



**Politecnico  
di Torino**

**Politecnico di Torino**

Laurea magistrale in ingegneria gestionale

A.a. 2024/2025

Sessione di Laurea Novembre 2025

**Vantaggi derivanti  
dall'implementazione della  
metodologia Agile in un progetto  
assicurativo internazionale**

Relatore: Giovanni Zenezini

Candidato: Arianna Ciaccia



## Sommario

1	Introduzione.....	4
2	Metodologie di project management nei progetti.....	6
2.1	La metodologia Waterfall .....	6
2.1.1	Le fasi.....	6
2.1.2	Applicazioni.....	10
2.1.3	Punti di forza e limiti .....	11
2.2	La metodologia Agile .....	13
2.2.1	Sviluppo della metodologia Agile.....	13
2.2.2	Il manifesto Agile .....	15
2.2.3	I principi dell'Agile .....	17
2.2.4	Applicazioni.....	18
2.2.5	Il framework Scrum .....	21
2.2.5.1	Scrum Team .....	24
2.2.5.2	Cerimonie .....	28
2.2.5.3	Artefatti .....	31
2.3	Confronto tra Waterfall e Agile .....	33
3	Metodologia .....	38
3.1	Il settore assicurativo: complessità e sfide nei progetti IT.....	38
3.2	La metodologia Agile nel settore assicurativo .....	41
3.3	Contesto aziendale .....	46
3.4	Descrizione del progetto .....	47
3.5	Applicazione della metodologia Agile nel progetto .....	50
3.5.1	Burndown Chart .....	53
3.5.2	Burnup chart .....	54
3.5.3	Velocity Chart.....	56
3.5.4	Cumulative Flow Diagram (CFD) .....	57
3.6	Definizione dei Key Performance Indicators (KPI) .....	58
4	Analisi dei KPI .....	61
4.1	Applicazione dei KPI al progetto .....	61
4.1.1	Analisi quantitativa .....	61
4.1.1.1	Efficienza temporale e aderenza al piano .....	62
4.1.1.2	Qualità tecnica e testing.....	66
4.1.2	Analisi qualitativa .....	71

4.1.2.1	Gestione dei requisiti e flessibilità .....	72
4.1.2.2	Collaborazione e coinvolgimento stakeholders.....	77
4.2	Vantaggi e raccomandazioni per l'adozione di Agile nei progetti assicurativi .....	81
5	Conclusioni.....	83
6	Riferimenti bibliografici .....	85

# 1 Introduzione

Negli ultimi anni il mondo dei progetti IT ha subito un'evoluzione profonda, spinta dalla digitalizzazione e dalla necessità per le imprese di adattarsi rapidamente a contesti in continua trasformazione.

In particolare, il settore assicurativo, tradizionalmente caratterizzato da strutture organizzative complesse, processi regolamentati e sistemi legacy, si trova oggi al centro di un cambiamento che richiede maggiore flessibilità, collaborazione e rapidità decisionale. In tale scenario, la metodologia Agile si configura come una risposta concreta alle esigenze di innovazione e adattabilità, consentendo alle organizzazioni di ridurre la complessità e migliorare la qualità dei risultati attraverso un approccio iterativo e orientato al valore.

In particolare, le metodologie di project management rappresentano strumenti fondamentali per garantire il successo dei progetti, ma la crescente imprevedibilità dei mercati e la velocità del cambiamento tecnologico rendono sempre più difficile applicare modelli tradizionali come il Waterfall, basati su pianificazioni rigide e sequenziali. L'approccio Agile, al contrario, privilegia la collaborazione, la trasparenza e la capacità di adattamento continuo, permettendo di ottenere risultati più rapidi e di maggiore qualità grazie al coinvolgimento costante degli stakeholder e al rilascio incrementale di valore.

Il presente elaborato nasce dall'esperienza maturata durante il tirocinio presso la società di consulenza Accenture, nell'ambito di un progetto assicurativo internazionale volto alla dismissione e ristrutturazione delle piattaforme di gestione dei sinistri motor e non motor. In tale contesto, è stata adottata la metodologia Agile come modello di esecuzione del lavoro.

La motivazione principale alla base di questo studio risiede nella volontà di analizzare quali vantaggi concreti derivano dall'implementazione di tale metodologia nel progetto assicurativo, valutando il suo impatto in termini di efficienza operativa, qualità del prodotto e coinvolgimento degli stakeholder. L'obiettivo è quello di comprendere come l'approccio Agile, opportunamente adattato alle peculiarità del settore assicurativo, possa rappresentare un fattore abilitante per la trasformazione digitale e la creazione di valore duraturo.

Dal punto di vista metodologico, la tesi combina una parte teorica, volta a descrivere le principali metodologie di project management e i principi fondanti dell'Agile, con una parte applicativa, che analizza il caso aziendale oggetto del tirocinio, mettendo in luce i KPI utilizzati per misurare le performance e l'efficacia del modello Agile implementato. Nel dettaglio, nel capitolo 2 vengono illustrate le principali metodologie di project management, analizzando in particolare le differenze tra i modelli Waterfall e Agile. Il Capitolo 3 approfondisce la metodologia Agile applicata al contesto assicurativo, descrivendo le peculiarità del settore, il contesto aziendale di riferimento e il progetto oggetto dello studio, nonché gli strumenti adottati. Nel Capitolo 4 viene svolta un'analisi quantitativa e qualitativa dei risultati ottenuti, valutando l'efficacia dell'approccio Agile attraverso i principali indicatori di performance (KPI) e discutendo i vantaggi emersi e le raccomandazioni per una sua adozione più efficace. Infine, nel Capitolo 5, vengono presentate le conclusioni, che sintetizzano i risultati dell'analisi.

In sintesi, questa tesi intende dimostrare come l'introduzione di un approccio Agile in un contesto complesso e regolamentato come quello assicurativo, possa rappresentare un passo decisivo verso una gestione più efficiente, collaborativa e orientata al valore dei progetti IT, contribuendo alla costruzione di organizzazioni più flessibili e resilienti.

## **2 Metodologie di project management nei progetti**

### **2.1 La metodologia Waterfall**

Nel vasto e dinamico panorama del project management, ogni metodologia offre una propria prospettiva e un insieme di valori che guidano il suo utilizzo. Soddisfare le aspettative degli stakeholder ed ottenere risultati concreti è il frutto di un buon lavoro avendo adottato un approccio strategico.

Waterfall e Agile sono le metodologie di project management maggiormente diffuse, entrambe valide nel gestire i progetti ma basate su principi profondamente diversi.

La metodologia Waterfall, introdotta all'interno dell'articolo "Managing the Development of Large Software Systems" dal Dr. Winston W. Royce nel 1970, rappresenta una delle più antiche e consolidate tecniche per lo sviluppo sequenziale dei progetti. Essa è particolarmente indicata in contesti che richiedono un ordine rigoroso: un esempio tipico è quello dell'edilizia, dove non è possibile costruire le pareti prima di completare le fondamenta. Infatti, originariamente questa metodologia trovava applicazione nel settore edile e manifatturiero, in secondo luogo è stata adattata anche ad altri ambiti. Un esempio è lo sviluppo software, dove però può risultare limitante per via della mancanza di flessibilità.

Il metodo Waterfall, noto anche come "modello a cascata", si distingue per la struttura sequenziale delle sue attività, che fluiscono progressivamente da una fase all'altra. Ogni fase deve essere completata prima di passare a quella successiva, poiché l'output di una diventa l'input della seguente, questa caratteristica infatti rende il sistema molto rigido.

L'importanza di una pianificazione dettagliata, una documentazione meticolosa, requisiti definiti inizialmente e rigidità verso cambiamenti significativi sono criteri ritenuti necessari per il raggiungimento del successo del progetto. Questo permette di creare una tempistica dettagliata e ottimizzare la gestione delle risorse. [1] [2]

#### **2.1.1 Le fasi**

Il modello a cascata tradizionale, come illustrato in figura 1, prevede le seguenti fasi:

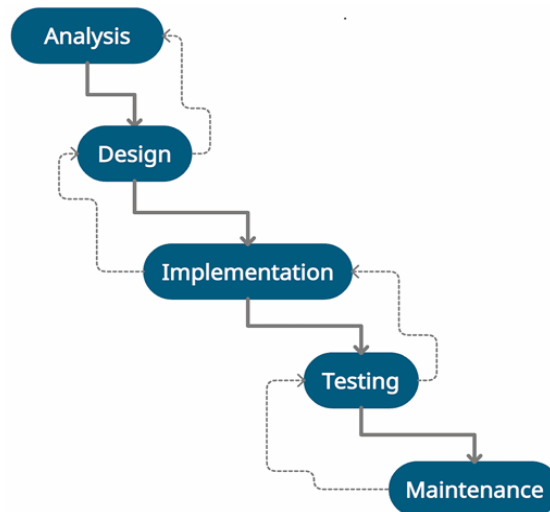


Figura 1: Le fasi della metodologia Waterfall

Analizzando nel dettaglio ciascuna fase:

1. **Analisi dei requisiti:** vi è il processo iniziale di pianificazione, in cui il team raccoglie quante più informazioni possibili. In questa fase è necessario delineare il progetto dall'inizio alla fine, si crea quindi un piano del progetto dettagliato, chiamato documento dei requisiti di progetto, in cui si includono le seguenti informazioni:
  - ogni fase del processo
  - quali saranno gli aspetti su cui lavorerà ogni membro del team e in quale fase
  - le dipendenze fondamentali
  - le risorse necessarie
  - una sequenza temporale per l'intero progetto, che delinea la durata di ogni fase[3] [4]

2. **Progettazione del sistema (o design):** questa fase è fondamentale per capire come sarà realizzato il sistema e quale struttura dovrà assumere. L'obiettivo è quello di tradurre i requisiti raccolti nella fase precedente in un piano tecnico e operativo.

La progettazione si sviluppa solitamente su due livelli distinti:

- Progettazione ad alto livello (high-level design): definisce la visione complessiva del sistema.
- Progettazione di dettaglio: specifica gli aspetti tecnici e implementativi.



Nel dettaglio, la fase di design comprende:

- Definizione dell'architettura del sistema: individuazione dei principali componenti, dei flussi informativi e delle modalità di interazione tra le varie parti del sistema.
- Sviluppo di prototipi: realizzazione di versioni preliminari del sistema per mostrare funzionalità e interfacce, utili per validare le scelte progettuali.
- Scelte tecnologiche: identificazione dei linguaggi di programmazione, dei database, degli strumenti di sviluppo e delle piattaforme da adottare.
- Pianificazione delle attività di test: elaborazione di una strategia di collaudo per garantire che ogni parte del sistema funzioni correttamente.

Ad esempio, in un progetto software, questa fase potrebbe includere la progettazione dell'interfaccia utente tramite wireframe, che rappresentano ogni schermata dell'applicazione, e la strutturazione del database per gestire in modo efficiente i dati degli utenti.

3. **Implementazione (o sviluppo):** In questa fase la realizzazione del prodotto o servizio prende forma concretamente, trasformando le idee e le specifiche progettuali in qualcosa di tangibile. Il team lavora con attenzione, rimanendo coerente con il documento dei requisiti stilato nella prima fase e con il processo di progettazione del sistema della seconda. In questa fase, infatti, è essenziale rispettare rigorosamente i requisiti e le specifiche definiti in precedenza.

Le attività principali includono:

- Implementazione del codice: sviluppo dei moduli e delle funzionalità secondo le specifiche tecniche definite nella fase di design.
- Integrazione dei componenti: unione dei diversi elementi sviluppati per formare un sistema completo, stabile e coerente.
- Monitoraggio del progresso: verifica continua dello stato di avanzamento dei lavori, per assicurare il rispetto delle tempistiche e degli obiettivi.
- Creazione della documentazione: produzione di manuali tecnici, guide utente e documenti di supporto per facilitare l'utilizzo, la gestione e la manutenzione del sistema. [5]

4. **Verifica e controllo qualità (fase di test):** l'obiettivo è prevenire il rilascio di un prodotto imperfetto e garantire un'esperienza affidabile. Il progetto passa dalle mani del developer team a quelle del team dedicato alla qualità, il cui compito è assicurarsi che il sistema funzioni correttamente in ogni sua parte.

È importante condurre test per tutti i casi d'uso per evitare di rilasciare un prodotto difettoso ai clienti e garantire un'esperienza utente positiva.

Di seguito sono elencate le attività principali del processo di testing:

- Creazione dei test case: vengono elaborati scenari specifici per simulare diverse situazioni d'uso e analizzare il comportamento del sistema.
- Registrazione di errori e malfunzionamenti: ogni anomalia individuata viene documentata con precisione, facilitando la risoluzione e contribuendo alla costruzione di una base di conoscenza utile per il futuro.
- Test modulari e mirati: le singole funzionalità vengono esaminate in modo isolato per garantire che ogni componente risponda correttamente alle specifiche.
- Monitoraggio degli indicatori di qualità: vengono definiti e analizzati parametri come la percentuale di errori, la copertura dei test e i tempi di correzione, per valutare l'efficienza complessiva del sistema. [4]

5. **Distribuzione:** il prodotto viene consegnato agli utenti finali e reso disponibile per l'utilizzo reale. La distribuzione in maniera efficace, affidabile e senza ulteriori disagi verso il cliente è l'obiettivo di questa fase.

Il lavoro del team viene valorizzato da una buona ricezione di feedback da parte del cliente che riceve un'esperienza appagante dovuta ad una distribuzione pianificata.

[1]

6. **Manutenzione:** una volta completato e lanciato il progetto, esso entra in una fase di manutenzione costante, in modo da mantenerne alte prestazioni e sicurezza nel tempo.

Tra le tipologie di manutenzione si distingue:

- Correttiva: gli strumenti di analisi rilevano malfunzionamenti o errori segnalati dagli utenti, al fine di intervenire e risolverli.

- **Adattativa:** aggiornamento di sistemi operativi, nuovi dispositivi o modifiche normative. Il sistema è costantemente aggiornato per rispondere in maniera corretta.
- **Evolutiva:** si introducono nuove funzionalità o si migliorano quelle esistenti per rispondere alle richieste del mercato o per offrire un'esperienza utente più ricca.
- **Preventiva:** interventi per prevenire futuri problemi.

Ad esempio, per un'app mobile, la manutenzione potrebbe includere il rilascio di aggiornamenti per correggere errori segnalati dagli utenti, migliorare la sicurezza e l'introduzione di nuove feature come l'autenticazione biometrica.

Nel settore dello sviluppo software è comune lavorare continuamente su questa fase.

[5]

## 2.1.2 Applicazioni

La caratteristica del modello Waterfall è che si basa su un approccio lineare, in cui ogni fase del progetto deve essere completata prima di passare alla successiva. Questo può rendere il lavoro poco flessibile in contesti in cui le esigenze evolvono in maniera rapida. Ne consegue che la metodologia Waterfall non è adatta a tutti i tipi di progetto.

Di seguito sono elencati alcuni esempi in cui è preferibile usare questo metodo:

- **Progetti con obiettivi chiari e stabili:** Quando il risultato finale è definito fin dall'inizio e non è soggetto a modifiche. Ogni fase si sviluppa partendo da requisiti chiari, permettendo un percorso ben tracciato fino al prodotto finito. Il modello Waterfall consente di costruire il progetto in modo ordinato e prevedibile.
- **Assenza di vincoli stringenti su tempo e costi:** In situazioni in cui il progetto non è limitato da scadenze rigide o budget ristretti, si può dedicare più tempo alla pianificazione e all'analisi iniziale. Inoltre, i membri del team possono modificare e adattare le necessità del progetto quanto vogliono, finché non ottengono un piano di progetto preciso e definito.

- **Esigenza di processi documentati e ripetibili:** Il modello waterfall richiede l'elaborazione di documenti durante quasi tutte le fasi del processo. In questo modo, un nuovo membro del team potrà ripetere facilmente il progetto; ogni passaggio è spiegato nei minimi dettagli, in modo che il processo sia ripetibile. Creare processi ripetibili semplifica la formazione dei nuovi membri del team e spiegare loro cosa devono fare esattamente in progetti simili. Ciò rende il processo waterfall un approccio efficace alla gestione dei progetti nell'ambito dei processi di standardizzazione. [3]
- **Progetti con requisiti stabili e contesto prevedibile:** Il modello Waterfall è meno indicato in contesti dinamici, dove è necessario adattarsi rapidamente alle nuove esigenze, ma è adatto a progetti con requisiti chiari, pochi cambiamenti attesi e un forte bisogno di controllo e documentazione. Mantenere il controllo sul processo ed evitare continue revisioni è possibile solo se l'ambiente di progetto è stabile e le richieste dei clienti non cambiano frequentemente. In questo caso, la rigidità del modello rappresenta un vantaggio. [4]

### 2.1.3 Punti di forza e limiti

L'implementazione della metodologia tradizionale offre una serie di vantaggi:

- **Struttura chiara del progetto:** L'approvazione dei requisiti sin dall'inizio, permette all'intero processo di avere un traguardo finale chiaramente identificato e condiviso, con un sistema progettuale sistematico e prevedibile, motivo per il quale l'incertezza è ridotta al minimo. [1]
- **Precisione nella programmazione di tempi e costi:** grazie ad una pianificazione rigorosa, effettuata prima dell'inizio effettivo del lavoro, tutte le attività, le risorse e i tempi sono definiti fin da subito, consentendo un controllo più efficace durante l'intero ciclo di vita del progetto e riducendo il rischio di ritardi e superamento dei budget.

- **Massima trasparenza:** ogni fase è accompagnata da una documentazione precisa, che facilita la comunicazione tra i membri del team tecnico e gli stakeholder senza competenze specifiche. La documentazione costituisce una risorsa fondamentale per progetti futuri analoghi, nonché per interventi di manutenzione e assistenza.
- **Facilità di gestione:** la linearità e la struttura chiara della metodologia Waterfall rendono il progetto facilmente gestibile sia per i team tecnici che per i project manager. Il monitoraggio dei progressi è semplice, poiché ogni fase ha una conclusione ben definita. [5]

Tra gli aspetti critici da considerare nell'utilizzo del modello Waterfall:

- **Tempi di consegna estesi:** La natura sequenziale del modello Waterfall può rallentare significativamente il processo complessivo, in quanto prevede il completamento di ogni fase prima di procedere alla successiva, con un impatto diretto sulla velocità di consegna del prodotto finale. [4]
- **Limitata interazione con il cliente:** Eventuali cambiamenti nelle esigenze del cliente durante il progetto difficilmente vengono intercettati e integrati in tempo utile. Il rischio di non soddisfare pienamente le aspettative aumenta in quanto i momenti di confronto con il cliente sono generalmente previsti solo all'inizio per la definizione dei requisiti e alla fine per la consegna. [1]
- **Bassa flessibilità:** uno dei principali svantaggi del modello a cascata è la scarsa capacità di adattarsi durante tutte le fasi del progetto, rendendo il processo troppo vincolato al piano originale e meno reattivo alle eventuali modifiche o nuove esigenze. Infatti, l'approccio tradizionale alla gestione dei progetti si basa principalmente su relazioni gerarchiche e lineari tra le attività e non riesce a riflettere adeguatamente tutta la complessità e le dinamiche dei progetti odierni, i quali sono sempre più complessi, con un numero maggiore di attività e interrelazioni complesse. [6]
- **Testing tardivo:** Il ritardo nel rilevamento degli errori può compromettere la qualità complessiva del prodotto. Le attività di testing sono concentrate nella fase finale del progetto e di conseguenza, eventuali errori o problemi progettuali

possono emergere troppo tardi, quando correggerli comporta costi elevati o richiede di rivedere intere sezioni già completate.

## **2.2 La metodologia Agile**

Le obiezioni all'approccio tradizionale di gestione dei progetti, unite alla domanda sempre più pressante di innovazione costante in ogni ambito, hanno favorito l'ascesa di un nuovo approccio di project management, denominato Agile.

Esso rappresenta un cambiamento significativo rispetto al modello Waterfall, infatti si distingue per la sua capacità di adattarsi ai cambiamenti e di conseguenza alle modifiche che si verificano durante l'intero svolgimento del progetto, oltre che per la sua applicabilità a progetti di diversa natura. Agile è stato ideato per gestire la complessità progettuale mediante iterazioni successive e miglioramenti progressivi, a differenza dell'approccio lineare del Waterfall. La consapevolezza che il mutamento è una costante è il principio fondante di questa metodologia, che rende quindi poco realistico pianificare in modo esaustivo sin dalle fasi iniziali del progetto. [6]

La cooperazione continua tra i membri del team, la capacità di adattamento e la prontezza nel reagire ai cambiamenti sono ideali di agile, che trova la sua massima efficacia in ambienti contraddistinti da forte instabilità, elevato contenuto tecnologico, scadenze ravvicinate e dinamicità.

