

POLITECNICO DI TORINO

Collegio di Ingegneria Gestionale

Corso di Laurea Magistrale in Ingegneria Gestionale

Tesi di Laurea Magistrale

**Proposta di un modello Hybrid Agile-Stage-Gate per la
gestione dello sviluppo prodotto nel processo di design
del settore automotive**



Relatore

Prof. Alberto De Marco

Candidato

Enrico Villa

AA. 2017-2018

Ringraziamenti

Desidero ringraziare il Professor Alberto De Marco per avermi assistito durante la stesura del mio elaborato fornendomi costante supporto, mostrandosi disponibile e dandomi nuovi spunti per condurre al meglio il mio lavoro.

Ringrazio la mia famiglia, i miei genitori e mia sorella per avermi sostenuto, sia a livello economico che morale, durante tutto il mio percorso di studi. Siete il mio punto di riferimento ed è anche grazie a voi che oggi ho raggiunto questo importante traguardo.

Ringrazio gli amici di sempre: Lorenzo, Simone, Nicolò e Federica e tutti gli altri compagni di vita, avventure e vacanze. Mi avete supportato e sopportato durante questi cinque lunghi anni e abbiamo condiviso insieme momenti di gioia e di difficoltà, sia legati all'Università ma soprattutto nella vita di tutti i giorni.

Ultima ma non ultima, ringrazio Blue Engineering srl, per avermi ospitato e dato l'opportunità di condurre il progetto per la mia Tesi Magistrale. In particolare, ringrazio l'Ing. Pierangelo Farina e l'Ing. Paolo Mirone per avermi quotidianamente aiutato, guidato e per avermi insegnato molto.

Sommario

1	Introduzione	1
1.1	Scopo del lavoro	2
1.2	Struttura della tesi	3
2	Analisi della letteratura	4
2.1	Stage-Gate Model	4
2.1.1	Idea Stage: Scoperta	7
2.1.2	Gate 1: Selezione dell'idea	8
2.1.3	Stage 1: Scoping	8
2.1.4	Gate 2: Seconda selezione	8
2.1.5	Stage 2: Creazione del Business Case.....	9
2.1.6	Gate 3: Via allo sviluppo	10
2.1.7	Stage 3: Lo Sviluppo.....	10
2.1.8	Gate 4: Via ai test	11
2.1.9	Stage 4: Test e validazione	11
2.1.10	Gate 5: Via al lancio	12
2.1.11	Stage 5: Il lancio.....	13
2.1.12	Post-Launch review	13
2.2	Scrum	14
2.2.1	I principi	15
2.2.2	Elementi di Scrum.....	16
2.2.3	Organizzazione	17
2.2.4	Attività time-boxed	18
2.2.5	Gli strumenti Scrum.....	20

2.2.6	Definizione di “done”	23
2.2.7	Casi studio	24
2.2.8	Confronto tra Scrum e project management tradizionale regolarmente.....	26
2.3	Modelli ibridi Agile-Stage-Gate.....	29
2.3.1	Applicazione a livelli differenti	29
2.3.2	Le 3 A	32
2.3.3	Criteri flessibili ai Gate.....	33
2.3.4	Express Stage-Gate	34
2.3.5	Vantaggi.....	35
3	Proposta di piano sviluppo prodotto	36
3.1	Soggetti della ricerca	36
3.2	Il modello	37
3.3	Agile nell’automotive.....	39
3.4	Panoramica del processo.....	40
3.5	Organizzazione.....	44
3.6	Lo sprint	47
3.7	I meeting	48
3.8	Ripetizione del ciclo	52
3.9	Principali differenze rispetto Scrum	53
3.10	Strumenti, documenti e template	55
3.10.1	Product backlog.....	55
3.10.2	Stima delle attività.....	57
3.10.3	Sprint backlog	59

3.10.4	Sprint board.....	59
3.10.5	Matrici di responsabilità.....	61
3.10.6	Burndown chart.....	62
3.11	Criteri passaggio gate.....	62
3.12	Vantaggi e svantaggi attesi	64
4	Il caso studio.....	67
4.1	Blue Engineering srl	67
4.2	Il progetto pilota	67
4.3	Preparazione.....	69
4.4	Product backlog	70
4.5	Day 1: sprint planning.....	72
4.6	Daily meeting.....	74
4.7	Scrum of scrum	75
4.8	Sprint review e deliverable.....	77
4.9	Sprint 2.....	77
4.10	Osservazioni	78
5	Conclusioni e sviluppi futuri	83
	Appendici.....	88
	Diagramma di flusso del processo	89

1 Introduzione

L'innovazione e lo sviluppo del prodotto costituiscono un importante fattore per la crescita di un'organizzazione. Al giorno d'oggi però le richieste e i cambiamenti del mercato sono in rapida evoluzione ed è richiesto un time to market sempre più basso in modo da soddisfare tali cambiamenti e battere la concorrenza.

Lo sviluppo di nuovi prodotti è un processo di difficile gestione, pieno di incertezze e cambiamenti inaspettati. Questo rende complicata la visione anticipata dei risultati e spesso la pianificazione, oltre a essere di difficile stima, diventa rapidamente obsoleta con l'insorgere di nuove informazioni relative al progetto.

Un problema comune a molte aziende risiede quindi nel loro piano di sviluppo prodotto, spesso troppo rigido e schematico per potersi adattare ai nuovi ritmi e bisogni del mercato. L'efficientamento di questi piani di sviluppo prodotto rappresenta una sfida di vitale importanza per queste aziende, soprattutto quelle di piccole medie dimensioni che vogliono emergere nel mercato.

All'inizio degli anni 2000 sono emersi nuovi metodi di gestire lo sviluppo prodotto all'interno dell'industria software. I creatori di tali metodi nello sviluppo software si sono riuniti e hanno dato vita nel 2001 a "The Agile Manifesto", che comprende i capisaldi dei nuovi metodi di sviluppo prodotto Agile. Le teorie seguite e che si sono sviluppate sono molte e differenti tra loro, ma tutte sono basate sullo sviluppo del prodotto in maniera iterativa e incrementale, team di sviluppo cross-funzionali e auto organizzati e una rapida risposta al cambiamento. [1]

Tra la grande varietà di metodi, quello maggiormente diffuso nell'industria software è Scrum, utilizzato attualmente anche da grandi compagnie come Microsoft e Google.

A prima vista sembrerebbe che questo schema sia applicabile esclusivamente al contesto di sviluppo software. Se si studia più nel dettaglio il framework di Scrum, soprattutto analizzando le guide ad esse dedicate (SBOK), ci si accorge come non vi sia nessun riferimento particolare all'industria software. [2] Si potrebbe pensare quindi che tale metodo possa essere applicato per la gestione dello sviluppo di qualsiasi tipologia di prodotto. Le evidenze nella letteratura sono purtroppo piuttosto scarse e i casi studio di aziende che applicano metodi di puro Agile al di fuori del contesto informatico sono praticamente nulli.

1.1 Scopo del lavoro

Lo scopo di questa tesi è investigare se Scrum possa essere adottato e utilizzato con successo nella gestione dello sviluppo del nuovo prodotto al di fuori del suo contesto tradizionale, valutare la necessità di modificarne alcuni aspetti per adattarlo al nuovo contesto e stabilire se la sua applicazione può davvero incrementare l'efficienza dello sviluppo prodotto. L'obiettivo della trattazione sarà quello dell'ideazione e implementazione di un piano di sviluppo prodotto non tradizionale, maggiormente flessibile e reattivo e con caratteristiche tipiche dell'Agile. Sarà obiettivo della ricerca anche l'organizzazione delle risorse, dei team di sviluppo e i metodi di cooperazione tra team.

La ricerca è stata effettuata individuando come soggetto le PMI fornitrici di ingegneria e design, con una particolare attenzione al settore automotive, per restringere il campo di applicazione e avere un'analisi maggiormente approfondita. Si tratta di aziende che non producono direttamente prodotti fisici ma che si occupano della parte di progettazione, ricerca di stile, analisi virtuale e test del prodotto.

Un caso studio dettagliato è stato condotto all'interno Blue Engineering srl, una società di Engineering&Design con sede a Rivoli (TO) e all'interno di questa tesi vi è un'analisi dei risultati ottenuti dalla sperimentazione dei nuovi metodi di

sviluppo prodotto all'interno di questa azienda limitatamente al campo automotive.

1.2 Struttura della tesi

Avendo stabilito gli obiettivi della tesi, la struttura dell'elaborato si sviluppa secondo il seguente schema:

- Il Capitolo 2 fornisce una panoramica sui metodi di gestione dello sviluppo prodotto maggiormente diffusi, risultato di un'approfondita analisi della letteratura. In particolare, sono descritti il modello tradizionale Stage-Gate, i principi dello sviluppo Agile, lo Scrum e alcuni modelli ibridi che fondono insieme metodi tradizionali e di gestione Agile. Per ciascun modello sono evidenziate le principali caratteristiche, i vantaggi e svantaggi della loro applicazione.
- Nel Capitolo 3 viene delineato il profilo delle aziende obiettivo della ricerca e successivamente proposto un modello di sviluppo prodotto adatto a tali realtà. Sono descritte le principali fasi, proposta la struttura organizzativa dei progetti e descritti metodi e strumenti necessari ad una corretta implementazione del piano.
- Il Capitolo 4 tratta il caso empirico condotto in Blue Engineering. Sono evidenziate le differenze rispetto allo studio teorico e sono mostrati esempi applicativi del piano di sviluppo nuovo prodotto, evidenziando benefici e difficoltà incontrate.
- Il Capitolo 5 riporta le conclusioni in merito al lavoro effettuato e si sofferma su possibili aspetti che potrebbero essere approfonditi in futuro al fine di migliorare l'efficienza dei processi di sviluppo del prodotto.

2 Analisi della letteratura

In questo capitolo sono mostrati i risultati dell'analisi della letteratura. È stata svolta questa ricerca per avere una panoramica su tutti i metodi di sviluppo prodotto e avere una solida base di conoscenze relative all'argomento. Di ciascun metodo sono indicati principali caratteristiche, contesti applicativi, vantaggi, svantaggi e alcune evidenze empiriche tratte da casi studio disponibili dalla letteratura. L'apprendimento delle diverse tipologie di gestione dello sviluppo prodotto ha permesso di modellare al meglio il piano di sviluppo prodotto all'interno del contesto applicativo selezionato, cioè le PMI fornitrici di Engineering e Design.

2.1 Stage-Gate Model

Il modello che viene definito Stage-Gate è stato idealizzato ed esposto per la prima volta nel libro "Winning at new products" (Robert G. Cooper, 1986) come evoluzione dei modelli di sviluppo prodotto presenti fino a quel momento, definiti *phased review process* e chiamati così prendendo il nome dal più famoso di questi modelli, quello della NASA (National Aeronautics and Space Administration). L'idea comune di tali modelli era la suddivisione del processo di sviluppo di un nuovo prodotto in periodi distinti, chiamati appunto *stage* o *fasi*, ognuno dei quali seguito da un momento decisionale, il *gate*. [3]

Il modello originale della NASA e quelli nati nello stesso periodo, ovvero intorno agli anni '60, prevedevano il frazionamento del processo di innovazione in cinque principali fasi, ognuna delle quali riferita ad una specifica funzione o dipartimento. La principale innovazione introdotta da Cooper nel 1986 fu l'inserimento di stage cross-funzionali e non più controllati da un singolo dipartimento, creando un processo che abbracciava tutte le aree di business dell'impresa; prevedeva attività svolte in parallelo all'interno degli stage mantenendo però la logica *waterfall* del processo nel suo insieme. Queste

migliorie portarono evidenti benefici in termini di time to market e furono seguite da altri perfezionamenti proposti dallo stesso Cooper in sue opere successive, seguendo i bisogni e le evoluzioni del mercato. [4]

Il modello di Stage-Gate rappresenta dunque una mappa concettuale e operativa per portare un nuovo prodotto dall'idea al lancio sul mercato. Si può applicare a qualsiasi contesto e settore industriale o di ricerca, indipendentemente dalla complessità del prodotto. Ha quindi una valenza universale. [5]

Si analizzeranno ora nel dettaglio le principali caratteristiche e peculiarità.

Uno stage è un periodo di tempo discreto e predefinito e il processo di sviluppo di un nuovo prodotto viene suddiviso tipicamente in sei stages. All'interno di ogni stage i vari attori eseguono azioni prestabilite che sono necessarie a completare le attività di sviluppo e raccogliere le informazioni utili per portare il progetto in avanti verso il successivo punto di decisione. Ogni stage è cross-funzionale: non esiste lo stage del marketing o lo stage di progettazione, ma in ogni stage troviamo un contributo essenziale di più funzioni aziendali (marketing, ingegneria, qualità, etc.). Le attività all'interno di ogni stage sono svolte parallelamente dai membri delle diverse funzioni. Tali attività servono a raccogliere informazioni e ridurre il livello di incertezza e rischio del progetto. Più il progetto avanza, più si riducono le incertezze e si hanno maggiori dati sui rischi che possono così essere evitati; allo stesso tempo, ogni stage ha un costo superiore a quello precedente, mitigato per l'appunto da una minore presenza di incertezza e rischio. Come si può notare dalla **Figura 1**, il processo appare molto lineare e segue una logica waterfall; in realtà, negli sviluppi più recenti del modello, all'interno di ogni stage vi sono iterazioni e cicli che rendono il processo più snello.

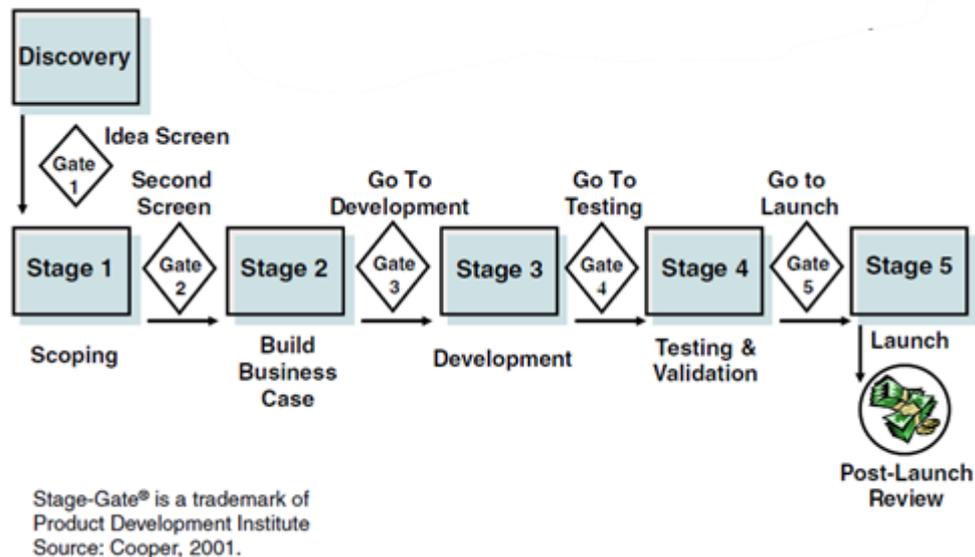


Figura 1. Flowchart illustrante il processo di sviluppo prodotto con il metodo Stage-Gate.

Al seguito di ogni stage troviamo un *gate*, ovvero un punto decisionale in cui l'esito può essere Go/Kill/Revise: nel primo caso il progetto di sviluppo prosegue, nel secondo il progetto viene interrotto perché non ritenuto più abbastanza profittevole mentre nel terzo caso la decisione prevede di rivedere alcuni aspetti incompleti o non perfetti prima di procedere allo stage successivo. La struttura di ogni gate è molto simile e prevede principalmente tre diversi elementi: deliverable, criteri e output.

I deliverable sono il risultato delle attività svolte durante lo stage precedente, quindi incrementi di progetto contenenti informazioni aggiuntive sullo sviluppo del nuovo prodotto. Sono oggetti o parti di progetto che devono essere tangibili o visibili per poter essere giudicati correttamente nel gate. Tali deliverable costituiscono quindi l'input al gate.

I criteri sono le regole secondo le quali i deliverable sono giudicati. Sono previsti criteri che devono essere rispettati assolutamente (*must meet criteria*), spesso sotto forma di check-list, che servono a eliminare e stoppare immediatamente progetti errati e fallimentari, e altri che sono giudicati opzionali e che portano valore aggiunto più elevato (*should meet criteria*).

Gli output di ogni gate sono numerosi e diversi tra loro. Il più importante è sicuramente la decisione Go/Kill/Revise che consente al prodotto di proseguire o no nel suo sviluppo; altri output importanti sono ad esempio una pianificazione dettagliata delle attività da svolgere durante lo stage successivo, indicando personale coinvolto, costi e tempi, oppure la lista dei deliverable di quel particolare stage, che costituiranno l'input del gate successivo.

Nei vari gate troviamo i cosiddetti *gatekeepers*, ovvero le persone incaricate di prendere la decisione Go/Kill. Le figure aziendali che rappresentano i gatekeepers vanno indicati nel piano di sviluppo prodotto dell'azienda. La regola per la loro scelta prevede che essi facciano parte del senior management dell'impresa o del particolare business coinvolto e che siano i responsabili delle risorse che servono al project leader e al team per proseguire con in progetto. Trattandosi di team cross-funzionali e di attività che coinvolgono diverse funzioni nell'azienda, gli stessi gatekeepers fanno parte di diverse funzioni. È utile che per ogni gate vi sia una matrice delle responsabilità, in cui vengono indicate le varie attività svolte durante lo stage e i diversi responsabili dell'approvazione.

Come anticipato in precedenza, il modello classico di Stage-Gate prevede l'alternanza di cinque stage e cinque gate, come mostrato in **Figura 1**, preceduti da uno stage di scoperta e conclusi da un gate terminale di revisione. Si analizzeranno ora nel dettaglio le attività e le caratteristiche di ciascuna fase.

2.1.1 Idea Stage: Scoperta

Viene anche definito Stage 0 ed è uno stage aggiuntivo rispetto ai canonici cinque. Molte aziende ritengono che l'attività di ideazione sia così importante da dedicarle uno stage completo, con un processo di generazione e acquisizione di idee ben definito. Le idee infatti costituiscono la materia prima del processo di Stage-Gate. Il grande bisogno di idee e l'elevato tasso di abbandono di esse durante i primi stage fanno sì che il processo di generazione delle idee sia

fondamentale: si ha bisogno di grandi idee e in gran numero. Le attività svolte in questo stage sono, ad esempio: ricerche tecniche per scoprire nuove possibilità tecnologiche; collaborazioni con i clienti e utenti principali e più innovativi alla ricerca di bisogni insoddisfatti; utilizzo di metodi di ideazione creativa, come i brainstorming; pianificazioni strategiche per individuare zone libere e non sfruttate del mercato che rappresentano opportunità da cogliere.

2.1.2 Gate 1: Selezione dell'idea

È la prima decisione che viene presa; da una serie di idee scaturite dal processo di generazione di idee ne viene scelta una e tale selezione costituisce la nascita formale del progetto. I criteri che vengono seguiti per la scelta sono di tipo qualitativo e sono ad esempio l'allineamento strategico, la fattibilità del progetto, l'attrattività del mercato, l'allineamento con le competenze delle risorse dell'impresa. In questo gate non vengono presi in considerazione criteri quantitativi come ad esempio quelli finanziari.

2.1.3 Stage 1: Scoping

Questo primo stage è piuttosto economico e di breve durata, in quanto viene solitamente completato in meno di un mese di lavoro. L'obiettivo di questo stage è l'identificazione dei meriti tecnici e di mercato del progetto, facendone una breve panoramica. Non vengono svolte ricerche approfondite ma solo valutazioni preliminari "a tavolino" e investigative; in particolare vi sono valutazioni preliminari del mercato, tecniche e dell'impresa.

2.1.4 Gate 2: Seconda selezione

Questo gate è praticamente la ripetizione del Gate 1, alla luce delle nuove ricerche e informazioni raccolte durante lo scoping. Il progetto viene rivalutato più rigorosamente e vengono utilizzati gli stessi criteri indicati precedentemente, con l'aggiunta di criteri finanziari di facile calcolo e interpretazione, come ad esempio il *payback period*. Se la decisione è Go, il

progetto prosegue verso gli stage successivi, che diventeranno sempre più costosi.

