ha provato emozioni diverse e ha individuato aspetti positivi differenti in questo viaggio che è appena iniziato.

Di seguito sono raccolte le riflessioni di ciascuna persona che ha fatto parte del progetto di Agile Transformation.

1. Perché partire con un progetto pilota di Agile Transformation? (Stefano Bonasegale – Direttore Generale)

Progetti e Soluzioni nasce nel 1999 con la missione di sviluppare soluzioni software per la gestione dei servizi a domanda individuale (in particolare per la refezione scolastica) rivolte agli Uffici Pubblica Istruzione dei Comuni e alle Società di Ristorazione Collettiva.

School E-Suite è il prodotto sviluppato per rispondere a tali esigenze: attualmente è utilizzato da oltre 360 Comuni in tutta Italia ma la sua nascita risale a 22 anni fa, in un mercato e in un contesto tecnologico profondamente diversi da quelli attuali; nel corso degli anni il prodotto è cresciuto moltissimo sia in termini di complessità che di funzioni, arrivando a soddisfare un livello notevole di esigenze.

School E-suite nasce come un prodotto monolitico e viene sviluppato con un approccio waterfall, pertanto, allo stato attuale, richiede costi di gestione e manutenzione molto elevati, soprattutto considerando che molte delle sue funzioni sono ormai superate e non più utilizzate dagli utenti.

Per tale ragione, nel 2014, Progetti e Soluzioni sceglie un progetto di totale riscrittura del software che portasse alla creazione di una soluzione flessibile, tecnologicamente avanzata e semplice da utilizzare, che tenesse conto delle nuove esigenze del cliente. A tal fine viene creato un gruppo di lavoro di 10 persone provenienti dai diversi settori dell'azienda (vendite, sviluppo, delivery e assistenza) che inizia a lavorare all'idea: il team si riunisce in un hotel per due settimane per potersi dedicare full time e senza interruzioni al progetto, e poi di nuovo per altre due settimane dopo due mesi, sempre in modalità full immersion, con l'obiettivo di definire tutte le funzioni necessarie al nuovo software; da questa prima attività di macro-analisi emergono le seguenti necessità:

- Serviranno dai 6 ai 9 mesi per le specifiche tecniche;
- Ulteriori 18-24 mesi di sviluppo da parte di 5-6 programmatori;
- Ulteriori 6 mesi di test.

In sostanza, dai 3 ai 4 anni di lavoro da aggiungersi alla normale attività di manutenzione di School E-Suite, in attesa di lanciare il nuovo prodotto sul mercato; queste valutazioni spingono l'azienda ad abbandonare il progetto in quanto non sostenibile.

Nel 2019 viene creato un nuovo gruppo di lavoro, composto stavolta non solo da persone chiave dell'azienda, ma anche da clienti: le attività di brainstorming si concentrano in 4 giorni distribuiti nell'arco di un mese, alla fine del quale emerge un elenco di oltre 200 voci da sviluppare; appare immediatamente chiaro che si tratterà di un lavoro estremamente impegnativo, ma in azienda è nel frattempo maturata una nuova sensibilità verso Agile, per cui si decide di valutarne l'approccio e le metodologie.

Il primo nodo da sciogliere riguarda la scelta delle persone da dedicare allo sviluppo del nuovo software: l'azienda, nel 2018, aveva acquisito un competitor diretto che sviluppava un prodotto molto simile per il medesimo mercato di riferimento, e

l'acquisizione aveva portato, di fatto, alla creazione di due team di lavoro in competizione interna tra di loro.

Di qui la necessità di individuare sviluppatori nuovi, estranei ad entrambi i gruppi, per comporre il team dei Developers [1]. Il ruolo di Product Owner [1] viene affidato alla responsabile del gruppo che si occupa di Delivery e Deployment di School E-Suite presso i clienti, persona in azienda da oltre 15 anni che conosce profondamente le esigenze dei clienti e del mercato e che sa relazionarsi con i diversi stakeholder. Il ruolo dello Scrum Master [1] viene ricoperto dalla nuova responsabile tecnica della azienda acquisita, persona nuova in azienda ma di grande esperienza nel ruolo di direttore tecnico che provvede sia alla selezione degli sviluppatori che degli Agile Coach. Si decide di ingaggiare questi ultimi perché le incertezze collegate all'utilizzo di nuovi strumenti e all'acquisizione di nuovi collaboratori rischiavano di obbligare a procedere per tentativi prima di arrivare alle competenze necessarie per implementare framework e mindset, con conseguente perdita di tempo e di denaro.