### **2.2.1 Sviluppo della metodologia Agile**

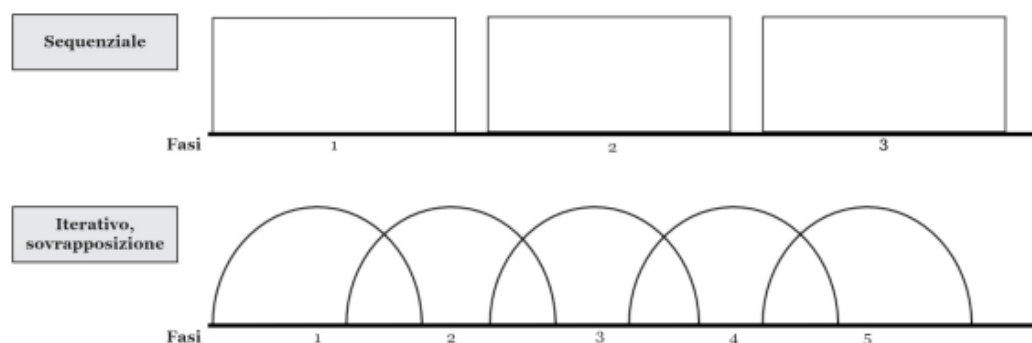
Negli anni Settanta lo sviluppo del software seguiva modelli di lavoro tradizionali, che si basavano su una pianificazione attenta e su una documentazione precisa. Questa idea, proposta da Winston W. Royce, richiedeva di pianificare tutto il progetto in anticipo, ma limitava la capacità di adattarsi ai cambiamenti.

Con il passare del tempo, tuttavia, il mondo industriale ha subito cambiamenti: già verso la fine degli anni '80 case automobilistiche di grande spessore come Toyota, avevano bisogno di un modello di produzione all'avanguardia, che potesse competere con i grandi colossi del mercato automobilistico americano ed in generale rispondere alle richieste del mercato che chiedevano aggiornamenti veloci e frequenti dei prodotti.

L'articolo "The New Product Development Game", pubblicato nel 1986 sulla Harvard Business Review [7] da Hirotaka Takeuchi e Ikujiro Nonaka, presenta un nuovo modo di gestire lo sviluppo dei prodotti. In particolare, l'articolo mostra come alcune aziende giapponesi, tra cui Toyota, Honda e Canon, stessero già rompendo gli schemi tradizionali dello sviluppo: invece di seguire processi rigidi e sequenziali, queste realtà avevano iniziato a puntare su squadre multidisciplinari e autonome, capaci di auto-organizzarsi, prendere decisioni rapide e lavorare in parallelo su più aspetti del progetto. Per descrivere questo approccio, Takeuchi e Nonaka usarono una metafora: lo sviluppo di nuovi prodotti viene affrontato in modo simile a una partita di rugby, in cui non si procede per passaggi rigidi ed ordinati, ma la "palla" (che rappresenta il progetto) viene fatta avanzare grazie al movimento coeso e sincronizzato di tutta la squadra. L'elemento chiave diventa la sovrapposizione delle fasi, che permette di accelerare i tempi e migliorare la capacità di adattamento.

Al contrario, l'approccio tradizionale sequenziale può essere paragonato ad una staffetta, in cui l'individuo passava il testimone solo dopo aver completato la distanza di competenza. Per questo motivo tale metodo potrebbe non risultare strategico in termini di velocità, adattamento e flessibilità.

In figura 2 sono rappresentati entrambi gli approcci, in alto quello sequenziale, in basso l'approccio iterativo, in cui è evidente il concetto di sovrapposizione.



*Figura 2: Confronto tra approccio sequenziale e iterativo*

Si tratta di un cambio di mentalità: meno documenti e piani rigidi, più collaborazione, sperimentazione e responsabilità condivisa.

Da quell'idea di lavoro in gruppo, flessibile e creativo, prese forma un modo diverso di intendere l'innovazione. Ed è proprio su quelle basi che, qualche anno dopo, nacque il

Manifesto Agile: un documento scritto che metteva al centro le persone, il lavoro di squadra e la capacità di rispondere al cambiamento.

## 2.2.2 Il manifesto Agile

La metodologia Agile nasce nel 2001, quando un gruppo di esperti di sviluppo software si sono incontrati in una località sciistica nello Utah, per confrontarsi su come rendere più efficiente la gestione dei progetti.

Durante tale incontro venne redatto il Manifesto Agile, un documento che ha rivoluzionato il modo di approcciare allo sviluppo software, introducendo così una prospettiva incentrata sulle persone, sulla qualità del prodotto finale, sulla cooperazione tra i team e sull'adattabilità ai cambiamenti.

Esso non è solo un manuale operativo, ma una raccolta di valori e linee guida concepite per favorire capacità di adattamento, maggiore flessibilità e cooperazione nel lavoro quotidiano. L'obiettivo consisteva nel superare le barriere imposte dalle metodologie tradizionali, che erano fortemente caratterizzate da rigidità e da un approccio burocratico, come il modello a cascata, che nel tempo aveva finito per ostacolare l'innovazione. [8]

In figura 3 sono mostrati i quattro principi guida concordati dai 17 sviluppatori del Manifesto Agile:



Figura 3: Il manifesto Agile



Analizzandoli nel dettaglio:

1. La prima affermazione propone una **visione centrata sulle persone e sulle dinamiche relazionali**, superando la tradizionale enfasi su strumenti e procedure. Questo orientamento valorizza la capacità dei membri del team di adattarsi, collaborare e prendere decisioni in autonomia, promuovendo una leadership meno rigida e più attenta alle esigenze individuali. L'obiettivo non è eliminare del tutto i processi e gli strumenti, ma riconoscerli come supporti e non come vincoli. In questo modo, il progetto assume un significato più profondo per chi vi partecipa, favorendo un ambiente di lavoro più umano, flessibile e orientato alla co-creazione.

2. Il secondo valore del Manifesto privilegia la **realizzazione concreta del prodotto rispetto alla produzione di documentazione esaustiva**. Questa prospettiva nasce come reazione ai modelli tradizionali, nei quali ogni fase richiede una mole significativa di documenti formali. L'intento non è quello di eliminare la documentazione, ma di evitare che assuma una centralità sproporzionata e spostando l'attenzione sull'effettiva funzionalità del software o del prodotto. In ambienti complessi e in continua evoluzione, la difficoltà nel definire requisiti dettagliati fin dalle prime fasi rende inefficace un approccio basato sulla documentazione. Per questo motivo, l'approccio Agile incoraggia la creazione di soluzioni operative e testabili, che permettano di adattarsi alle esigenze reali degli utenti, riducendo il rischio di errori derivanti da un'eccessiva dipendenza dalla documentazione.

3. Il terzo valore dell'approccio Agile si concentra sulla **valorizzazione della collaborazione con il cliente rispetto alla negoziazione contrattuale**. Questo orientamento nasce dalla consapevolezza che, in contesti progettuali dinamici e soggetti a frequenti cambiamenti, è più efficace costruire un rapporto di fiducia e comunicazione continua con gli stakeholder piuttosto che affidarsi esclusivamente a clausole contrattuali rigide. Sebbene la gestione formale delle modifiche e dei vincoli di budget rimanga importante, l'interazione diretta con il cliente consente di identificare tempestivamente obiettivi condivisi e di adattare il lavoro alle esigenze emergenti. In

questo modo, il progetto non solo risponde meglio alle aspettative, ma favorisce anche una maggiore coerenza tra le soluzioni sviluppate e i bisogni reali dell'utente finale.

4. Il quarto ed ultimo principio guida del manifesto Agile ribadisce l'**accoglienza dei cambiamenti**, anche nelle fasi avanzate del progetto. Il cambiamento non rappresenta un ostacolo, bensì uno strumento che, se gestito in modo collaborativo e trasparente, può rafforzare il valore del progetto. E' quindi essenziale stabilire sin dall'inizio una comprensione condivisa su come tali variazioni verranno affrontate, così da mantenere equilibrio tra flessibilità e controllo. [9]

### 2.2.3 I principi dell'Agile

Nei mesi successivi, gli autori hanno esteso le idee del Manifesto Agile con i 12 principi riportati in figura 4, che ampliano i concetti espressi dalle poche frasi che costituiscono i valori. [10]

<b>1</b>	La nostra massima priorità è quella di soddisfare il cliente attraverso la consegna tempestiva e continua di software di valore.	<b>7</b>	Il software funzionante è la misura principale del progresso.
<b>2</b>	Accogli con favore i requisiti in evoluzione, anche in fase avanzata di sviluppo. I processi agili sfruttano il cambiamento per ottenere un vantaggio competitivo del cliente.	<b>8</b>	I processi agili promuovono lo sviluppo sostenibile. Gli sponsor, gli sviluppatori e gli utenti dovrebbero essere in grado di mantenere un ritmo costante a tempo indeterminato.
<b>3</b>	Consegna il software funzionante frequentemente, da un paio di settimane a un paio di mesi, con una preferenza per i tempi più brevi.	<b>9</b>	La continua attenzione all'eccellenza tecnica e al buon design migliora l'agilità.
<b>4</b>	Gli uomini d'affari e gli sviluppatori devono lavorare insieme quotidianamente durante tutto il progetto.	<b>10</b>	La semplicità, l'arte di massimizzare la quantità di lavoro non svolto, è essenziale.
<b>5</b>	Costruisci progetti attorno a persone motivate. Offri loro l'ambiente e il supporto di cui hanno bisogno e fidati di loro per portare a termine il lavoro.	<b>11</b>	Le migliori architetture, requisiti e progetti emergono da team auto-organizzati.
<b>6</b>	Il metodo più efficiente ed efficace per trasmettere informazioni a e all'interno di un team di sviluppo è la	<b>12</b>	A intervalli regolari, il team riflette su come diventare più efficace, quindi sintonizza e regola il proprio comportamento

Figura 4: I principi dell'Agile

## 2.2.4 Applicazioni

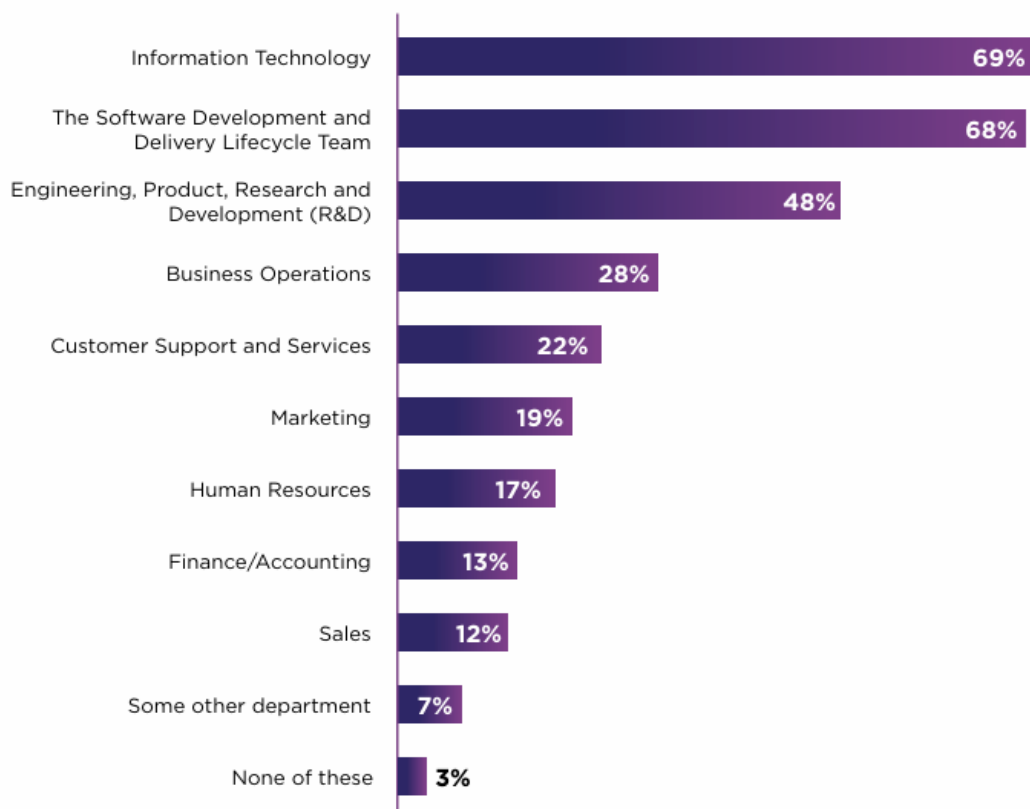
Nell'aprile 2023 è stato pubblicato il diciassettesimo rapporto annuale sullo stato della metodologia Agile. Esso ha offerto una visione aggiornata e dettagliata sull'integrazione delle pratiche Agile all'interno delle aziende, infatti è considerato tra le indagini più rilevanti del settore.

All'indagine hanno preso parte 788 professionisti, di cui il 31% operava in organizzazioni con oltre 20.000 dipendenti e il 29% risultava impiegato in realtà con meno di 1.000 collaboratori. Questi dati hanno evidenziato che l'adozione dell'Agile tende a concentrarsi soprattutto nelle imprese di dimensioni molto contenute o particolarmente estese.

Inoltre, circa il 71% delle aziende ha affermato di averlo integrato nel proprio processo di sviluppo software.

Il questionario condotto ha permesso di analizzare anche le aree di applicazione dell'Agile. Come illustrato in figura 5, le aree IT e i team di sviluppo risultano essere i principali utilizzatori (quasi il 70%), il che non è sorprendente considerando che l'Agile è nato all'interno dello sviluppo software e, di conseguenza, è questo il settore in cui se ne fa un maggiore utilizzo.

Tuttavia, l'adozione dell'Agile sta progressivamente interessando anche ambiti differenti, estendendosi ben oltre il contesto originario dello sviluppo software: dai team di ingegneria, prodotto e R&D (48%), alle operazioni aziendali (circa il 28%). Anche i team di marketing (19%) e delle risorse umane (17%) mostrano un incremento verso l'adozione di questo approccio.



*Figura 5: Le aree di implementazione dell'approccio Agile*

Le motivazioni che spingono le organizzazioni verso l'adozione dell'Agile sono strettamente legate all'aumento del valore aziendale. Il 41% dei partecipanti ha indicato come priorità la misurazione del valore per i clienti/valore aziendale e la prioritizzazione della consegna, mentre un ulteriore 41% ha posto l'accento sulla necessità di accelerare il time to market. Tra gli altri elementi strategici segnalati figurano la trasformazione digitale (34%) e la capacità di prevedere con maggiore precisione le tempistiche di rilascio (30%). I risultati del sondaggio sono mostrati in figura 6.



Figura 6: I driver chiave dell'approccio Agile

La soddisfazione del cliente è emersa per il 43% degli intervistati come la priorità principale per quanto riguarda gli obiettivi operativi. Il 39% ha attribuito importanza alla riduzione dei tempi di consegna (time to delivery), il 34% ha puntato sul rafforzamento del vantaggio competitivo. L'aumento dei ricavi dei prodotti, l'espansione del mercato e l'acquisizione di nuovi clienti sono alcune delle altre priorità, secondo quanto dichiarato da oltre un quarto degli intervistati (25%).

Tali risultati, mostrati in figura 7, confermano quanto sia fondamentale nel contesto odierno produrre un valore che sia tangibile per il cliente, evidenziando così una chiara tendenza verso modelli orientati alla qualità e alla reattività. [11]

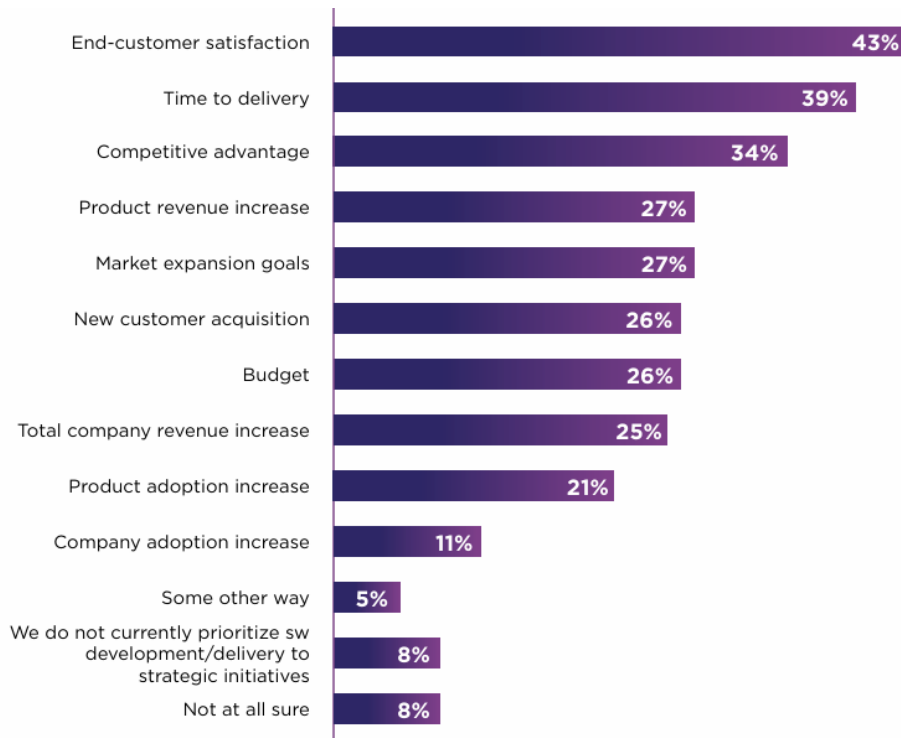


Figura 7: Gli obiettivi operativi dell'approccio Agile

## 2.2.5 Il framework Scrum

Un framework Agile è un approccio specifico alla pianificazione, alla gestione e all'esecuzione del lavoro. All'interno dell'universo delle metodologie Agile, la letteratura ha identificato diversi framework tra cui figurano Scrum, Kanban, Extreme Programming, Crystal, Feature Driven Development, Dynamic Systems Development Method, Adaptive Software Development e Agile Modeling. Ciascuno di questi modelli offre strumenti utili e funzionali in determinati contesti, pur non assicurando sempre una copertura esaustiva dell'intero processo di gestione dei progetti.

Tra questi, Scrum ha assunto nel tempo un ruolo di primo piano, affermandosi come il framework Agile più adottato a livello globale. Diversi studi e report, tra cui il "State of Agile Report" [11], confermano il suo primato, rilevando che circa il 63% degli intervistati utilizza Scrum, un dato che evidenzia la sua posizione di forte stabilità e continuità nel tempo, infatti questa metodologia fin dal 2006 ha mantenuto una presenza costante e significativa. Questa diffusione rende Scrum il punto di riferimento principale per l'analisi che segue.

Esso si presenta come un'implementazione specifica dei principi Agili, offrendo un modello strutturato sia dal punto di vista organizzativo che operativo per affrontare progetti complessi all'interno di intervalli temporali definiti (time-boxed). Si distingue, infatti, per la sua organizzazione in cicli di lavoro noti come sprint, che generalmente hanno una durata compresa tra due e quattro settimane ciascuno. Durante ogni sprint, il team Scrum si dedica al completamento di un insieme specifico di attività, favorendo così la coerenza operativa e permettendo iterazioni brevi e frequenti. Questo approccio consente di adattare sia le modalità di esecuzione che gli obiettivi del lavoro. Al termine di ogni sprint, ne viene avviato immediatamente uno nuovo, garantendo continuità e ritmo costante nel processo di sviluppo. [12]

Scrum adotta una logica di gestione che favorisce una distribuzione progressiva dei risultati e una pianificazione flessibile rispecchiando pienamente i principi fondanti dell'Agile, esso infatti consente ai team di collaborare con maggiore efficacia, ed affrontare nella migliore maniera le sfide complesse generando valore attraverso rilasci gradualmente. Un grande vantaggio è la capacità di rispondere ai cambiamenti, migliorando la soddisfazione del cliente e il successo del progetto. [13]

Iterazione dopo iterazione, Scrum mira a perfezionare progressivamente l'idea iniziale del cliente, basandosi su ciò che viene rilasciato e sul relativo feedback. Per raggiungere tale obiettivo, i gruppi di lavoro analizzano regolarmente le modalità operative e le procedure al fine di intervenire prontamente di fronte a imprevisti o modifiche che si presentano in tempo reale. Questo approccio si contrappone al modello classico di project management, in cui si attende la conclusione di una fase pianificata per esaminare poi i risultati (momento difficile per apportare eventuali correzioni).