2.1.5 Stage 2: Creazione del Business Case

Il business case spalanca le porte allo sviluppo del prodotto vero e proprio. Durante lo Stage 2 viene condotta una dettagliata attività investigativa e di ricerca per definire il prodotto e la sua attrattività prima delle ingenti spese di sviluppo. In alcuni casi viene chiamato *critical homework stage* per dar enfasi alla sua criticità e alla caratteristica di non essere ancora entrati nell'attività di sviluppo prodotto, ma ci si trova nell'ultima fase di ricerca a tavolino (da cui la parola homework).

Le attività principali di questo stage sono:

- Ricerche di mercato, previsioni dei volumi e interviste ai potenziali clienti (la cosiddetta *Voice of Customer, VoC*) per individuare i bisogni e le preferenze così da identificare il prodotto vincente;
- Analisi dei competitors del mercato;
- Test sui concept per convalidare i concept del prodotto scelti;
- Analisi della fattibilità tecnica del progetto;
- Analisi della fattibilità produttiva del progetto (sistemi di produzione e fonti di approvvigionamento);
- Definizione del nuovo prodotto vincente che sarà successivamente sviluppato. Vengono definiti il mercato di destinazione, il concept del prodotto, la strategia di posizionamento, la value proposition e gli attributi, i requisiti e le specifiche ad alto livello del prodotto;
- Analisi finanziaria dettagliata comprendente l'analisi dei flussi di cassa e il calcolo del NPV (Net Present Value) e dell'IRR (Interest Return Rate), completa di analisi dei potenziali rischi.

Il risultato finale è un business case completo del progetto. Il prodotto definito viene quindi accettato e viene elaborata una prima pianificazione delle attività di sviluppo. Lo Stage 2 comporta uno sforzo maggiore rispetto agli stage precedenti, sia in termini di ore-uomo che in termini economici. Viene svolto da un team cross-funzionale per gestire al meglio la varietà di aspetti del progetto.

2.1.6 Gate 3: Via allo sviluppo

È l'ultimo gate prima dello stage relativo allo sviluppo del prodotto; in questo gate si decide se proseguire allo sviluppo entrando quindi nelle fasi in cui i costi crescono esponenzialmente, oppure decidere di interromperlo perdendo i costi già sostenuti ma risparmiando le ingenti uscite future.

Nel Gate 3 vengono controllate e revisionate tutte le attività svolte durante lo Stage 2, verificando che le attività previste siano state effettivamente svolte e valutandone il compimento. I criteri utilizzati sono a grandi linee gli stessi utilizzati nel gate precedente, ma ora sono molto più rigorosi dal momento che si dispone di molte informazioni aggiuntive. Hanno grande importanza ora i risultati delle analisi finanziarie, dal momento che la decisione di Go comporta un grande impegno economico durante le successive fasi.

In caso di esito positivo del gate (Go), viene accordata la pianificazione delle attività future. Vengono approvati i piani di sviluppo, le operazioni preliminari e i piani di marketing.

2.1.7 Stage 3: Lo Sviluppo

Durante lo Stage 3 vengono messe in atto le attività pianificate al Gate 3 e che portano allo sviluppo fisico del prodotto. Durante questa fase vengono fatti molti test di laboratorio e test interni per verificare che il prodotto soddisfi i requisiti sotto certe condizioni. Se si tratta di progetti molto lunghi o di prodotti complessi da sviluppare, possono essere previste numerose *milestone*

intermedie o *design review* periodiche per verificare il corretto andamento del progetto; non si tratta di veri e propri gate in cui vengono prese le decisioni di Go/Kill, ma sono molto utili per controllare e gestire al meglio il progetto. Il maggior sforzo di questo stage è incentrato sullo sviluppo tecnico del prodotto, e il deliverable principale è un prototipo del prodotto funzionante e che abbia superato i test interni.

Parallelamente allo sviluppo tecnico, procedono le attività di marketing e controllo del mercato, con frequenti incontri con i clienti per avere feedback aggiornati che possano essere integrati nello sviluppo del prodotto. Vengono aggiornati i piani di lancio e vengono portate al cliente i progressi dello sviluppo, affinché possa valutarli e fornire un importante feedback. Le analisi finanziarie e dei rischi vengono aggiornate periodicamente utilizzando le maggiori informazioni disponibili. Durante questo stage vengono anche sviluppati piani di produzione e si studia il processo produttivo, in contemporanea con lo sviluppo del prodotto.

2.1.8 Gate 4: Via ai test

Vengono verificati e valutati i progressi dello sviluppo e l'attrattività del prodotto. Tramite apposite checklist, il lavoro di sviluppo viene controllato, assicurandosi della completezza delle attività svolte e che il prodotto sviluppato sia coerente alle specifiche tecniche e ai requisiti stabiliti durante la definizione del prodotto. Vengono rivisti gli aspetti economici del progetto, valutando l'analisi finanziaria aggiornata con i nuovi dati emersi durante lo sviluppo. Infine, vengono approvati i piani di test e convalida che verranno svolti durante lo stage successivo e i piani di marketing e del processo produttivo vengono revisionati.

2.1.9 Stage 4: Test e validazione

Questo stage testa e convalida l'intero progetto in tutti i suoi aspetti: il prodotto stesso, il processo produttivo, l'accettazione del cliente e tutti gli aspetti

economici del progetto. In questo stage continuano i test interni e iniziano numerosi test esterni per convalidare il prodotto:

- Test interni sul prodotto per verificare la qualità e le prestazioni del prodotto in condizioni controllate e prestabilite;
- Prova del cliente finale sul campo per verificare che il prodotto si comporti bene in condizioni di utilizzo effettive. Serve anche a conoscere le opinioni dell'utente, che potrebbe essere un potenziale cliente futuro;
- Produzione di prova, per testare i macchinari e il processo produttivo e determinare più nel dettaglio i costi e la capacità produttiva. È in questo momento che vengono acquistati o costruiti i macchinari per la produzione;
- Simulazione test di mercato e vendite di prova, per valutare le reazioni dei clienti, misurare l'efficacia del piano di lancio e stimare la quota di mercato, i volumi di vendita e i relativi ricavi;
- Revisione delle previsioni e analisi finanziarie, sempre più accurate alla luce dei nuovi dati sui costi di produzione e sui ricavi attesi.

Spesso i risultati dei test non sono positivi, e ciò comporta un arretramento del progetto allo Stage 3 di sviluppo, per affinare e modificare il prodotto sulla base delle indicazioni fornite dai test.

2.1.10 Gate 5: Via al lancio

Questo gate dà il via al lancio commerciale e alla piena produzione del prodotto. È l'ultimo momento in cui il gate può avere esito Kill, nonostante i grandi sforzi economici fin qui sostenuti. Viene valutata la qualità e i risultati dei test svolti in precedenza. I criteri per il superamento del gate sono principalmente di tipo finanziario e operativo; vengono valutati i rendimenti economici attesi derivati dai test di mercato e sulla base di questi si studia la profittabilità del prodotto. Viene inoltre valutata l'adeguatezza del processo

produttivo e degli impianti che dovranno essere utilizzati per la produzione in massa. Vengono esaminati ed approvati i piani di marketing per il lancio commerciale.

2.1.11 Stage 5: Il lancio

È l'ultimo stage previsto dal modello di Stage-Gate. Viene implementato il processo produttivo e inizia la produzione in serie del prodotto. Parallelamente viene condotta la fase di lancio sul mercato, con campagne pubblicitarie mirate e la consegna dei primi prodotti ai clienti o punti vendita. Se le attività svolte fino a questo momento sono state svolte correttamente e le decisioni prese ai gate sono state coerenti, il prodotto lanciato sul mercato dovrebbe rispettare le aspettative ed essere un successo commerciale.

2.1.12 Post-Launch review

Dopo qualche mese dalla commercializzazione del prodotto, tipicamente da 3 a 6 mesi, il progetto relativo al nuovo prodotto può considerarsi ufficialmente terminato. Vengono rivisti i ricavi, costi, le spese e le tempistiche e le si paragonano alle stime effettuate durante la creazione dei business case. Questo processo è utile a valutare le forze e le debolezze del progetto e a imparare la lezione per portare a termine nel futuro un progetto migliore.

2.2 Scrum

Le tecniche di project management Agile si sono sviluppate a partire dagli anni '90 ed hanno avuto una maggior diffusione a seguito della pubblicazione, nel 2001, de "The Agile Manifesto", nel quale vengono presentati i quattro capisaldi di queste nuove metodologie, adatte ad ambienti di lavoro turbolenti e fortemente instabili, con frequenti cambi di requisiti e bisognosi di risposte rapide. [1]

I quattro fondamenti tratti da "The Agile Manifesto" sono:

- Le persone e le loro interazioni più importanti dei processi e degli strumenti.
- Un software funzionante più importante di una documentazione omnicomprensiva.
- La collaborazione con il cliente più importante della negoziazione di un contratto.
- Rispondere al cambiamento più importante di seguire un piano.

Come si evince dall'elenco precedente, queste caratteristiche sono tipiche degli ambienti di sviluppo software, ed è in questo settore in cui Agile ha riscontrato da subito il maggior successo. Negli anni si sono sviluppati diversi metodi Agile tra cui Scrum, Crystal, Agile Modeling etc, ognuno con i suoi vantaggi e corredato da una serie di strumenti da utilizzare. Di tutti questi, Scrum è stato esplicitamente creato per gestire i progetti lungo il loro processo di sviluppo.

Nato originariamente per lo sviluppo di prodotti nell'industria software, ha iniziato a suscitare interesse nei dipartimenti di R&S di molte aziende manifatturiere per lo sviluppo di prodotti fisici trattandosi di una metodologia adattiva, iterativa, veloce, flessibile ed efficace, finalizzata alla realizzazione di valore in tempi rapidi e lungo tutta la durata del progetto. Scrum assicura la trasparenza delle comunicazioni e massimizza la responsabilità collettiva

dell'ambiente di lavoro. Scrum prevede l'utilizzo di team cross-funzionali, auto organizzati e investiti di potere che suddividono le attività lavorative in brevi cicli dalla durata prestabilita, chiamati *sprint*. Le tecniche di Scrum, così come descritte nella Guida SBOK (Scrum Body Of Knowledge), sono pensate per essere adottate da qualsiasi tipo di progetto, di qualsiasi dimensione e di qualsiasi settore industriale.

Si cercherà ora di illustrare brevemente le principali caratteristiche di tale metodologia, indicando gli artefatti, i ruoli, i principi e gli strumenti di Scrum.
[2]

2.2.1 I principi

I principi dello Scrum sono sei, e devono essere mantenuti il più possibile integri per assicurare che il framework Scrum possa far raggiungere gli obiettivi del progetto.

1. **Controllo empirico di processo.** Questo principio enfatizza il cuore della filosofia Scrum, basata sulle tre idee di:
 - **Trasparenza:** è permesso a chiunque di osservare tutti gli aspetti di qualsiasi processo/progetto Scrum all'interno dell'impresa. Ciò garantisce un flusso trasparente di informazione e promuove la creazione di una cultura di lavoro aperta.
 - **Ispezione:** è rappresentata dall'utilizzo di una scrumboard comune in cui vengono mostrati i progressi dello sprint in corso o dalla raccolta di feedback dal cliente e da altri stakeholder.
 - **Adattamento:** avviene quando tramite la trasparenza e l'ispezione i membri dello scrum team acquisiscono le informazioni necessarie e si adattano incorporandole nel lavoro che stanno portando avanti.
2. **Auto-organizzazione.** Negli ambienti di lavoro attuali, i lavoratori realizzano un valore significativamente maggiore quando hanno la possibilità di auto-organizzarsi; ciò si traduce in un'assunzione di

responsabilità condivisa da parte del team e in un ambiente di lavoro innovativo che stimola maggiormente la crescita.

3. **Collaborazione.** Il team di sviluppo collabora per trarre vantaggio dai reciproci contributi per creare e consegnare deliverable che hanno un valore superiore alla somma dei contributi di ogni singolo individuo. La collaborazione si basa sui concetti di consapevolezza, articolazione e appropriazione.
4. **Prioritizzazione basata sul valore.** Questo principio evidenzia l'intenzione di Scrum di realizzare il massimo valore dalle primissime fasi del progetto, nel minor tempo possibile e farlo continuamente nel tempo. Per consegnare il massimo valore nel minor tempo possibile si deve utilizzare lo strumento della prioritizzazione.
5. **Time-boxing.** In Scrum il tempo è considerato un vincolo limitante e da rispettare assolutamente. Tale vincolo viene utilizzato per gestire efficacemente la pianificazione e l'esecuzione del progetto. Gli elementi per i quali il tempo è predeterminato sono gli *sprint*, i *daily standup meetings*, gli *sprint planning meetings* e gli *sprint review meetings*.
6. **Sviluppo iterativo.** Questo principio definisce lo sviluppo iterativo ed evidenzia come gestire al meglio i cambiamenti dei bisogni del cliente e dei requisiti da sviluppare. Il responsabile di questi cambiamenti è il *Product Owner* che si interfaccia con il cliente e porta a conoscenza il suo feedback.

2.2.2 Elementi di Scrum

Nella **Figura 2** è mostrata una mappa che riassume graficamente gli elementi principali di un progetto Scrum. Scrum si suddivide in tre elementi che sono organizzazione, attività time-boxed e strumenti (chiamati anche artefatti).

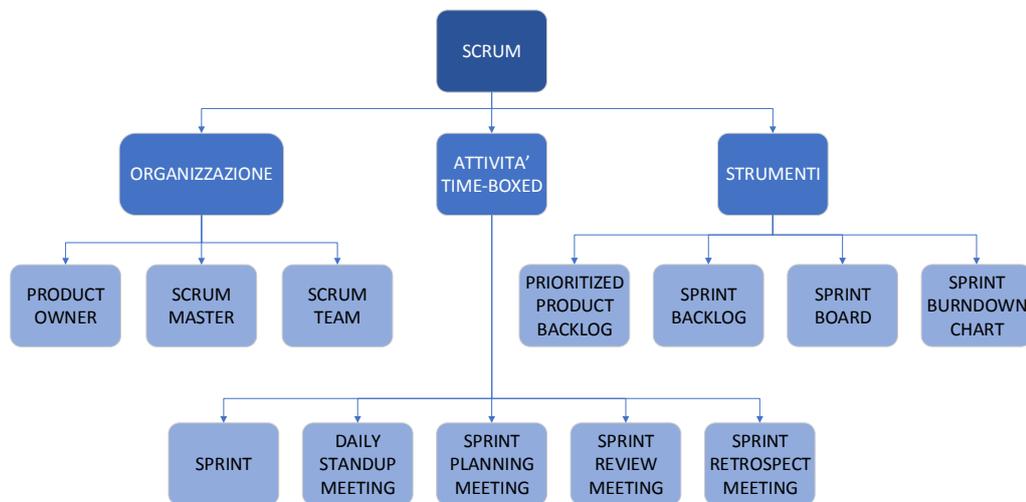


Figura 2. Diagramma riassuntivo di tutti gli elementi tipici di un progetto gestito tramite Scrum.

2.2.3 Organizzazione

In Scrum gli attori che all'interno dell'organizzazione si dividono in *ruoli core* e *ruoli non core*. In questo paragrafo si descriveranno i ruoli core, la cui presenza è necessaria per tutti i progetti che seguono metodi Scrum.

- **Product Owner.** Il Product Owner è il responsabile che ha l'obiettivo di trarre il massimo valore dal progetto. Deve far sì che i requisiti di qualità e funzionalità desiderati dal cliente siano comunicati correttamente allo Scrum Team. Rappresenta quindi gli interessi degli *stakeholder* di fronte al team di sviluppo. Deve sempre mantenere un duplice punto di vista, facendo rispettare gli interessi di tutti gli stakeholder e allo stesso tempo comprendendo le esigenze dello Scrum Team. Rappresenta la Voice of Customer e non necessariamente deve far parte del team di sviluppo, ma può essere un membro del management aziendale.
- **Scrum Master.** È un facilitatore che insegnando le tecniche e le pratiche di Scrum, crea un ambiente di lavoro che favorisca il completamento dello sviluppo del prodotto. Nello SBOK viene definito come *servant leader*, cioè un leader servitore, allo stesso livello degli altri membri del team; non ha il compito di gestire il team ed essere il leader del progetto

come nel project management waterfall classico, ma si limita a favorire i processi e limare ogni contrasto fungendo da motivatore. Qualsiasi membro del team che conosca bene le tecniche Scrum può svolgere il ruolo di Scrum Master per l'intero progetto o per un singolo sprint.

- **Scrum Team.** È il team di sviluppo vero e proprio, cross-funzionale e responsabile dello sviluppo del prodotto. È generalmente composto da poche persone, non più di dieci, che preferibilmente devono lavorare a stretto contatto, nella stessa stanza o in un ambiente a loro dedicato, per favorire la comunicazione e l'utilizzo di alcune tecniche Scrum che altrimenti sarebbe difficile utilizzare. Può essere flessibile e cambiare membri durante la durata del progetto. Ha come obiettivo finale la creazione dei deliverable.

Altri ruoli non core ma comunque importanti all'interno del mondo Scrum sono gli stakeholder, termine generico che comprende tutti coloro i quali sono interessati allo svolgimento e allo sviluppo del prodotto. Sono compresi naturalmente i clienti, gli utenti finali, gli sponsor. Spesso alcune di queste figure possono coincidere, ad esempio cliente e sponsor possono essere la stessa persona.

2.2.4 Attività time-boxed

In Scrum sono molto importanti le attività a durata predeterminata, in quanto il tempo viene considerato una risorsa limitata. Il time-boxing porta alcuni vantaggi come l'efficienza dei processi, minori spese generali e alta velocità del team di sviluppo. Inoltre, essendo a conoscenza della durata delle attività, il team di sviluppo prende in carico solo la quantità di lavoro che sa di essere in grado di portare a termine, evitando sprechi di risorse o ritardi nelle consegne.

Le attività a durata predeterminata di Scrum sono:

- **Sprint.** Lo sprint è il cuore di tutta la metodologia Scrum. Si tratta di un periodo di tempo dalla durata fissa e prestabilita che va dalle due alle sei settimane, anche se si raccomanda una durata di quattro settimane per massimizzare i benefici, in cui lo Scrum Team lavora a un incremento di prodotto usabile o potenzialmente consegnabile al cliente. Durante uno sprint non possono essere fatti cambiamenti che possano mettere a rischio l'obiettivo dello sprint e le attività svolte sono quelle che vengono decise all'inizio di ogni sprint.
- **Daily standup meeting.** Anche chiamato Daily Scrum, è una breve riunione quotidiana della durata di circa 15 minuti in cui i membri del team di riuniscono per fare un resoconto sull'avanzamento del progetto, rispondendo a tre semplici domande:
 - cosa ho completato ieri che ha aiutato il team a raggiungere l'obiettivo dello sprint?
 - cosa farò oggi per aiutare il team a raggiungere l'obiettivo dello sprint?
 - quali ostacoli o problemi ho riscontrato?

Questa riunione, condotta e facilitata dallo Scrum Master, favorisce la collaborazione del team e permette di prevedere le prossime attività da completare.

- **Sprint planning meeting.** È una riunione che viene fatta all'inizio di ogni sprint ed ha la durata massima di 8 ore per uno sprint di un mese. È portata avanti come attività collaborativa di tutto il team: il Product Owner presenta i requisiti che hanno la priorità più alta e il team decide quindi l'obiettivo dello sprint e quali requisiti possono essere sviluppati e incrementati. Nella seconda parte della riunione si fa una pianificazione mensile delle attività da svolgere per raggiungere l'obiettivo dello sprint.