È in questo contesto che nasce il prodotto Omnibus: la nuova soluzione software diventa il progetto pilota di transizione e adozione Agile di Progetti e Soluzioni SpA.

2. Cosa fa un Product Owner di diverso rispetto ad un approccio tradizionale?

Alessia Pozzoli (Product Owner): “Dopo venti anni di sviluppo software in modalità Waterfall, i dubbi iniziali erano davvero tanti; soprattutto avevo due grandi perplessità:

- come poter garantire lo sviluppo efficace di un prodotto, non facendo prima un'analisi completa di tutte le esigenze, sulla base dell'approccio Waterfall che fino a quel momento avevo applicato,
- come individuare al meglio l'MVP da implementare per il target di clienti individuato.

In questo senso, il corso di formazione è stato di grande aiuto per affrontare questa nuova sfida perché oltre a fornirci conoscenze e strumenti per sviluppare con il framework Scrum, abbiamo approfondito e sviscerato bene l'aspetto del mindset Agile, molto importante a mio avviso per affrontare un cambiamento così significativo, in particolare rispetto alla collaborazione e alle dinamiche all'interno dello Scrum Team.

Inoltre, la possibilità di essere affiancati da un coach nella gestione dei primi sprint, ci ha permesso di applicare fin da subito quanto appreso dalla teoria in modo efficace.

Questo ci ha permesso di rilevare che il nuovo approccio Agile aveva dei punti di forza significativi rispetto all'approccio precedente perché in Agile la centralità del cliente è massima; direi che gli elementi più impattanti sono stati tre:

- lo sviluppo ha come principale obiettivo quello di generare sempre Valore per il cliente;
- lo sviluppo iterativo con approccio empirico per implementare quanto realmente richiesto e necessario, al fine di dare un riscontro immediato al cliente;

- il coinvolgimento e la condivisione puntuale del cliente da cui raccogliere i feedback per capire se quanto realizzato soddisfa realmente le esigenze di chi utilizza il software.

Nello Scrum Team il PO indica l'evoluzione che dovrà avere il prodotto e le priorità degli sviluppi, dopo un lavoro di periodico e puntuale confronto con i clienti e con i vari stakeholder interni dell'azienda, dal Business Owner, ai commerciali, al marketing, ecc.. È stato gratificante riscontrare come le mie capacità relazionali e di negoziazione maturate negli anni per la gestione dei clienti e delle loro richieste, fossero competenze estremamente utili per un PO, oltre – ovviamente - alla profonda conoscenza dei nostri clienti e dell'ambito in cui operano.”

3. Lo Scrum Master non è un tecnico, quindi qual è il suo contributo al team?

Monica Semprini (Scrum Master): “Ho un'ampia esperienza alle spalle, con svariati anni trascorsi in passato alla guida di ampi progetti anche molto complessi, con il ruolo di Project Manager (parliamo di progetti waterfall di durata anche pluriennale). Queste esperienze già mi avevano portato a maturare una serie di considerazioni sui limiti di questo approccio (a puro titolo esemplificativo, tempi di rilascio troppo lunghi rispetto alle esigenze del cliente di business, parziale ignoranza da parte di IT di quello che è la reale esigenza e relativo valore di business, mancata reattività a fronte di eventuali mutate esigenze del business nel tempo).

In tal senso ho percepito da subito, sulla carta, i vantaggi che avrebbero potuto derivare da un approccio totalmente diverso, quale è Agile.

Nel ruolo di Scrum Master mi sono dovuta però misurare con una serie di comportamenti consolidati in anni di lavoro come Project Manager, che non sono in linea con il ruolo di Scrum Master (ne sono l'antitesi): sono nata e cresciuta in area tecnica (e con un ruolo attualmente di direzione tecnica) e mi sono trovata più volte a dovermi letteralmente mordere la lingua, per non intervenire durante le varie sessioni di refinement (e/o gli eventi). Questo non solo relativamente a questioni tecniche di alto livello, ma organizzative/analitiche trattate a livello di team dei Developer (o Scrum Team): ho dovuto mettere da parte le mie attitudini di guida attiva, per mettermi al servizio del team, anche nei momenti in cui ho intravisto possibili scelte non congrue e ho scelto di non intervenire (e questo grazie anche al supporto iniziale del coach).