Nella "Guida a Scrum - La guida definitiva a Scrum: Le regole del gioco", redatta nel 2010 e aggiornata nel Novembre 2020, Ken Schwaber e Jeff Sutherland, entrambi autori dell'Agile Manifesto e ideatori di Scrum, propongono una definizione chiara di tale framework. [14]

Scrum viene descritto come "un quadro all'interno del quale le persone possono affrontare problemi adattivi complessi, fornendo al contempo prodotti del più alto valore possibile in modo produttivo e creativo".

Per definizione, Scrum è pensato per essere snello ed intuitivo. Tuttavia, è difficile da padroneggiare:

- È leggero, in quanto i team Scrum lavorano per eliminare gli sprechi pianificando il lavoro con solo poche settimane di anticipo. Ciò consente una maggiore adattabilità rispetto ad altri metodi, garantendo inoltre flessibilità.
- È semplice da capire, in quanto si basa su alcuni principi molto basilari:
  - Sviluppa in modo iterativo
  - Ottimizza la prevedibilità
  - Controlla il rischio
  - Pratica il controllo dei processi attraverso la trasparenza (il personale deve conoscere il lavoro e il progresso), l'ispezione (è necessario verificare l'avanzamento dei lavori) e l'adattamento (effettuare modifiche qualora ce ne fosse bisogno in base alla risposta dei clienti).
- È difficile da padroneggiare, perché i valori Scrum di impegno, coraggio, concentrazione, apertura e rispetto richiedono che gli individui e i team si attengano a uno standard di condotta elevato e disciplinato. [15]

Tali valori, presenti all'interno della Guida Scrum, rappresentano la base su cui si fonda il successo nell'impiego di Scrum, poiché forniscono indicazioni sul lavoro, sulle azioni e sul comportamento del team.

- **Impegno:** Poiché i team Scrum sono piccoli e agili, ogni membro svolge un ruolo significativo nel successo del team. Quest'ultimo si impegna nei confronti dell'organizzazione durante ogni riunione di pianificazione dello sprint, al termine della quale ogni membro dovrebbe avere lo stesso livello di comprensione di ciò che il team si impegna a realizzare durante quello sprint. Dovrebbe esserci una comunicazione frequente riguardo all'avanzamento del lavoro, spesso in stand-up.
- **Coraggio:** consiste nella capacità di affrontare le criticità nonostante le proprie paure. I membri del team Scrum dovrebbero sentirsi abbastanza sicuri di essere trasparenti su ostacoli, avanzamento del progetto, ritardi e così via.
- **Concentrazione:** Focus significa fare tutto il necessario per consentire al team di concentrarsi sulla consegna in corso per l'intera durata di un dato sprint. Al centro del flusso di lavoro per i team Scrum c'è lo sprint, che fornisce



una struttura ma anche la messa a fuoco per completare la quantità di lavoro pianificata.

- **Apertura:** Lo stand-up quotidiano favorisce un'apertura che consente ai team di parlare apertamente dei lavori in corso e degli ostacoli. Questo aiuta a evidenziare i progressi e identificare le criticità. Aiuta anche a rafforzare il team quando tutti condividono i progressi.
- **Rispetto:** I team Scrum ad alte prestazioni si fidano abbastanza l'uno dell'altro da ammettere gli ostacoli. Il rispetto per i propri compagni di squadra, o la sua mancanza, può decretare il successo o il fallimento di un progetto. La collaborazione e il riconoscimento delle potenzialità di ogni membro del team contribuiscono al successo di ogni sprint, in questo infatti risiede la forza di un team agile. [16]

Quando questi valori sono incarnati dallo Scrum Team e dalle persone che lavorano con loro, i pilastri empirici di Scrum (trasparenza, ispezione e adattamento) prendono vita, creando fiducia.

Nonostante il modello Scrum si presenti come intuitivo e lineare, la sua efficacia può risultare compromessa se non viene adottato correttamente. Per questo motivo, la metodologia Scrum si struttura attraverso specifici ruoli, eventi e artefatti, come illustrato nei paragrafi seguenti.

### 2.2.5.1 Scrum Team

Un team Scrum è un gruppo di individui che lavorano insieme per fornire gli incrementi di prodotto richiesti. Il framework Scrum promuove un alto livello di comunicazione tra i membri del team, in modo che possa:

- Inseguire uno scopo condiviso
- Rispettare le stesse norme e regole
- Mostrare rispetto reciproco

I team Scrum sono interfunzionali, poiché possiedono tutte le competenze necessarie per svolgere il lavoro associato al progetto senza dipendere da persone esterne al team. Sono anche auto-organizzati, poiché scelgono il modo migliore per eseguire il loro lavoro,

anzichè ricevere indicazioni da qualcuno al di fuori del team. Tutti i membri del team si supportano a vicenda per garantire il successo dello sprint. [15]

Scrum definisce chiaramente 3 ruoli all'interno di un team. Come è evidente in figura 8, vi sono Product Owner, Scrum Master e Developers.

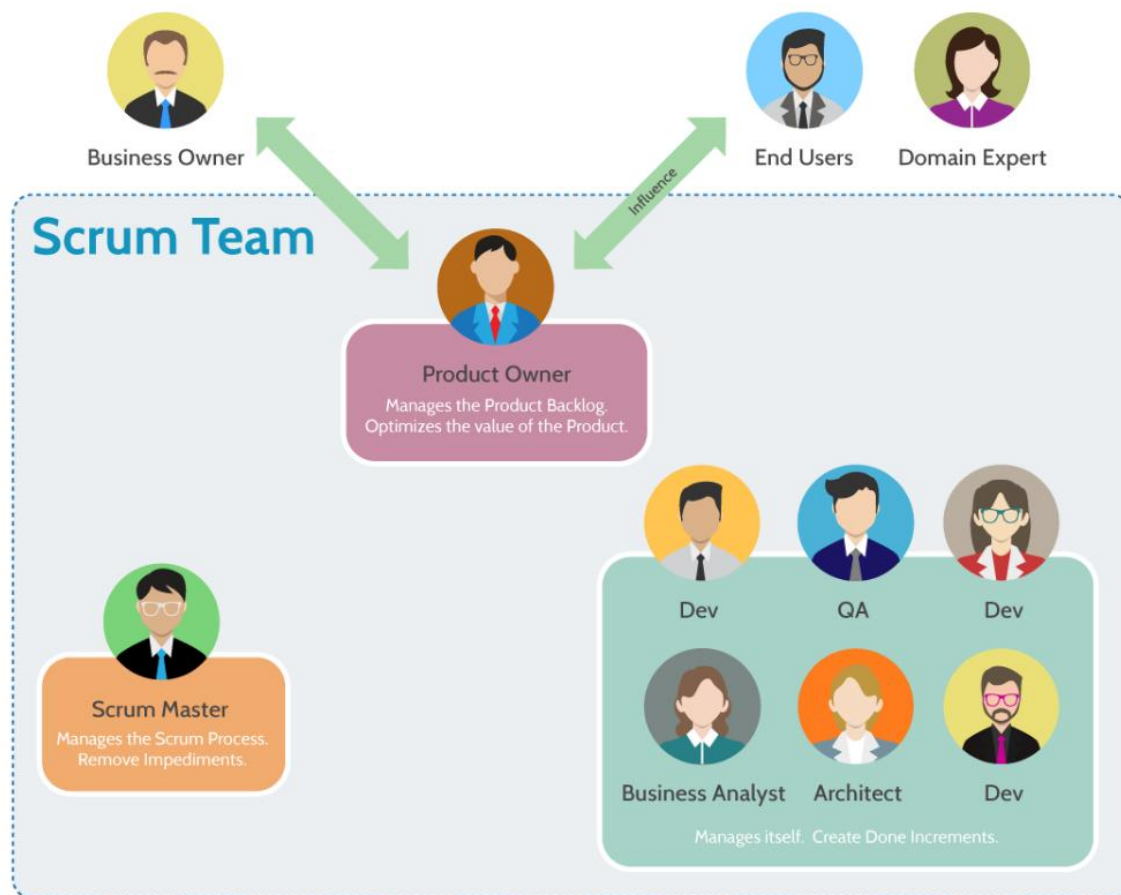


Figura 8: Scrum Team

- **Product owner:** Si concentra sulla comprensione delle esigenze aziendali, dei clienti e del mercato. Nel dettaglio si occupa di:
  - Creare e gestire il Product Backlog (elenco accurato e dettagliato di tutti i requisiti che devono essere sviluppati): deve essere a conoscenza di tutto ciò che è nel backlog e le altre persone che aggiungono elementi al backlog del prodotto devono assicurarsi di comunicare con il PO. E' responsabile dell'aggiornamento del medesimo, che permette quindi l'aderenza a nuove richieste e cambiamenti.
  - Garantire che il backlog del prodotto sia visibile, trasparente e chiaro a tutti.

- Fornire al team indicazioni precise su quali funzionalità fornire successivamente.
- Decidere quando rilasciare il prodotto con una predisposizione verso consegne più frequenti.
- Assicurare che il team di sviluppo offra il massimo valore all'azienda.

E' l'unico a mantenere contatti all'esterno dello Scrum Team: fa da ponte con Business Owner, utenti finali e domain expert: riceve esigenze e feedback, li traduce in requisiti e priorità. [14]

- **Scrum Master** agisce da facilitatore, garantendo che il team segua i principi e le pratiche Scrum, rimuovendo eventuali ostacoli e favorendo la comunicazione. E' un vero leader al servizio dello Scrum Team e dell'organizzazione in generale. Nel dettaglio, gli scrum master aiutano i product owner a comprendere e comunicare meglio il valore, a gestire il backlog, a pianificare il lavoro con il team. Al servizio del team di sviluppo, lo scrum master li supporta nel concentrarsi sui risultati nella gestione dei blocchi. Lo Scrum Master serve anche l'organizzazione in generale, aiutandola a capire cos'è Scrum e a creare un ambiente che sostenga Scrum.

Lo scrum master si concentra su:

- Trasparenza: Affinchè il team possa agire adattando ed ispezionando il lavoro in modo efficace, è di fondamentale importanza che le informazioni siano accessibili alle persone competenti. Lo scrum master si impegna a garantire che il team scrum agisca con trasparenza, rendendo visibile progressi e minacce/ostacoli.
- Empirismo: Il miglior modo per pianificare consiste nel lavorare concretamente apprendendo dai risultati. Questo è un approccio fondamentale per Scrum, che si basa sull'esperienza diretta e richiede che lo Scrum Master formi il team sulla suddivisione del lavoro, sulla definizione di obiettivi misurabili e sulla valutazione costante degli output generati.

- Valori: come riportato nel paragrafo 2.2.5, Scrum si basa su 5 valori di impegno, coraggio, concentrazione, apertura e rispetto, che creano un ambiente di sicurezza e fiducia. Seguire i valori è responsabilità di tutti i membri dello Scrum Team, ma lo Scrum Master assume un ruolo attivo nell'incoraggiare e ricordare a tutti l'importanza di tali valori. [17] [18]
- Il **Team di sviluppo (developer team)** ha il compito di portare a termine le attività pianificate durante lo sprint. Al termine di ogni ciclo, il loro obiettivo è consegnare un incremento completo del prodotto. Il gruppo di lavoro è costituito da professionisti molto competenti, come Quality Assurance/tester, Business Analyst, architetti, UX/UI designer, Ops/DevOps Engineer, oltre agli sviluppatori. Questi profili non rappresentano ruoli distinti, bensì abilità che il Developer team deve possedere collettivamente.

Analizzandole nel dettaglio:

- **Developer** si dedicano alla scrittura del codice, implementano funzionalità e risolvono anomalie. Possono avere diverse specializzazioni (front-end, back-end, data engineer, ecc.).
- **QA (Quality Assurance)** si impegnano a preservare la qualità del prodotto. Supportano il team nel mantenere una copertura adeguata dei test, identificano malfunzionamenti redigendo test automatici e manuali.
- **BA (Business Analyst)** hanno competenze di analisi dei requisiti e processi di business e collaborano con il Product Owner per chiarire le esigenze e con i Dev per tradurle in dettagli tecnici.
- **Architect** hanno competenze tecniche, utili a definire e guidare le decisioni architetturali. Aiutano il team a mantenere un prodotto scalabile, sicuro e coerente con la visione tecnica aziendale.
- **UX/UI Designer:** si concentrano sull'interfaccia grafica e sull'esperienza dell'utente curando l'aspetto visivo.
- **Ops/DevOps Engineer** gestiscono automazione, cloud e infrastruttura.

La composizione ottimale del Team di Sviluppo deve essere abbastanza piccola da rimanere agile e abbastanza grande da completare attività rilevanti all'interno dello Sprint. Un team troppo ridotto rischia di non avere delle skill necessarie, mentre team costituiti da un numero elevato di persone possono risentire di problemi nella gestione e nel coordinamento. [16]

### 2.2.5.2 Cerimonie

Le cerimonie (o riunioni) Scrum sono eventi in cui lo Scrum Master, il Product Owner e il team di sviluppo si incontrano per organizzare le attività, confrontarsi sull'avanzamento del lavoro e raccogliere feedback. Le cerimonie Scrum facilitano quindi la gestione dello sprint, offrendo una struttura chiara per una comunicazione costante, contribuendo a mantenere il team allineato al fine di migliorare i propri processi. Ogni sprint è costituito da 4 cerimonie fondamentali: lo sprint planning, lo stand-up giornaliero (daily scrum), lo sprint review e lo sprint retrospective. Tale processo è mostrato in figura 9.

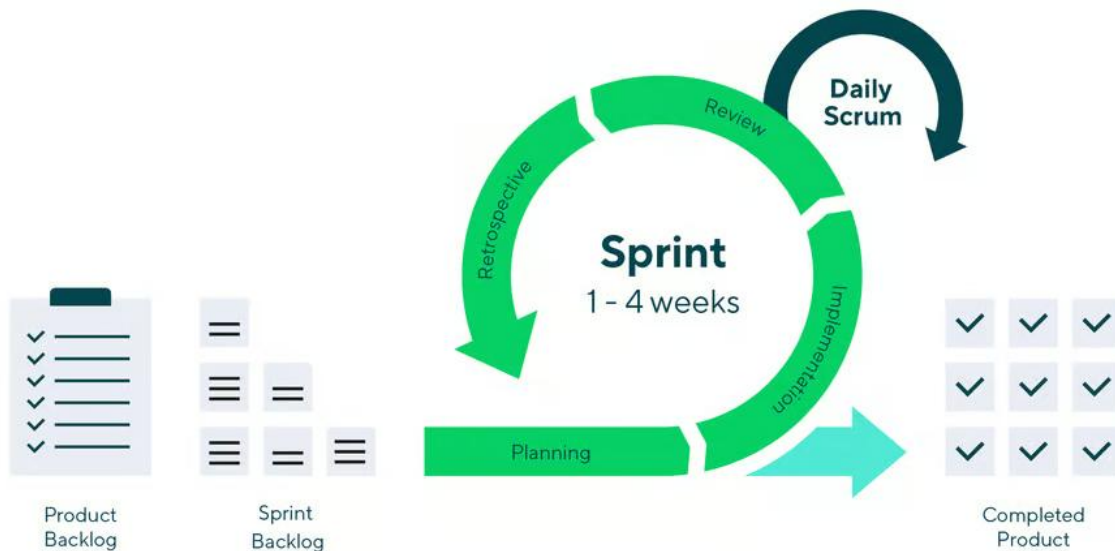


Figura 9: Processo Scrum

- **Sprint Planning:** segna l'inizio dello Sprint, definendo il lavoro da portare avanti nel corso dello stesso. Quindi l'obiettivo è determinare gli elementi del backlog del prodotto su cui si lavorerà durante lo sprint. In questa fase tutti i membri dello Scrum Team discutono e stabiliscono ciò che il team di sviluppo può realizzare

concretamente e, una volta raggiunto il consenso, viene fissato l'obiettivo dello sprint (sprint goal).

Lo Sprint Planning tratta i seguenti argomenti:

- **Valorizzazione dello Sprint:** l'intero Scrum Team collabora per definire uno Sprint Goal capace di trasmettere chiaramente il motivo per cui lo Sprint apporta beneficio agli stakeholder. Lo Sprint Goal deve essere finalizzato prima della conclusione dello Sprint Planning.
- **Ambito operativo dello Sprint:** Durante questa fase, il team ha la possibilità di affinare e chiarire diversi elementi, migliorando la comprensione comune e rafforzando la fiducia reciproca. Infatti, attraverso la discussione con il Product Owner, i Developer selezionano gli elementi del Product Backlog da includere nell'attuale Sprint. La scelta di ciò che può essere completato all'interno di uno Sprint può risultare complessa. Tuttavia, quanto più i Developer hanno familiarità con le prestazioni passate, le capacità previste e la Definition of Done (ossia il criterio che stabilisce quando un incremento può considerarsi concluso), tanto maggiore sarà la sicurezza nelle loro stime di lavoro.
- **Modalità di esecuzione del lavoro selezionato:** per ogni elemento selezionato del Product Backlog, i Developer pianificano il lavoro necessario per realizzare un Incremento che sia conforme alla Definition of Done. Questo viene spesso fatto scomponendo gli elementi del Product Backlog in elementi di lavoro più piccoli della durata di un giorno o meno. Il modo in cui questa suddivisione viene effettuata è completamente a discrezione del team di sviluppo: nessun altro stabilisce come i Developer debbano trasformare gli elementi del backlog in incrementi di valore.

Lo Sprint Goal, gli elementi del Product Backlog selezionati per lo Sprint, più il piano per la loro consegna sono insieme indicati come lo Sprint Backlog.

- **Daily Scrum:** I Daily Scrum consentono una comunicazione rapida e trasparente tra i membri del team, facilitando il confronto sui progressi evidenziando eventuali ostacoli e promuovendo così decisioni rapide ed efficaci. Questa riunione

quotidiana, della durata di 15 minuti, coinvolge i Developer del team Scrum, i quali hanno piena libertà di scegliere la struttura e le modalità più adatte, purché il focus rimanga sull'avanzamento verso l'obiettivo dello Sprint e sulla definizione di un piano operativo per la giornata successiva. L'obiettivo principale è quello di verificare i progressi verso lo Sprint Goal e di adattare lo Sprint Backlog secondo le esigenze, aggiustando il lavoro pianificato in arrivo. Ciò crea concentrazione e migliora l'autogestione.

- **Sprint Review:** si svolge alla fine di ogni sprint ed ha una durata massima di quattro ore per uno Sprint di un mese, mentre per Sprint più brevi, l'evento è proporzionalmente ridotto. Lo Scrum Team presenta agli stakeholder gli output del lavoro svolto, discutendo il livello di avanzamento rispetto al Product Goal. Lo scopo dello Sprint Review è quello di esaminare i risultati ottenuti e di determinare i futuri adattamenti. Sulla base di queste informazioni, i partecipanti collaborano per definire le azioni successive e individuare strategie per massimizzare il valore nei futuri Sprint. Anche il Product Backlog viene aggiornato tenendo conto di nuove opportunità emerse.
- **Sprint Retrospective:** è l'incontro che conclude lo Sprint ed ha una durata massima di tre ore per uno Sprint di un mese, mentre per Sprint più brevi, è proporzionalmente ridotto. Lo Scrum Team riflette sull'andamento dello Sprint precedente, considerando aspetti legati agli individui, alle interazioni, ai processi, agli strumenti e alla Definition of Done. Lo scopo è quello di analizzare ciò che ha funzionato durante lo Sprint ed identificare possibili miglioramenti, con l'intento di aumentare la qualità e l'efficienza del lavoro. Durante la riunione, il team discute dei successi, delle difficoltà riscontrate e delle modalità con cui eventuali problemi sono stati risolti (o meno). Le proposte di miglioramento vengono eventualmente integrate nello Sprint Backlog per lo Sprint successivo. [14] [19]

### 2.2.5.3 Artefatti

Gli artefatti sono informazioni che un team Scrum e le parti interessate utilizzano per dettagliare il prodotto in fase di sviluppo, le azioni per produrlo e le azioni eseguite durante il progetto.

Scrum genera artefatti accessibili e controllabili da tutti i membri del team, basandosi su un approccio fondato su trasparenza, verifica e adattamento. Per garantire che tali artefatti offrano dati utili a migliorare la trasparenza, ciascuno di essi è legato a un impegno specifico.