- **Sprint review meeting.** Ha una durata di 4 ore per uno sprint di un mese. Si svolge al termine dello sprint ed ha lo scopo di presentare al Product Owner gli incrementi e i deliverable completati durante lo sprint appena concluso. Il Product Owner rivede il prodotto e in base ai criteri di accettazione concordati decide di accettare o rifiutare il lavoro completato. Il risultato di questo incontro è un *product backlog* rivisto e riprioritizzato che sarà la base dei successivi sprint. Tale backlog può essere rivisto sia per il completamento di alcune attività che prima erano presenti, sia per delle modifiche e nuove richieste del cliente emerse solo durante lo sprint.
- **Retrospect sprint meeting.** Ha anch'esso una durata prestabilita di 4 ore per uno sprint di un mese. Durante questa riunione lo Scrum Team si riunisce per guardarsi alle spalle e riflettere sugli aspetti positivi e negativi dello sprint appena concluso. Vengono analizzati i processi seguiti, gli strumenti utilizzati, la comunicazione e la collaborazione all'interno del team per evidenziare eventuali problemi e trovare soluzioni, ma anche per sottolineare aspetti positivi o problemi passati cui si è trovata una soluzione. Questa riunione è guidata dallo Scrum Master, che con il suo ruolo di facilitatore agevola la comunicazione tra i membri del team e favorisce il confronto.

2.2.5 Gli strumenti Scrum

La metodologia di sviluppo prodotto secondo Scrum utilizza numerosi strumenti, spesso chiamati *artefatti*, che servono a massimizzare la trasparenza, migliorare il flusso di informazioni, il tutto con il fine ultimo di consegnare al cliente il maggior valore nel minor tempo possibile.

Gli strumenti principali utilizzati in Scrum sono:

- **Prioritized product backlog.** Il product backlog è un elenco di tutto ciò che è necessario completare per lo sviluppo del prodotto: requisiti e

bisogni del cliente, requisiti funzionali, correzioni. Come suggerisce il nome, i requisiti o le caratteristiche del prodotto sono prioritizzate secondo diverse logiche, soprattutto basandosi sul valore. Il responsabile del product backlog è il Product Owner, sia del suo contenuto sia dell'ordine delle attività. Il product backlog è uno strumento dinamico, i cui contenuti e il cui ordine cambia e si evolve in base alle nuove informazioni derivanti dal progetto, dai cambiamenti richiesti dal cliente e dagli insegnamenti appresi dal team di sviluppo. Questa caratteristica, la dinamicità, è la caratteristica core di Scrum e permette una risposta rapida ai cambiamenti e l'adattabilità ad ogni tipo di progetto.

I metodi per prioritizzare gli elementi del product backlog sono molti e sono principalmente basati sul valore che i requisiti sviluppati andranno a conferire al prodotto finale. Il criterio più efficace è lo schema MoSCoW, che classifica i requisiti da quelli più importanti e che non possono mancare (Must have), passando per Should have, Could have e Won't have, per quei requisiti che hanno un'importanza sempre minore.

A differenza dei metodi classici in cui per la definizione del prodotto si indicavano caratteristiche tecniche e/o di performance molto dettagliate, in Scrum si preferisce mantenerli a livello più alto e descrittivo, indicando bisogni e preferenze del cliente affinché il team di sviluppo sia più libero nel progettare le giuste soluzioni.

Gli elementi all'interno del backlog sono scritti sotto forma di User Story, ovvero brevi frasi che dicono tre aspetti fondamentali per ciascun requisito: *Chi*, *Cosa* e *Perché*. Sono dichiarazioni brevi e di facile interpretazione e seguono il seguente schema:

"Nella mia qualità di <Ruolo> devo essere in grado di <Requisito/Attività> in modo da <Beneficio>"

- **Sprint backlog.** È il corrispettivo del product backlog per un singolo sprint. Contiene i requisiti e le caratteristiche da sviluppare all'interno di quello sprint e viene definito all'inizio di ogni sprint durante la riunione di sprint planning. Lo sprint backlog tendenzialmente contiene gli elementi in cima al product backlog, cioè quelli a più alta priorità. Lo scrum team però prende in carico soltanto il numero di attività che sa di poter svolgere in un mese.

La regola generale vieta di modificare lo sprint backlog dopo che lo sprint è partito, per non creare confusione; è possibile però che alcune attività possano essere inserite in un secondo momento se sono state fatte gravi dimenticanze o errori.

- **Sprint board.** Anche chiamata kanban board, è un'estensione dello sprint backlog. È un metodo di pianificazione del lavoro visiva. È una lavagna o parete suddivisa in tre colonne: to do, work in progress, done; le attività e i requisiti contenuti nello sprint backlog sono scritti su post-it e assegnati a un membro del team che, man mano che li completa, li sposta lungo le varie colonne. Può essere utile utilizzare anche delle righe con indicati ciascun membro del team così da incrociare l'attività svolta con chi la sta sviluppando.
- **Sprint burndown chart.** È un grafico che mostra l'andamento dei progressi durante uno sprint e ne monitora gli obiettivi. Sull'asse delle ordinate viene mostrato il lavoro ancora da compiere misurato in ore-uomo, giorni-uomo o story points; sull'asse delle ascisse invece viene mostrato il tempo dello sprint. Giorno dopo giorno il lavoro rimanente diminuisce per arrivare a zero idealmente l'ultimo giorno dello sprint seguendo una linea retta. Nel grafico viene quindi rappresentata la reale situazione dell'avanzamento del progetto e tale misura viene fatta quotidianamente; osservando la posizione della linea corrispondente al lavoro svolto rispetto alla retta ideale si può capire se si è in anticipo

rispetto alla pianificazione (a sinistra della retta azzurra), oppure in ritardo (a destra della retta). Un esempio è mostrato in **Figura 3**.

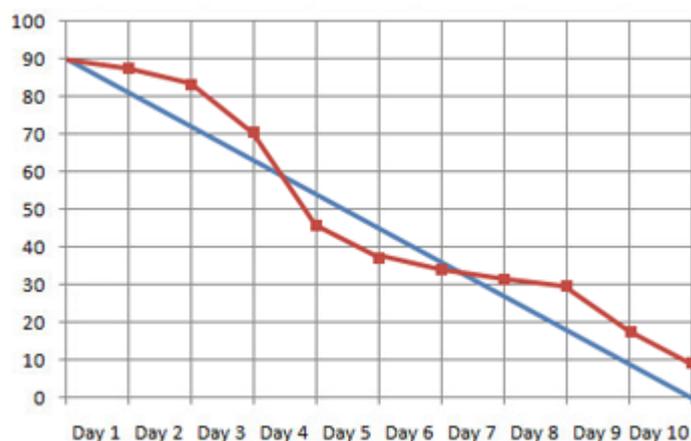


Figura 3. Esempio di burndown chart. La linea azzurra rappresenta l'avanzamento ideale, quella rossa l'avanzamento effettivo. Sull'asse Y l'effort residuo è misurato in story points.

Se vengono osservati dei ritardi, è compito dello Scrum Master identificare le situazioni che ostacolano e impediscono di completare il lavoro con successo e provare a rimuoverle.

Un grafico simile ma con un punto di vista opposto è il burnup chart, in cui viene mostrato la quantità di lavoro svolto durante lo sprint.

2.2.6 Definizione di “done”

Uno degli aspetti più complessi nella teoria di Scrum è determinare quando un'attività eseguita all'interno di uno sprint può essere definita completata e quindi consegnata e presentata al cliente o stakeholder di riferimento. I membri del team devono essere a conoscenza di quale sia la definizione di “done” all'interno del progetto. Spesso vi sono delle linee guida aziendali che lo definiscono, altre volte vengono definiti i criteri di accettazione per lo specifico progetto. In ogni caso è il Product Owner che è responsabile di tali criteri e che decide se un'attività svolta ha soddisfatto a pieno i requisiti e può essere considerata terminata o se occorre rivederla e ri-svilupparla in uno o più ulteriori sprint.

2.2.7 Casi studio

Uno studio condotto da Begel e Nagappan è stata una delle prime ricerche il cui focus era di misurare le performance dei metodi di sviluppo prodotto Agile all'interno di un'azienda. Poiché come spiegato queste tecniche sono nate nel mondo dell'IT, è stata scelta come azienda su cui condurre la ricerca un'importante software house statunitense. Sono stati consegnati dei questionari anonimi per indagare quali fossero le tecniche utilizzate dagli sviluppatori, quali vantaggi o svantaggi avessero riscontrato nell'utilizzo di tali tecniche e quali fossero eventuali migliorie proposte.

I risultati dell'indagine sono stati superiori alle aspettative e hanno rivelato che circa un terzo degli intervistati adottava metodologie Agile allo sviluppo software, e di questi quasi il 70% adottava Scrum, di gran lunga la più utilizzata.

Tra i principali benefici riscontrati dagli utenti di Scrum e delle altre tecniche Agile si trova un significativo miglioramento della comunicazione e della coordinazione. Questo risultato è stato favorito indubbiamente dalla collocazione degli individui del team di sviluppo; in oltre l'80% dei casi, infatti, gli sviluppatori che lavoravano allo stesso progetto erano situati nella stessa stanza o comunque a brevi distanze. I daily meeting inoltre aiutavano molto a far emergere problemi e risolverli in tempi rapidi.

Il secondo beneficio maggiormente segnalato dagli intervistati è stata la velocità di rilascio dei demo dei software, che ha avuto netti miglioramenti dopo l'introduzione delle metodologie Agile. La continua integrazione e la durata prestabilita degli sprint ha facilitato lo sviluppo di funzionalità in maniera più veloce; inoltre la trasparenza di questi metodi ha favorito il monitoraggio delle attività che avveniva molto più frequentemente e non solo in presenza di milestone importanti, a distanza di mesi.

Un altro vantaggio dell'applicazione di Scrum è stato la flessibilità nella progettazione dei software. I frequenti feedback del cliente e la facilità di cambiamento dei requisiti nel product backlog ha portato a una migliore agilità ed efficienza nella risposta ai cambiamenti del progetto.

Altri intervistati hanno infine trovato utile Scrum e l'Agile per migliorare la qualità del servizio offerto, per incrementare la produttività e migliorare il focus sul cliente.

La ricerca ha mostrato anche degli aspetti negativi di Scrum. La preoccupazione principale degli sviluppatori era la scarsa adattabilità di Scrum a progetti di grandi dimensioni. Infatti, tali tecniche sono pensate per team di sviluppo modesti, con una decina di membri al massimo. Trattandosi di una grande società, alcuni progetti avrebbero potuto coinvolgere centinaia di sviluppatori, e Scrum forse difficilmente si può adattare a queste dimensioni. Un altro aspetto legato a quest'ultimo è stata la difficoltà di comunicazione con gli altri team, soprattutto quelli che non utilizzano metodologie Agile.

Per alcuni intervistati i frequenti meeting anziché rappresentare un vantaggio, erano fonte di ostacolo e considerati inefficienti. Se lo Scrum Master e in generale tutto il team aveva poca familiarità con Scrum, le riunioni giornaliere erano fuorvianti e si riducevano alla domanda *"cosa hai fatto nelle ultime 24 ore?"*, aumentando la pressione degli sviluppatori che così vedevano come obiettivo la consegna di una certa quantità di righe di codice, anziché puntare alla qualità del prodotto.

La poca esperienza con la cultura Agile, data la sua relativamente recente diffusione, rendeva infine molti sviluppatori e manager poco familiari con tali tecniche e così facendo le performance lavorative peggioravano. [6]

2.2.8 Confronto tra Scrum e project management tradizionale regolarmente

Tabella 1. Riassunto comparativo tra Scrum e project management tradizionale.

	Scrum	Project management tradizionale
L'attenzione è concentrata su	Persone	Processi
Documentazione	Minima, solo se necessario	Omnicomprendiva
Tipo di processo	Iterativo	Lineare
Pianificazione anticipata	Bassa	Alta
Prioritizzazione dei requisiti	Basata sul valore aggiornata regolarmente	Fissata nel piano del progetto
Garanzia di qualità	Incentrata sul cliente	Incentrata sul processo
Organizzazione	Auto-organizzata	Centralizzato
Stile di gestione	Decentrato	Centralizzato
Cambiamento	Aggiornamento del product backlog	Sistema formale di gestione del cambiamento
Leadership	Collaborativa, leader servitore	Comando e controllo
Misurazione della prestazione	Valore di business	Conformità al piano
Ritorno sull'investimento	Anticipato/lungo tutta la durata del progetto	Alla fine del progetto
Coinvolgimento del cliente	Elevato durante tutta la durata del progetto	Variabile, elevato soprattutto all'inizio

Nella **Tabella 1** sono mostrate le principali differenze tra i metodi di Scrum di gestione di un progetto e sviluppo prodotto rispetto alle tecniche di project management tradizionale. Su molti di questi aspetti si è già discusso precedentemente e questa tabella funge da sommario. In questo paragrafo si vuole approfondire una caratteristica importante di Scrum che è la quasi assenza di pianificazione anticipata. Le attività svolte durante uno sprint infatti non sono mai pianificate in dettaglio precedentemente, così come le attività da fare lungo tutto l'arco di sviluppo del prodotto. All'inizio di ogni sprint, durante lo sprint planning meeting, si decidono quali requisiti e caratteristiche sviluppare durante il mese di sprint, ma non quando svolgerle, cioè in quale preciso istante o giorno del mese. La pianificazione viene fatta quasi quotidianamente, durante i daily standup meeting, utilizzando la scrum board e decidendo insieme quali attività svolgere e in che modo. Non vi è la presenza di un project manager ad un livello gerarchico più alto che decide e smista il lavoro.



Figura 4. Schema di uno Sprint in cui sono evidenziati i principali elementi di Scrum.

Nella **Figura 4** è illustrato uno schema del funzionamento di uno sprint: si parte da un product backlog contenente la lista prioritizzata dei requisiti da sviluppare, da questi se ne scelgono alcuni da completare o incrementare nello

sprint e che vanno a formare lo sprint backlog; i requisiti vengono scomposti e suddivisi in attività elementari e di breve durata, tipicamente non più di un paio di giorni; lo sprint ha una durata prestabilita di 4 settimane nel maggior numero di casi e ogni giorno avvengo i daily meeting per discutere dei problemi; alla fine dello sprint viene rilasciato al Product Owner un incremento del prodotto con nuove funzionalità sviluppate.

È importante sottolineare che essendo lo sprint un'attività time-boxed, al termine delle quattro settimane esso si considerato concluso sia nel caso le attività previste siano state totalmente completate, sia nel caso queste non lo siano. In tal caso, si cerca di capire cosa non ha funzionato all'interno del team e quali sono i motivi che hanno portato al ritardo, e si posizionano le caratteristiche non sviluppate in cima al product backlog con massima priorità per poterli prendere in carico e completarli nello sprint successivo.

2.3 Modelli ibridi Agile-Stage-Gate

Il modello di Stage-Gate originale e nelle sue prime evoluzioni segue fondamentalmente una logica waterfall, con gli stage e le attività al loro interno che si susseguono a cascata. I principali vantaggi di questo modello sono una maggiore velocità di sviluppo e una diminuzione del time to market, una qualità superiore del prodotto, una maggior disciplina e in generale delle performance migliori rispetto ai processi di sviluppo prodotto non formalizzati. La sua forte disciplina e rigidità, se da un lato può essere vista come un punto di forza, dall'altro rappresenta anche un grande limite, non consentendo una risposta rapida e flessibile ad eventuali cambiamenti del mercato o delle richieste del cliente. Partendo da queste necessità, molte aziende hanno iniziato a sviluppare e adottare piani di sviluppo prodotto leggermente diversi, mischiando il modello Stage-Gate classico con tecniche di project management Agile, in particolar modo lo Scrum descritto nel paragrafo precedente. Sono nati i cosiddetti Hybrid Agile-Stage-Gate System. Questi nuovi metodi permettono di gestire correttamente le iterazioni e i cicli per rendere il processo di sviluppo prodotto più flessibile e snello.

2.3.1 Applicazione a livelli differenti

La peculiarità di questi sistemi è il diverso livello a cui vengono applicati i metodi di Stage-Gate e Scrum. La gestione dei progetti di sviluppo prodotto, e in generale qualsiasi attività legata all'operational management, si può suddividere in tre principali livelli: strategico, tattico e esecutivo. Al livello strategico, quello più alto, vengono prese le decisioni di lungo periodo e si effettua una pianificazione strategica che ha come orizzonte l'intera durata del progetto, che può durare anni. Il livello tattico è quello in cui viene definita la pianificazione di attività e risorse settimanale o mensile, che deve rispettare le linee guida e le scadenze decise al livello strategico. L'ultimo livello, chiamato

esecutivo, ha come contenuto le decisioni sulle attività giornaliere da completare.

Nei modelli ibridi Agile-Stage-Gate sviluppati in molte aziende prevedono che a livello strategico si segua il modello Stage-Gate tradizionale: un comitato direttivo, per pianificare e controllare l'avanzamento del progetto, cambiamenti di budget e i costi, usa il metodo di Stage-Gate. Utilizzare tale metodo si sposa bene con le necessità del top management aziendale, che deve monitorare le attività svolte e approvare l'avanzamento del processo attraverso il superamento dei vari gate. Dal momento che il superamento di un gate comporta un impegno piuttosto oneroso in termini di utilizzo di risorse e soprattutto di costo, poiché ogni stage ha costi notevolmente superiori al precedente, il comitato direttivo, attraverso lo strumento dei gate, ha la facoltà di valutare la bontà e la completezza delle attività svolte e stabilire se il proseguimento del progetto è conveniente o meno.

A livello tattico, e ancor più al livello esecutivo, vengono invece adottate le tecniche di Agile project management, in maggior modo Scrum. Durante le riunioni tattiche, viene definito il carico di lavoro e viene rivista la prioritizzazione delle attività nel product backlog: tali riunioni sono gli sprint planning meeting e i review meeting. Il livello esecutivo invece prevede l'assegnazione e la programmazione giornaliera del lavoro, cioè quello che avviene durante i daily meeting proposti dalla teoria Scrum. Questo fa capire come si adatti bene la teoria Agile a questi livelli di gestione dei progetti. La suddivisione dei vari stage, che spesso possono durare decine di mesi, in sprint della durata prestabilita di quattro settimane, facilita la partizione del lavoro e diminuisce la variabilità del processo. [7]

Sebbene siano metodi nati e utilizzati maggiormente in contesti fortemente innovativi e iterativi quali l'industria software, vi sono alcuni casi in cui sistemi

ibridi sono stati adottati da aziende operanti nei più svariati settori. I vantaggi di questi piani sono molteplici. [8]

Questa versione rapida, iterativa e incrementale di concept, design e prototipi fornisce un rapido riscontro da parte dei clienti, che viene integrato nello sprint successivo per avvicinare il prodotto a ciò che i clienti desiderano e di cui hanno bisogno. I clienti, di contro, perfezionano le definizioni dei loro bisogni attraverso la loro partecipazione al processo. La voce del cliente diventa così un driver dinamico per tutto il progetto.

Gli stage e i gate rimangono comunque una parte fondamentale di questi modelli ibridi. I gate forniscono importanti punti decisionali in cui viene determinato il futuro dei progetti e vengono tagliati i progetti più deboli e meno redditizi e permettono al senior management di verificare il livello di avanzamento del progetto a punti chiave prestabiliti. Gli stage invece forniscono una linea guida di alto livello sulle attività raccomandate e sui deliverable che ci si aspetta alla fine dello stage. Tuttavia, i deliverable previsti nei modelli ibridi sono più snelli e flessibili rispetto alla teoria classica e sono costituiti da incrementi tangibili, sia disegni che prototipi, ma non più report inutili o presentazioni da fare al management.

Nella **Figura 5** è illustrato il modello Agile-Stage-Gate ibrido con le diverse tecniche di gestione del processo di sviluppo prodotto nei veri livelli strategico, tattico ed esecutivo.



Figura 5. Schema che mostra il diverso livello a cui vengono applicati lo Stage-Gate e l'Agile nei modelli ibridi.

2.3.2 Le 3 A

I processi di sviluppo prodotto di nuova generazione e che adottano sistemi ibridi devono essere caratterizzati dalle cosiddette tre A: Adattabile e flessibile, Agile e Accelerato.