In altri termini ho cercato di essere presente ma al servizio, in reazione a problemi, esigenze espresse dallo Scrum Team o comunque per prevenire impedimenti. Ho riposto fiducia nella capacità di auto-organizzarsi da parte del Developer e nella collaborazione, e ho potuto apprezzarne i risultati. Mettere in pratica comportamenti e linee guida che sulla carta sembravano di semplice interpretazione, è stato in tal senso molto faticoso. Imparare ad esempio a gestire i silenzi, per lasciare spazio alle riflessioni, senza riempire i vuoti temporanei che a volte si creano. Parimenti, ci tengo a sottolineare che la guida Scrum è essenziale (nel senso di stringata) e, nonostante i corsi, gli

approfondimenti e lo studio fatti, per la certificazione prima e per il consolidamento delle conoscenze poi, interpretare realmente il ruolo su un progetto mi ha messo di fronte a situazioni inattese, in cui ho dovuto mettermi in gioco e soprattutto provare strade per declinare la teoria. Ho avuto la fortuna di lavorare con uno Scrum Team molto motivato e tornare a vivere in prima persona (seppure in un ruolo non più di guida attiva ma come servant-leader al servizio del team) le sfide di un progetto, i momenti belli (i successi) e quelli difficili (i fallimenti) è stato motivante: il tutto però da una prospettiva del tutto diversa dal passato e questa è stata per me una grande opportunità.

Con in testa un grande insegnamento: il lavoro non deve essere solo fatica ma deve essere anche ‘divertimento’. E soprattutto non deve diventare ripetitivo e portarci a vivere la quotidianità (degli eventi ad esempio) in maniera vuota e noiosa, con il pilota automatico inserito: e qui in alcuni casi ho dovuto dare spazio alla fantasia (per rendere ad es. diverso il formato della retrospettiva, sulla base però delle evidenze emerse durante lo sprint, per mantenerla sempre interessante ma capace di offrire spunti di riflessione pertinenti).

Ho comunque potuto vedere tra i risultati di questo approccio, uno Scrum Team veramente motivato e sotto pressione non per interventi esterni, ma per voglia di fare e di migliorare.

Vorrei inoltre sottolineare il fatto che il team dei Developer (nato con persone che non si conoscevano e mai avevano lavorato insieme prima di allora) ha da subito cominciato a collaborare in maniera efficace con una sinergia da team rodato, dimostrando nel contempo notevoli capacità in termini di auto-organizzazione, inoltre non ci sono state (almeno fino ad ora) pressioni e tentativi di influenza esterni, ragion per cui mi sono trovata per lo più a fare interventi di coaching mirati allo Scrum Team in relazione al framework Scrum, interventi per prevenire ostacoli (organizzando il supporto ad hoc da parte di esperti), interventi per favorire la comunicazione con colleghi esterni al team.

I benefici ottenuti sono legati in prima battuta alla tempestività del riscontro ed all'efficacia del risultato, con un team di Developer (o meglio uno Scrum Team) che ritengo si senta spalleggiato, mai solo di fronte alle difficoltà, incoraggiato a prendere decisioni e provare strade, nella consapevolezza di una responsabilità comunque condivisa; ma soprattutto si è creato un ambiente sereno (in senso ampio, che coinvolge l'azienda) in cui affrontare insieme le difficoltà, in un clima di collaborazione, condivisione e trasparenza, con uno Scrum Team coeso teso al miglioramento continuo, ove ciascuno fornisce il suo contributo, ma lavora in sinergia con gli altri per un obiettivo comune.”

4. Cosa è cambiato nel processo di sviluppo utilizzando un approccio Agile?

Claudio Fioretti (Developer): “Il mio approccio è stato molto propositivo e aperto al cambiamento. Avevo infatti già vissuto positivamente alcuni aspetti della metodologia Agile durante esperienze lavorative precedenti, ed ero molto interessato ad approfondire il framework Scrum.

Adottando questa metodologia nel lavoro, ho avuto subito un riscontro positivo: i costanti feedback di fine Sprint si sono

dimostrati molto preziosi, in quanto hanno permesso agli stakeholder di dare maggiore importanza a ciò che per loro era prioritario, e allo Scrum Team di proseguire il lavoro, senza la necessità di un'analisi dettagliata delle attività da svolgere.

Inevitabilmente il metodo di sviluppo è cambiato, dovendosi adattare al framework: in primo luogo le interfacce grafiche sono diventate una priorità e in alcuni casi ci hanno permesso di rimuovere o modificare delle features, prima di averle completate lato back-end. Questo ci ha così portato ad un risparmio di diversi giorni di lavoro.