I principali artefatti Scrum sono:

- **Product Backlog:** insieme delle funzionalità e requisiti del prodotto associato al Product Goal.
- **Sprint Backlog:** elenco delle attività selezionate per lo Sprint accompagnato dall'impegno dello Sprint Goal.
- **Incrementi:** risultati tangibili ottenuti durante lo Sprint, il cui impegno è rappresentato dalla Definition of Done, ovvero il criterio che definisce quando un lavoro può considerarsi completato.

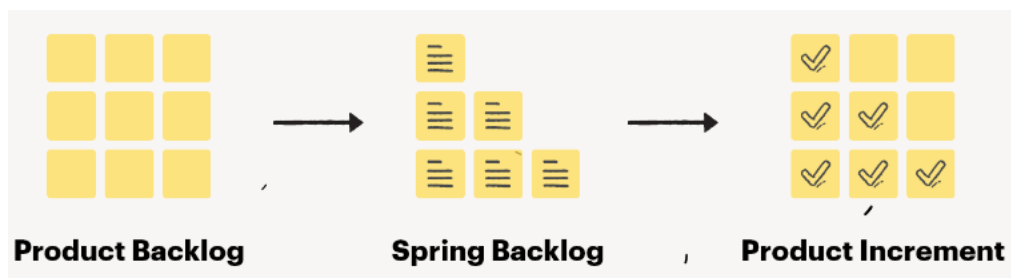


Figura 10: Gli artefatti Scrum

Il **Product Backlog** rappresenta la fonte primaria di informazioni su tutte le attività necessarie per lo sviluppo del prodotto: è un elenco di nuove funzionalità, ottimizzazioni, correzioni di bug, attività o requisiti di lavoro indispensabili per la realizzazione del prodotto. Il backlog del prodotto è un artefatto "attivo" perché è costantemente aggiornato in base alle nuove informazioni disponibili. La responsabilità della sua gestione ricade sul Product Owner, che è incaricato di curarne i contenuti e stabilirne le



priorità in modo continuativo. Tutti gli elementi contenuti nel backlog, infatti, devono essere classificati in ordine di priorità e ordinati in base al valore aziendale. Gli elementi con la massima priorità definiscono le successive attività di sviluppo ed hanno un aspetto più chiaro e informazioni più dettagliate rispetto agli elementi con priorità più bassa.

Il Product Goal è l'obiettivo a lungo termine per lo Scrum Team, descrivendo una visione futura del prodotto che può servire da orientamento per lo Scrum per la pianificazione.

Lo **Sprint Backlog** è composto dallo Sprint Goal (il perché), l'insieme degli elementi del Product Backlog selezionati per lo Sprint (il cosa), e da un piano attuabile per realizzare l'Increment (il come). In pratica, rappresenta la lista di attività del Product Backlog che sono state selezionate per essere sviluppate durante lo Sprint in corso, ovvero una lista delle “cose da fare” su cui il team di sviluppo deve concentrarsi durante lo Sprint in corso. Per la creazione dello Sprint Backlog, ogni attività del product backlog selezionata viene scomposta in task più piccoli, così da facilitare la gestione operativa. Questo livello di dettaglio deve essere sufficiente affinché i Developer possano ispezionare i propri progressi nel Daily Scrum. Lo Sprint Backlog costituisce quindi un quadro dinamico e trasparente del lavoro che i Developer intendono realizzare per raggiungere lo Sprint Goal, aggiornato costantemente per riflettere l'evoluzione delle attività. Se emergono scostamenti rispetto allo Sprint Goal, i Developer collaborano con il Product Owner per valutare eventuali modifiche all'ambito (scope) dello Sprint Backlog, purché non venga compromesso lo Sprint Goal.

Durante la Sprint Planning, il Product Owner supporta il team nella definizione dello Sprint Goal. Successivamente, il backlog dello Sprint può essere modificato dal team man mano che il lavoro progredisce, ma solo i Developer hanno l'autorità di modificare lo Sprint Backlog durante lo Sprint.

Un **incremento** è la somma di tutti gli elementi del backlog del prodotto completati nel corso dello sprint, più il valore cumulativo di tutti gli incrementi completati in precedenza. All'interno di uno stesso Sprint possono essere generati più incrementi. La somma degli Increment è presentata durante la Sprint Review. Un Increment, comunque, può essere consegnato agli stakeholder prima della fine dello Sprint. Affinché un'attività possa

essere considerata parte di un incremento, deve rispettare la Definition of Done: una descrizione formale che stabilisce quando un elemento è da considerarsi “completo” secondo gli standard qualitativi richiesti per il prodotto.

La Definition of Done garantisce trasparenza e allineamento all’interno del team, offrendo quindi una visione condivisa su cosa significa “lavoro completato”. Se un elemento non raggiunge questo livello di completezza, non può essere né rilasciato né presentato durante la Sprint Review. In tal caso, viene reinserito nel Product Backlog soggetto ad una valutazione futura. [14] [20]

## 2.3 Confronto tra Waterfall e Agile

Dopo aver condotto un’analisi accurata delle metodologie Waterfall e Agile, è necessario evidenziare le differenze tra i due approcci. Sebbene entrambi mirino alla consegna di un prodotto di qualità, differiscono profondamente per una serie di aspetti, come mostrato in tabella 1.

APPROCCIO	WATERFALL	AGILE
STILE DI PROCESSO	Sequenziale	Iterativo
DOMINIO	Prevedibile	Imprevedibile
DOCUMENTAZIONE	Completa	Minima-solo se necessaria
REQUISITI	<ul style="list-style-type: none"><li>• Stabiliti all’inizio del progetto</li><li>• Basso tasso di variazione</li></ul>	<ul style="list-style-type: none"><li>• Basati sul valore aziendale</li><li>• Regolarmente aggiornati</li></ul>
ENFASI	Processo	Persone

<b>MISURAZIONE DELLE PRESTAZIONI</b>	Conformità al piano	Valore aziendale
<b>PIANIFICAZIONE ANTICIPATA</b>	Alta	Bassa
<b>PROSPETTIVA VERSO IL CAMBIAMENTO</b>	Sostenibile	Adattabile
<b>LEADERSHIP</b>	Comando e controllo	Collaborativa
<b>ORGANIZZAZIONE</b>	Gestita	Auto-organizzata
<b>UTENTI</b>	Non coinvolti	Collaborazione stretta e frequente

*Tabella 1: Confronto tra Waterfall e Agile*

Uno degli aspetti che distingue maggiormente le metodologie Waterfall e Agile è lo stile di processo.

Il modello Waterfall si basa su un approccio lineare e sequenziale dove ogni fase del progetto viene eseguita una sola volta e deve essere completata prima di passare alla successiva. Questa applicazione rende il processo più rigido, ma anche più prevedibile. L'enfasi è posta sulla conformità al piano iniziale, con il successo che mira al rispetto di tempi, costi e requisiti stabiliti.

Agile adotta un approccio iterativo e incrementale dove il lavoro è gestito in brevi cicli (Sprint), durante i quali il piano viene costantemente adattato e modificato in base ai risultati ottenuti e alle nuove esigenze. La struttura ciclica di Agile implica che tutte le fasi del processo si ripetano ad ogni Sprint.

Nel modello Waterfall, i requisiti sono definiti fin dall'inizio e sono difficilmente modificabili. Di conseguenza, la documentazione assume un ruolo centrale: ogni fase è formalizzata e tracciata, garantendo controllo ma limitando la flessibilità.

Agile, invece, parte dal presupposto che il contesto sia mutevole, infatti i requisiti hanno la possibilità di evolversi nel tempo, in funzione del valore aziendale e la documentazione è ridotta al minimo indispensabile. La comunicazione diretta con il cliente e il valore reale

generato dal prodotto sono fattori in cui vi è posta una grande attenzione. Il progresso viene misurato non solo rispetto al piano, ma soprattutto in base al valore effettivo che il prodotto rilasciato apporta agli stakeholder.

Questo cambio di prospettiva rappresenta una differenza sostanziale: Waterfall premia la pianificazione, Agile premia la capacità di adattamento, considerandolo come parte integrante del processo. Infatti, nel modello a cascata la pianificazione è alta e avviene all'inizio del progetto, definendo in dettaglio tempi, costi e requisiti. Nel modello Agile, la pianificazione iniziale è bassa e progressiva: ogni iterazione serve per affinare il backlog e aggiornare le priorità.

Un altro elemento di netta divergenza tra i due modelli è il cambiamento: nel metodo tradizionale Waterfall, i requisiti vengono definiti e documentati in modo rigido all'inizio del progetto, rendendo difficile qualsiasi modifica successiva.

Agile, invece, accoglie il cambiamento come parte integrante del processo: se emergono nuove esigenze, il team non interrompe il lavoro, ma valuta come integrare al meglio le variazioni nel flusso operativo.

Queste differenze si riflettono anche nella struttura organizzativa, infatti mentre Waterfall si basa su una leadership gerarchica, dove le decisioni sono centralizzate e i team eseguono quanto stabilito dai manager, Agile promuove una leadership distribuita e collaborativa dove i team sono auto-organizzati e gestiscono autonomamente le proprie attività, favorendo responsabilità condivisa e flessibilità.

Il coinvolgimento degli utenti è un altro elemento distintivo: nel modello tradizionale, gli utenti partecipano solo nelle fasi iniziali (raccolta requisiti) e finali (collaudo); nel modello Agile, invece, esiste una collaborazione continua con gli stakeholder dove il processo è costantemente modellato sulle loro esigenze, grazie a feedback frequenti che orientano lo sviluppo del prodotto.

Infine, i due approcci si differenziano anche nella gestione dei vincoli di progetto, come mostrato in figura 11.

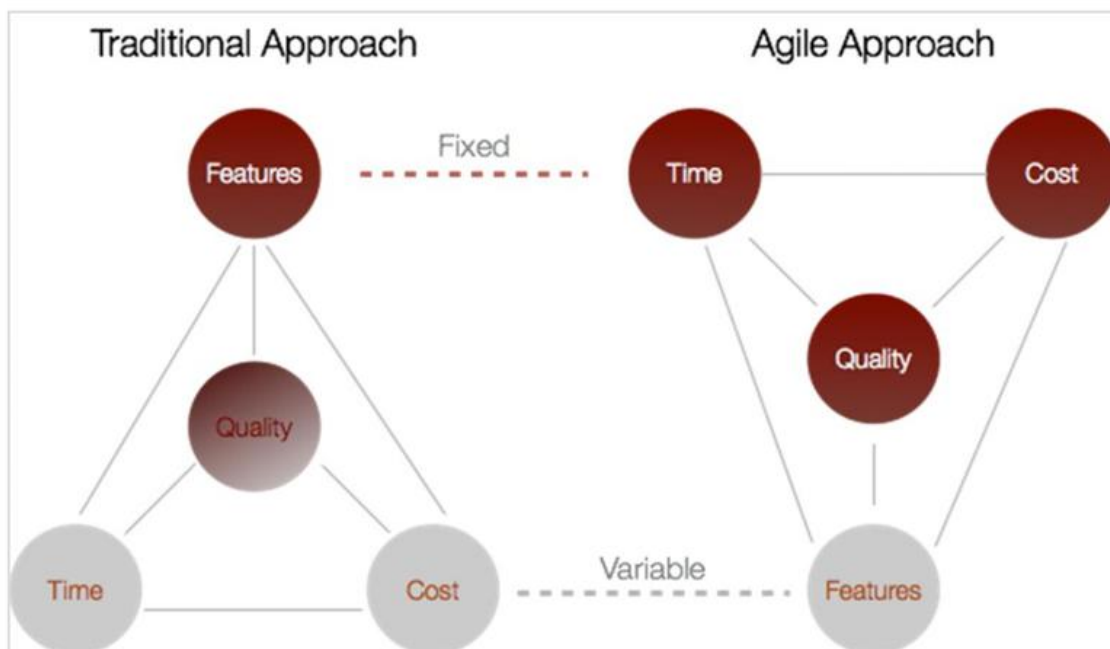


Figura 11: Confronto dei triangoli di progetto per gli approcci tradizionale e Agile

Il modello tradizionale è di tipo “plan-driven”, ovvero i requisiti (o funzionalità) sono considerati fissi: tutte le funzionalità previste devono essere consegnate, indipendentemente dalle variazioni di tempo o costo. In questo schema, infatti, tempo e costo sono variabili: possono aumentare se necessario per garantire la consegna completa del prodotto. Quindi, in primis bisogna fissare lo scopo del progetto ed in seguito si definisce il piano di lavoro. Tuttavia, questo approccio può portare a ritardi e sforamenti di budget.

Nel modello Agile, invece, la logica è rovesciata: è di tipo “value-driven”, dove tempi e costi sono fissi, mentre i requisiti possono variare in base al valore e alla priorità. Infatti, l'ambito non solo può cambiare, ma è anche prevedibile che cambi. Questo consente di mantenere costante la qualità, focalizzandosi su ciò che genera il maggiore impatto per il business e gli utenti.

In sintesi, Waterfall è indicato per progetti stabili, con requisiti chiari e non mutabili, dove la documentazione e la pianificazione sono fondamentali. Agile, al contrario, eccelle in ambienti dove la complessità e la mutevolezza sono caratteristiche alla base e dove il valore si costruisce attraverso il feedback continuo.

La scelta dell'approccio migliore dipende sempre dalle caratteristiche specifiche del progetto.

Guardando al futuro, la necessità di risposte rapide ai cambiamenti di mercato e l'aumento delle complessità dei progetti favoriranno l'adozione di approcci agili. [21]

## **3 Metodologia**

### **3.1 Il settore assicurativo: complessità e sfide nei progetti IT**

Il settore assicurativo è nato con l'obiettivo di proteggere individui e imprese dai rischi economici, nel corso del tempo ha dovuto adattarsi a un contesto profondamente mutato, caratterizzato da digitalizzazione, globalizzazione dei mercati e cambiamenti normativi sempre più stringenti rappresenta uno dei comparti più complessi e regolamentati dell'intero panorama finanziario. Ad oggi le compagnie assicurative operano in un ecosistema competitivo, tecnologicamente avanzato e guidato dai dati, in cui l'innovazione digitale non è più un'opzione strategica, ma una condizione necessaria per restare sul mercato. [22]

Secondo Majukumar e Hernandez [23], la trasformazione digitale che il settore assicurativo sta attraversando è determinata soprattutto dalla necessità di rispondere in modo più rapido e flessibile alle aspettative dei clienti e all'intensificarsi della competizione. I clienti si aspettano di interagire tramite diversi canali come social media, dispositivi mobili, contact center, siti web aziendali e vogliono scegliere autonomamente il tipo di interazione e di esperienza.

Oltre a doversi misurare con i concorrenti tradizionali e con la difficoltà di preservare i clienti acquisiti, le compagnie assicurative devono battersi contro nuovi competitor, i cosiddetti InsurTech, che si presentano sotto forma di start-up tecnologiche più piccole e più agili che offrono prodotti molto mirati e personalizzati. [24]

Le imprese assicurative devono ripensare i propri modelli di business, puntando a migliorare la customer experience ed allo stesso tempo spostando il focus verso una maggiore centralità del cliente ("customer centricity") ed un uso più esteso delle tecnologie digitali.

"Digital transformation" per il settore assicurativo significa ottimizzare i processi e progredire avendo idee innovative. La definizione di una priorità strategica incentrata su uno solo di questi ambiti non è sufficiente per restare competitivi nell'economia digitale. Si rende necessario fare in modo che le funzioni principali, così come quelle di supporto

dell'azienda, come Finance e HR, abbiano una visione chiara della strategia aziendale e del loro ruolo nel processo.

La digital transformation non può limitarsi all'implementazione tecnologica, ma deve essere un processo integrato che coinvolga l'intera organizzazione: dai processi decisionali alla cultura aziendale, fino ai modelli di governance e collaborazione interna.

Nonostante i progressi compiuti, permangono una serie di ostacoli alla trasformazione digitale:

- **Sistemi legacy e fonti di dati frammentate:** molte compagnie assicurative hanno investito nella creazione di nuovi canali digitali, ma spesso non riescono a trarne un effettivo vantaggio a causa dell'assenza di esperienze completamente interattive. Questo è dovuto al fatto che i loro attuali sistemi non supportano l'integrazione di dati interni ed esterni da cui trarre informazioni utili per un coinvolgimento dei clienti positivo, significativo e trasparente. La qualità dei dati è penalizzata dai tentativi di conciliare sistemi frammentati e formati di memorizzazione diversi. I sistemi causano problemi di scarsa trasparenza, qualità e velocità di condivisione dei dati, e di conformità, limitando la capacità delle compagnie di innovare in tempi brevi, aumentando di conseguenza i costi di manutenzione e riducendo la flessibilità operativa necessaria per adattarsi ai cambiamenti del mercato
- **Mancanza di strategia e collaborazione a livello aziendale:** spesso le iniziative di trasformazione vengono implementate a livello di singola business unit, senza una visione strategica di lungo periodo. Questa frammentazione si manifesta spesso anche nella separazione tra funzioni di business e IT, che operano con obiettivi e priorità diverse, rendendo complessa l'integrazione dei processi e rallentando l'innovazione. Tale approccio ostacola la creazione di un ecosistema aziendale integrato, che va a ridurre la capacità di offrire customer experience coerenti e di condividere dati e conoscenze tra i vari reparti. I clienti nel mercato di oggi prediligono l'apertura e la cooperazione, infatti apprezzano l'intuitività del servizio, la possibilità di scegliere i canali e di controllare i propri dati e rapporti. La consapevolezza a livello di azienda e di ecosistema dell'importanza dei dati e del modo in cui dovrebbero essere trattati è oggi fondamentale per fornire offerte



aggregate contestuali e incentrate sul valore. Un'azienda in grado di condividere internamente i dati in modo efficace ha maggiori probabilità di migliorare le proprie capacità di partnership e la customer experience aprendosi alla collaborazione con terzi [23]

- **Complessità dei flussi informativi:** i sistemi assicurativi devono interagire con una molteplicità di piattaforme interne ed esterne: sistemi di gestione polizze, piattaforme sinistri, CRM, moduli di pagamento, sistemi antifrode, applicazioni per agenti e broker, nonché interfacce con enti regolatori e partner esterni. La necessità di garantire continuità operativa e coerenza dei dati su scala internazionale aumenta ulteriormente il livello di difficoltà nella gestione dei progetti IT
- **Vincoli normativi e culturali:** le compagnie devono affrontare un quadro regolatorio sempre più complesso, in particolare in ambito protezione dei dati (GDPR), antiriciclaggio e sicurezza informatica, che richiede investimenti significativi in compliance e cybersecurity. A ciò si aggiungono resistenze interne al cambiamento, legate a una cultura organizzativa tradizionalmente orientata alla stabilità e al controllo, piuttosto che alla sperimentazione e all'agilità. [23]  
[24]

Secondo McKinsey [25], per superare queste barriere è necessario adottare un modello operativo agile, in cui team multidisciplinari autonomi collaborano per generare risultati aziendali commisurati alle esigenze del momento. La metodologia di lavoro agile è basata, inoltre, su processi decisionali rapidi e una governance flessibile. Questo approccio riduce la complessità organizzativa, andando a migliorare la velocità di rilascio dei prodotti ed aumentando la capacità di risposta ai cambiamenti del mercato, oltre che l'innovazione. [23]

Infine, come sottolineano Majukumar e Hernandez, le sfide legate alla digital transformation rappresentano anche un'opportunità di rinnovamento strategico, infatti le compagnie che combinano innovazione tecnologica, agilità organizzativa e capacità di partnership con player tecnologici trasformeranno la complessità del settore in un vantaggio competitivo sostenibile, costruendo così un modello assicurativo più resiliente, trasparente e centrato sul cliente.

## 3.2 La metodologia Agile nel settore assicurativo

L'adozione di un approccio più flessibile come la metodologia Agile assume un ruolo strategico nel contesto del settore assicurativo, contraddistinto da rigidità infrastrutturali, elevati vincoli normativi e una forte separazione tra funzioni IT e business. Come evidenziato nel paragrafo precedente, la complessità e la lentezza dei processi IT tradizionali rendono necessario un cambio di paradigma gestionale. Le compagnie assicurative che intendono innovare, migliorare la collaborazione tra stakeholder e soddisfare le mutevoli esigenze dei clienti stanno progressivamente considerando l'Agile come modello operativo in grado di mettere a sistema efficienza, velocità e orientamento al valore.

La ricerca "Scaling agility: A new operating model for insurers" condotta da McKinsey & Company [25] evidenzia che l'adozione dell'Agile nel settore assicurativo consente di rispondere più efficacemente alle esigenze dei clienti e di adattarsi a un contesto in continuo cambiamento. Team cross-funzionali, iterazioni brevi, feedback rapidi e rilascio incrementale diventano strumenti essenziali per aumentare la produttività e favorire un miglior allineamento tra business e IT. In particolare, la stretta collaborazione tra le due aree permette di ridurre i tempi di allineamento e di migliorare la qualità delle decisioni strategiche, favorendo una cultura aziendale più trasparente e partecipativa.