Adattabile e flessibile perché spesso il prodotto può non essere completamente definito all'inizio del processo, ma si può iniziare comunque a svilupparne le caratteristiche chiave con meno della metà dei requisiti disponibili. Durante lo sviluppo poi evolverà e si adatterà alle nuove informazioni e richieste che il cliente comunicherà durante la fase di sviluppo. Per favorire l'adattabilità del processo è necessario che esso sia iterativo e che permetta di presentare in poco tempo e molto frequentemente al cliente alcuni risultati dello sviluppo. Tali iterazioni vengono definite *build-test-revise*. La flessibilità del processo è garantita anche dal fatto che ogni stage e ogni gate non presenta attività o deliverable prestabiliti per ogni progetto, ma in base al prodotto che si sta sviluppando e al contesto del mercato, si avranno attività e deliverable specifici per quell'unico progetto.

Agile perché, come si è evidenziato nel paragrafo precedente, vengono adottate tecniche di project management Agile; da alcune ricerche è emerso che le tecniche più usate sono quelle appartenenti a Scrum. Il tempo viene

frazionato in periodi di durata prestabilita (sprint e daily meeting ad esempio) in cui vengono sviluppati incrementi tangibili del progetto. Si cerca infine di diminuire le attività inutili e che non aggiungono valore, come ad esempio la documentazione eccessiva.

Accelerato perché tali sistemi hanno un importante focus sull'accelerazione del processo di sviluppo di un nuovo prodotto, per riuscire ad arrivare presto sul mercato, diminuendo il time-to-market. Per favorire questa accelerazione, viene ridotta la durata dei vari stage sovrapponendo e mettendo in parallelo le varie task all'interno degli stage. Spesso gli stessi stage sono leggermente sovrapposti tra loro, rendendo il concetto di stage meno rigido rispetto ai modelli tradizionali. Per favorire le sovrapposizioni di attività, è necessaria l'integrazione con un sistema IT robusto che tenga traccia dell'avanzamento del processo e favorisca la comunicazione tra i team.

2.3.3 Criteri flessibili ai Gate

Nei modelli Stage-Gate originali, le decisioni che venivano prese ai Gate erano spesso basate su criteri di origine finanziaria: se il progetto aveva un elevato ROI o generava flussi di cassa positivi veniva fatto proseguire, mentre un progetto economicamente meno appetibile veniva interrotto non appena la sua condizione finanziaria non favorevole veniva evidenziata. Spesso però tali criteri non sono molto corretti, non perché siano teoricamente sbagliati, ma perché spesso mancano dati consistenti su cui fondare tali criteri. In progetti fortemente innovativi infatti è difficile fare previsioni accurate su quali possano essere i ricavi e i flussi di cassa futuri. È utile quindi basare la scelta di Go/Kill su regole più flessibili, che comprendano anche criteri finanziari ma non solo, aggiungendo l'adattamento strategico del progetto, i potenziali benefici futuri in termini di reputazione o alleanze, il vantaggio competitivo e il potenziale di mercato.

2.3.4 Express Stage-Gate

In molte aziende infine il processo di sviluppo di un nuovo prodotto è scalabile e si adatta alla tipologia di progetto da portare a termine. Come mostrato in **Figura 6**, per i progetti più complessi e caratterizzati da un elevato rischio, viene adottato lo schema classico a cinque step, descritto nel dettaglio nel Paragrafo 2.1.

Per progetti con un livello di rischio moderato, in cui il prodotto sia già quasi interamente definito dallo start, oppure prodotti rischiosi ma standard, su cui l'azienda ha una importante esperienza, si utilizza un processo più leggero composto di sole tre macro-fasi: lo stage di scoping e creazione del business case sono uniti tra loro, così come vengono fusi insieme gli stage di sviluppo e test.

Per i progetti dal valore economico modesto, con rischi minimi o di durata breve, si adottano spesso processi Stage-Gate a due step: lo stage 1 e 2 vengono uniti, e se al gate la decisione è positiva e approva l'esecuzione del progetto, si prosegue con un unico stage in cui sono completati lo sviluppo del prodotto, i relativi test e il lancio nel mercato. [9]

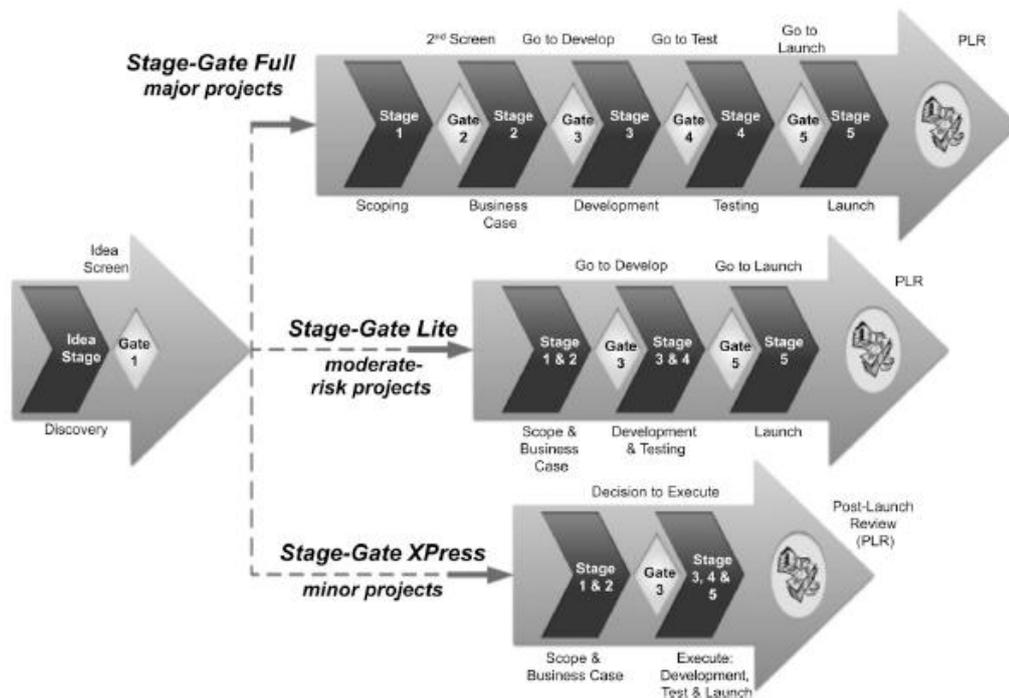


Figura 6. Schema mostrante tre diverse tipologie di processi Stage-Gate, ciascuno di diversa lunghezza a seconda della complessità e del rischio del progetto.

2.3.5 Vantaggi

I modelli ibridi presentati offrono numerosi vantaggi. Il principale, testimoniato da alcune ricerche, è un miglioramento della velocità di consegna del valore al cliente, derivato direttamente dalle tecniche Scrum.

Inoltre, si migliora la comunicazione e la collaborazione tra i vari team che lavorano Agile, ma si hanno benefici anche se non tutti i team di sviluppo adottano tali tecniche. Infatti, l'utilizzo di un modello Stage-Gate tradizionale per la pianificazione strategica consente a tutti i team di parlare lo stesso linguaggio, confrontandosi con le stesse tecniche di programmazione. [8]

Infine, il modello tradizionale è adottato dal top management dell'azienda, non sempre favorevole a grandi innovazioni e cambiamenti. Così facendo si mantiene il metodo con il quale i manager sono stati abituati a lavorare e si facilita il controllo dell'avanzamento del progetto.

3 Proposta di piano sviluppo prodotto

In questo capitolo viene presentata una proposta di piano di sviluppo prodotto adatta alle aziende obiettivo della ricerca. Vengono inizialmente definiti il profilo e le caratteristiche di tali società e come esse si inseriscono all'interno della filiera del settore automotive. Successivamente si propone nel dettaglio una metodologia di sviluppo prodotto, descrivendo le principali fasi, la struttura organizzativa dei progetti, metodi e strumenti necessari a una corretta implementazione del piano. Infine, vengono mostrate le differenze e analogie con la letteratura e quali sono i vantaggi previsti da tale piano di sviluppo prodotto.

Il metodo utilizzato nella creazione del modello di gestione dello sviluppo prodotto è stato un metodo induttivo che, partendo da alcuni aspetti peculiari della teoria analizzata e studiata a fondo, ha portato alla redazione di un piano di sviluppo prodotto che potesse essere adottato da tutte le aziende precedentemente selezionate come soggetti della ricerca. La caratteristica principale di questo modello di gestione è quindi la sua universalità, in quanto è stato pensato per poter essere adottato da diverse società per diversi progetti.

Nel Capitolo 4 dell'elaborato sarà applicato tale modello ad un progetto reale in una di queste società, con l'obiettivo di testarne la validità e valutare l'opportunità di modificare alcuni elementi al fine di rendere migliore e più efficiente il piano di sviluppo.

3.1 Soggetti della ricerca

Le società che sono state obiettivo della ricerca e per le quali si è ideato un piano di sviluppo prodotto innovativo sono, come anticipato nel Capitolo 1, società di ingegneria e design, all'interno del settore automotive. Nel processo di ideazione e sviluppo di un nuovo prodotto automobilistico, spesso le case

automobilistiche, OEM (Original Equipment Manufacturer), si affidano a fornitori esterni altamente specializzati per quanto riguarda l'attività di progettazione e ingegnerizzazione della vettura e dei suoi componenti. L'esternalizzazione può riguardare la progettazione della vettura intera o di un solo componente, può prevedere la fase di ideazione in cui si decide quale tipologia e modello di auto sviluppare ed è quindi un processo che varia in base alle esigenze dell'OEM e del singolo progetto. La nostra ricerca è rivolta quindi alle società che forniscono ingegneria e design alle OEM.

All'interno della filiera automobilistica, la tipologia di società presa in considerazione si posiziona come fornitore di primo livello, il cosiddetto tier-1. Si tratta di fornitori direttamente a contatto con la casa automobilistica produttrice e che a loro volta possono avere dei sub fornitori, i tier-2.

Dal momento che tali società possono occuparsi di progetti molto differenti tra loro, il piano di sviluppo prodotto ideato dovrà essere flessibile e dovrà adattarsi alle diverse esigenze dei progetti.

3.2 Il modello

Dopo un'approfondita analisi della letteratura e avendo compreso a fondo le caratteristiche e peculiarità dei diversi metodi di gestione di sviluppo prodotto si è ideato un modello che possa essere applicato alle società obiettivo della ricerca.

Il modello di base che si è scelto di sviluppare è un modello ibrido Agile-Stage-Gate, perché si ritiene che la giusta combinazione di modelli Stage-Gate tradizionali con tecniche di project management Agile possa fornire al piano di sviluppo prodotto la flessibilità necessaria per affrontare le diverse esigenze.

Da un lato, essendo l'automobile un prodotto altamente complesso, è necessario avere uno schema rigido e ben definito che suggerisca le attività da svolgere all'interno degli stage e che consenta il proseguimento dei progetti

solo se vengono soddisfatti determinati criteri. Queste caratteristiche sono tipiche del modello Stage-Gate tradizionale, che verrà utilizzato per pianificare le attività sul lungo periodo, fissare milestone e design review e monitorare l'avanzamento dei vari progetti. I Gate serviranno quindi come punto di controllo della qualità del progetto, come momento di revisione da parte del gruppo dirigente e forniranno una valutazione del progetto necessaria a stoppare e chiudere eventuali progetti non profittevoli. L'approccio Stage-Gate può essere utile anche nei rapporti con il cliente, inteso come committente del progetto, che deve essere coinvolto e approvare il passaggio del gate.

Dall'altro lato, il processo di progettazione di un'automobile è molto lungo ed è soggetto ad un'elevata variabilità: possono cambiare le richieste del cliente, sia inteso come utente finale del veicolo che come committente del progetto (OEM), possono cambiare le condizioni economiche del mercato in cui è destinata la vettura, possono essere sviluppate nuove tecnologie o possono variare le norme a cui devono sottostare i veicoli sviluppati. Questa elevata incertezza e frequenza dei cambiamenti può essere mitigata dall'utilizzo di tecniche di gestione dei progetti Agile, che si basano appunto sullo sviluppo del prodotto già dalle prime fasi senza che tutti i requisiti siano necessariamente definiti. La flessibilità e la velocità di reazione di Scrum infatti si adatta bene a contesti innovativi e incerti come lo sviluppo di nuovi prodotti nel settore automotive. Inoltre, i frequenti feedback forniti dal cliente consentono lo sviluppo di un prodotto che rispetta molto più fedelmente la Voice of Customer.

I due modelli, in sostanza, vengono adottati a livelli differenti: per decisioni e pianificazioni di lungo periodo, e per valutazioni sulla bontà dei progetti, cioè per tutte le decisioni prese a livello alto e strategico, si utilizza il modello Stage-Gate; per la pianificazione quotidiana ed esecutiva, con orizzonte temporale più limitato, quindi a un livello più basso e tattico, si adottano le tecniche Agile. L'approccio ibrido lo si nota anche nei Gate, che non saranno più rigidi come

nel modello tradizionale di Cooper, ma avranno dei criteri di superamento più flessibili, ispirati alla gestione dei progetti Agile.

Ovviamente, né il modello Stage-Gate né Scrum possono essere adottate così come vengono descritti dalla teoria, ma necessitano di adattamenti che si illustreranno all'interno di questo capitolo.

3.3 Agile nell'automotive

Nel modello ibrido proposto, la componente Agile è rappresentata da Scrum. La scelta è dovuta al fatto che Scrum permette di progettare e procedere allo sviluppo di un prodotto nonostante tutti i requisiti non siano ancora stati definiti e alcuni di essi non siano ancora stati nemmeno individuati. La natura di Scrum inoltre consente lo sviluppo completo della vettura in tutti i suoi aspetti secondo cicli di lavoro iterativi, gli sprint, alla fine dei quali viene consegnato al cliente un incremento di prodotto tangibile e funzionante. Ricordando che il framework Scrum deriva dall'industria software, è facile capire che in questo settore è molto più agevole presentare al cliente un incremento di prodotto funzionante al termine di uno sprint, trattandosi di linee di codice che vanno a soddisfare determinati pre-requisiti. Per una società di design che progetta veicoli e componenti meccaniche, è molto più complesso consegnare un veicolo o un prototipo tangibile al termine di uno sprint. La *definition of done*, come viene chiamata in Scrum, necessita quindi di una ridefinizione. L'incremento di prodotto di uno sprint è infatti costituito da disegni tecnici 2D o 3D, studi di fattibilità, risultati di calcoli e analisi virtuali. Soltanto nelle ultime fasi di sviluppo prodotto verranno consegnati e mostrati al cliente prodotti fisici, quali ad esempio prototipi su cui verranno effettuati test di vario tipo.

Un'altra differenza rispetto allo schema canonico di Scrum è il livello di dettaglio del Product Backlog. Secondo Scrum, i requisiti inseriti nel backlog devono essere di alto livello, non troppo dettagliati, lasciando ai singoli

sviluppatori la scelta di cosa fare e come fare per soddisfare quel particolare requisito. Questo favorisce l'aumento della creatività del progettista e lo responsabilizza, essendo lui solo l'artefice di quella particolare progettazione. Nella progettazione di un'automobile invece, i requisiti da sviluppare (i Basic Requirements) sono collegati a precise specifiche tecniche che gli sviluppatori sono obbligati a rispettare; inoltre, alcune attività sono soggette a vincoli di tipo tecnologico e/o normativo. Questi aspetti diminuiscono la libertà dei membri del team di sviluppo e portano alla creazione di un product backlog abbastanza dettagliato, almeno per le attività poste più alto e con priorità più alta. Per facilitare il completamento dei requisiti e delle attività all'interno di un singolo sprint e avere un incremento finito del prodotto, è necessario che le macro-attività di progettazione, che nella loro totalità durano mesi, vengano scomposte in pacchetti di task elementari, della durata di qualche decina di ore, per poter essere presi in carico in uno sprint e per facilitare l'utilizzo di kanban board e burndown chart.

3.4 Panoramica del processo

Il processo di sviluppo di un nuovo prodotto automobilistico è stato suddiviso in cinque macro-fasi, che coincidono e danno il nome ai cinque stage del processo che si è ideato, più uno stage iniziale di valutazione della commessa, definito Stage 0.



Figura 7. Schema illustrante la panoramica del processo, suddiviso in cinque stage, in successione allo Stage 0 di valutazione.

- **Stage 0 Job Evaluation.** La natura delle società prese in considerazione è quella di essere fornitrici di ingegneria e progettazione. Essendo fornitori, il processo di ideazione non avviene all'interno dell'azienda

che si limita a lavorare su commesse esterne. Il cliente infatti fornisce una RFQ, Request For Quotation, e la società deve predisporre un preventivo per la realizzazione del progetto. Lo Stage 0 ha il compito di valutare la commessa, stimando le macro-attività da svolgere, valutandone la fattibilità, la convenienza economica e strategica. Queste valutazioni vengono fatte da un team di lavoro provvisorio costituito da un project manager coadiuvato da rappresentanti delle funzioni Marketing e Finance, oltre a direttori tecnici che aiutano il PM nella stima delle attività. Il preventivo viene presentato al cliente e viene preparata una bozza di contratto contenente i tre elementi fondamentali di un progetto: i tempi di consegna, il prezzo e il contenuto tecnico. Dopo un'eventuale trattativa con il cliente si arriva al **Gate 0** che prevede, in caso di Go, la firma del contratto e l'inizio dell'attività di progetto; nel caso opposto, Kill, il progetto si stoppa e la commessa non viene presa in carico.

Questo Gate è l'unico che rispecchia l'approccio tradizionale e rigido del modello Stage-Gate: non sono previsti criteri flessibili per l'accettazione, ma solo una decisione Go, Kill. Se si supera il Gate, viene firmato il contratto che impegna la società. Se il Gate non viene superato, il progetto viene immediatamente abbandonato.

- **Stage 1 Product Definition.** Durante questo Stage vengono effettuati studi di mercato, analisi normative, analisi benchmark e posizionamento del prodotto nel mercato, per capire la profittabilità del prodotto una volta lanciato sul mercato. Tali attività consentono di definire il prodotto che dovrà essere sviluppato, cioè quale tipologia di veicolo e con quali caratteristiche. Spesso queste attività vengono svolte direttamente dall'OEM che richiede l'RFQ e fornisce una sorta di Capitolato Tecnico contenente tutte le caratteristiche da sviluppare. L'attività fondamentale di questo Stage è la definizione dei Basic

Requirements che saranno poi inseriti nel backlog del prodotto. Partendo dalle analisi di mercato e normative o estraendo dal Capitolato Tecnico vengono elencate tutte le caratteristiche che il prodotto automobilistico deve avere, raggruppate in diversi livelli. Il **Gate 1**, che deve approvare i Basic Requirements, è superato se una percentuale prefissata (80%, ma può variare in base ai progetti) di requisiti sono stati correttamente individuati. Trattandosi di attività di progettazione di una vettura, si dà tendenzialmente precedenza ai requisiti di stile, essendo una delle prime attività della successiva fase di sviluppo. Viene aggiornato il Backlog del prodotto e prioritizzati i contenuti al suo interno.

- **Stage 2 Concept Alternatives.** È da questo Stage in avanti che le caratteristiche tipiche di Scrum portano i maggiori benefici. Viene ufficialmente costituito il team cross-funzionale che si occuperà dello sviluppo. Viene sviluppato lo stile interno ed esterno del veicolo, vengono fatti studi approfonditi di fattibilità e sono elaborate diverse alternative di concept riguardanti diversi aspetti e componenti del veicolo. La cross-funzionalità del team e la presenza di membri con competenze differenti facilita lo sviluppo del prodotto a 360°, in tutti i suoi aspetti. Lo sviluppo congiunto e i feedback del cliente consentono un allineamento più preciso alle richieste del cliente. Il Gate 2, seguendo la logica Agile, può essere superato anche se non sono state completate tutte le attività dello Stage 2; la logica di Scrum secondo la quale si può iniziare a sviluppare un software senza che tutte le funzionalità e requisiti siano stati definiti è qui fortemente richiamata. I criteri di superamento dei Gate sono descritti nel **Paragrafo 3.11**.
- **Stage 3 Product Development.** È lo stage in cui la nuova automobile inizia a prendere forma; si progettano il telaio, la carrozzeria, le parti meccaniche e tutte le altre componenti della vettura. Si producono

disegni tecnici 2D e 3D da presentare al cliente. Allo stesso tempo vengono effettuate prove di calcolo e simulazioni virtuali per validare la progettazione. Infine, vi è un periodo di test effettuato su prototipi appositamente costruiti. Nuovamente, la cross-funzionalità del team di sviluppo permette l'avanzamento del prodotto in tutte le sue componenti e sezioni. Dal momento che le diverse parti della vettura sono strettamente collegate tra loro, la co-locazione aiuta la comunicazione e lo sviluppo del prodotto in parallelo. Ad esempio, chi progetta componenti meccaniche deve sapere da chi sviluppa il telaio dove siano i punti duri e le caratteristiche della struttura portante. Scrum facilita questo. L'intero progetto viene validato al **Gate 3**.