D'altro canto, per mostrare più funzionalità e ricevere il maggior numero di feedback in anticipo, ci è capitato di lasciare alcuni dettagli incompleti o slegati dal resto del prodotto. Questo atteggiamento porta inevitabilmente ad accumulare del lavoro non fatto, che dovrebbe essere smaltito in uno sprint di refactoring, che ad oggi non è stato pianificato poiché viene considerato un rallentamento sulla tabella di marcia, senza dare evidente valore di business.

Come sviluppatore ho ancora qualche difficoltà ad adattarmi alla necessità di scrivere codice utile solo alla US in corso, senza dover tenere in considerazione gli inevitabili intrecci con altre US già pianificate nel Product Backlog.

In poche parole, questo approccio ha portato, talvolta a dover riscrivere più volte le stesse porzioni di codice, aumentando il tempo necessario per "completare" una funzionalità, e in altri casi invece a risparmiare tempo su attività non più necessarie.

Non è stato semplice quantificare la mole di lavoro in grado di portare a termine come Developer, ma grazie alle statistiche del Product Backlog, nel tempo stiamo riuscendo a prendere coscienza della nostra velocity senza sopravvalutarci e senza sottovalutare complessità e incognite. Sprint dopo sprint, le nostre stime stanno diventando sempre più accurate e le negoziazioni con il PO sempre meno necessarie."

5. Il framework Scrum è solo forma o anche sostanza?

Luciano Taurino (Developer): "Sono entrato a far parte del gruppo Progetti e Soluzioni a giugno 2020 e, in particolare, del team deputato allo sviluppo del progetto "Omnibus".

La mia esperienza professionale era stata consolidata per la maggior parte in contesti ampiamente strutturati e normati dove la pianificazione predittiva, sia qualitativa che quantitativa, era requisito imprescindibile di qualsiasi progetto o attività.

Da 5 anni mi occupo ormai in maniera esclusiva di progettazione e sviluppo di piattaforme informatiche web-based e, sin da subito, mi sono reso conto che un approccio completamente predittivo era molto difficile da attuare. Ad ogni nuovo progetto si cercava di realizzare un'analisi migliore della precedente ma, puntualmente, poco dopo l'avvio dell'implementazione la stessa veniva messa da parte per essere trattata come un mero canovaccio. Lentamente, ma inesorabilmente il progetto deviava dal suo percorso originario per problematiche tecniche o

adattamenti del cliente e alla fine l'analisi tornava utile in fase di controversie (che puntualmente prima o poi si proponevano).

In passato ho incontrato il framework Scrum in diverse sue declinazioni ma l'esperienza è sempre stata deludente. Avevo ricevuto l'impressione che il framework consistesse in una mera liturgia in grado, nel migliore dei casi, di portare in sistema alcune good practice relative alla gestione del team e del progetto. Il più delle volte neppure si era consapevoli di aver spostato l'attenzione dallo scope allo schedule, perdendo il controllo delle risorse necessarie.

Appena arrivato nel Gruppo Progetti e Soluzioni sono stato inserito nel corso Scrum condotto da Anna e Alessandro in cui, per la prima volta, ho avuto modo di vedere e capire i principi fondanti del framework e toccare con mano la fase di avvio di un progetto imponente quale "Omnibus". Pian piano ho capito come Scrum possa essere veramente un valido strumento per la gestione di progetti complessi/dematerializzati e, quindi, applicabile con profitto nella produzione del software.

Durante il corso ho capito che la liturgia che avevo sempre visto è sì forma che incapsula una serie di buoni principi, ma è anche sostanza in quanto li eleva a metodo.

Solitamente le riunioni sono viste come una perdita di tempo che si protrae a tempo indeterminato senza un ritorno alcuno; gli eventi Scrum, al contrario sono brevi e frequenti costringendo i vari attori coinvolti ad un confronto continuo e fruttuoso. Lo Sprint, nella sua brevità, costringe a rilasci frequenti e ravvicinati e, quindi, a feedback immediati facilmente accolti. Finalmente il software viene inteso come un prodotto vivo che viene continuamente adattato alle esigenze degli stakeholder o del business owner e che, in fondo, non potrà mai considerarsi giunto alla sua versione definitiva.