L'approccio Agile trova oggi applicazione in diversi ambiti chiave del settore assicurativo:

- Sviluppo di piattaforme digitali per la gestione di polizze, sinistri e relazioni con il cliente. L'approccio Agile permette di rilasciare tali piattaforme in modo incrementale, introducendo funzionalità progressivamente testate con gli utenti finali [26]
- Trasformazione operativa e automazione dei processi interni, come underwriting (valutazione e accettazione del rischio assicurativo) e la gestione sinistri (dalla denuncia alla liquidazione), i quali sono due dei processi più complessi e regolamentati in ambito assicurativo. Tradizionalmente sono stati gestiti con procedure manuali, lente e frammentate tra diversi dipartimenti. L'approccio iterativo, invece, consente una revisione continua delle soluzioni e una maggiore collaborazione tra business e IT. Ad esempio, un caso pubblicato da PilotFish Technology mostra un progetto Agile

per automatizzare il processo di underwriting in una compagnia americana: integrando i sistemi legacy con un motore decisionale in tempo reale, i team hanno potuto testare e adattare progressivamente le regole di business, migliorando la velocità e la precisione delle decisioni assicurative. [27]

Tali applicazioni portano benefici tangibili: riduzione del time-to-market, semplificazione della governance, miglioramento della qualità del prodotto/servizio; sul piano organizzativo si osservano una maggiore cooperazione tra i vari team (business, IT, operations e compliance), maggiore trasparenza con conseguente incremento del coinvolgimento dei dipendenti, che comprendono meglio le priorità aziendali e partecipano attivamente ai processi decisionali. [25] [28]

Lo studio condotto da McKinsey [25] riporta, ad esempio, che una compagnia assicurativa asiatica che ha completato la propria trasformazione Agile ha registrato un incremento di quasi 20 punti nella soddisfazione dei clienti e di oltre 20 punti nel coinvolgimento dei dipendenti dopo un anno di adozione del modello Agile, come riportato in figura 12.

#### **Agile companies show superior operational, health, and financial results.**

Time to market, months

**5–7× faster**

Productivity

**20–30%  
improvement**

Customer-satisfaction score

**20–30 p.p.  
improvement**

Employee-engagement score

**20–25 p.p.  
improvement**

Note: p.p. = percentage point.

*Figura 12: Impatto della trasformazione Agile (McKinsey)*

Questi risultati trovano riscontro anche nello studio “Agile adoption challenges in insurance: a systematic literature and expert review” condotto da Krishna Yudhokusuma, Putra Munandar e Teguh Raharjo [29], che analizza le principali sfide e soluzioni nell’adozione dell’Agile nel settore assicurativo. Gli autori evidenziano che le difficoltà maggiori non risiedono negli aspetti tecnici, ma nelle dimensioni organizzative e culturali tipiche delle compagnie assicurative: una governance fortemente gerarchica, la presenza di processi di

compliance stringenti e una cultura orientata al controllo possono ostacolare la piena realizzazione dei principi Agile.

Munandar e Raharjo classificano tali sfide in otto domini principali, derivati dal PMBOK 7th Edition:

- **Stakeholders:** resistenza al cambiamento e difficoltà di coinvolgimento dei portatori di interesse
- **Team:** mancanza di fiducia, scarsa collaborazione, incomprensione su cosa significhi “Done”
- **Development approach & life cycle:** difficoltà nell’adattare i framework agili a strutture organizzative tradizionali e gerarchiche, scaling in grandi organizzazioni
- **Planning:** carenza di strumenti di monitoraggio coerenti con i principi Agile
- **Project work:** comunicazione inefficace, scarsa integrazione tra team e gestione inefficace dei cambiamenti
- **Delivery:** mancanza di strategia ICT condivisa sul miglioramento continuo dello sviluppo, requisiti incompleti/contraddittori e coinvolgimento non appropriato di stakeholder nelle decisioni tecniche
- **Measurement:** mancanza di azioni tempestive per mantenere i progetti nei tempi (sottostima tempi, feedback tardivi)
- **Uncertainty:** complessità legata a sistemi legacy, interdipendenze e normative variabili.



*Figura 13: Dominio delle prestazioni del progetto*

Per superare queste barriere, gli autori suggeriscono:

- Maggiore comunicazione e trasparenza, adottare pratiche di governance che supportino agilità (quindi governance adattativa, non gerarchica)
- Sviluppo di soft skills e coaching per costruire team agili maturi, definizioni condivise di “Done”
- Personalizzare i framework, introducendo Agile in modo graduale e utilizzare l'Enterprise Architecture Management (EAM) per coordinare l'adozione su larga scala. Infatti, EAM svolge un ruolo fondamentale nel garantire coerenza tra la flessibilità dei team agili e la stabilità dei processi aziendali di lungo periodo
- Migliorare canali di comunicazione, includere rappresentanti multi-team, definire priorità chiare

- Focalizzarsi sulle esigenze chiave, investire in gestione dei requisiti e nel ruolo del Product/Project Owner, aumentare trasparenza con gli stakeholder
- Ridurre lead time, introdurre feedback più rapidi e metriche agili che siano utili e tempestive [29]

Ulteriori conferme provengono dallo studio “Agile Transformation in the European Insurance Sector” di The Transformation Alliance [28], che sottolinea come il successo dell’adozione Agile richieda la definizione chiara della visione di trasformazione e la misurazione dei risultati tramite KPI dedicati. Lo studio, basato su interviste ad importanti compagnie assicurative europee, evidenzia vantaggi quali riduzione dei costi operativi, maggiore produttività ed incremento della trasparenza interna che consente ai dipendenti di comprendere meglio il proprio apporto ai risultati.

L’adozione dell’Agile in ambito assicurativo, nonostante i benefici, deve essere accompagnata da una trasformazione del modello operativo complessivo, che vada ad includere la revisione delle strutture di governance, l’allineamento con i requisiti di compliance locali (come le normative paese) e l’utilizzo di chart e metriche Agile adattati al contesto assicurativo e internazionalizzato. [25] [28]

In sintesi, la metodologia Agile rappresenta per il settore assicurativo un fattore abilitante di innovazione e competitività, capace di migliorare la velocità decisionale, la collaborazione, la capacità di risposta alle evoluzioni del mercato. Tuttavia, il suo successo dipende dalla capacità delle organizzazioni di integrare la flessibilità del modello Agile con la stabilità e i requisiti di compliance del settore assicurativo, trasformando le sfide strutturali in leve per un miglioramento continuo.

Le evidenze riportate trovano riscontro anche nel progetto oggetto di questa tesi, dove l’adozione dell’approccio Agile ha richiesto un bilanciamento tra la flessibilità metodologica e le esigenze di compliance e coordinamento multi-paese.

### 3.3 Contesto aziendale

Accenture è una delle principali aziende globali nel settore di servizi professionali, che offre una vasta gamma di soluzioni nei campi Strategy, Consulting, Technology, Operations e Digital.

Fondata nel 1989 e con sede principale a Dublino, la società opera oggi in oltre 120 nazioni e vanta circa 779.000 dipendenti nel mondo. La mission di Accenture è assistere le organizzazioni pubbliche e private nella costruzione di un futuro digitale e sostenibile, supportandole nei processi di trasformazione digitale, innovazione tecnologica e ottimizzazione dei processi operativi. Accenture, quindi, aiuta le organizzazioni a migliorare le loro performance e a creare valore sostenibile per gli stakeholder.

L'azienda è organizzata in business unit: Strategy & Consulting (che assiste le aziende nelle decisioni strategiche e organizzative), Technology (che si dedica allo sviluppo software, system integration e implementazione di piattaforme digitali), Operations (incentrata sull'ottimizzazione dei processi aziendali) e Industry X (che unisce capacità ingegneristiche e digitali per incentivare l'innovazione industriale). Questo modello consente una copertura trasversale delle esigenze dei clienti.

Al centro della cultura aziendale vi è la valorizzazione delle persone, dell'innovazione e della sostenibilità. Accenture promuove la diversità, la formazione continua e l'adozione responsabile delle tecnologie emergenti, come l'intelligenza artificiale. Gli investimenti e le attività dell'azienda rispecchiano l'impegno verso un modello di business che considera gli impatti sociali e ambientali, oltre alla performance economica.

In Italia Accenture opera da decenni con sedi principali a Milano, Roma, Torino, Napoli e Cagliari, oltre ad ulteriori uffici e centri operativi. L'Italia rappresenta un nodo strategico nel network internazionale dell'azienda, grazie anche alla presenza di diversi centri di innovazione specializzati (ad esempio nei settori automotive, industria 4.0, telecomunicazioni) e Delivery Center che agevolano la gestione operativa e il rilascio di soluzioni digitali.

Nel settore assicurativo, Accenture si posiziona come partner strategico di riferimento per la modernizzazione dei sistemi core e per l'introduzione di modelli digitali basati su architetture modulari, piattaforme cloud e intelligenza artificiale.

L'obiettivo è rendere le compagnie più agili, efficienti e in grado di rispondere rapidamente ai mutamenti del mercato. [30] [31] [32]

### **3.4 Descrizione del progetto**

Nel corso del tirocinio curriculare presso la società Accenture, leader nel settore della consulenza strategica e tecnologica, ho preso parte ad un progetto assicurativo internazionale finalizzato alla dismissione e ristrutturazione delle piattaforme di gestione dei sinistri Motor e Non Motor di una delle più importanti compagnie assicurative europee. Le piattaforme preesistenti della compagnia assicurativa, gestite da sistemi separati, presentavano criticità in termini di interoperabilità, scalabilità e manutenzione.

L'obiettivo principale del progetto analizzato nella presente tesi è quello di modernizzare e integrare i sistemi di gestione dei sinistri, dismettendo le vecchie piattaforme e ricostruendole su un'architettura moderna, modulare e flessibile.

Il nuovo modello prevede la suddivisione della piattaforma in verticali funzionali (o moduli), ciascuno dedicato a un'area specifica del processo di gestione dei sinistri, ad esempio Motor, Non-Motor, Pagamenti, Workflow approvativo, Gestione documentale e Reportistica KPI.

Questa struttura consente di sviluppare i componenti in modo indipendente, garantendo al contempo l'interoperabilità grazie all'uso di servizi API, integrazioni con sistemi esterni come sistemi di pagamento e documentali (es. PDM) e piattaforme cloud-native.

Quindi l'obiettivo del progetto è duplice:

- Ottimizzare l'architettura tecnologica, rendendola più moderna, scalabile e semplice da mantenere nel tempo
- Migliorare la qualità operativa e la tracciabilità dei processi di gestione sinistri, riducendo i tempi di lavorazione e aumentando la capacità di risposta alle richieste del business.

Il progetto è stato gestito seguendo la metodologia Agile, nello specifico tramite il framework Scrum. Questo approccio ha permesso di gestire la complessità tecnica e organizzativa del



progetto, assicurando un flusso di comunicazione costante tra i vari gruppi e una pianificazione iterativa degli sviluppi.

Il ciclo di lavoro segue quindi la struttura tipica del framework Scrum:

- Sprint Planning, in cui vengono definite le User Stories prioritarie e gli obiettivi di sprint (Sprint Goal)
- Daily Scrum, incontri giornalieri di 15 minuti per condividere i progressi, discutere eventuali ostacoli e riallineare le priorità
- Sprint Review, al termine di ogni sprint, con la presentazione dei risultati agli stakeholder di business e la raccolta dei feedback
- Sprint Retrospective, dedicata all'analisi dei punti di forza e delle criticità del ciclo appena concluso, per introdurre azioni di miglioramento continuo.

Le attività quotidiane e l'avanzamento dei lavori sono gestiti e monitorati tramite strumenti come Jira (per la gestione del backlog e dei report Agile), Confluence (per la documentazione condivisa) ed EazyBI (per la visualizzazione e l'analisi dei KPI di progetto).

Nel dettaglio, il lavoro è organizzato in diversi team cross-funzionali, ciascuno focalizzato su un verticale specifico e composto da professionisti con ruoli complementari:

- Product Owner (PO): responsabile della definizione del backlog e della prioritizzazione delle User Stories, nonché dell'allineamento continuo con gli stakeholder di business. Ogni verticale ha un proprio PO, incaricato di definire gli obiettivi di sprint (Sprint Goal)
- Scrum Master (SM): facilitatore del team, garante dell'applicazione dei principi Scrum e della rimozione degli ostacoli operativi
- Development Team: gruppo multidisciplinare di analisti, sviluppatori e tester incaricati di implementare le funzionalità pianificate nello sprint
- Solution Architect e Technical Lead: responsabili della coerenza architetturale complessiva e delle integrazioni tecniche tra i vari verticali
- Project Manager e Governance Team: con funzioni di coordinamento inter-team, gestione delle milestone e controllo dei KPI di avanzamento.

Questa organizzazione riflette il principio Agile della responsabilità distribuita: ogni team opera in autonomia sulla propria area, ma contribuisce a un obiettivo comune, mantenendo

una comunicazione costante attraverso meeting trasversali durante i quali vengono allineati i progressi, le dipendenze e i rischi di progetto.

Il lavoro è stato organizzato in sprint di tre settimane, ognuno dei quali mira alla realizzazione e al rilascio incrementale di nuove funzionalità. Nel dettaglio, il periodo di osservazione su cui si concentra per semplicità questo elaborato di tesi è da aprile a giugno, in cui il team ha completato quattro sprint consecutivi, raggiungendo in ciascuno di essi gli obiettivi di sprint prefissati.

Ogni sprint si è concluso con un rilascio incrementale, corrispondente a una milestone funzionale del programma:

- Sprint 1: realizzazione della base del sistema e della gestione dei ruoli di accesso
- Sprint 2: aggiunta delle funzionalità di pagamento e gestione documentale
- Sprint 3: implementazione del workflow approvativo (flussi di approvazione sinistri e delle notifiche automatizzate)
- Sprint 4: introduzione dei moduli di monitoraggio e ottimizzazione del sistema.

Durante il tirocinio, ho ricoperto un ruolo di supporto alla governance del progetto, partecipando attivamente al monitoraggio delle attività e all'analisi dei KPI di performance.

In particolare, mi sono occupata di:

- Monitorare l'andamento operativo dei verticali, verificando il rispetto della pianificazione rispetto alla roadmap definita
- Analizzare i principali Agile report, valutando l'efficienza dei team e la regolarità dei rilasci
- Collaborare con i Product Owner nella revisione degli obiettivi di sprint e nell'individuazione di eventuali criticità tecniche o organizzative
- Partecipare ai meeting settimanali di avanzamento e agli allineamenti cross-team, interfacciandomi con stakeholder di diverse aree (architetturale, business)
- Contribuire alla raccolta e alla formalizzazione dei dati di performance, successivamente utilizzati per l'elaborazione dei Key Performance Indicators (KPI) descritti nel Capitolo 4.

Questa esperienza mi ha permesso di approfondire concretamente i meccanismi di governance Agile in un contesto aziendale di grandi dimensioni, osservando come la

collaborazione, la trasparenza e la misurazione continua delle performance siano elementi centrali per garantire il successo di un progetto complesso.

All'interno del periodo di osservazione sono stati monitorati i principali Agile report (Burndown, Burnup, Velocity e Cumulative Flow Diagram).

Questi strumenti hanno consentito di analizzare l'andamento del lavoro, la velocità di delivery e la stabilità del flusso operativo, fornendo la base per la successiva analisi dei Key Performance Indicators descritta nel capitolo 4.

### 3.5 Applicazione della metodologia Agile nel progetto

Il progetto oggetto di analisi ha una durata pluriennale e un'elevata complessità organizzativa e tecnica. Per semplicità e per rendere più chiara l'analisi dei risultati ottenuti e dell'applicazione della metodologia Agile, si è scelto di considerare un periodo campione di tre mesi, da aprile a giugno, corrispondente a quattro Sprint di tre settimane ciascuno.

L'obiettivo di questa analisi è mostrare, attraverso i principali Agile Report disponibili in Jira/EazyBI, piattaforma comunemente impiegata in azienda, l'andamento del progetto e la capacità del team di rispettare le pianificazioni, gestire lo scope e migliorare la propria efficienza nel tempo.

Nel contesto Agile, il lavoro è organizzato su più livelli di granularità: alla base vi sono le Epiche, che rappresentano i macro-obiettivi funzionali del progetto, come ad esempio l'integrazione del modulo Motor Claims o il rilascio della nuova interfaccia Non-Motor. Ogni epica è stimata ad alto livello tramite la T-Shirt Size, una convenzione che consente di tradurre l'epica in un valore numerico espresso in Person Days (giornate/uomo).

Nel progetto analizzato è stata adottata la seguente tabella di conversione:

T-shirt size	PDs
XXS	5
XS	20
S	65
M	100

L	150
XL	230
XXL	460

Tabella 2: Conversione da T-shirt size a PDs

Ciascuna epica viene poi scomposta in User Stories (US), ovvero elementi di lavoro più piccoli e più concreti, che descrivono specifiche funzionalità o requisiti da realizzare. Le user stories vengono stimate in Story Points (SP). In particolare, nel progetto si assume che uno story point coincide con un person day (1 SP = 1 PD).

Come rappresentato in tabella 3, durante il trimestre analizzato il team ha gestito 12 epiche, di dimensioni variabili tra XS e L, per un totale di circa 1.000 person days complessivi. Da queste epiche sono derivate circa 40 user stories, che nel periodo considerato hanno totalizzato 350 story points complessivi, distribuiti sui quattro sprint analizzati. Ogni sprint ha incluso una selezione di user stories provenienti da epiche diverse, con una pianificazione media compresa tra 80 e 95 story points per iterazione, in funzione delle priorità e della complessità del lavoro.

Di seguito è mostrato il contributo in termini di effort (person days) di ciascuna delle dodici epiche e i relativi story points pianificati negli sprint considerati:

Epica	T-Shirt Size	PDs totali	US nel trimestre	SP allocati nel trimestre	Sprint di riferimento
Motor Claims (integrazione)	M	100	4	35	S1, S2
UI Non-Motor (ristrutturazione)	S	65	4	35	S1, S2, S3
Pagamento automatico	L	150	5	45	S2, S3, S4
Workflow di approvazione	M	100	5	40	S3, S4
Reportistica	XS	20	2	20	S4
Ruoli e permessi	S	65	3	30	S1, S2

Performance DB sinistri	M	100	4	35	S1, S2, S3
Documentali e allegati	S	65	2	20	S2, S3, S4
Notifiche ed eventi	XS	20	2	20	S3, S4
Integrazione PDM	L	150	5	40	S2, S3, S4
Portale partner/officine	M	100	3	23	S1, S2, S3
Automazione Test & QA	S	65	1	7	S2, S3
<b>Totale</b>	<b>-</b>	<b>1.000</b>	<b>40</b>	<b>350</b>	<b>-</b>

Tabella 3: Contributo delle epiche in PDs e SP

E' importante precisare che in totale le epiche distinte nel trimestre sono 12, ma non tutte attive contemporaneamente. Alcune epiche sono sviluppate su più sprint (es. Motor Claims, Workflow di approvazione), altre sono brevi e concluse in un singolo sprint (es. Reportistica). Il numero di epiche "attive" per sprint rappresenta la varietà funzionale del lavoro gestito (quante epiche hanno avuto user stories attive in quello specifico sprint), non la quantità totale di epiche presenti nel periodo.

La tabella seguente riassume i dati principali utilizzati per la costruzione dei grafici Agile (Burndown, Burnup, Velocity chart e Cumulative Flow Diagram), mostrati nei paragrafi seguenti.

Sprint	Epiche coinvolte	SP pianificati (Backlog Sprint)	SP completati (incrementali)	SP completati (cumulativi)	Scope totale (scope creep)	SP rimanenti
<b>Sprint 0</b>	-	0	0	0	350	350
<b>Sprint 1</b>	5	80	75	75	355	280
<b>Sprint 2</b>	9	85	80	155	365	210
<b>Sprint 3</b>	9	90	88	243	368	125
<b>Sprint 4</b>	6	95	92	335	368	33

Tabella 4: Sintesi dei dati di progetto per sprint

Lo Sprint Backlog contiene quindi solo una selezione di user stories appartenenti a diverse epiche, rappresentando il lavoro pianificato dal team. Di conseguenza, gli Story Points totali stimati nel backlog dello sprint non coincidono con la somma complessiva di tutte le epiche, ma solo con la porzione effettivamente gestita in quel periodo.