- **Stage 4 Process Development.** In questo stage viene progettato, testato e costruito il processo produttivo necessario alla fabbricazione dei prototipi prima e della produzione in massa poi. Anche se la numerazione è progressiva, questo stage è svolto quasi totalmente in parallelo allo Stage 3, sia per risparmiare tempo e accelerare il processo, sia per garantire l'effettiva fattibilità e realizzabilità industriale delle componenti progettate. Il processo produttivo riceve una valutazione circa la sua bontà al **Gate 4**.
- **Stage 5 Manufacturing & Product Launch.** Dopo aver approvato prodotto e processo, si può procedere alla produzione e al lancio commerciale. Le attività in questo stage esulano leggermente da quelle di una società di ingegneria e design, ma tali società collaborano e sono responsabili di eventuali problemi che possono accadere durante le prime fasi di manufacturing, avendo loro progettato il prodotto e processo. In questa fase si lanciano le attrezzature, si testano gli stampi e si eseguono prove di assemblaggio e altri test per verificare il corretto funzionamento della linea produttiva. Parallelamente avviene il lancio commerciale con comunicazione, advertising e attività promozionali. Il

processo di sviluppo prodotto si chiude con il **Gate 5**, in cui vengono convalidati i test di produzione e si può procedere con la produzione di massa, di cui il responsabile non è più la società di design ma è il car-maker.

3.5 Organizzazione

L'adozione di modelli ibridi Agile-Stage-Gate per la gestione dello sviluppo del prodotto ha un impatto anche sull'organizzazione aziendale. Con Scrum infatti viene data molta più responsabilità allo Scrum Team, che è un team fondamentalmente auto organizzato. Trattandosi di un modello ibrido proposto ad aziende la cui organizzazione gerarchica è tipicamente molto radicata, si propongono alcune modifiche al modello Scrum classico. Si nota subito un diverso utilizzo dei nomi: si è pensato di mantenere i nomi delle figure più tradizionali all'interno del mondo automotive per mantenere una certa familiarità nell'ambiente. Allo stesso tempo, pur mantenendo gli stessi nomi, le figure che si andranno a descrivere avranno dei tratti in comune con la teoria Agile Scrum.

Il Product Owner è rappresentato dal Project Manager (PM). È il PM infatti che si occupa dei rapporti con il cliente ed è lui o lei che è responsabile del soddisfacimento dei requisiti imposti dal cliente e da tutti gli stakeholder, in termini di tempi, costi e qualità del prodotto; rappresenta pertanto la Voice of Customer. Durante gli Stage iniziali del progetto redige una macro-pianificazione delle attività richieste, che sarà la base per formulare l'offerta al cliente. Basandosi sulla disponibilità di risorse indicata dai responsabili di funzione, sceglie i membri del team di sviluppo. Successivamente, durante le fasi di sviluppo del prodotto, è la persona incaricata di suggerire al team di sviluppo quali attività fare all'interno di uno sprint e di definirne i criteri di accettazione. Al termine dello sprint infine dovrà decidere se l'incremento di prodotto rispetta tali criteri, se ha effettivamente aumentato il valore del

prodotto e se può essere consegnato o presentato al cliente o stakeholder di riferimento. È incaricato della creazione del product backlog e di revisionarlo in base ai feedback ricevuti dal cliente. All'interno dell'organizzazione di un progetto, il Project Manager è quello al livello gerarchico più elevato e con la maggiore autorità.

Lo Scrum Master, all'interno dei team di sviluppo prodotto prende il nome di Team Leader. Come lo Scrum Master svolge il ruolo di facilitatore, insegnando e favorendo l'adozione di tecniche Scrum. Deve essere per questo motivo un esperto di tale metodologia e deve averla compresa in tutti i suoi aspetti essenziali. Come suggerisce il nome, lui o lei è un leader all'interno del team di sviluppo: modera i daily meeting, organizza il lavoro ed è il portavoce del team durante le riunioni con il PM (Scrum of Scrum meeting).

Lo Scrum Team, infine, all'interno di una società di engineering è chiamato più semplicemente Team di Sviluppo o Team di Progetto. È un team cross-funzionale, con membri appartenenti a diverse funzioni o comunque esperti di progettazione di diverse parti della vettura. La cross-funzionalità aiuta a sviluppare il prodotto a 360° fin da subito e permette una rapida risoluzione dei problemi. Per facilitare ulteriormente il lavoro di sviluppo, il Team di Sviluppo deve essere possibilmente co-locato, affinché la comunicazione all'interno del team funzioni e si riescano a sfruttare al meglio i meeting e gli strumenti di Scrum. È preferibile infine che il team sia concentrato su di un solo progetto alla volta. Alcuni membri devono essere presenti per tutta la durata del progetto, altri invece possono essere presenti solo in alcune fasi specifiche per portare all'interno del gruppo le loro competenze caratteristiche.

Ad un livello superiore rispetto allo Scrum Team, troviamo il cosiddetto Scrum of Scrum Team: si tratta di un team composto dal Project Manager del progetto e dai Team Leader dei vari team. Tale team viene chiamato anche Core Team di progetto e ne fanno parte anche altri attori che possono svolgere la funzione di

staff durante la vita del progetto e lavorare “a chiamata”: ad esempio un rappresentante marketing deve essere un punto di riferimento durante tutta la fase di definizione del prodotto e svolge un ruolo in parte simile al Product Owner, ma con una presenza marginale nelle fasi successive, non occupandosi direttamente dello sviluppo del prodotto. I membri di queste funzioni (Marketing, Finance) possono dedicarsi a più progetti contemporaneamente, dato il loro impiego marginale. Vista la natura delle società e delle attività svolte, non è necessario che i rappresentanti dell’area finance o commerciale facciano parte del team cross-funzionale. È preferibile avere il loro supporto ad un livello leggermente più alto, all’interno del Core Team.

L’organizzazione dei progetti, se di grandi dimensioni, può dotarsi di un livello aggiuntivo: può essere previsto un coordinatore tecnico dei Team Leader o Project Engineer che coordini i diversi Team Leader. È lui o lei che fa parte del Core Team. Tale scelta vien fatta per snellire leggermente il Core Team nel caso in cui la presenza di molti Team di Sviluppo creasse una piattaforma troppo numerosa.

Uno schema sintetico dei diversi livelli gerarchici dell’organizzazione dei progetti di sviluppo prodotto è mostrato in **Figura 8**.

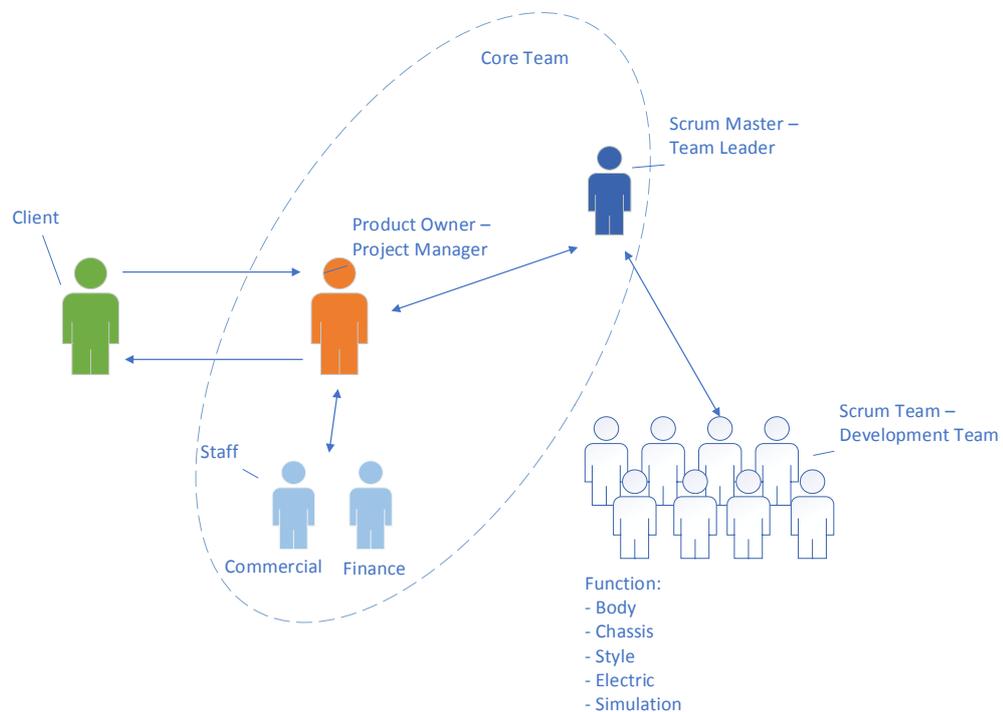


Figura 8. Schema della proposta di organizzazione delle risorse all'interno di un progetto. Si nota la cross-funzionalità a due livelli, sia all'interno del Team di Sviluppo che nel Core Team.

3.6 Lo sprint

Il modello ibrido Agile-Stage-Gate che si sta proponendo prevede l'adozione di tecniche Scrum a livello tattico ed esecutivo, per la pianificazione e organizzazione delle attività di sviluppo all'interno dei diversi stage. Uno degli elementi principali di Scrum è lo Sprint. Per sfruttare al massimo i vantaggi delle tecniche dell'Agile è necessario fissare una durata dello Sprint, che sia la stessa per tutta la durata del progetto e per tutti i progetti, per favorire una facile diffusione di queste metodologie all'interno dell'azienda. Si è scelto di utilizzare sprint di 4 settimane.

I motivi di questa scelta sono molteplici. Innanzitutto, è la durata maggiormente consigliata dalla teoria Scrum e quella più diffusa all'interno delle aziende che utilizzano Scrum al di fuori del contesto software. Essendo il

design di veicoli un'attività ad alta complessità, sprint di durata inferiore rischierebbero di concludersi con poche attività svolte correttamente.

Le quattro settimane proposte rappresentano una sorta di soglia psicologica, durante le quali i membri del Team di Sviluppo riescono a mantenere la giusta concentrazione per raggiungere senza problemi l'obiettivo dello sprint. Sprint più lunghi rischierebbero di far perdere la giusta attenzione al Team, in quanto troppo lontana la fine del periodo. Sprint più brevi invece potrebbero portare confusione all'interno del team per il poco tempo disponibile a completare le attività e farebbero fatica ad entrare a regime. Inoltre, una durata di quattro settimane è facilmente riconducibile ad un mese del calendario; non si tratta di un vero e proprio vantaggio derivato dalla metodologia Scrum, ma è una comodità che facilita la programmazione e la visione d'insieme del progetto per tutti i membri del team.

3.7 I meeting

Lungo tutta la durata del progetto è importante che siano stabiliti dei meeting a vari livelli in cui si decide quali attività svolgere, quando e in che modo. Scrum ritiene che la comunicazione all'interno del team, tra i vari team e con le figure gerarchicamente superiori sia di fondamentale importanza per una corretta riuscita del progetto. All'interno del Piano di Sviluppo Prodotto proposto, si è scelto di inserire quindi delle riunioni con una frequenza prestabilita per favorire la comunicazione a tutti i livelli. Prendendo spunto dalla teoria Scrum, i meeting previsti durante lo sviluppo di un prodotto sono:

- **Sprint planning meeting.** Viene effettuato una volta per sprint, quindi una volta ogni quattro settimane, all'inizio del periodo. Ha una durata variabile che va dalle 4 alle 8 ore. A questa riunione partecipa il Team di Sviluppo al completo, compreso di Team Leader, e il Project Manager responsabile del progetto. L'obiettivo di questo incontro è stabilire quali attività fare durante lo sprint che sta per partire. Il PM seleziona dal

product backlog dei requisiti e caratteristiche del prodotto da sviluppare, tendenzialmente quelle posizionate in cima a cui è stata data una priorità più elevata. Il PM quindi propone al Team Leader di portare a termine alla fine dello sprint tali attività. A questo punto si instaura una discussione e un confronto a cui partecipano tutti i membri del Team di Sviluppo: si stima la durata delle attività analizzate, si cerca di capire quali rischi e quali imprevisti potrebbero accadere durante lo sprint, il Project Manager decide i criteri di accettazione. Per stimare correttamente la quantità di lavoro da prendere in carico, le attività di lunga durata vengono suddivise in task di breve durata, dell'ordine di poche decine di ore al massimo; ciò consentirà anche una maggior precisione di utilizzo di strumenti come scrum board e burndown chart. Dopo una fase di confronto si raggiunge a un accordo e il Team prende in carico la lista di requisiti e attività da completare; vengono schedate le task dei primi giorni dello sprint, così che il Team di Sviluppo possa da subito iniziare a creare incrementi di prodotto. La lista dei requisiti da sviluppare presi in carico in un uno sprint costituisce lo sprint backlog. Durante questa riunione viene creata la *sprint board*, la lavagna che permette la gestione e programmazione visiva delle attività tramite kanban.

- **Daily standup meeting (daily scrum).** È una riunione che si svolge tutti i giorni, dalla breve durata prestabilita di 15 minuti; deve svolgersi preferibilmente al mattino. A questa partecipano il Team di Sviluppo con il Team Leader in qualità di moderatore. Ha lo scopo di fare il punto della situazione riguardo alle attività svolte e da svolgere. Ogni componente della squadra deve rispondere a tre semplici domande: *cosa ho fatto ieri? Cosa devo fare oggi? Quali eventuali impedimenti mi hanno ostacolato?* Il Team Leader prenderà nota di tali impedimenti e durante il corso della giornata cercherà di trovarne una soluzione,

eventualmente coinvolgendo anche attori di livello superiore. Durante questa riunione il Team Leader organizza anche il lavoro quotidiano, aggiornando la scrum board.

- **Sprint review meeting.** Questa riunione si svolge una volta ogni 4 settimane, al termine del periodo dello Sprint. Vi partecipa il Team al Completo con il Project Manager. Ha una durata di circa 4 ore. Durante tale incontro il PM, in qualità di Product Owner, decide se le attività svolte durante lo sprint rispettano i criteri di accettazione stabiliti 4 settimane prima e decreta quindi se ritenere l'incremento di prodotto accettabile e consegnabile al cliente. In caso negativo, i requisiti non soddisfatti vengono posizionati in cima al product backlog e saranno completati nel primo sprint utile. Anche nel caso di esito positivo della revisione, il PM riorganizza il product backlog, alla luce delle attività completate e delle nuove informazioni disponibili comunicate dal cliente. Infine, vengono analizzati i processi seguiti, gli strumenti utilizzati, la comunicazione e la collaborazione all'interno del team per evidenziare eventuali attriti e difficoltà all'interno del team e proporre soluzioni.
- **Scrum of scrum meeting.** È un incontro che viene tenuto con cadenza settimanale e della durata di massimo 2 ore. È previsto soltanto per i progetti di medie-grandi dimensioni che abbiano più di un Team di Sviluppo. Vi partecipano il Project Manager responsabile dello sviluppo prodotto, tutti i Team Leader di team che lavorano a tale progetto e i membri delle funzioni staff come Marketing o Finance. Durante questo meeting si verifica l'allineamento degli obiettivi dei vari team, accertandosi che non vi siano incongruenze tra lo sviluppo di diverse componenti del prodotto. Tale riunione può essere svolta più di una volta a settimana se qualcuno dei Team Leader ne ha urgente necessità, per evitare che tutto il progetto esca dai giusti binari.

Vi possono inoltre essere degli incontri o delle chiamate tra membri di diversi team, per risolvere piccoli dubbi e avere chiarimenti su alcuni requisiti che sta sviluppando un altro team. Si tratta di incontri molto informali, che avvengono normalmente senza bisogno di schedulazioni particolari.

I vari meeting previsti all'interno del Piano di Sviluppo Prodotto sono riassunti nella **Tabella 2**.

Tabella 2. Tabella riassuntiva dei meeting proposti secondo il piano di sviluppo prodotto automotive.

Meeting	Durata	Frequenza	Partecipanti	Argomenti
Sprint planning	4 – 8 h	1 per sprint, all’inizio	PM, Team Leader, Team di Sviluppo	Cosa fare nello sprint, sprint backlog, sprint board, criteri di accettazione
Daily standup	15 min	Ogni giorno	Team Leader, Team di Sviluppo	Cosa fare nella giornata, risoluzione eventuali problemi di ieri
Sprint review	4 h	1 per sprint, alla fine	PM, Team Leader, Team di Sviluppo	Accettazione incrementi di prodotto, revisione product backlog, risoluzione problemi
Scrum of scrum	2 h	1 a settimana, o a richiesta	PM, Team Leader, Staff	Allineamento attività vari team

3.8 Ripetizione del ciclo

La caratteristica fondamentale di Scrum è l’iteratività. Tutte le attività descritte fino a questo momento vengono ripetute fino a termine del progetto, ognuna con la sua frequenza.

Ricapitolando, durante uno sprint i membri del Team di Sviluppo collaborano per creare una serie di incrementi di prodotto. Ogni incremento deve

soddisfare i criteri di accettazione del Project Manager. Il lavoro parte dal product backlog, di cui è responsabile il PM. Ogni sprint inizia con uno Sprint Planning per generare uno sprint backlog. Il team si auto organizza per realizzare lo sviluppo, usando i Daily stand-up meeting per coordinarsi e garantire un miglior incremento di prodotto. Il coordinamento tra diversi team è effettuato durante gli Scrum of Scrum meeting e, a richiesta, con alcuni incontri informali tra membri dei vari team. Infine, durante lo Sprint review il Project Manager stabilisce se gli incrementi di prodotto soddisfano i criteri di accettazione, viene rivisto e modificato il product backlog e si rivede il processo. Al termine dello Sprint review, si svolge un nuovo Sprint planning meeting e il processo si ripete dall'inizio.

3.9 Principali differenze rispetto Scrum

Una delle principali differenze rispetto alla teoria Scrum è una minor cross-funzionalità all'interno del Team di Sviluppo. Scrum prevede infatti che all'interno del team vi siano figure professionali derivanti da ogni funzione aziendale, affinché in ogni momento si possano affrontare e risolvere i più diversi problemi, dalla progettazione all'analisi della profittabilità. Nel framework che si propone invece la cross-funzionalità è intesa in maniera differente: nel team sono rappresentate diverse funzioni, ma sono tutte legate allo sviluppo del prodotto. Si trovano infatti progettisti della carrozzeria, del telaio, stilisti, esperti di calcolo e simulazioni. Si tratta di membri del team che sviluppano parallelamente diversi aspetti della vettura. Nel Team di Sviluppo non vi sono invece membri del marketing o dell'area finance, al contrario di quanto suggerisce la teoria Scrum. Queste figure sono comunque presenti all'interno dell'organizzazione dei progetti ma ad un livello più alto, all'interno del cosiddetto Core Team.

Un'altra differenza rispetto alla teoria classica Scrum è la presenza di una gerarchia all'interno del Team di Sviluppo. Lo Scrum Master o Team Leader

infatti, ha una funzione di leadership all'interno del team. Non è soltanto il *leader servitore* indicato nella teoria, che si limita a facilitare e favorire l'applicazione delle tecniche Scrum, ma è lui che organizza il lavoro giornaliero e si interfaccia con il PM del Team di Piattaforma nelle riunioni di sprint planning e sprint review, tenendo comunque sempre fortemente in considerazione le opinioni e le esigenze dello Scrum Team, che infatti partecipano attivamente a tali riunioni.