Per quanto mi riguarda, però, la vera illuminazione è stata scoprire la funzione degli Story Point e del metodo di stima delle story e degli sprint, della stima delle fasi successive del progetto. Contrariamente a quanto avevo visto in precedenza, ho capito che si rinuncia alla "stima esatta" per procedere con un approccio euristico in cui lo story point è un'unità di misura relativa che valuta il rendimento medio passato e quindi può stimare il lavoro futuro.

Dal punto di vista tecnico, il mio stile di scrittura del software ha subito alcuni aggiustamenti dettati dalla maggior consapevolezza che non è ammissibile avere uno scope a lungo termine. Il mio codice è ora maggiormente Object Oriented al fine di essere scalabile e manutenibile. La mia progettazione, però, ha uno sguardo al futuro orientata ad un più breve periodo al fine di evitare, come è successo molto spesso in passato, di dover cestinare interi moduli divenuti inutili o completamente in contrasto con le mutate esigenze del business owner.

Il Product Owner riveste un ruolo essenziale in questo processo in quanto conosce bene il prodotto e stabilisce la direzione cui tendere. Ogni scelta progettuale è stata quindi condotta nel confronto: dopo aver letto la story e i relativi criteri di accettazione il più delle volte l'ambito di implementazione era abbastanza definito, ma con poche domande è stato possibile chiarire anche gli aspetti incerti. Il problema vero è riuscire a

limitare la solita smania del programmatore che vuole prevedere sin dall'inizio tutte le possibili problematiche e limitare lo sviluppo al solo contesto della story.

La produzione del software resta comunque un'attività complessa in cui l'interazione di numerosi sistemi differenti porta all'imprevedibilità nella progressione e quindi il raggiungimento degli obiettivi non è mai assicurato. Con il susseguirsi degli sprint, la velocity media diventa un dato via via sempre più affidabile che consente di stimare in maniera ragionevole il quantitativo di lavoro gestibile dal team. Ma a volte, specialmente nei primi sprint dei Developer, le cose non vanno come desiderato. Più volte a noi è capitato di dover cercare insieme al Product Owner un punto di equilibrio tra le difficoltà tecniche e le esigenze progettuali. Il più delle volte è stato sufficiente cambiare il metodo nello sprint goal o ridimensionare i criteri di accettazione di alcune story riuscendo a realizzare quasi tutto il valore portato da essa. In alcuni casi più gravi, purtroppo, la negoziazione ha portato alla rimozione di un'intera story dallo sprint (ed al non raggiungimento dello sprint goal).

Inoltre, grazie soprattutto alla disponibilità dell'azienda che ha abbracciato il framework nella sua interezza, per la prima volta ho avuto la possibilità di sperimentare all'interno del ciclo produttivo formazione continua e test automatici che assicurano qualità continua e crescente, con un ritorno a lungo termine per tutti gli attori coinvolti nel progetto.”

6. Quali vantaggi ha dato la formazione prima dell'avvio del progetto?

Marco Spanò (Developer): “Prima di questo progetto la metodologia Agile Scrum mi appariva davvero interessante ma, per certi versi, di difficile applicazione; infatti, nelle precedenti esperienze lavorative i miei interlocutori utilizzavano il termine Agile in modo confuso, spesso solo per fare riferimento a nuove metodologie senza conoscerle.

La formazione ha svolto un ruolo fondamentale nell'introdurmi nel mondo Agile e Scrum, una formazione orientata alla concretezza e dei coach che ci hanno seguito passo dopo passo nella fase di avvio del progetto condividendo la loro conoscenza e le loro esperienze.

Un team alla pari permette di confrontarsi continuamente durante lo svolgimento del progetto, in questo modo ogni membro del team porta in campo le proprie esperienze a favore degli obiettivi di gruppo. Si tratta di una modalità di lavoro fortemente caratterizzata da corresponsabilità, autonomia, cooperazione e confronto.

Le scelte tecniche sono principalmente a carico dei Developer, ad ogni modo il Product Owner contribuisce esprimendo le necessità e gli obiettivi da raggiungere, integrandosi e avvicinandosi il più possibile al linguaggio tecnico degli sviluppatori.

Stiamo migliorando nel prendere consapevolezza delle nostre capacità di gestione del carico lavorativo. Le volte che ci siamo

trovati in debito rispetto al tempo a disposizione abbiamo negoziato con il PO eventuali features comprese nello sprint, proponendo soluzioni che ci hanno permesso di raggiungere lo Sprint Goal, alcune volte, però, queste scelte hanno costretto noi Developer a dover modificare nuovamente il codice in un secondo momento per garantire l'integrità del software.”