Lo Sprint 0 indica lo stato iniziale, ovvero il punto di partenza, prima dell'inizio effettivo della release. Non vi sono epiche attive, né backlog pianificato o completato. Lo scope iniziale è 350 SP, e l'intero backlog è ancora "da fare", quindi gli SP rimanenti sono 350.

### 3.5.1 Burndown Chart

Il Burndown Chart è una rappresentazione grafica del lavoro rimanente nel tempo. Vengono messi a paragone il lavoro svolto con quello che il team si era prefissato di svolgere.

Sull'asse orizzontale sono riportati gli Sprint, mentre sull'asse verticale viene indicata la quantità di lavoro residua, espressa in Story Points.

Il grafico consente di monitorare visivamente la velocità con cui il team riduce il backlog e di prevedere il completamento delle attività. E' importante sottolineare che l'andamento è decrescente, in quanto tale grafico è una decumulata delle attività, indicando il lavoro che manca per raggiungere l'obiettivo.

Nel caso analizzato, la linea arancione tratteggiata rappresenta l'andamento previsto (ideale), ossia la riduzione lineare del backlog da 350 SP fino a 0 al termine del quarto sprint. Il backlog ideale rimanente a fine sprint rappresenta la traiettoria teorica di riduzione del backlog se il team mantenesse un ritmo perfettamente costante per tutta la release. In particolare, la riduzione costante ideale è pari a 87,5, valore ottenuto dividendo il backlog iniziale della release per il numero di sprint totali.

La linea blu continua, invece, mostra l'andamento reale: gli Story Points rimanenti, ottenuti come differenza tra Scope totale (fine sprint) e SP Completati (cumulativi), si riducono in modo costante, arrivando ad un totale di 33 SP nel quarto sprint, evidenziando un miglioramento continuo nella capacità di completamento.

L'andamento quasi sovrapposto tra le due linee evidenzia l'efficienza del team nel mantenere il ritmo pianificato e vicino all'andamento ideale, con scostamenti minimi dovuti a complessità tecniche o allineamenti funzionali, mostrando un'ottima prevedibilità del processo di sviluppo e gestione del carico di lavoro.

Il ritmo di lavoro del team molto vicino all'ideale, che mostra un miglioramento graduale della produttività, è coerente con i dati del Velocity Chart, mostrato nel paragrafo 3.5.3. (da 75 a 92 SP/sprint).

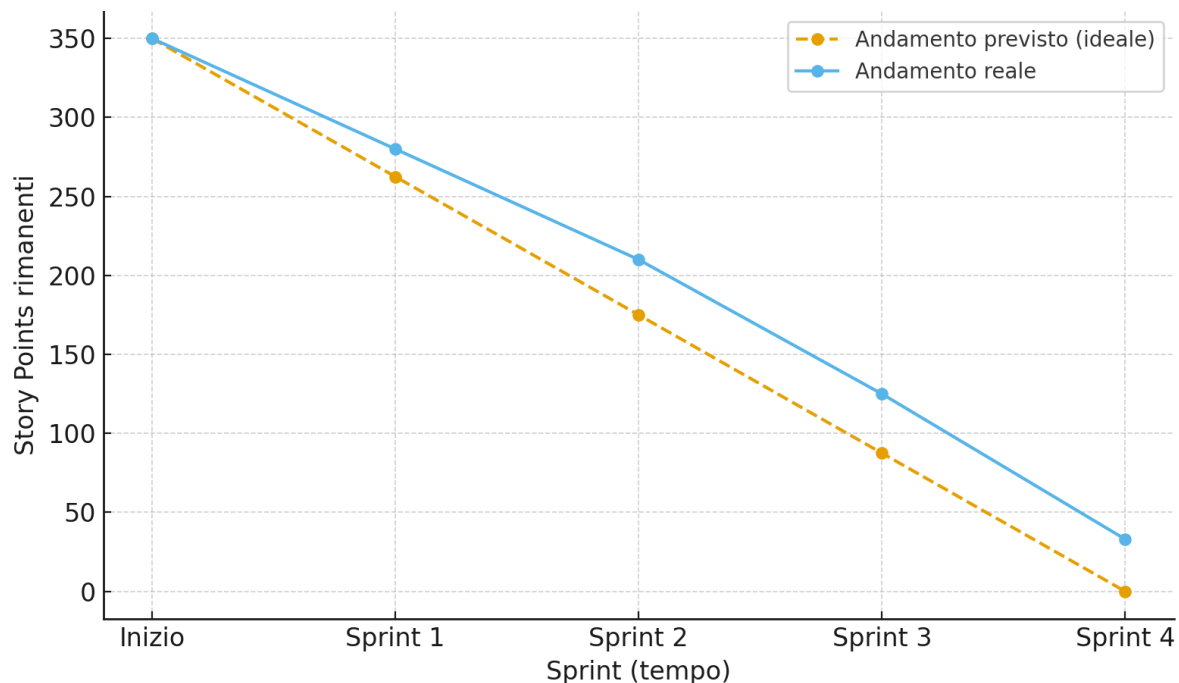


Figura 14: Burndown Chart

### 3.5.2 Burnup chart

Il Burnup Chart ha l'andamento opposto del Burndown, infatti è una cumulata delle attività, cioè rappresenta il progresso cumulativo del lavoro completato e, quindi, il lavoro effettivamente svolto.

Sull'asse verticale mostra il lavoro complessivo in Story Points e sull'asse orizzontale il tempo (Sprint).

Le due linee presenti nel grafico indicano:

- il totale dello scope di progetto (linea arancione tratteggiata);

- il lavoro completato cumulativamente (linea blu continua).

Nel periodo analizzato, lo scope complessivo cresce leggermente da 355 a 368 SP per effetto dell'inserimento di alcune attività minori, modifiche dei requisiti e all'introduzione di nuove funzionalità (fenomeno di scope creep). Infatti, lo scope creep si verifica quando vengono aggiunti requisiti non previsti in fase avanzata del progetto, come ad esempio nuove feature richieste dopo aver definito i requisiti iniziali. Un incremento dell'ambito può portare a ritardi nelle consegne, costi sovrastimati, blocchi e problemi di gestione delle risorse.

Per monitorare e gestire questi cambiamenti, i grafici burn-up sono strumenti particolarmente utili: permettono di visualizzare le variazioni di ambito e di valutare il loro impatto sul progresso complessivo, aiutando a intervenire modificando o riducendo il carico di lavoro e a ridistribuire le risorse, mantenendo sotto controllo eventuali ritardi e garantendo una gestione più efficace del progetto.

Nel grafico lo scope creep è visibile nel leggero scostamento tra la curva ideale e la curva reale dopo lo sprint 2, ma non ha impattato negativamente l'andamento complessivo, grazie alla crescita costante della velocity. Infatti, gli Story Points completati cumulativi aumentano progressivamente fino ad arrivare a 335 nello sprint 4: alla fine dello sprint 4 vi è un quasi completo allineamento tra le due linee, questo è indice del fatto che il team riesce a completare circa 91% dello scope pianificato, con una progressione costante e sostenuta.

Questo andamento evidenzia un'elevata efficienza del team e una buona capacità di adattamento e gestione degli imprevisti, nonostante l'aumento di complessità dovuto all'introduzione di nuove funzionalità. Il grafico dimostra che, pur con l'aumento del carico di lavoro, il team è riuscito a mantenere un ritmo sostenuto e a ridurre il divario tra completato e desiderato.



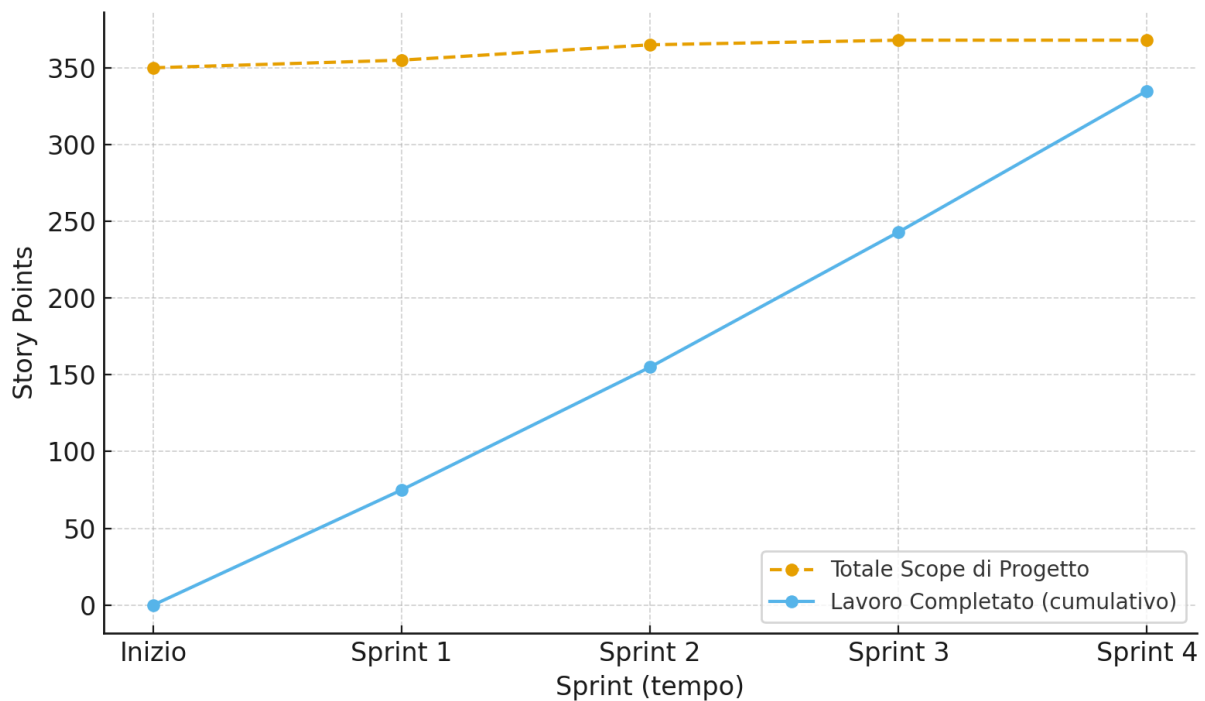


Figura 15: Burnup Chart

### 3.5.3 Velocity Chart

Il Velocity Chart misura la quantità media di lavoro completato in ciascuno sprint, espresso in Story Points.

Si tratta di un indicatore che mostra l'andamento delle performance del team, quindi è fondamentale per valutare la capacità del team di mantenere un ritmo stabile nel tempo e per stimare il volume di lavoro di cui un team può farsi carico per gli sprint futuri.

Nel periodo analizzato, la velocity risulta crescente: nel primo sprint il team ha completato 75 SP, mantenendo una buona aderenza alla pianificazione. Gli sprint successivi mostrano una progressiva crescita della capacità produttiva, con una media di 83,8 SP completati per sprint. Quest'ultimo rappresenta un valore stabile, che costituisce un riferimento utile per stimare la capacità futura del team nei successivi rilasci.

In particolare, la crescita della velocity indica che il team ha progressivamente migliorato la propria produttività, grazie ad un miglior coordinamento interno e un affinamento della stima delle User Stories.

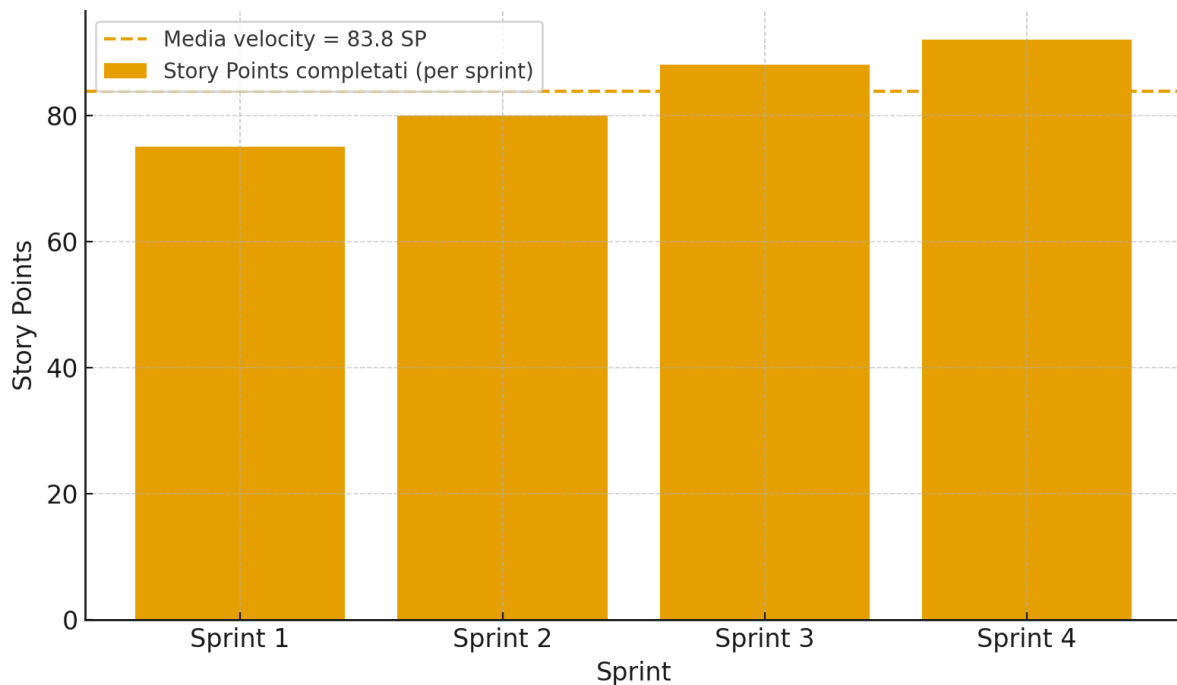


Figura 16: Velocity Chart

### 3.5.4 Cumulative Flow Diagram (CFD)

Il Cumulative Flow Diagram (CFD) fornisce una visione complessiva dello stato di avanzamento del lavoro nel tempo, suddiviso per stato:

- To Do, ossia le attività ancora da avviare
- In Progress, le attività in lavorazione
- Done, le attività completate.

Si usano dei colori differenti per indicare i vari stati del flusso di lavoro, in questo modo si evidenziano i colli di bottiglia. Il CFD, quindi, consente di avere rapidamente una visione di quanto sta accadendo nel proprio team e anticipare azioni correttive a supporto.

Nel grafico si nota una riduzione costante dell'area "To Do" (da 350 SP iniziali a 33 SP finali), una fascia "In Progress" stabile che oscilla tra 30 e 45 SP, e una crescita regolare della fascia "Done" (da 0 a 335 SP).

Questo andamento dimostra che il team ha mantenuto un flusso di lavoro equilibrato, con un Work In Progress (WIP) controllato e una consegna continua di valore incrementale, coerente con i principi Agile.

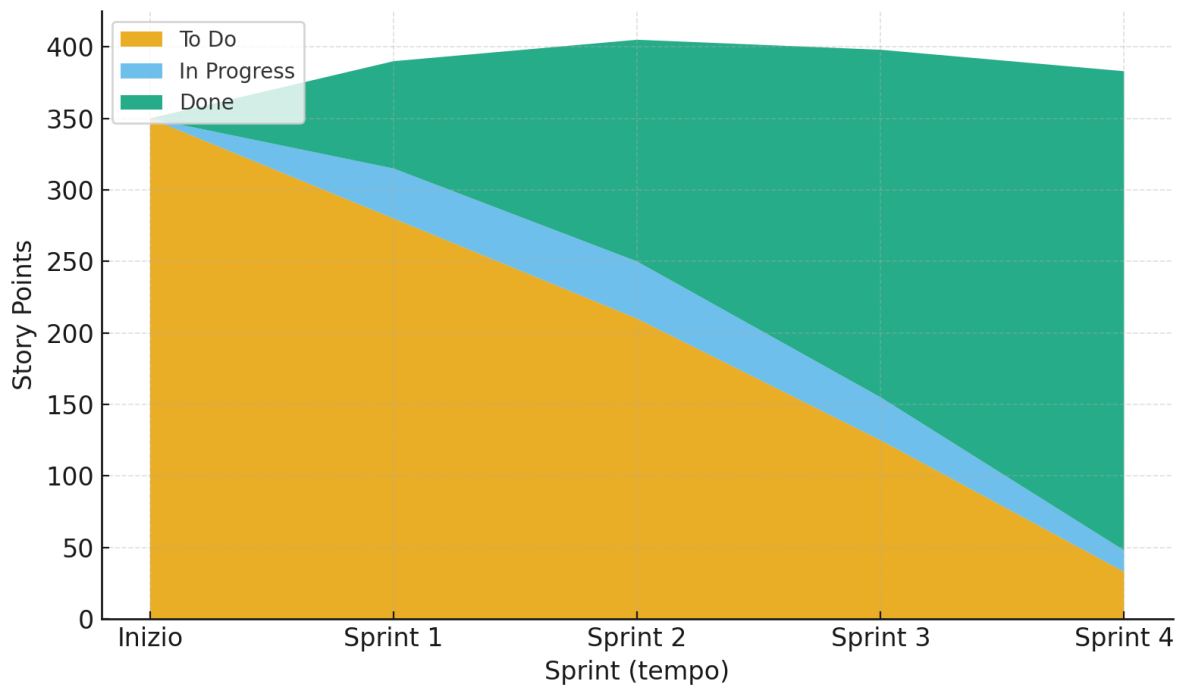


Figura 17: Cumulative Flow Diagram

L'insieme dei quattro grafici fornisce un quadro coerente dell'andamento del progetto durante il trimestre analizzato.

Il team ha mostrato un'evoluzione positiva nella gestione del carico di lavoro, migliorando la propria efficienza e la qualità del flusso produttivo. Ha saputo pianificare, stimare e completare il lavoro con continuità, riducendo progressivamente il backlog e migliorando la propria velocity sprint dopo sprint.

In particolare, il Burndown e il Burnup Chart testimoniano un'elevata affidabilità delle stime e una capacità costante di raggiungere gli obiettivi prefissati; il Velocity Chart dimostra un incremento misurabile della produttività; infine, il Cumulative Flow Diagram evidenzia un flusso continuo e regolare, con un equilibrio efficace tra lavoro in corso e completato.

### 3.6 Definizione dei Key Performance Indicators (KPI)

In un progetto complesso come quello oggetto di questa tesi, l'utilizzo di indicatori chiave di performance (Key Performance Indicators, o KPI) consente di monitorare in modo

sistematico non solo l'avanzamento delle attività, ma anche la capacità del team di generare valore reale per il business.

Si tratta di misure che consentono di valutare il progresso verso uno specifico obiettivo strategico o operativo, garantendo trasparenza e supportando decisioni basate su dati oggettivi.

Per essere realmente efficaci, i KPI devono rispettare i criteri SMART, un approccio consolidato nella definizione di obiettivi e metriche di performance.

Secondo Tableau [34], i KPI sono considerati SMART quando rispettano le seguenti condizioni:

- **Specific** (Specifico): il KPI deve essere chiaramente definito e mirare a un obiettivo preciso, evitando ambiguità o interpretazioni soggettive
- **Measurable** (Misurabile): deve poter essere quantificato attraverso dati oggettivi, in modo da valutare in modo concreto i progressi
- **Achievable** (Raggiungibile): deve essere realistico e realizzabile in base alle risorse, alle tempistiche e alle capacità del team
- **Relevant** (Rilevante): deve contribuire direttamente al raggiungimento degli obiettivi strategici dell'organizzazione o del progetto
- **Time-bound** (Temporale): deve essere collocato entro un periodo di tempo definito, per permetterne la verifica e la valutazione periodica.

In conclusione, l'applicazione dei criteri SMART consente di strutturare un sistema di misurazione più solido e orientato ai risultati. [33]

Nell'ambito del progetto analizzato, l'insieme dei KPI è strutturato in due macro-categorie:

- **KPI quantitativi**: misurano l'efficienza operativa e temporale del team, quindi quanto velocemente e aderentemente alle previsioni viene consegnato il lavoro
- **KPI qualitativi**: valutano aspetti legati alla collaborazione, alla soddisfazione degli stakeholder e alla flessibilità del team nel rispondere ai cambiamenti.

Questo capitolo di definizione dei KPI funge da base metodologica per il Capitolo 4, nel quale verranno presentati e analizzati i dati empirici raccolti durante l'esperienza di tirocinio.

Attraverso le metriche definite, sarà possibile valutare l'impatto dell'adozione della metodologia Agile nel contesto assicurativo e sviluppare raccomandazioni per future implementazioni.