Il Project Manager responsabile del particolare progetto, all'interno del Core Team o Scrum of Scrum Team svolge una duplice funzione: rimane sempre con il suo ruolo di Product Owner responsabile del soddisfacimento dei requisiti del cliente, ma è anche Chief Scrum Master in quanto facilita, coordina e collabora con gli Scrum Master dei team di sviluppo dei livelli inferiori.

Un'ulteriore differenza riguarda i nomi che sono stati utilizzati: si è scelto di mantenere i nomi dei ruoli maggiormente diffusi nelle organizzazioni delle società di engineering e design, senza sostituirli con quelli tipici della cultura Scrum. Tale scelta è stata fatta principalmente per rendere l'ambiente e i ruoli più familiari ai progettisti e lavoratori che altrimenti potrebbero trovarsi disorientati ad interfacciarsi con delle figure con nomi cui non erano abituati.

Non vi è poi una riunione dedicata all'analisi retrospettiva, quello che in Scrum viene chiamato Retrospect Meeting. L'analisi delle problematiche che emergono all'interno del team rispetto all'utilizzo di Scrum o in generale nello sviluppo del prodotto viene infatti discussa durante lo Sprint Review, che si compone di due parti: la prima, durante la quale vengono presentati, valutati e accettati i deliverable e la seconda, durante la quale avviene appunto un'analisi retrospettiva su problemi e consigli per lo sprint successivo.

Infine, gli elementi del product backlog non sono scritti sotto forma di user story, principalmente per una maggior facilità di compilazione del backlog.

3.10 Strumenti, documenti e template

Per facilitare l'adozione delle tecniche Scrum, in particolare alcuni strumenti fondamentali come il Product e lo Sprint Backlog, si propongono alcuni template ed esempi che possano essere utilizzati dai team. Si ricorda che uno degli obiettivi del project management Agile è l'abolizione della documentazione non necessaria; si è cercato pertanto di prevedere documenti e file solo dove è strettamente necessario avere un documento come promemoria o prova della avvenuta attività.

3.10.1 Product backlog

Il product backlog è un documento, fatto sotto forma di tabella, ad esempio un file in formato excel o compatibile con i sistemi informativi dell'azienda, contenente tutti i requisiti del prodotto e le macro-attività che devono essere completate per soddisfarli. Questo file può essere visualizzato da tutto il team e dai PM di altri progetti, ma può essere modificato esclusivamente dal PM che lo ha creato. Le attività al suo interno sono ordinate seguendo una priorità decrescente. Il responsabile della prioritizzazione del backlog è il Project Manager e, partendo dai Basic Requirements e dal macro-planning fatto a inizio progetto, ordina le attività seguendo fondamentalmente tre criteri: valore, rischio e dipendenze.

Scrum enfatizza al massimo la prioritizzazione basata sul valore, quindi vanno completati per primi i requisiti e le attività che creano maggior valore al cliente. Il PM, agendo da Product Owner, collabora con il cliente e deve capire cosa vuole e a cosa dà valore, in modo da organizzare gli elementi del product backlog. Oltre a sviluppare prima i requisiti che portano più valore, la consegna del valore al cliente deve seguire un flusso teso, il più possibile costante durante tutta la durata del progetto: ogni sprint deve consegnare la stessa quantità di valore al cliente. Il problema più grande in questo caso è rappresentato dal modo in cui viene calcolato il valore per il cliente: nella maggioranza dei casi,

per semplicità di calcolo, il valore è pari alle ore uomo lavorate durante lo sprint.

Un altro criterio seguito per la prioritizzazione è il rischio legato allo sviluppo di particolari requisiti. Devono essere poste in alto le attività che hanno un livello di incertezza più elevato, e si devono prevedere delle attività di mitigazione del rischio.

Infine, la regola di prioritizzazione più importante per una società di design di autoveicoli è sicuramente il rispetto di vincoli tecnologici. Per poter progettare al meglio un componente o parte di un veicolo, è necessario aver progettato e approvato un'altra parte. Anche se molte attività possono essere svolte in parallelo, alcuni vincoli tecnologici sono da rispettare categoricamente.

ID	Requirement/task	Stage	Priorità	Stima
AB123	Bozzetto stile	2	5	20h

Figura 9. Esempio proposto di template per il product backlog.

Il template proposto si compone di diverse colonne: ID rappresenta un codice alfanumerico identificativo che riconduca alla particolare attività, che per semplicità potrebbe essere lo stesso codice della WBS utilizzato in MS Project; requirement/task è una breve descrizione dell'attività; Stage indica lo Stage in cui è prevista quella attività ed è necessario per calcolare la percentuale di completamento per il superamento del Gate (si veda paragrafo 3.11); priorità

è un numero su una scala da 1 a 5 che mostra il livello di priorità dell'attività, tiene conto di diversi fattori ed è stabilita dal PM; stima indica la durata stimata dell'attività, espressa in giorni o ore lavorative o story point.

Il product backlog è un documento in continuo aggiornamento. Durante lo sviluppo del prodotto il PM e tutto il team hanno maggiori informazioni sul progetto, su quali attività sono più importanti e su come possono essere evitati i rischi; pertanto, il Project Manager modifica la priorità del product backlog adattandolo alle nuove informazioni. È molto importante anche il feedback fornito dal cliente: la consegna di piccoli incrementi di prodotto ogni quattro settimane dà la possibilità al cliente di verificare spesso l'avanzamento del prodotto, correggendo e suggerendo eventuali modifiche. Il PM, nell'aggiornare il product backlog, deve tener conto delle risposte dei clienti.

All'interno del product backlog le attività sono dettagliate in modo diverso a seconda della loro posizione: le attività poste più in alto, ovvero quelle che verosimilmente saranno completate per prime, hanno un elevato livello di dettaglio, in quanto si tratta di task di breve durata, di cui si dispongono abbastanza informazioni per stimarne la durata; le altre attività poste più in basso sono descritte in modo più approssimativo, e la loro stima è quella fatta durante la fase di macro planning.

3.10.2 Stima delle attività

Un metodo molto utile nella stima delle attività o requisiti è quello degli *story points*. Uno Story Point è un'unità di misura arbitraria utilizzata dagli Scrum Team per misurare l'effort necessario per la realizzazione di una task. I valori che gli story point possono assumere sono solitamente appartenenti all'insieme [1, 2, 4, 8, 16, Big?], alla serie di Fibonacci [1, 2, 3, 5, 8, 13, 21, 34, Big?] o una sua versione modificate [1, 2, 3, 5, 13, 40, Big?] dove "Big?" rappresenta il punteggio di una task di difficile stima per la sua grandezza e/o complessità o su cui il team non ha subito trovato un accordo sull'effort. La

peculiarità della stima tramite story point è che essa consente solo misurazioni relative tra diverse task, ma non dà informazioni sulla durata effettiva di tali attività.

Il Planning Poker, chiamato anche Estimation Poker, è una tecnica di stima che utilizza il consenso per stimare le dimensioni relative delle task. Nel Planning Poker, viene assegnato un mazzo di carte a ciascun membro del team. Ogni carta è numerata secondo una delle sequenze precedentemente descritte e i numeri rappresentano la complessità del problema, in termini di tempo o di impegno, per come stimato dal membro del team. Il Project Manager sceglie un'attività dal Prioritized Product Backlog e la presenta al team. I membri dello Scrum Team la valutano e cercano di capirla più a fondo prima di fornire la propria stima per il suo sviluppo. In seguito, ognuno prende dal mazzo una carta che rappresenta la propria stima per quella attività. Se la maggioranza di tutti i membri del team sceglie la stessa carta allora la stima indicata da quella carta sarà la stima stabilita per quella task. Se non c'è accordo, i membri del team discutono le ragioni della scelta delle differenti carte o stime. Dopo questa discussione scelgono di nuovo le carte. Questa sequenza si ripete fino a quando sono ben compresi i presupposti, sono risolte le incomprensioni e si raggiunge il consenso unanime o un accordo. Il Planning Poker promuove una maggiore interazione e una migliore comunicazione fra i partecipanti. Questa tecnica facilita il pensiero indipendente dei partecipanti, evitando così il fenomeno del pensiero di gruppo.

La somma degli story point che si prevede siano completati in uno sprint definisce la velocità di sviluppo. È importante che essa sia costante lungo tutta la durata del progetto per garantire un flusso teso che porti alla consegna di deliverable ad ogni sprint. La velocità dello sprint può essere diversa per diversi team, perché può cambiare leggermente il metodo di stima delle attività, essendo un metodo basato sulla soggettività. Pertanto, un team con velocità

più alta non indica un maggior impegno del team, ma soltanto una stima fatta utilizzando valori tendenzialmente più alti.

3.10.3 Sprint backlog

Durante lo sprint planning meeting, vengono decise le attività da svolgere durante lo sprint e viene creato lo sprint backlog. Ricordando l'obiettivo di snellezza della documentazione, lo sprint backlog è quindi un documento di una pagina che viene stampato, conservato dal Team Leader e archiviato una volta concluso lo sprint, in modo tale da poter ripercorrere a ritroso in un secondo momento le attività svolte. Questo documento riassume quello che viene decretato durante lo sprint planning: data di inizio sprint, data di fine, attività prese in carico, obiettivo dello sprint e deliverable, eventuali cambiamenti della composizione del Team di Sviluppo.

3.10.4 Sprint board

Il documento Sprint backlog verrà poi trasferito all'interno della sprint board, e utilizzando i kanban, sarà la base per la pianificazione quotidiana del lavoro all'interno del team. La sprint board è strutturata come una sorta di matrice: vi sono 3 colonne che suddividono le attività in To Do, In Progress e Done, ovvero attività ancora da svolgere, in corso di svolgimento e completate e pronte per essere consegnate al Project Manager prima e al cliente poi; le righe rappresentano invece i diversi membri del team, ciascuno dei quali è responsabile del completamento di una serie di requisiti, assegnati a lui dal Team Leader. I requisiti da sviluppare e le attività da completare sono segnati su post-it, che vengono spostati e mossi all'interno della sprint board nelle varie celle. Ogni post-it è caratterizzato da il nome dell'attività o il codice identificativo e dal numero di ore lavorative stimate per completarlo. Ogni giorno, al termine del Daily meeting, viene aggiornata la sprint board con le nuove informazioni disponibili, spostando i post-it se le attività su di essi cambiano fase e aggiustando il monte ore previsto se la stima risulta errata.

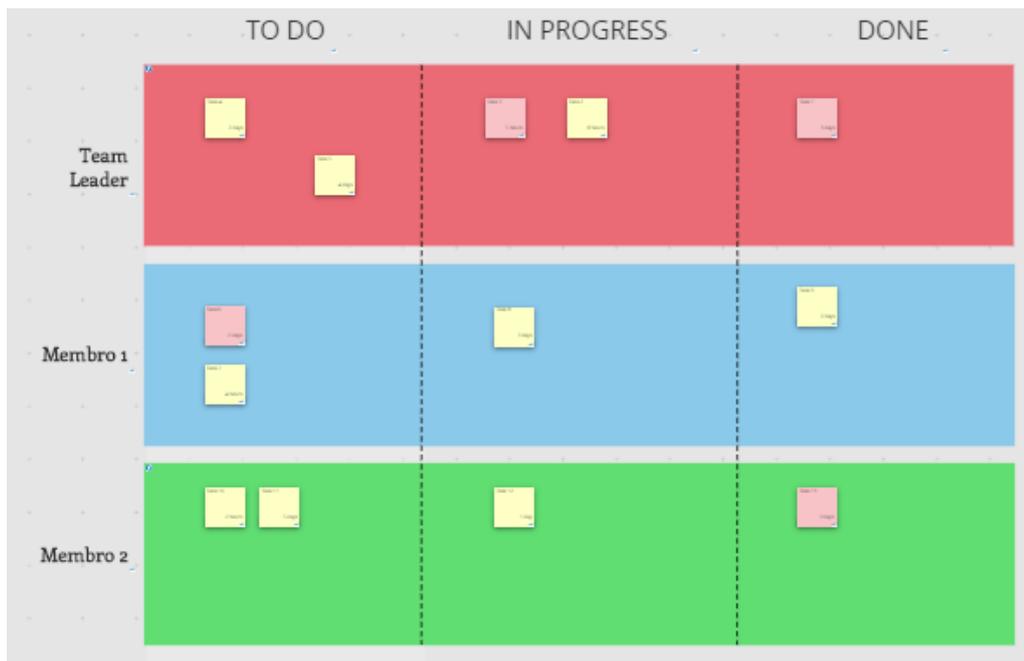


Figura 10. Schema illustrativo di una sprint generica, con post-it per ciascuna task suddivisi nelle tre diverse fasi di completamento.

La gestione delle attività tramite kanban può essere fatta sia tramite kanban fisici (post-it su lavagna), sia utilizzando software dedicati. Tra i più conosciuti vi sono Freedcamp, un sito con funzionalità freemium che permette la gestione dei kanban, assegnandoli ai membri del Team di Sviluppo, modificandone la priorità e lo stato di avanzamento, aggiungendo commenti e note per gli altri membri del team e definendo scadenze. Una feature che può essere molto utile è la possibilità di allegare dei file alle singole task, ad esempio un file CAD, per mostrare l'avanzamento del progetto, tuttavia può risultare rischioso condividere online tali file, per questioni di cyber security. Risulta comunque ottimo come strumento di comunicazione. Un altro software disponibile su licenza è Jira, che contiene le funzionalità di Freedcamp ma dedicato particolarmente al mondo Agile. [10, 11]

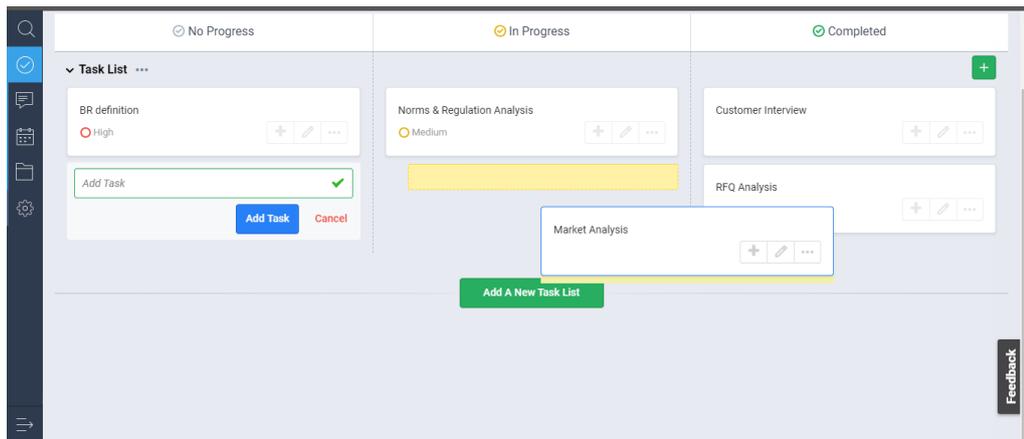


Figura 11. Screenshot di una schermata di Freedcamp, software che può essere utilizzato per l'informatizzazione dei kanban.

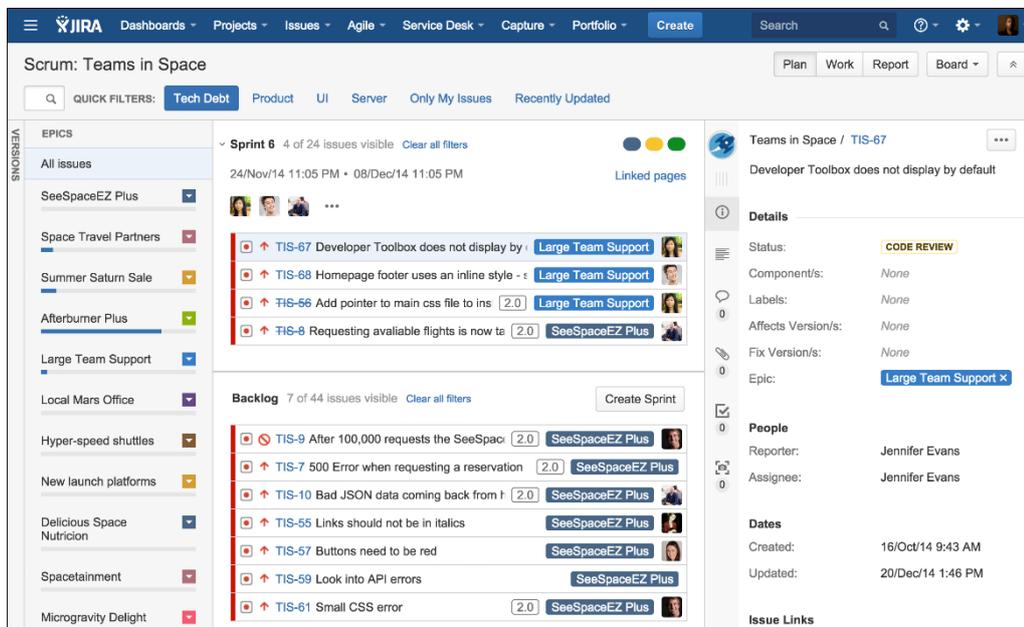


Figura 12. Screenshot di una schermata di Jira, software dedicato alla gestione dei progetti con tecniche Agile.

3.10.5 Matrici di responsabilità

All'interno di ogni gate deve essere chiaro chi è il responsabile delle decisioni da prendere e dell'avanzamento del progetto. Ogni attività deve avere il suo responsabile e un momento decisionale adeguato. Per favorire ciò, nel Piano di Sviluppo Prodotto sono state indicate delle cosiddette *matrici di responsabilità*. Si tratta di tabelle a doppia entrata con sulle colonne diverse figure di

riferimento dell'organizzazione del progetto e vertici aziendali (Responsabili di funzione, CEO), mentre sulle righe vi sono le macro-attività svolte durante lo Stage. Incrociando righe e colonne vi è indicato in ruolo che ha la particolare figura rispetto a quell'attività: può esserne responsabile, co-responsabile, può essere soltanto informato di tale decisione o è colui che deve approvarla.

3.10.6 Burndown chart

Come indicato in tutte le guide Scrum, uno strumento utile per monitorare l'andamento delle attività durante uno sprint è il burndown chart. Come spiegato nel Paragrafo 2.2.5, questo grafico deve essere aggiornato quotidianamente dal Team Leader dopo i daily meeting tenendo conto delle attività che il team ha completato. È utile per capire se si è in ritardo o in anticipo con la schedulazione. Il Project Manager può creare un burndown chart con orizzonte temporale maggiore di uno sprint, ad esempio uno stage o l'intero progetto, per monitorare la commessa nel suo complesso.

La frammentazione di grandi attività in piccole task quantificabili in poche ore-uomo di lavoro permette di ottenere un grafico burndown molto più accurato e consente al Team Leader di correggere tempestivamente i carichi di lavoro se la pianificazione non è perfettamente rispettata.

3.11 Criteri passaggio gate

La presenza dei metodi di gestione dei progetti Agile all'interno del framework Stage-Gate ha permesso la trasformazione dei criteri di superamento degli stessi Gate.

Se nei modelli tradizionali, per superare un Gate era necessario aver completato al 100% tutte le attività e soddisfatto tutti i requisiti previsti in quel determinato Stage e solo in quel momento potevano essere iniziate attività relative allo Stage successivo, nel modello ibrido proposto questo non è più vero. È possibile avanzare lungo il processo, superare un Gate e iniziare uno

Stage anche se le attività di quello precedente non sono state completamente terminate. La soglia per il superamento del Gate è stata fissata all'80% delle attività previste.