7. Anna Di Girolamo (Senior Agile Coach)

Il progetto Omnibus è stato molto sfidante; all'incertezza che normalmente accompagna l'avvio di un progetto pilota si aggiungeva la difficoltà data dal fatto che avevamo a che fare con un remote team e che tutte le attività si sarebbero svolte da remoto, collegati in video-conferenze. Mi occupo di comunicazione e relazioni organizzative, per cui mi era chiaro che sarebbe stato importante gestire questo elemento con molta attenzione, soprattutto considerando che eravamo in piena pandemia e non sapevo quale potesse essere l'impatto che questa avrebbe avuto sugli stati emotivi delle persone.

Ciò ha comportato certamente uno sforzo in più per cogliere tutti quegli elementi della comunicazione non verbale che sono molto chiari quando siamo in co-presenza ma che possono diventare sfuggenti quando ci troviamo dietro a un monitor; la scelta di lavorare in coppia con il mio partner d'aula, che è dettata da ragioni di efficacia formativa, si è dimostrata vincente anche sotto questo punto di vista perché di fatto questo ha permesso a me e al collega di poter cogliere (ed accogliere) precocemente e produttivamente tutto quello che accadeva durante le nostre sessioni.

Il mio lavoro consiste nell'aiutare le persone a sviluppare le proprie soft skills, sia in un'ottica di potenziamento individuale che di team; questo tema viene spesso sottovalutato quando si parla di trasformazione e adozione Agile, ma Alessandro ed io lo abbiamo sempre considerato fondante. Ovviamente non tutte le aziende colgono l'importanza di avere due Coach che lavorano in co-presenza e in maniera sinergica, anche perché si tratta di un investimento significativo. Il punto è che “installare” Scrum senza aver lavorato prima sulle persone, si rivela infinitamente più oneroso e non produce i risultati attesi. In Progetti e Soluzioni noi abbiamo trovato persone molto sensibili a questo tema, che hanno compreso le ragioni di un progetto pilota sostenuto da due Coach, dimostrando un elevatissimo livello di interesse e coinvolgimento.

Il mio obiettivo è quello di rendermi indispensabile il prima possibile, ma non posso sapere predittivamente quando questo accadrà; con lo Scrum Team di Omnibus i progressi sono stati molto veloci, tutte le persone coinvolte mostravano un grande desiderio di lavorare insieme e di farlo al meglio, per cui le competenze che ho trasferito io sono state velocemente assorbite, tanto che le ho viste atterrare in maniera sistematica, e molto produttiva.

Al termine del progetto pilota erano pronti a muoversi in autonomia e questo, per chi sceglie di fare l'Agile Coach, è un momento di grande soddisfazione umana e professionale.

8. Alessandro Ingrosso (Senior Agile Coach)

Il progetto Omnibus è stato per me unico sotto molti aspetti. Come già più volte scritto, il progetto si è svolto durante la pandemia con un team che ha lavorato da remoto. In 20 anni di esperienza questa è stata la prima volta in cui ho collaborato con un team virtuale. Ovviamente lavorare con un team in presenza è diverso, sono diverse le dinamiche, le interazioni, i comportamenti, ma se il team è in grado di adattare i propri comportamenti alla dinamica collaborativa da remoto, allora il risultato non cambia. È scontato che un team da remoto sia diverso da un team in presenza, quello che facciamo in presenza non possiamo farlo a distanza. Ma è anche vero che a distanza possiamo fare cose che non si possono fare in presenza. Quindi se cambiamo il paradigma di lavoro dobbiamo anche cambiare comportamenti, strumenti, pratiche, interazioni, abitudini, prendendo il meglio che è possibile ottenere dal lavoro da remoto. Questo forte vincolo del team da remoto mi ha consentito di avere la prova empirica che questo cambiamento è possibile. Non è necessario attendere tanto per far andare a regime il team, ma soprattutto i risultati sono gli stessi che si sarebbero ottenuti in presenza. Non dimenticando che sia in presenza che a distanza l'ingrediente più importante sono le persone, sono loro che superano gli ostacoli, trovano nuove soluzioni e consentono di raggiungere i risultati. Solo grazie a loro il lavoro da remoto ha funzionato.