## 4 Analisi dei KPI

### 4.1 Applicazione dei KPI al progetto

Dopo aver fornito nel capitolo precedente la definizione ed aver definito i criteri di selezione dei Key Performance Indicators (KPI), questo capitolo si propone di applicare concretamente tali indicatori al progetto assicurativo internazionale analizzato, con l'obiettivo di misurare in modo oggettivo l'efficacia dell'approccio Agile adottato.

L'utilizzo dei KPI ha consentito di monitorare le performance del team in termini sia quantitativi che qualitativi, fornendo un quadro completo dell'andamento del progetto e permettendo di valutare l'impatto dell'introduzione di Agile sulla gestione operativa, sui tempi di consegna e sulla collaborazione tra i diversi stakeholder.

L'analisi è stata condotta attraverso un duplice approccio:

1. una **valutazione quantitativa**, volta a misurare le dimensioni oggettive del progetto, come l'efficienza temporale, l'aderenza al piano e la qualità tecnica dei rilasci
2. una **valutazione qualitativa**, finalizzata a comprendere gli aspetti relazionali, gestionali e di collaborazione che hanno influenzato il raggiungimento degli obiettivi prefissati.

I KPI selezionati rispondono a criteri di misurabilità, rilevanza e coerenza con gli obiettivi strategici del progetto. Essi consentono di rappresentare con chiarezza lo stato di avanzamento dei lavori, evidenziando sia le aree di eccellenza sia i punti di miglioramento.

Questa struttura di analisi riflette l'impostazione metodologica tipica dei progetti Agile, in cui il monitoraggio continuo e la trasparenza dei dati costituiscono elementi fondamentali per guidare le decisioni e favorire un miglioramento costante delle performance.

#### 4.1.1 Analisi quantitativa

La prima parte dell'analisi dei KPI si concentra sugli indicatori quantitativi, ossia su quelle metriche che permettono di misurare in modo oggettivo la produttività e l'efficienza del team.

Nel contesto Agile, queste metriche assumono un ruolo particolarmente rilevante poiché consentono di comprendere quanto il team riesca a rispettare gli obiettivi pianificati, a mantenere un ritmo di lavoro sostenibile e a garantire la qualità tecnica del prodotto finale.

L'analisi quantitativa è organizzata in due dimensioni principali:

- **Efficienza temporale e aderenza al piano**, che considera indicatori come il Sprint Goal Success Rate (SGSR), il Lead Time e il Sprint Time Accuracy Index (STAI)
- **Qualità tecnica e testing**, che comprende gli indicatori legati all'attività di test e controllo della qualità del software, quali l'UAT Test Success (UTS), il Test Pass Rate (TPR) e il Completamento UAT.

L'obiettivo di questa analisi è duplice, poiché da un lato consente di misurare la capacità del team di rispettare tempi e pianificazioni, dall'altro permette di verificare la solidità tecnica delle soluzioni rilasciate.

Questi indicatori, analizzati nel periodo di osservazione (aprile–giugno), consentono di valutare in modo oggettivo l'efficacia dell'approccio Agile nel garantire continuità e prevedibilità ai rilasci, aspetti fondamentali in un contesto assicurativo caratterizzato da complessità tecnica e forte interdipendenza tra moduli applicativi.

#### **4.1.1.1 Efficienza temporale e aderenza al piano**

Uno degli obiettivi centrali del framework Agile è quello di assicurare un flusso di lavoro costante e prevedibile. Per questo motivo, la valutazione dell'efficienza temporale e dell'aderenza al piano rappresenta un indicatore chiave.

In un progetto articolato in più verticali funzionali e caratterizzato da interdipendenze complesse, come quello in oggetto, il rispetto delle scadenze e la capacità di pianificare in modo accurato gli sprint diventano elementi determinanti per il successo complessivo.

Di seguito sono analizzati i KPI analizzati relativi al monitoraggio dell'efficienza temporale e aderenza al piano.

### Sprint Goal Success Rate (SGSR)

Il primo indicatore considerato è lo Sprint Goal Success Rate, che misura la percentuale di obiettivi di sprint effettivamente raggiunti rispetto a quelli pianificati, calcolato secondo la formula:

$$SGSR = \frac{NOR}{NTS} \times 100$$

dove:

- **NOR** (Number of Objectives Reached) è il numero di sprint in cui gli obiettivi stabiliti sono stati raggiunti
- **NTS** (Number of Total Sprints) è il numero totale di sprint analizzati.

Nel progetto in analisi, uno sprint goal è raggiunto se almeno il 90% degli Story Points pianificati viene completato e se la User Story principale associata allo Sprint Goal è conclusa e rilasciata.

Per ognuno dei quattro sprint analizzati, è stata quindi monitorata la quantità di Story Points pianificati e di Story Points effettivamente completati, come riportato in tabella 5:

Sprint	SP pianificati	SP completati	% completamento
S1	80	75	93,8
S2	85	80	94,1
S3	90	88	97,8
S4	95	92	96,8

Tabella 5: Tasso di completamento delle attività per sprint

Applicando la formula:

$$SGSR = \frac{4}{4} \times 100 = 100\%$$

si ottiene un SGSR pari al 100%, che indica che tutti e quattro gli sprint hanno raggiunto l'obiettivo previsto.

Infatti, un valore elevato di SGSR indica un buon livello di pianificazione, di coordinamento e di affidabilità del team.

Di conseguenza, nel periodo analizzato il valore del SGSR (maggiore del 90%) in ciascuno sprint evidenzia che la stima degli Story Points è stata accurata, e che la collaborazione tra Product Owner e team di sviluppo è stata efficace nella definizione di backlog realistici.



## Lead Time (LT)

Il Lead Time rappresenta il tempo complessivo necessario per completare un processo o un rilascio, calcolato dal momento in cui viene formulata una richiesta fino al momento in cui il risultato è effettivamente consegnato e validato. Tale indicatore, che comprende sia le fasi operative (gli sprint), che le attività preliminari di pianificazione e allineamento che precedono l'avvio dello sviluppo, consente quindi di misurare la reattività del team.

Nel progetto assicurativo analizzato, il Lead Time è stato calcolato considerando:

- Settimane di setup iniziale, pari a 3, dedicate alla pianificazione, alla definizione del backlog e alla predisposizione degli ambienti di test e integrazione
- Le settimane operative relative ai quattro sprint effettivi (ognuno della durata di tre settimane, quindi 12 settimane complessive)
- Un margine di sicurezza, previsto di 2 settimane, legato a possibili ritardi dovuti a dipendenze inter-sistema o allineamenti esterni (es. ritardi nelle interfacce PDM o nella validazione dei test UAT da parte degli stakeholder).

Di conseguenza:  $\text{Lead Time previsto} = 3 + 12 + 2 = 17$  settimane

Al termine del periodo osservato, però, il Lead Time effettivo, pari a 16 settimane, è stato quindi inferiore di una settimana rispetto al previsto, con una riduzione del 6% sui tempi di completamento complessivi. Questo dato evidenzia un miglioramento del coordinamento inter-team e una più efficace gestione dei test e delle fasi di rilascio.

Infatti, il guadagno temporale si è concentrato nella fase di setup iniziale, grazie a un'ottimizzazione delle attività di setup, completate in anticipo rispetto al piano, grazie alla stretta collaborazione tra Scrum Master, Product Owner e architetti tecnici.

Il confronto tra Lead Time previsto ed effettivo mostra, inoltre, un miglioramento della capacità previsionale del team: la pianificazione iniziale, pur includendo margini prudenziali, si è rivelata affidabile, segno che il processo di stima basato sugli story points e sulla velocity era ben calibrato.

In termini gestionali, ciò si traduce in una maggiore affidabilità delle roadmap trimestrali e in una riduzione del rischio progettuale, elementi fondamentali in contesti assicurativi complessi dove ogni ritardo può impattare le fasi successive di integrazione e testing.

### **Sprint Time Accuracy Index (STAI)**

Infine, lo Sprint Time Accuracy Index valuta l'accuratezza con cui i team rispettano la durata pianificata dello sprint. La formula di riferimento è la seguente:

$$STAI = \frac{ES}{ED}$$

dove:

- **ES**= durata effettiva dello sprint
- **ED**= durata programmata

Nel progetto in esame, tutti e quattro gli sprint sono stati completati nel rispetto del timebox di 3 settimane (tutti gli sprint durano 3 settimane come da piano):

$$STAI = \frac{3}{3} = 1$$

Un valore di STAI maggiore di 1 indicherebbe sprint prolungati oltre la durata prevista, sintomo di possibili criticità nella pianificazione iniziale, carichi di lavoro eccessivi, indisponibilità di risorse o interferenze esterne (es. ritardi nel rilascio di ambienti o permessi). Viceversa, un valore minore di 1 suggerirebbe sprint conclusi in anticipo, potenzialmente a causa di una pianificazione troppo conservativa, di una stima eccessiva degli Story Points, o di una momentanea riduzione del carico operativo.

Invece, il valore di STAI ottenuto, pari ad 1, indica che i tempi pianificati sono stati rispettati, senza ritardi o prolungamenti rispetto alla calendarizzazione originale. Ciò evidenzia una pianificazione temporale accurata, un'elevata capacità di gestione del tempo da parte del team e di coordinamento tra i verticali.

Questo aspetto è particolarmente significativo in un contesto multi-team come quello analizzato, in cui le dipendenze tra moduli diversi possono facilmente generare ritardi se non opportunamente monitorate.

L'aderenza temporale osservata è anche coerente con i risultati dei grafici analizzati nel capitolo 3.5: il Cumulative Flow Diagram mostra un flusso stabile e privo di accumuli significativi di "work in progress" e il Burndown Chart evidenzia una progressiva riduzione del backlog in linea con l'andamento ideale.

In sintesi, gli indicatori di efficienza temporale e aderenza al piano confermano che l'applicazione della metodologia Agile ha favorito una pianificazione più accurata, un controllo continuo dei tempi di sviluppo e una maggiore prevedibilità dei rilasci.

#### 4.1.1.2 Qualità tecnica e testing

Un altro aspetto fondamentale per valutare l'efficacia della metodologia Agile adottata nel progetto riguarda la qualità tecnica del prodotto rilasciato e l'efficienza dei processi di testing.

La filosofia Agile si fonda sull'idea che la qualità non debba essere controllata alla fine del ciclo di sviluppo, ma monitorata costantemente durante tutto il processo, grazie a un approccio iterativo e integrato in cui sviluppo, test e validazione procedono di pari passo.

A tal fine, sono stati presi in esame tre indicatori principali, che insieme forniscono una visione complessiva dello stato qualitativo delle componenti software e della capacità del team di rilasciare soluzioni affidabili e conformi agli standard.

##### Code Quality Assurance (o Unit Test Success UTS)

Il Code Quality Assurance rappresenta una misura della copertura, qualità e adeguatezza dei test automatizzati applicati al codice sviluppato.

La qualità del codice è uno degli elementi più rilevanti in un progetto Agile, poiché determina la stabilità e la velocità di rilascio del software. In un contesto iterativo come quello analizzato, dove ad ogni sprint vengono introdotte nuove funzionalità e modifiche, la presenza di test automatizzati accurati consente di garantire una continuità di integrazione.

L'indicatore UTS viene calcolato secondo la seguente formula:

$$UTS = \left( \frac{CC \times TU}{CT} \right) \times 100$$

dove:

- **Copertura del codice (CC):** indica la percentuale di codice effettivamente eseguita dai test automatici. In pratica, quando i test vengono eseguiti, strumenti come SonarQube o JUnit Coverage analizzano il codice e verificano quali righe, funzioni o condizioni logiche vengono "toccate" dai test.

Un valore elevato ( $\geq 80\%$ ) significa che quasi tutto il codice è stato testato e, di conseguenza, il rischio di bug non rilevati è più basso, con conseguente maggiore affidabilità del sistema.

- **Adeguatezza della suite di test (TU):** misura quanto la quantità e la varietà dei test siano proporzionate alla complessità del codice. In altre parole:
  - se il codice è semplice, bastano pochi test
  - se il codice è complesso (molte condizioni, integrazioni, eccezioni), serve una suite di test più ampia e articolata.

Un valore superiore a 1 indica una suite ben strutturata (copre casi limite e scenari complessi); un valore inferiore a 1 indica una suite carente o incompleta.

- **Complessità del codice (CT):** rappresenta la complessità media delle funzioni o moduli. Un valore alto implica maggiore rischio di bug e quindi richiede test più estesi.

Nel periodo di osservazione (da aprile a giugno), i valori calcolati per i quattro sprint sono i seguenti:

Sprint	Copertura codice (CC)	Complessità (CT)	Adeguatezza test (TU)	UTS (%)
S1	74%	1	0,98	72,5
S2	77%	1	1	77
S3	80%	1	1,02	81,6
S4	81%	1	1,01	81,8

Tabella 6: Metriche di copertura, complessità e test per sprint

E' evidente che il valore di UTS cresce costantemente passando da 72,5 % nel primo sprint fino a 81,8 % nel quarto, grazie a:

- Aumento di copertura del codice (CC): la % di codice testato è cresciuta dal 74% all'81%, con una media pari al 78%, segnale di un processo di test più completo
- Irrobustimento della suite di test (TU): il valore dell'adeguatezza è cresciuto fino a superare 1 negli ultimi due sprint, indicando una suite di test leggermente sovradimensionata rispetto alla complessità del codice. Questo è un segnale positivo, poiché evidenzia un approccio preventivo e orientato alla qualità

- **Stabilità architetturale (CT):** la costanza della complessità del codice (CT = 1) suggerisce che la base architetturale è rimasta stabile, senza variazioni strutturali significative.

La crescita costante dell'indice UTS evidenzia quindi una progressiva maturità del team di sviluppo nella gestione della qualità del software. Tale valore si riflette, di conseguenza, anche sul miglioramento della fiducia del team e degli stakeholder nei confronti delle funzionalità rilasciate, poiché la qualità del codice è diventata un parametro oggettivamente monitorato e condiviso.

In prospettiva, mantenere l'UTS in un range compreso tra 75% e 85% rappresenta un equilibrio ottimale tra qualità e costi di test, poiché livelli di copertura troppo elevati (>90%) potrebbero introdurre un effort sproporzionato rispetto ai benefici effettivi.

L'analisi dell'indicatore conferma quindi che la strategia di testing adottata ha garantito una qualità del codice elevata e sostenibile, pienamente coerente con i principi di efficienza e miglioramento continuo propri della metodologia Agile.

### **Test Pass Rate (TPR)**

Il Test Pass Rate (TPR) misura la percentuale di test superati con successo sul totale dei test eseguiti e si calcola secondo la seguente formula:

$$TPR = \frac{NTS}{NTE} \times 100$$

dove:

- **NTS (Number of Tests Successful)** è il numero di test superati con esito positivo
- **NTE (Number of Tests Executed)** è il numero totale di test eseguiti.

Questo indicatore consente di valutare la stabilità funzionale e la qualità complessiva del software, fornendo una misura diretta dell'efficacia del processo di validazione.

Nel progetto analizzato sono stati pianificati complessivamente 268 test, di cui 260 effettivamente eseguiti e 244 superati con esito positivo, per un TPR medio pari al 93,8%.

La distribuzione dei volumi di test per sprint è riportata nella tabella 7:

<b>Sprint</b>	<b>Test pianificati</b>	<b>Test eseguiti</b>	<b>Test superati</b>	<b>TPR (%)</b>
S1	62	60	55	91,7
S2	66	64	60	93,80
S3	70	68	64	94,1
S4	70	68	65	95,6
<b>Totale</b>	<b>268</b>	<b>260</b>	<b>244</b>	<b>93,8% (media)</b>

*Tabella 7: Risultati dei test per sprint*

L'indicatore mostra un miglioramento costante nel corso del progetto: da 91,7% del primo sprint al 95,6% nell'ultimo, segno di una crescente efficacia del processo di test.

Un Test Pass Rate elevato (superiore al 90%), come in questo caso, indica un buon livello di qualità del software e una stabilità funzionale crescente del sistema. Ciò significa che la maggior parte delle funzionalità testate ha superato con successo le verifiche di validazione, dimostrando che il codice rilasciato è affidabile e ben integrato con gli altri componenti applicativi.

In un contesto complesso come quello assicurativo, caratterizzato da forti interdipendenze tra sistemi (Motor, Non-Motor, Pagamenti e piattaforma documentale PDM), tale valore rappresenta una soglia di eccellenza.

Un valore alto di TPR può derivare da:

- Pianificazione accurata delle attività di test
- Efficace collaborazione tra team di sviluppo e QA (Quality Assurance)
- Corretta gestione dei difetti, risolti rapidamente all'interno dello sprint stesso.

Al contrario, un Test Pass Rate basso indicherebbe la presenza di problemi di qualità del codice, instabilità del sistema, o inefficienze nel processo di testing.

In questi casi, il team dovrebbe analizzare le cause dei fallimenti e adottare azioni correttive mirate.

In sintesi, il Test Pass Rate conferma l'efficacia del processo di validazione e la capacità del team di garantire rilasci affidabili e coerenti con le aspettative di business.

### **Completamento UAT (TC)**

Il Completamento UAT (TC) rappresenta un indicatore fondamentale per la valutazione della qualità e dell'affidabilità del software rilasciato, poiché misura la percentuale di casi di User Acceptance Test (UAT) completati con esito positivo rispetto al totale pianificato per il

trimestre. In particolare, gli User Acceptance Test (UAT) sono test eseguiti con il coinvolgimento del business per confermare che le funzionalità rilasciate soddisfino i requisiti.

La formula utilizzata è:

$$TC = \frac{CC}{CP} \times 100$$

dove:

- **CC** indica i casi UAT superati
- **CP** i casi complessivamente pianificati.

Quindi questo KPI consente di verificare il livello di accettazione delle funzionalità da parte del business, fornendo una misura concreta del grado di allineamento tra sviluppo tecnico e requisiti utente.

Nel progetto analizzato, su un totale di 50 casi UAT pianificati, 46 sono stati completati con successo, con un tasso di completamento pari al 92%, come è evidente nella seguente tabella:

<b>Sprint</b>	<b>Funzionalità rilasciate</b>	<b>Casi UAT pianificati (CP)</b>	<b>Casi UAT superati (CC)</b>	<b>% Successo</b>
S2	Motor + prime integrazioni PDM	10	9	90
S3	Workflow + Pagamenti + Notifiche	18	16	89
S4	Reportistica + stabilizzazione + Portale partner	22	21	95
<b>Totale</b>	-	<b>50</b>	<b>46</b>	<b>92(media)</b>

*Tabella 8: Risultati dei test UAT per sprint*

Tale risultato evidenzia un'elevata qualità del prodotto e una gestione efficace della fase di validazione. Inoltre, il valore elevato dell'indice dimostra un forte allineamento tra team tecnico e business.

E' importante specificare che gli UAT si svolgono solo sugli sprint in cui vengono rilasciate funzionalità business complete, cioè dal secondo sprint in poi (quando il software è già utilizzabile).

La distribuzione dei test ha previsto 10 casi nel secondo sprint, 18 nel terzo e 22 nel quarto, con percentuali di successo rispettivamente pari al 90%, 89% e 95%. Tali risultati dimostrano la progressiva stabilizzazione del sistema e la crescente maturità del team nella gestione della qualità e del rischio. In particolare, il picco del 95% nello Sprint 4 riflette la capacità del team di garantire un prodotto sempre più solido, grazie all'esecuzione regolare di demo di fine sprint e al coinvolgimento diretto degli stakeholder nelle attività di accettazione.

In contesti assicurativi complessi, questo approccio iterativo e collaborativo consente di ridurre il rischio di rilascio, garantendo che ogni funzionalità venga validata in modo incrementale e che le decisioni di go-live siano basate su evidenze oggettive. Il valore di TC ottenuto, superiore al 90%, rappresenta quindi una soglia di eccellenza, poiché conferma la solidità del processo di validazione e l'efficacia dell'approccio Agile adottato, che ha permesso di rilasciare soluzioni affidabili, validate progressivamente e pienamente rispondenti alle esigenze operative e strategiche del cliente.

In conclusione, l'analisi dei KPI di qualità tecnica e testing evidenzia che il progetto ha mantenuto un'elevata qualità tecnica, con un trend positivo in tutti gli sprint. Inoltre, il superamento del 90% dei test UAT dimostra come l'approccio Agile, basato su test continui e cicli brevi di feedback, garantisca risultati concreti anche in contesti complessi come quello assicurativo.