Il lettore potrebbe chiedersi come sia possibile iniziare step successivi dello sviluppo del prodotto se non tutti i requisiti e le attività su cui si basa tale sviluppo siano completati, e perché sia stata scelta esattamente la soglia dell'80%. La risposta alla prima domanda è insita nel funzionamento del Product Backlog: vengono sviluppati e completati prima i requisiti che hanno una priorità più alta, che portano più valore e che soprattutto diventano input per successivi sviluppi. La corretta priorizzazione del backlog permette la realizzazione anticipata delle attività fondamentali. Raggiungendo la percentuale prefissata, si hanno tutti gli elementi sufficienti per poter proseguire lo sviluppo; le attività rimanenti saranno comunque completate nei primi sprint del nuovo Stage oppure, con l'emergere di nuove informazioni e feedback del cliente, requisiti che avevano bassa priorità potrebbero essere addirittura cancellati.

Per quanto riguarda la scelta della soglia dell'80%, questa resta sicuramente una questione aperta. Si è scelto di utilizzare tale soglia in attesa di verificarne il corretto funzionamento applicando tali criteri ad un progetto reale. Non mancano comunque i fondamenti teorici per questa scelta. Se l'attività di priorizzazione delle task è stata condotta in maniera corretta, le attività sicuramente completate durante uno Stage sono quelle che forniscono maggior valore al prodotto e sono propedeutiche e/o necessarie ad attività che si svolgeranno successivamente in altri Stage, costituendone gli input. Aver completato l'80% di queste attività è un modo matematico e di stima per dire "si è a buon punto", si può quindi superare il Gate perché sono disponibili tutti gli input indispensabili per le attività future; quelli non ancora completati saranno comunque terminati nei primi sprint del nuovo Stage oppure,

trattandosi di requisiti di poco valore, potrebbero essere eliminati dal backlog con il sopraggiungere di nuove informazioni.

Ricordando che Scrum dà molta enfasi alla priorizzazione basata sul valore, è necessario sottolineare che spesso il completamento dei requisiti ha un andamento paretiano: completando il 20% dei requisiti si genera l'80% del valore. Ponendo la soglia di completamento all'80% delle attività svolte si è quasi sicuri di aver estratto quasi totalmente il valore disponibile in quel determinato Stage.

Il Gate spesso coincide con un importante pagamento del cliente che quindi vorrebbe soddisfatti tutti i requisiti, non solo buona parte di essi. Per superare questo problema, il framework di Scrum è molto utile: la frequente interazione con il cliente, i deliverable presentati a ogni sprint e i costanti feedback permettono al cliente di conoscere precisamente lo stato di avanzamento del prodotto, diventando più favorevole ad accettare un superamento del Gate con l'80% di attività completate.

3.12 Vantaggi e svantaggi attesi

La proposta di questo nuovo Piano di Sviluppo Prodotto Agile-Stage-Gate porta con sé diversi vantaggi, derivati quasi totalmente dalle metodologie di project management Agile di nuova adozione.

Innanzitutto, la frammentazione del lavoro in piccole task e soprattutto la scomposizione dei lunghi stage in brevi sprint dalla durata prestabilita di quattro settimane costituisce una semplificazione al processo di sviluppo prodotto. Attività molto complesse e molto lunghe vengono suddivise in varie unità di lavoro elementari che facilitano molto la stima e la pianificazione del lavoro, diminuendo l'incertezza sulle stime, che rispecchieranno di più la durata effettiva delle attività. Una volta entrato a regime consentirà una gestione più snella del processo di sviluppo nuovo prodotto.

Alla fine di ogni sprint vengono creati dei deliverable o incrementi di prodotto potenzialmente consegnabili al cliente. Ciò consente di avere un feedback continuo da parte degli stakeholder, che possono molto più frequentemente controllare l'avanzamento dello sviluppo del prodotto e verificare che i loro requisiti siano pienamente soddisfatti. La consegna di aggiornamenti frequenti e regolari consentirà alla società di ricevere pagamenti più regolari, che potranno, ad esempio, essere schedulati alla fine di ogni sprint secondo una logica SAL (Stato Avanzamento Lavori).

La suddivisione del tempo in sprint e il frequente feedback del cliente comporta un ulteriore vantaggio: una maggior velocità di risposta al cambiamento. Il cliente infatti può potenzialmente fornire risposte e cambiare le proprie richieste ogni sprint; ciò si traduce in un aggiornamento del product backlog e quindi già dallo sprint successivo il Team di Sviluppo può prendere in carico le modifiche richieste dal cliente. Tutto questo si converte in una riduzione dei tempi di sviluppo e del time to market.

La presenza di gate con criteri di superamento flessibili, che permettono l'avanzamento allo stage successivo anche se non sono stati terminate completamente tutte le attività dello stage precedente, consente di evitare fasi di stallo in cui il processo non avanza perché alcune attività non fondamentali devono essere completate. Evitando queste pause del processo, si riduce ulteriormente il time to market.

Infine, con l'adozione all'interno dei Team di Sviluppo delle tecniche di project management agile, ci si attende un miglioramento della comunicazione all'interno del team e tra i diversi team di sviluppo; sono disponibili infatti numerosi strumenti e riunioni dedicati esclusivamente a migliorare la comunicazione e allineare lo sviluppo del prodotto, evitando che i singoli team seguano percorsi diversi.

Dall'adozione del piano di sviluppo prodotto ibrido ci si aspettano anche alcune difficoltà e svantaggi.

Le prime difficoltà previste sono relative proprio all'adozione di tali tecniche. Alcuni dipendenti potrebbero trovare difficoltà nel cambiamento del metodo di organizzazione; sarà compito del PM e del Team Leader trovare delle persone dinamiche, motivate e favorevoli al cambiamento per i primi progetti. Successivamente, dopo la prima fase di sperimentazione, saranno condotti degli incontri per formare tutti i dipendenti e mostrare i vantaggi di queste nuove metodologie.

Il feedback del cliente continuativo e ripetuto nel tempo costituisce sicuramente un vantaggio. Guardandolo da un diverso punto di vista, però, si possono incontrare diverse difficoltà nel trovare clienti o stakeholder che siano davvero disponibili a ricevere aggiornamenti ogni 4 settimane e fornire un feedback rapido. Spesso infatti i clienti preferiscono avere scadenze molto meno frequenti. Questo non deve però costituire un problema per le aziende che dovranno comunque continuare a seguire internamente gli schemi del Piano di Sviluppo Prodotto, cercando di coinvolgere il più possibile il cliente.

4 Il caso studio

In questo capitolo si mostreranno i risultati del caso studio condotto in Blue Engineering srl, società di engineering & design automotive. Verrà applicato il piano di sviluppo prodotto che si è proposto nel Capitolo 3 e se ne valuterà sul campo l'effettiva applicabilità. Sarà presentato il progetto pilota preso in esame e si mostreranno alcuni elementi e momenti importanti della gestione del progetto.

4.1 Blue Engineering srl

Blue Engineering Srl è una società con sede a Rivoli (TO) le cui attività principali sono l'ingegneria e la progettazione di componenti e veicoli appartenenti principalmente ai settori dell'automotive e ferroviario. Blue Engineering è una società in forte espansione sia sul territorio italiano, ma soprattutto in campo internazionale, dal momento che la quota maggioritaria è detenuta da soci cinesi.

La tendenza a una rapida innovazione e la necessità di avere un piano di sviluppo prodotto molto flessibile, vista la varietà di progetti presi in carico, fa di Blue Engineering un'ottimo modello per l'applicazione del framework proposto. Il caso studio sarà condotto limitatamente all'area automotive, ma è interesse della società espanderlo a tutte le attività progettuali dopo averne valutata la bontà.

4.2 Il progetto pilota

Il progetto al quale si è applicato il piano di sviluppo prodotto Hybrid Agile-Stage-Gate è un progetto di pesante restyling di una vettura di segmento C destinata a un mercato emergente. In particolare, sono mantenuti il telaio e alcuni elementi meccanici principali come motore, trasmissione, sospensioni, sterzo, impianto frenante. Della vettura originale si conservano inoltre la struttura dei sedili, parabrezza e finestrini laterali. La fase di restyling riguarda

invece tutta la carrozzeria esterna (portiere, entrambi i paraurti, cofano motore, baule, ecc), il cruscotto, la console centrale, e in generale tutti i pannelli interni e la forma dei sedili.

Si tratta di un progetto molto impegnativo che terrà impegnata la società per 2 anni abbondanti secondo il plan iniziale.

Le tempistiche del tirocinio in Blue Engineering hanno permesso all'autore di questa tesi di applicare la proposta di piano di sviluppo prodotto soltanto a progetto già avviato. Quando una bozza del piano di sviluppo prodotto era pronta, il progetto si trovava all'inizio dello Stage 2 Concept Alternatives. La valutazione circa la fattibilità economica e strategica della commessa era già stata effettuata precedentemente e si era già terminata la definizione dei Basic Requirements, seguendo il piano di sviluppo prodotto precedentemente adottato in Blue Engineering.

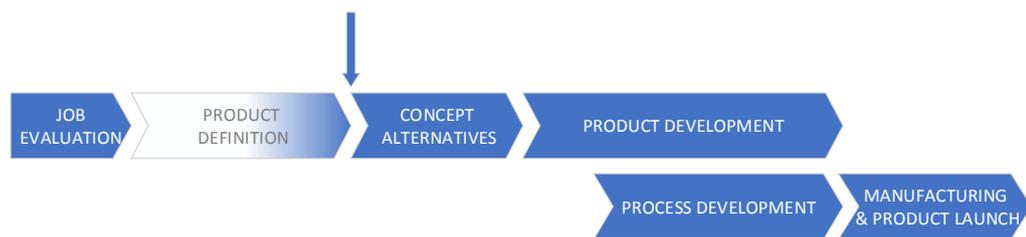


Figura 13. Schema illustrante il momento di partenza della sperimentazione del nuovo piano di sviluppo prodotto all'interno di Blue Engineering srl. La freccia all'inizio dello Stage 2 indica l'inizio della sperimentazione.

L'applicazione del modello ibrido al progetto pilota ha reso necessarie alcune modifiche all'organizzazione del progetto, ma sono state di facile risoluzione dal momento che il progetto non era ancora entrato nella fase di sviluppo più profonda.

Data l'imponente mole di lavoro e complessità delle attività di progettazione di una vettura, si è deciso di creare due Team di Sviluppo dedicati al progetto: il primo dedicato alle attività di sviluppo concept e design dell'esterno della vettura, mentre il secondo dedicato all'interno. Entrambi i team, come

suggerito nella proposta di piano sviluppo proposto, erano cross-funzionali e con membri multidisciplinari. Entrambi i team erano composti da 10-15 membri ciascuno, alcuni dei quali presenti solo in alcuni momenti o sprint. La scelta di affidare a due team distinti è stata fatta per evitare di avere troppe persone all'interno dello stesso team, situazione che avrebbe certamente causato problemi organizzativi e di comunicazione. La suddivisione Team di Sviluppo Interni e Esterni è sembrata la più logica, vista la tipologia di attività previste.

4.3 Preparazione

La prima attività portata a termine prima di proseguire con le attività dello Stage 2 è stata la costituzione dei Team di Sviluppo. Come anticipato, sono stati costituiti due team, uno per gli interni e uno per gli esterni del veicolo, la cui composizione, durante il primo sprint, è stata per entrambi la stessa e elencata di seguito:

- 4 Stilisti
- 4 disegnatori CAS (Computer Aided Surfaces)
- 3 progettisti CAD (Computer Aided Design)
- 1 esperto CAE (Computer Aided Engineering)

Stilisti e disegnatori CAS appartengono alla divisione aziendale di Stile, i progettisti CAD all'area Engineering mentre l'esperto CAE a quella di Calcolo e Simulazioni. L'appartenenza a diverse aree aziendali evidenzia fin da subito la *cross-funzionalità* dei Team di Sviluppo, una delle caratteristiche fondamentali del metodo di organizzazione dei progetti proposto.

Dopo aver istituito i team, sono stati definiti gli altri due ruoli principali. Il Project Manager è rimasto lo stesso della prima parte del progetto, continuando la sua opera di comunicazione con il cliente, rappresentandone gli interessi all'interno del progetto. Per ogni team infine si è identificato il Team Leader che, oltre ad avere una profonda esperienza tecnica, è stato

responsabile della corretta diffusione delle metodologie Scrum e dell'organizzazione del lavoro all'interno del Team. Essendo tecniche di recente adozione all'interno dell'azienda, l'autore della tesi ha assunto inoltre il ruolo di Scrum Master così come viene definito all'interno della Guida Scrum, ovvero leader servitore con il solo compito di facilitare l'adozione e il giusto utilizzo degli strumenti Scrum all'interno del team. Tale figura è stata prevista per entrambi i Team nel breve periodo in cui l'autore è stato presente in azienda.

Sono state infine individuate le figure di staff previste. Un rappresentante dell'area Finance e uno dell'area commerciale sono stati assegnati al progetto per partecipare alle riunioni Scrum of Scrum e supportare il progetto.

Nella Figura è mostrato uno schema sintetico dell'organizzazione del progetto.

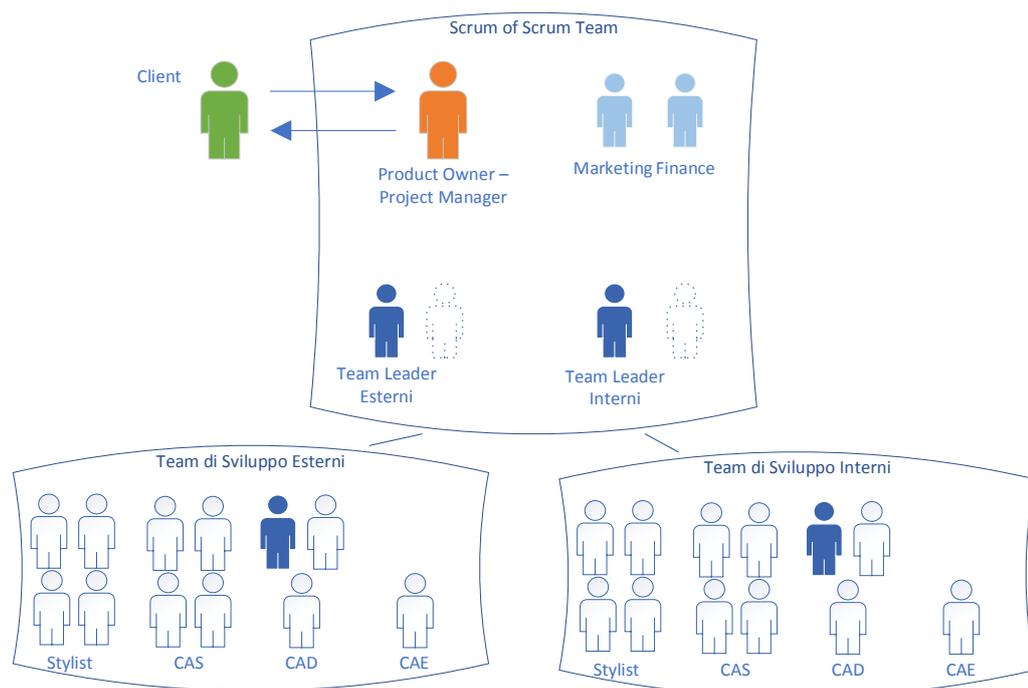


Figura 14. Schema dell'organizzazione del progetto di re-styling. Sono presenti due team, uno per interni vettura e uno per esterni, coordinati da un unico Project Manager.

4.4 Product backlog

Basandosi sul master plan preparato dal PM in fase di pianificazione è stato preparato un grande product backlog contenente tutte le macro-attività

previste. Esse sono state ordinate seguendo una priorità decrescente, usando come criterio principale di prioritizzazione i vincoli temporali del progetto, ovvero la sequenza logica e tecnologica di attività previste. Una volta creato tale backlog, sono state evidenziate le attività relative all'interno della vettura e quelle all'esterno, così da facilitare la pianificazione delle attività per i due Team di Sviluppo.

Successivamente i due team hanno cercato di aumentare il dettaglio del product backlog scomponendo le macro-attività in task più piccole, stimandone la durata e aggiornando di fatto il product backlog. Questa attività è stata svolta dai due team in maniera piuttosto autonoma, con il PM che ha fornito le date del planning che i team dovevano rispettare. La parcellizzazione del lavoro è stata fatta per le attività dei primi tre mesi, corrispondenti quindi ai primi tre sprint dello Stage 2.

Nella fase di stima delle attività, per una questione di abitudine aziendale e di maggior familiarità con il metodo, non sono stati utilizzati gli story points bensì un metodo di stima più tradizionale, basato esclusivamente sulla durata delle task. I membri del team quindi, basandosi sulla loro esperienza passata, hanno stimato la durata delle singole task, per avere una stima più oggettiva e non solo relativa delle attività da svolgere durante ciascuno sprint.

Per dettagliare molto le attività del backlog e facilitare le attività di progettazione degli sviluppatori, si è cercato di associare a ciascuna task i Basic Requirement derivanti dallo Stage 1 e le specifiche tecniche. Sono stati riscontrati molti problemi, soprattutto in quanto il backlog diventava un documento davvero pesante e di difficile lettura. Pertanto, si è deciso di mantenerli separati e inserire nel backlog soltanto la lista delle attività priorizzate, cercando di parcellizzarle il più possibile.

Task/Requirements	Stima	Priorità	Stage	Int/Ext
Style	60 days	5	2	
Sketches realization exterior	7 days	5	2	Ext
5x front view	2 days	5	2	Ext
5x rear view	2 days	5	2	Ext
5x side view	1 days	5	2	Ext
5x full vehicle view	3 days	5	2	Ext
3x headlight	4 days	5	2	Ext
3x rear light	3 days	5	2	Ext
Sketches realization interior	7 days	5	2	Int
5x dashboard	4 days	5	2	Int
3x tunnel console	2 days	5	2	Int
3x front seats	3 days	5	2	Int
3x rear seats	2 days	5	2	Int
5x full vehicle	3 days	5	2	Int
Power point preparation ext	1 day	5	2	Ext
Power point preparation int	1 day	5	2	Int
Sketch choice	1 day	5	2	Int/Ext
Rendering ext	5 days	5	2	Ext
Rendering int	5 days	5	2	Int
C-Class Exterior	46 days	5	2	Ext
C-Class Closure	46 days	5	2	Ext
Front door	24 days	5	2	Ext

Figura 15. Schermata del Product Backlog prioritizzato del progetto.

Durante la giornata il PM ha progressivamente aiutato e validato le stime fatte dai membri dei team, arrivando a fine giornata ad avere un backlog pronto.

4.5 Day 1: sprint planning

Il giorno seguente sono ufficialmente iniziati gli sprint con le due riunioni di Sprint Planning. Al mattino il PM ha sovrinteso il planning per gli interni, mentre al pomeriggio per gli esterni. Entrambe le riunioni sono durate 4 ore. Per facilità di gestione, gli sprint sono partiti di lunedì.

Partendo dal product backlog completato il giorno precedente, il PM ha proposto al team una serie di attività da completare durante lo sprint, quelle poste in cima al backlog. Dopo aver discusso ancora sulla stima di tali task, è stato stabilito l'obiettivo del primo sprint. L'obiettivo del primo sprint è stato

quello di creare i bozzetti di stile, renderizzarli cioè creare i modelli di stile 3D, e iniziare l'attività di CAS, fare i primi studi di macro-fattibilità, definendo i componenti nuovi e carry-over, ovvero derivati direttamente dal vecchio modello da rinnovare.

Successivamente, il Team Leader ha assegnato a ciascun membro le prime attività da svolgere, ascoltando consigli e accogliendo proposte di cambiamento che provenivano dai componenti del team. È stato completato lo Sprint Planning Report, quindi creato lo Sprint Backlog e la Sprint Board fisica, utilizzando i post-it.



Figura 16. Estratto dallo Sprint Planning Report, mostrante date inizio e fine sprint, il backlog e i deliverables previsti.



Figura 17. Foto della Sprint Board preparata al termine dello Sprint Planning meeting.

4.6 Daily meeting

I daily meeting o daily scrum si sono svolti tutte le mattine alle 9:30. L'unica eccezione è stata fatta durante le prime settimane dello sprint, in cui i due meeting avvenivano a distanza di mezz'ora l'uno dall'altro per permettere all'autore della tesi di assistere ad entrambe in qualità di Scrum Master, dando qualche consiglio su come svolgerla e portando le proprie conoscenze di Scrum. Dopo i primi giorni, una volta entrati a regime, i team conducevano regolarmente e autonomamente le riunioni.