In passato ho sempre avuto esperienze di Agile Coaching con team di Developers che lavoravano insieme in azienda da diversi anni. In questo caso mi sono trovato di fronte un team che non aveva mai lavorato insieme e che si conosceva per la prima volta durante la fase di formazione del progetto pilota. Sapevo che non avevo elementi per fare alcuna predizione sul risultato, ma ero molto curioso di sperimentare questa nuova dinamica. Quello che ho osservato dalle loro dinamiche di collaborazione è stata la rapidità con cui hanno iniziato a sviluppare in modo Agile. Non avendo sviluppato software insieme in passato non avevano un processo di lavoro già definito, hanno così potuto determinarlo per la prima volta utilizzando valori e principi dell'Agile Manifesto for Software Development [2].

L'Agile Mindset e le metodologie Agile sono ancora frequentemente associate al solo mondo dello sviluppo software. Questo è uno dei motivi per cui spesso manca il sostegno del management aziendale e degli altri stakeholder. La conseguenza di questo disallineamento porta alla presenza di due lingue differenti nella stessa azienda, con effetti che si possono immaginare.

Tutto questo non è accaduto in Progetti e Soluzioni, sin dai primi incontri il management aziendale è stato presente e ha sostenuto l'iniziativa. Durante lo sviluppo del progetto pilota è stato presente nei momenti chiave, ha partecipato attivamente ai momenti formativi sviluppando un linguaggio condiviso con lo Scrum Team. L'apprendimento e l'uso del linguaggio condiviso ha portato ad un efficace ed efficiente allineamento tra business e developers, evitando la tipica dinamica disfunzionale del "noi" e "voi" ma si sono sempre sentiti appartenenti ad un unico team.

Il progetto pilota Omnibus è stato anche il primo progetto a sperimentare il framework FCO. Anna ed io abbiamo portato

avanti ogni singolo intervento partendo sempre dall'outcome desiderato, identificando i gap formativi e abbiamo servito il team nel mettere in pratica le conoscenze acquisite. Iterativamente abbiamo adattato gli strumenti e i metodi alle differenti esigenze e raccolto i feedback dallo Scrum Team per migliorare costantemente. Il risultato è stato sorprendente anche per noi, abbiamo visto emergere concretamente un team ad alte prestazioni. Speriamo di poter applicare presto questo framework ad altri casi.

V. CONCLUSIONI E SVILUPPI FUTURI

L'utilizzo del framework FCO per supportare l'Agile Transformation in Progetti e Soluzioni si è dimostrato funzionante. I metodi e le pratiche utilizzate all'interno del framework FCO si sono rilevati utili nel governare con consapevolezza il fare Agile e l'essere Agile. Questo risultato ha consentito di mantenere al centro le persone e lavorare avendo sempre in mente come obiettivo quello di massimizzare il valore per il cliente. Mantenendo allenati questi comportamenti si è potuto far crescere il lavoro collaborativo, si è creato un clima di fiducia e trasparenza all'interno del team, che ha affrontato le incertezze e i cambiamenti come opportunità per sperimentare nuove soluzioni per far felici i propri clienti. Il team ha saputo affrontare i conflitti in ottica win-win, sapendo gestire le proprie emozioni e non perdendo mai di vista i bisogni del cliente.

Le persone che hanno fatto parte di questo progetto pilota sono state certamente fondamentali per raggiungere il risultato, l'utilizzo del framework FCO ha aiutato ad arrivare rapidamente all'autonomia del team e a rendere gli Agile Coach esterni dispensabili. Quest'ultimo è proprio il principale obiettivo dell'attività di Agile Coaching.

Questo risultato positivo apre a diversi scenari futuri non mutualmente esclusivi. Un follow-up ad un anno di distanza sul progetto Omnibus potrebbe valutare quale sia stata l'evoluzione dell'Agile Mindset, come si è evoluto il framework Scrum, quali tecniche siano in uso per lo sviluppo della value proposition e soprattutto quale sia la soddisfazione dei clienti nell'uso del prodotto. Un altro scenario riguarda la possibilità di supportare lo Scrum Team del progetto Omnibus nel far scalare Scrum; per aumentare il numero di feature rilasciate potrebbe essere gradualmente aumentato il numero di developers. Questo consentirebbe gradualmente di passare da uno a due team, osservando e governando la complessità del lavoro all'aumentare del numero delle persone.

Infine, l'ultimo scenario potrebbe essere quello di estendere l'uso dell'Agile Mindset e delle metodologie Agile ad altri team aziendali, facendo partire nuovi progetti pilota.

Queste ipotesi di scenario di sviluppo rendono chiaramente l'idea di come l'Agility possa gradualmente, lentamente e con determinazione diffondersi all'interno dell'organizzazione.

Un'organizzazione Agile che opera in un contesto che cambia rapidamente nell'incertezza ma che è quotidianamente ossessionata dalla soddisfazione dei bisogni dei propri clienti.

RIFERIMENTI

- [1] K. Schwaber, J. Sutherland, The Scrum Guide, The Definitive Guide To Scrum: The Rules of the Game, November 2020
- [2] Agile Manifesto for Software Development: <http://agilemanifesto.org>, 2001
- [3] F. Lisca, Business Agility, Che cosa è, come funziona e perché oggi è necessaria, FrancoAngeli, 2019
- [4] J. Job, 7 vital differences between Agile adoption and agile transformation, <https://responsiveadvisors.com/blog/7-vital-differences-between-agile-adoption-and-agile-transformation/>, 2019
- [5] A. Mersino, 5 key differences between agile adoption & transformation, <https://vitalitychicago.com/blog/5-key-differences-between-agile-adoption-and-agile-transformation/>, 2017
- [6] T. J. Gandomani, M. Z. Nafchi, Agile Transition and Adoption human-related challenges and issues: A Grounded Theory approach, Elsevier, 2016
- [7] N. Ganesh, S. Thangasamy, Lesson learned in transforming from traditional to agile development, J. Comput. Sci. 8, 389-392, 2012
- [8] A. Qumer, B. Henderson-Sellers, A framework to support the evaluation, adoption and improvement of agile methods in practice, J. Syst. Softw. 81, 1899-1919, 2008
- [9] J. Abdelnour-Nocera, H. Sharp, Understanding conflicts in agile adoption through technological frames, Int. J. Sociotechnol. Knowl. Dev. 4, 29-45, 2012
- [10] V. B. Mišić, Perceptions of extreme programming: a pilot study, In: Proceeding of IEEE International Engineering Management Conference (IEMC2005), pp.307-312, 2005
- [11] M. Pikkariainen, O. Salo, R. Kuusela, P. Abrahamsson, Strengths and barriers behind the successful agile deployment – insights from the three software intensive companies in Finland, Emp. Softw. Eng. 17, 675-702, 2012
- [12] A. Qumer, B. Henderson-Sellers, T. McBride, Agile adoption and improvement model, In: Proceeding of 4th European and Mediterranean Conference on Information Systems, EMCIS2007, pp.21-29, 2007
- [13] O. Salo, P. Abrahamsson, An iterative improvement process for agile software development, Softw. Process Improv. Prac. 12, 81-100, 2007
- [14] A. Sidky, A Structured Approach to Adopting Agile Practices: The Agile Adoption Framework, Faculty of Engineering, Virginia Polytechnic Institute and State University, Virginia, USA, 2007
- [15] J. Srinivasan, K. Lundqvist, Agile in India: Challenges and lessons learned, In: Proceeding of 3rd India Software Engineering Conference, ISEC'10, ACM, pp.125-130, 2010
- [16] M. Cruth, Discover The Spotify Model, What the most popular music technology company can teach us about scaling agile, <https://www.atlassian.com/agile/agile-at-scale/spotify> , Atlassian
- [17] T. J. Gandomani, H. Zulzalil, A. A. Ghani, A. M. Sultan, M. Z. Nafchi, Obstacles to moving to agile software development: at a glance, J. Comput. Sci. 9, 620-625, 2013
- [18] S. Nerur, R. Mahapatra, G. Mangalaraj, Challenges of migrating to agile methodologies, Commun. ACM 48, 72-78, 2005
- [19] T. J. Gandomani, M. Z. Nafchi, An empirically-development framework for Agile transition and adoption: A Grounded Theory approach, Elsevier, 2015
- [20] D. Pupek, Chicken and Pig Make Breakfast, <http://www.agilejedi.com/chickenandpig> , blog “The Agile Jedi”, data sconosciuta
- [21] E. Ries, The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, Currency, 2011
- [22] S. Wolpers, Scrum Anti-Patterns Guide, <https://age-of-product.com/scrum-anti-patterns/>, data sconosciuta

AUTORI

Alessandro Ingresso, Senior Agile Coach, a.ingrosso@me.com,
<https://www.alessandroingrosso.com/>,
<https://www.linkedin.com/in/aingrosso/>

Anna Di Girolamo, Senior Agile Coach,
digirolamoanna@yahoo.it, <https://www.linkedin.com/in/anna-di-girolamo-627756125/>