Ciascun membro, come suggerito dalla teoria, ha risposto alle tre domande: cosa ho fatto ieri?, cosa farò oggi?, che impedimenti ho trovato? cercando di essere più breve e conciso possibile. Al termine delle domande, veniva aggiornata la Sprint board, spostando i post-it con su scritte attività che avevano cambiato fase.

Alcuni giorni la riunione ha sfiorato i 15 minuti prestabiliti, ma nel complesso i meeting hanno funzionato.



Figura 18. Foto della Sprint Board scattata circa a metà sprint.

4.7 Scrum of scrum

Il venerdì di ogni settimana è stato previsto lo Scrum of Scrum meeting per allineare e coordinare il lavoro dei due Team di Sviluppo. Durante questo meeting l'esperienza diretta ha suggerito alcune modifiche da apportare all'organizzazione.

Innanzitutto, la presenza dei soli Team Leader come rappresentanti del team non è sufficiente. Questa riunione serve a fare il punto della situazione, valutare l'avanzamento del lavoro e risolvere alcuni problemi emersi. Se per le attività di controllo e coordinamento il Team Leader era sufficiente, per l'analisi dei problemi e la loro risoluzione spesso non aveva le competenze necessarie.

Nella fattispecie, i due Team Leader erano due progettisti esperti dell'area Engineering, con parecchi anni di esperienza pregressa. Non hanno avuto alcuna difficoltà a rispondere a dubbi dei colleghi circa le attività di fattibilità ingegneristica svolte, ma quando è emerso un problema relativo alla fase di stile, si sono entrambi resi conto di avere delle difficoltà a proporre soluzioni valide. Pertanto, è stato stabilito che a tali riunioni potessero partecipare, in caso di necessità, anche altri membri del team, dei "rappresentanti" per ciascuna area di attività, ad esempio uno per lo stile esterni e uno per simulazioni interni, per portare all'interno della riunione Scrum of Scrum le loro competenze altamente specifiche e trovare soluzioni più corrette.

Un'altra novità che si è resa necessaria in Blue Engineering è la presenza ancora più forte del cliente all'interno delle attività di sviluppo prodotto, soprattutto nei primi Stage in cui le attività sono piuttosto rapide. Ad esempio, le attività di preparazione bozzetti di stile sono attività abbastanza veloci, con un'alta variabilità poiché è molto importante la vena artistica e l'ispirazione degli stilisti. Nonostante questo, in una decina di giorni, 4 stilisti possono presentare un numero considerevole di idee di stile della vettura. A questo punto per procedere al rendering e successivo CAS della vettura è necessario che il cliente scelga uno dei concept di stile, e che solo quello venga successivamente sviluppato. Sarebbe uno spreco di risorse procedere allo sviluppo parallelo di molti concept in attesa della convalida del cliente a fine sprint; ancora peggio sarebbe stare inoperativi fino a fine sprint aspettando il feedback. Per evitare queste inefficienze, il cliente è stato invitato a partecipare allo Scrum of Scrum meeting di metà sprint. Durante tale riunione, il cliente ha potuto valutare i diversi concept proposti e ne ha scelto uno per gli interni e uno per gli esterni su cui il team ha successivamente lavorato e sviluppato. Bisogna far notare però che questo feedback anticipato può capitare solo in alcuni sprint con delle particolari necessità. La riunione Scrum of Scrum con il cliente presente ha

avuto una durata molto superiore alle due ore proposte, occupando quasi tutta la giornata lavorativa.

4.8 Sprint review e deliverable

A fine sprint, il PM ha incontrato i due Team di Sviluppo, interni al mattino ed esterni al pomeriggio, per la consegna dei deliverable. I Team Leader dei due team hanno presentato l'avanzamento del progetto e il PM ha potuto valutare lo stato di completamento delle attività e il loro allineamento con l'obiettivo dello sprint. La valutazione ha avuto esito positivo e il PM ha proceduto a presentare al cliente i deliverable prefissati. È stato chiesto al cliente un tempo di risposta piuttosto rapido, di un giorno al massimo, così da avere la massima velocità di risposta ad un eventuale nuova richiesta.



Figura 19. Alcune immagini raffiguranti deliverables del primo sprint: sono presenti bozzetti di stile, C-Class e definizione dei componenti carry-over

4.9 Sprint 2

Il lunedì successivo è iniziato il secondo sprint, e il ciclo si è nuovamente ripetuto. Durante il secondo Sprint Planning Meeting è stato ufficializzato un

cambiamento della composizione del team che era già stata preventivata: si è ridotto il numero degli stilisti in entrambi i Team, essendo terminata la preparazione dei bozzetti, mentre è stato aumentato il numero di progettisti CAD per aver il numero di risorse sufficienti al completamento delle attività di fattibilità ingegneristica (totali di 8 progettisti per l'interno, 5 esterno). Tale modifica dei due Team di Sviluppo è stata riportata anche all'interno dello Sprint Planning Report.

4.10 Osservazioni

In questo paragrafo si faranno delle osservazioni sugli aspetti positivi e negativi emersi durante i primi due sprint di utilizzo del piano di sviluppo prodotto e si raccoglieranno le prime impressioni e risultati qualitativi.

Il primo problema incontrato in fase di preparazione è stata la creazione del product backlog. Per facilitare la programmazione giornaliera del lavoro e l'utilizzo della Scrum board, era necessario parcellizzare il più possibile le attività in sub-task abbastanza dettagliate e di breve durata. Per molte attività (ad esempio le verifiche omologative) questo è stato possibile e piuttosto facile. Per altre attività invece tutti i membri del team hanno trovato serie difficoltà nella suddivisione. Ad esempio, le attività di sviluppo delle superfici C-Class, ovvero le superfici 3D di stile necessarie a dare una corretta forma agli esterni e interni della vettura, non possono essere suddivise in piccoli e brevi task. L'unica scomposizione che è stata fatta è stata geometrica, dividendo le parti della vettura da progettare. Nel product backlog sono stati inseriti ad esempio C-Class front bumper o C-Class dashboard-driver, ma la durata di queste attività era comunque nell'ordine di 10-20 giorni. Per valutarne l'avanzamento, si è pensato di misurarlo come una percentuale di completamento, utile soprattutto per alcune C-Class il cui sviluppo cadeva a cavallo dei due sprint. Tale metodo di calcolo però si è rivelato di difficile stima, e tale stima è stata affidata all'esperienza dei membri del team e del PM.

Sempre relativamente al product backlog, un'ulteriore difficoltà riscontrata dallo staff durante la sua preparazione è stata l'inserimento delle specifiche tecniche come attributi delle task. La dimensione eccessiva del documento e la poca facilità di lettura hanno spinto i Team di Sviluppo a mantenere separate le attività dalle specifiche del prodotto, inserendo nello stesso file Excel, ma in due fogli di lavoro diversi, il product backlog e le specifiche tecniche.

Un'ultima variazione effettuata in fase di stima rispetto al modello proposto nel Capitolo 3 è stato un diverso metodo di stima. Aniché utilizzare gli story point suggeriti dalla letteratura Scrum, si è optato per un metodo di stima più tradizionale, indicando l'effort di ogni task come una durata, espressa in giorni o ore lavorative. È stata fatta tale scelta per una maggior familiarità dei progettisti a questo metodo, con l'obiettivo di sperimentare gli story point in un prossimo futuro.

Come già mostrato, si è reso necessario far partecipare più frequentemente il cliente ad alcune riunioni. Questo è avvenuto durante lo Sprint 1, ma non nello Sprint 2. È importante la partecipazione del cliente all'interno del framework Scrum, ma è altrettanto importante capire se si è trattato di un'"eccezione alla regola". La consegna dei deliverable dello sprint viene fatta ufficialmente solo una volta che lo sprint è terminato. Sono ovviamente ammessi contatti tra il PM e il cliente durante tutta la durata dello sprint, ma non si devono confondere questi con la consegna dei deliverable. Se fosse sempre più frequente la necessità di coinvolgere il cliente durante lo Scrum of Scrum di metà sprint, sarebbe sintomo di un'errata durata dello sprint stesso. A quel punto sarebbe più corretto diminuire la durata dello sprint a due settimane, ricalibrando le attività al suo interno e avere così un feedback del cliente ogni due settimane.

Le riunioni Scrum of Scrum, come già detto, sono state aperte a più persone. In particolare, è stata favorita la partecipazione di alcuni portavoce delle funzioni

presenti all'interno del Team di Sviluppo e non rappresentate dal Team Leader. La loro presenza è stata richiesta per portare all'interno del Core Team alcune competenze che solo i Team Leader non avrebbero potuto portare. Questi portavoce sono stati convocati a chiamata, ovvero solo quando la particolare situazione richiedeva una competenza specifica. Nei due Sprint esaminati durante il caso studio, questa convocazione è avvenuta 2 volte: la prima durante la prima settimana, per verificare che i bozzetti di stile interno ed esterno fossero in armonia tra loro, e la seconda a metà secondo sprint, per risolvere un problema relativo alle sezioni tipiche.

Riguardo a questo progetto di restyling, essendo di dimensioni comunque ridotte rispetto alla progettazione di un veicolo completo, un solo PM si è rivelato sufficiente. Per progetti di più ampie dimensioni sono stati avanzati dubbi sull'impegno di un solo PM a gestire lo sviluppo del prodotto e i rapporti con il cliente. Per progetti di grandi dimensioni e elevata complessità è possibile individuare dei collaboratori che assistano il PM e che si dividano il lavoro di gestione, magari mantenendo la suddivisione in vettura interna ed esterna.

Durante la fase di preparazione del product backlog, alcuni progettisti hanno espresso la volontà di creare un backlog separato per ciascun Team di Sviluppo; l'attività di creare e successivamente dettagliare le attività è stata svolta dai due Team di Sviluppo in maniera separata. Una volta creati i due backlog con un livello di dettaglio sufficiente, si è comunque scelto di unirli all'interno di un unico backlog, cosicché tutto il personale coinvolto nel progetto, indipendentemente dal team di appartenenza, potesse avere un quadro generale di tutte le attività da fare e con quale priorità e ordine i requisiti del cliente dovessero essere sviluppati. Per venire incontro ai bisogni dei membri del team, si è deciso di aggiungere una colonna all'interno del backlog con un flag "Int" o "Ext", per filtrare facilmente gli elementi del backlog a seconda delle esigenze e del team di appartenenza.

L'affidamento della creazione del backlog direttamente ai Team di Sviluppo è stata essa stessa una novità, in quanto la teoria suggerisce che il responsabile della creazione del backlog sia il Product Owner, qui rappresentato dal Project Manager; ma anche in questo caso c'è stata una mancanza di competenze e di esperienza tecnica del PM, che ha preferito delegare l'attività, pur sovrintendendola e facendo rispettare le date e milestone fissate in fase di pianificazione.

Dopo i due sprint esaminati, l'autore della tesi ha condotto delle brevi interviste ai membri del team, al PM e Team Leader, per chiedere le prime impressioni dopo l'adozione del nuovo metodo di gestione dello sviluppo prodotto. Non avendo sufficienti dati per procedere ad un'analisi quantitativa dettagliata, come ad esempio un'analisi costi-benefici, ci si è limitati a riportare i principali risultati qualitativi emersi, qui di seguito illustrati.

La maggior parte degli intervistati ha riscontrato un miglioramento della comunicazione sia all'interno del team, che a livello più verticale. Questo è stato sicuramente favorito dai daily meeting in cui tutti i membri del team erano incoraggiati a parlare ed esporre i propri problemi. Anche durante le riunioni di inizio e fine sprint i singoli membri sono stati fortemente coinvolti, sono state ascoltate le loro proposte e la stima delle attività è stata fatta da tutti gli sviluppatori. Essendo il Team Leader non soltanto un "capo" unilaterale ma anche un ottimo motivatore, il team si è sentito più coinvolto e responsabilizzato, con un beneficio positivo sulle performance.

Dalle interviste è emersa anche una migliore efficienza del processo. La scomposizione delle attività e la rapida validazione del cliente ha permesso di evitare fasi di stallo che si verificavano con i vecchi sistemi di gestione, in cui spesso si doveva attendere un data prestabilita per poter avere un incontro con il cliente, mentre con il nuovo metodo almeno una volta al mese si riceve un'opinione degli stakeholder.

Una novità prevista dal piano che si è proposto ma che purtroppo non è stata possibile valutare è il funzionamento dei Gate flessibili. Per questioni di tempistiche, il progetto pilota che è stato analizzato partendo dallo Stage 2 non è arrivato al Gate 2 nel periodo in cui l'autore di questa tesi era presente in azienda. Resta quindi un aspetto importante da verificare nel prossimo futuro, per valutare la convenienza dei Gate con criteri di superamento possibile, che dovrebbero snellire ulteriormente il processo evitando fasi di stallo, mantenendo comunque un certo rigore e rigidità dello schema Stage-Gate.

5 Conclusioni e sviluppi futuri

Nel presente capitolo ci si propone di trarre le conclusioni in merito all'esito delle attività di preparazione di un piano di sviluppo prodotto automotive, attraverso metodi di gestione dei progetti tipici del mondo Agile, in particolare Scrum.

Come anticipato inizialmente, l'obiettivo prefissato consisteva nel valutare se Scrum potesse essere utilizzato con successo in un contesto diverso dallo sviluppo software, ambiente in cui si è diffuso per primo ed è in gran parte utilizzato. Lo scopo finale della tesi era l'ideazione e implementazione di un piano di sviluppo prodotto Agile, seguendo uno schema definito Hybrid Agile-Stage-Gate.

Al termine delle attività descritte nei precedenti capitoli, è possibile affermare che l'obiettivo è quasi totalmente raggiunto. Al momento si dispone di un piano di sviluppo prodotto che comprende tutte le fasi che una società di design automotive potrebbe dover portare a termine. Dall'analisi della letteratura e dalla collaborazione con esperti del settore, è stato ideato un modello ibrido di Agile-Stage-Gate, mantenendo un framework rigido ad alto livello ma dotandolo di una serie di strumenti e metodi di gestione snella e presa direttamente dal mondo Scrum, adattandola alle particolari esigenze del mondo automotive.

Dall'esperienza sul campo, si è notato che molte intuizioni che erano valide teoricamente si sono mostrate corrette anche all'applicazione pratica. La suddivisione delle attività all'interno di sprint dalla durata predeterminata ha effettivamente incrementato l'efficienza dello sviluppo del prodotto. La cross-funzionalità dei team di sviluppo ha permesso di sviluppare il prodotto in molteplici aspetti allo stesso tempo, minimizzando errori e incomprensioni dal momento che all'interno del team vi erano tutte le competenze necessarie allo

sviluppo. I frequenti meeting e il maggior coinvolgimento del personale ha migliorato notevolmente la comunicazione tra membri dello stesso team e tra team diversi, il personale del progetto è stato responsabilizzato, con benefici sullo sviluppo.

Alcune idee suggerite nella teoria si sono rivelate invece in parte sbagliate durante l'applicazione pratica. Si fa riferimento ad esempio alla presenza del cliente non solo a fine sprint con la consegna dei deliverables ma anche durante altri meeting all'interno delle 4 settimane, per fornire feedback ancora più frequenti e a volte resi necessari dalla natura delle attività; ne è un esempio l'approvazione dello stile della vettura a metà Sprint 1, che ha consentito al team di procedere immediatamente allo sviluppo dei render aumentando l'efficienza delle risorse. Un altro aspetto che è stato necessario correggere riguarda in qualche modo la cross-funzionalità del team: la presenza di diverse competenze all'interno del team erano coordinate e rappresentate da un solo Team Leader che in alcuni casi non aveva tutte le competenze necessarie a fornire consulenza su problemi altamente specifici. Un Team Leader dell'area Engineering difficilmente potrebbe risolvere problemi relativi allo stile. Per questo motivo gli Scrum of Scrum meeting sono stati aperti a più persone, convocati in base alla necessità, per aiutare il Team Leader fornendogli le competenze utili.

Per creare al meglio un product backlog dettagliato, è necessario suddividere attività molto lunghe in sub-task di breve durata, per favorire anche l'utilizzo della Scrum Board. L'evidenza pratica ha mostrato dei limiti, in quanto per alcune attività non si può scendere sotto un certo livello di dettaglio. Collegato al backlog, vi è anche l'inserimento al suo interno delle specifiche tecniche, risultato di difficile implementazione. Per risolvere tale problema si potrebbe suggerire di implementare un sistema informativo interno, che colleghi ciascuna task ad un elenco di requisiti specifici per lo svolgimento di quella

attività. Allo stesso tempo tale sistema dovrebbe aiutare nel calcolo della stima delle percentuali di completamento di alcune attività, compito che finora è stato affidato agli stessi componenti dei team.

Un ulteriore suggerimento per migliorare questo problema potrebbe essere un'applicazione più fedele di Scrum, che suggerisce di inserire i contenuti nel backlog sotto forma di user story, brevi frasi che contengono al loro interno anche sintetiche indicazioni sui requisiti da sviluppare. Si ritiene comunque che questo approccio non sia molto adatto al mondo automotive, in cui la quantità di specifiche è molto elevata, e si rischierebbe di avere user story troppo lunghe, complesse da scrivere e di non immediata comprensione.

Infine, alcuni aspetti sono tuttora da verificare. Uno su tutti il funzionamento dei Gate, visti non più come momenti di sbarramento con criteri rigidi come nelle teorie classiche, ma dotati di criteri flessibili, per permettere al processo di sviluppo prodotto di essere più snello per adattarsi meglio ad eventuali cambiamenti.

Scrum si è dimostrato adatto ad essere utilizzato in contesti diversi dalla sua storica tradizione informatica e, sebbene l'analisi effettuata abbia avuto come principale soggetto le società di engineering e design nel settore automotive, si ritiene che il framework proposto possa essere adottato, con le opportune modifiche, a qualsiasi settore industriale.

Fonti bibliografiche e sitografia:

[1] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M. et al. 2001. Manifesto for Agile Software Development. <http://agilemanifesto.org/>

[2] Guida al Corpo di Conoscenze di Scrum (Guida SBOK™) – edizione 2016

[3] Robert G. Cooper, *Winning at new products*, Holt, Rinehart and Winston of Canada, 1986

[4] Cooper, R. G. (2008), Perspective: The Stage-Gate® Idea-to-Launch Process—Update, What's New, and NexGen Systems. *Journal of Product Innovation Management*, 25: 213-232. DOI: 10.1111/j.1540-5885.2008.00296.x

[5] Robert G. Cooper (2016) *The Stage-Gate® System: A Road Map from Idea to Launch*, Gemba Insights

[6] A. Begel and N. Nagappan (2007) Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study, *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)*, Madrid, 2007, pp. 255-264. doi: 10.1109/ESEM.2007.12

[7] Robert G. Cooper & Anita Friis Sommer (2018) Agile-Stage-Gate for Manufacturers, *Research-Technology Management*, 61:2, 17-26, DOI:10.1080/08956308.2018.1421380

[8] Anita Friis Sommer, Christian Hedegaard, Iskra Dukovska-Popovska & Kenn Steger-Jensen (2015) Improved Product Development Performance through Agile/Stage-Gate Hybrids: The Next-Generation Stage-Gate Process?, *Research-Technology Management*, 58:1, 34-45, DOI: 10.5437/08956308X5801236

[9] Robert G. Cooper (2014) What's Next?: After Stage-Gate, Research-Technology Management, 57:1, 20-31, DOI: 10.5437/08956308X5606963

[10] <https://freedcamp.com/dashboard/home>

[11] <https://it.atlassian.com/software/jira>

[12] Guilherme Canuto da Silva, Paulo Carlos Kaminski & Duc Pham (2017) Proposal of framework to managing the automotive product development process, Cogent Engineering, 4:1, DOI: 10.1080/23311916.2017.1317318

[13] Karlström, D and Runeson, P (2005) Combining agile methods with stage-gate project management, IEEE Software, 22:3, 43-49, DOI:10.1109/MS.2005.59

Appendici

Diagramma di flusso del processo