## **4.1.2 Analisi qualitativa**

Dopo aver analizzato nel paragrafo precedente le dimensioni quantitative legate alla performance del team e alla qualità tecnica del prodotto, la presente sezione approfondisce gli aspetti qualitativi del progetto, con particolare riferimento ai fattori organizzativi, comunicativi e relazionali che hanno inciso sul raggiungimento degli obiettivi.

In un contesto Agile, i risultati non dipendono soltanto dall'efficienza dei processi o dal rispetto dei tempi di rilascio, ma anche dalla capacità del team di adattarsi al cambiamento, di mantenere una collaborazione costante con gli stakeholder e di assicurare una gestione trasparente dei requisiti.

L'analisi qualitativa si basa dunque su due categorie di indicatori:



- **Gestione dei requisiti e la flessibilità nella risposta al cambiamento**, rappresentati dai KPI Requirements Accuracy Rate (RAR), Project Change Control Ratio (PCCR) e Average Project Change Time (APCT)
- **Collaborazione, comunicazione e coinvolgimento degli stakeholder**, sintetizzati dai KPI Stakeholder Satisfaction Score (SSS) e Meeting Regularity Index (MRI).

Questi indicatori forniscono una prospettiva complementare ai KPI quantitativi, permettendo di comprendere come il metodo di lavoro Agile abbia influenzato la qualità del processo decisionale, la sinergia tra i team e la reattività dell'organizzazione di progetto.

#### 4.1.2.1 Gestione dei requisiti e flessibilità

Uno dei principi cardine della metodologia Agile è la capacità di rispondere efficacemente al cambiamento, bilanciando la stabilità del piano con la flessibilità nella gestione dei requisiti.

In un progetto complesso come quello assicurativo analizzato, caratterizzato da continue evoluzioni regolamentari, la tracciabilità e la gestione dinamica delle modifiche rappresentano elementi essenziali di successo.

A tale scopo, sono stati utilizzati tre indicatori chiave, analizzati di seguito.

##### **Requirements Accuracy Rate (RAR)**

Il Requirements Accuracy Rate (RAR) misura la qualità dei requisiti in ingresso agli sprint, ossia quanto i requisiti risultano chiari, completi e comprensibili al team di sviluppo. Quindi è un indice che non misura la quantità di lavoro, ma la qualità della preparazione del lavoro. In altre parole:

- Ogni requisito deve essere chiaro, completo e privo di ambiguità
- Se un requisito è incompleto o impreciso, il team perde tempo durante lo sprint per chiarirlo, causando ritardi o errori nel Burndown.

Per calcolare tale indice, si rapporta la quota di requisiti accurati e completi (**NRA**) rispetto al totale (**NTR**) trattato nel periodo. I requisiti accurati (NRA) sono quelli che non hanno richiesto modifiche o chiarimenti durante lo sprint (quindi chiari fin da subito).

Nel trimestre analizzato, in particolare, su 120 requisiti totali, 111 sono risultati accurati e completi:

$$RAR = \frac{111}{120} \times 100 = 92,5\%$$

In generale, più è alto il RAR, più lo sprint è fluido e prevedibile: il valore maggiore del 90% conferma la chiarezza delle User Stories, infatti indica che la maggior parte dei requisiti è stata definita bene già prima di iniziare lo sprint. Questo è positivo perché evita che il team perda tempo a chiarire i dettagli durante lo sviluppo.

Analizzando nel dettaglio i dati nel corso dei quattro sprint:

<b>Sprint</b>	<b>Requisiti trattati (NTR)</b>	<b>Requisiti accurati (NRA)</b>	<b>RAR sprint</b>
S1	38	35	92,1%
S2	32	30	93,8%
S3	28	26	92,9%
S4	22	20	90,9%
<b>Totale</b>	<b>120</b>	<b>111</b>	<b>92,5% (media)</b>

*Tabella 9: Andamento del RAR per sprint*

Nello sprint 4 si osserva un leggero calo del RAR: questo è realistico, considerando che negli sprint finali sono emersi requisiti nuovi e più complessi (es. reportistica che richiede una struttura dati precisa), dove è più facile avere incomprensioni. Questi tipi di attività generano spesso requisiti meno chiari all'inizio. Tuttavia, il valore del RAR resta comunque molto buono (sopra il 90%).

Nel complesso, il team ha dimostrato una solida capacità di analisi dei requisiti che ha consentito di ridurre i rischi di rielaborazione e di mantenere una pianificazione stabile.

Questo è coerente con l'andamento regolare dei grafici Burndown e Burnup, che non mostrano fluttuazioni anomale, e con la crescita costante della Velocity nel corso degli sprint.

### **Project Change Control Ratio (PCCR)**

Il Project Change Control Ratio (PCCR) misura il rapporto tra le modifiche non formalizzate (**UC**) e quelle gestite tramite processo ufficiale di Change Control (**TC**). In altre parole, è un indicatore che misura la capacità del team di gestire in modo controllato le modifiche ai requisiti o allo scope.

Come è evidente in tabella 10, nel corso dei quattro sprint sono state rilevate 18 modifiche complessive ai requisiti o allo scope del progetto, per un totale di 18 Story Points aggiunti al backlog. Di queste, 15 sono state gestite formalmente attraverso il processo di Change Control (TC), con valutazione di effort, impatto tecnico e priorità, 3 modifiche (UC) sono invece state introdotte in modo informale, a seguito di micro-aggiustamenti operativi.

Sprint	Modifiche totali	Formalizzate (TC)	Non formalizzate (UC)	Incremento di scope (SP)	Scope totale (fine sprint)
S1	4	3	1	5	355
S2	6	5	1	10	365
S3	5	4	1	3	368
S4	3	3	0	0	368
<b>Totale</b>	<b>18</b>	<b>15</b>	<b>3</b>	<b>18</b>	<b>-</b>

Tabella 10: Andamento delle modifiche per sprint

Le variazioni si concentrano nei primi due sprint (55% del totale), quando il progetto ha affrontato la fase di integrazione tra i moduli Motor, Non-Motor e Pagamenti, caratterizzate da un'elevata complessità tecnica, dove era "normale" un certo grado di aggiustamento dei requisiti. Tuttavia, la successiva stabilizzazione dei cambiamenti e l'azzeramento dell'incremento di scope nello Sprint 4 dimostrano che il backlog è diventato più stabile e i requisiti sono stati consolidati.

Applicando la formula:

$$PCCR = \frac{UC}{TC} = \frac{3}{15} = 0,2$$

si ottiene un indice pari a 0,2, che indica una bassa incidenza di cambiamenti non controllati rispetto a quelli formalizzati. Infatti, un valore basso del rapporto, come in questo caso, suggerisce che il progetto è stabile e ben gestito, e che le modifiche sono state controllate, approvate e tracciate in modo coerente con il piano di lavoro, riducendo il rischio di scope creep e garantendo coerenza tra la pianificazione e l'avanzamento effettivo.

Questo riflette, quindi, la capacità del team di adattarsi alle nuove esigenze senza compromettere i tempi di consegna, la qualità del prodotto o la coerenza del backlog.

### Average Project Change Time (APCT)

Il KPI Average Project Change Time (APCT) misura la rapidità con cui i cambiamenti vengono analizzati, approvati e implementati, rispetto ai tempi standard stabiliti (SLA - Service Level

Agreement). Cioè misura il tempo che intercorre tra la richiesta di modifica (nuovo requisito o variazione di priorità) e la sua implementazione completa (rilascio in ambiente di test o produzione).

In un contesto assicurativo, dove i processi di business coinvolgono sistemi complessi e fortemente integrati, la gestione del cambiamento deve essere bilanciata: rapida, per rispondere alle esigenze emergenti, ma anche controllata, per garantire stabilità applicativa e coerenza tra moduli.

Il valore dell'indicatore riflette dunque il grado di maturità del team nel trasformare le richieste di modifica in azioni concrete, senza rallentare il flusso di sviluppo.

L'analisi dei dati per sprint evidenzia un trend di miglioramento costante:

<b>Sprint</b>	<b>Cambiamenti applicati</b>	<b>Entro SLA</b>	<b>Oltre SLA</b>	<b>Tempo medio effettivo (giorni)</b>	<b>Commento</b>
S1	3	2	1	6,5	Ritardi iniziali dovuti a sincronizzazione moduli Motor-PDM
S2	3	2	1	7	Piccole deviazioni per testing integrato con Pagamenti
S3	2	2	0	5	Stabilizzazione e flussi, tempi rientrati nello SLA
S4	2	2	0	4,5	Efficienza massima raggiunta, piena automazione test
<b>Totale</b>	<b>10</b>	<b>8</b>	<b>2</b>	<b>6,2</b>	<b>-</b>

*Tabella 11: Monitoraggio dei tempi di implementazione dei cambiamenti per sprint*

E' fondamentale specificare che secondo la Service Level Agreement (SLA) interna al progetto, il tempo massimo previsto per la gestione di un cambiamento, dal momento della

sua approvazione fino al completamento del rilascio in ambiente di test, è di 7 giorni lavorativi.

Durante il periodo analizzato sono stati applicati 10 cambiamenti significativi, ossia modifiche che hanno avuto un impatto effettivo su requisiti o funzionalità e che hanno richiesto attività di sviluppo e testing dedicate. Di questi, 8 sono stati completati entro la soglia dei 7 giorni, mentre 2 hanno superato tale limite, principalmente a causa di complessità tecniche legate all'integrazione con sistemi esterni, in particolare il modulo Pagamenti e il sistema documentale PDM. Infatti, nel dettaglio, nei primi due sprint la gestione dei cambiamenti ha richiesto più tempo a causa delle attività di integrazione tra i sistemi legacy e le nuove componenti verticali, come il modulo Pagamenti e il sistema PDM. A partire dallo Sprint 3, i processi di build e testing sono stati ottimizzati e in parte automatizzati, di conseguenza i tempi medi di completamento dei cambiamenti si sono ridotti.

Il calcolo è il seguente:

$$APCT = \frac{\text{Cambiamenti completati entro SLA}}{\text{Cambiamenti applicati totali}} \times 100 = \frac{8}{10} \times 100 = 80\%$$

Il valore dell'indicatore (APCT = 80%) mostra quindi che la quasi totalità dei cambiamenti è stata gestita entro i tempi pianificati, confermando una buona reattività del team nel recepire e integrare modifiche senza impatti negativi sui rilasci.

La progressiva riduzione del tempo medio evidenzia inoltre che, con l'esperienza accumulata e l'aumento dell'automazione nei test, le decisioni di modifica sono state implementate più velocemente, migliorando la fluidità del flusso di sviluppo. Questo indica una crescente efficienza operativa e una piena adozione dei principi Agile.

Il progetto ha quindi dimostrato che è possibile gestire modifiche frequenti e complesse anche in un ambiente altamente regolato come quello assicurativo, garantendo al contempo stabilità, prevedibilità e continuità nel rilascio del valore al business.

L'analisi dei KPI relativi alla gestione dei requisiti e della flessibilità ha evidenziato che la metodologia Agile ha contribuito a migliorare la gestione dei requisiti rispetto a modelli più tradizionali, riducendo i tempi di risposta ai cambiamenti e favorendo una maggiore chiarezza comunicativa tra team tecnico e business.

In un contesto complesso e multinazionale, tale approccio si è dimostrato essenziale per garantire la continuità operativa, la qualità dei rilasci e la piena tracciabilità delle decisioni progettuali.

#### **4.1.2.2 Collaborazione e coinvolgimento stakeholders**

La seconda dimensione qualitativa analizzata riguarda la collaborazione e il livello di coinvolgimento degli stakeholder, aspetti che rappresentano il cuore pulsante dei progetti Agile.

Infatti, l'approccio Agile promuove una collaborazione costante e iterativa, in cui gli stakeholder partecipano attivamente alla definizione delle priorità, alla validazione dei rilasci e al miglioramento dei processi.

Inoltre, nei progetti complessi come quello in analisi, caratterizzato dall'integrazione di più sistemi, la qualità del dialogo continuo tra le diverse funzioni è determinante per garantire il successo dell'iniziativa.

Il grado di coinvolgimento e di collaborazione effettiva può essere misurato attraverso indicatori che quantificano la partecipazione e la soddisfazione delle pratiche agili.

##### **Stakeholder Satisfaction Score (SSS)**

Il Stakeholder Satisfaction Score (SSS) misura il grado di soddisfazione percepita dagli stakeholder a fine sprint, valutando la qualità dei rilasci, la trasparenza del processo e la capacità del team di rispondere ai feedback in modo tempestivo. Il punteggio, espresso su una scala da 1 a 5, viene calcolato attraverso una survey somministrata dopo ogni Sprint Review.

Nel progetto analizzato, la survey è stata compilata considerando come campione un gruppo di 12 stakeholder. I dati raccolti sono sintetizzati nella tabella 12:

Sprint	Media (1–5)	Dev. std.	Commento qualitativo
S1	4,3	0,4	Apprezzata la trasparenza iniziale e la chiarezza sugli obiettivi di dismissione piattaforme
S2	4,4	0,4	Riconosciuti i progressi nel modulo Motor e l'efficacia della comunicazione
S3	4,6	0,3	Apprezzate le demo integrate e la maggiore stabilità della piattaforma
S4	4,7	0,3	Elevata soddisfazione per la qualità dei rilasci e la chiarezza sui KPI di progetto

Tabella 12: Risultati della survey somministrata agli stakeholders

**SSS medio trimestre** = 4,5 / 5

L'andamento del punteggio mostra una crescita costante della soddisfazione nel corso dei quattro sprint (da 4,3 a 4,7), con una deviazione standard decrescente (da 0,4 a 0,3). Ciò oltre ad indicare un miglioramento percepito nel tempo, evidenzia anche una maggiore uniformità di giudizio tra gli stakeholder.

Il punteggio medio di 4,5 su 5 può essere interpretato come indice di alta fiducia nel modello Agile adottato e nella capacità del team di fornire valore continuo. In particolare, gli stakeholder hanno evidenziato una percezione positiva rispetto alla qualità delle demo di fine sprint e alla chiarezza nella comunicazione degli obiettivi, elementi che hanno rafforzato la fiducia nel processo e nel prodotto finale.

La riduzione della varianza nelle risposte suggerisce, inoltre, che il team è riuscito a stabilizzare la comunicazione e la collaborazione tra le varie aree aziendali (Motor, Non-Motor, Pagamenti, PDM).

Il punteggio positivo ottenuto nell'indice SSS è perfettamente coerente con i dati di performance del team riportati nel capitolo 3.5:

- La velocity è cresciuta da 75 a 92 story points per sprint
- Il burndown chart mostra un riallineamento all'andamento ideale dopo lo Sprint 2.

Inoltre, il tempo medio di applicazione dei cambiamenti (APCT) si è ridotto a 6,2 giorni, e la percentuale di casi di User Acceptance Test (UAT) completati con esito positivo ha raggiunto il 92%.

Tutti questi indicatori riflettono una delivery più prevedibile e una maggiore affidabilità del processo, due aspetti che incidono direttamente sulla soddisfazione degli stakeholder.

Questo dimostra che il framework Agile, quando applicato con costanza, non solo migliora la delivery tecnica, ma favorisce la fiducia e la partecipazione attiva degli stakeholder, elementi chiave in un progetto assicurativo di integrazione e trasformazione digitale.

### **Meeting Regularity Index (MRI)**

Il Meeting Regularity Index (MRI) misura la regolarità e la qualità delle riunioni di coordinamento, differenziandole per frequenza e livello di governance.

Nel contesto del progetto analizzato, l'indicatore è costituito da tre componenti principali:

- **RQ (Riunioni Trimestrali):** misurano la costanza e la puntualità degli incontri strategici di governance, in cui vengono analizzati i KPI complessivi, lo stato del portafoglio epiche e le milestone
- **RM (Riunioni Mensili):** includono gli incontri di allineamento tra Product Owner, Project Manager e stakeholder di business/IT, finalizzati alla definizione delle priorità e alla risoluzione di impedimenti interfunzionali
- **RB (Riunioni quindicinali e più frequenti):** rappresentano la regolarità delle cerimonie operative di Scrum (Planning, Daily, Review, Retrospective), fondamentali per il monitoraggio dell'avanzamento e la comunicazione interna.

La formula utilizzata è la seguente:



$$MRI = (RQ \times 40\%) + (RM \times 30\%) + (RB \times 30\%)$$

Le ponderazioni riflettono l'importanza relativa di ciascun livello: le riunioni più frequenti (RB) hanno il peso maggiore, in quanto garantiscono la continuità e la fluidità del flusso di lavoro.

Come è evidente nella tabella 13, l'indicatore mostra un andamento crescente, segno di un progressivo miglioramento nella regolarità e qualità della comunicazione multi-livello.

<b>Sprint</b>	<b>RQ</b>	<b>RM</b>	<b>RB</b>	<b>MRI</b>
S1	0,9	0,88	0,92	0,9
S2	0,93	0,9	0,94	0,92
S3	0,95	0,93	0,96	0,95
S4	0,96	0,94	0,97	0,96

*Tabella 13: Evoluzione dell'indice MRI per sprint*

Nello Sprint 1, la frequenza delle riunioni trimestrali (RQ) e mensili (RM) era ancora in fase di consolidamento, poiché l'organizzazione stava adattando la propria governance tradizionale al nuovo modello Agile. Dallo Sprint 2 in poi, si osserva un miglioramento costante dovuto ad una crescente familiarità con il framework Agile.

Con un valore medio pari al 93%, infatti, l'MRI evidenzia una governance comunicativa stabile, con riunioni svolte regolarmente e con alto grado di efficacia. Il miglioramento dell'indice risulta coerente con il Stakeholder Satisfaction Score (SSS), che è aumentato fino a 4,7 su 5.

La regolarità delle riunioni ha permesso di mantenere allineati i diversi attori coinvolti nel progetto, migliorando la prevedibilità del lavoro, riducendo il rischio di conflitti decisionali e mantenendo un costante allineamento tra i diversi verticali del progetto.

Nel complesso, i risultati evidenziano come la collaborazione e la partecipazione attiva degli stakeholder abbiano rappresentato un fattore determinante per il successo del progetto, consentendo di ridurre le inefficienze, migliorare l'allineamento tra team e business e trasformare la cooperazione in un vero vantaggio competitivo per l'organizzazione.

## **4.2 Vantaggi e raccomandazioni per l'adozione di Agile nei progetti assicurativi**

L'analisi condotta mostra come l'adozione della metodologia Agile possa generare benefici concreti e misurabili nei progetti assicurativi, sia in termini operativi che organizzativi.

In primo luogo, l'approccio iterativo e incrementale consente di ridurre il time-to-market e di garantire un rilascio continuo di valore, permettendo di rispondere rapidamente alle evoluzioni normative, tecnologiche e di mercato. Ciò è particolarmente rilevante in un contesto come quello assicurativo, dove la complessità dei sistemi e la rigidità dei processi tradizionali rendono difficile adattarsi ai cambiamenti.

In secondo luogo, la maggiore trasparenza e collaborazione interfunzionale favoriscono una comunicazione più efficace tra le diverse aree aziendali (business, IT, operations e compliance), ed in generale un coinvolgimento costante degli stakeholders, con feedback immediati, migliorando la qualità delle decisioni e riducendo il rischio di incomprensioni o rilavorazioni. Il focus, quindi, non è solo sugli output tecnici, ma sul valore percepito dal cliente finale.

Dal punto di vista organizzativo, l'introduzione della metodologia Agile favorisce la creazione di una cultura collaborativa e orientata al miglioramento continuo, che valorizza le competenze del team e incoraggia la condivisione delle responsabilità.

Questo processo di evoluzione culturale, se accompagnato da una leadership consapevole e da una governance flessibile, diventa un fattore determinante per il successo delle iniziative di trasformazione digitale nel settore assicurativo.

Tuttavia, l'esperienza analizzata conferma che l'adozione di Agile richiede un percorso graduale e strutturato, che tenga conto delle specificità del contesto assicurativo. È necessario, in particolare:

- Impegno concreto da parte della leadership nel sostenere il cambiamento e nel superare eventuali ostacoli organizzativi o culturali. I manager, quindi, devono agire come facilitatori del lavoro dei team, supportandoli e creando le condizioni per lavorare in modo efficace
- Investire nella formazione dei team e nello sviluppo di competenze trasversali, al fine di diffondere una piena comprensione dei principi Agile

- Definire indicatori chiave (KPI) chiari e condivisi per monitorare i progressi, misurare l'efficacia delle pratiche introdotte ed utilizzare tali indicatori non solo come controllo, ma come strumento di miglioramento continuo.
- Promuovere una governance adattiva, che combini i principi di agilità e collaborazione tipici del metodo Agile, con il necessario livello di controllo e conformità richiesto nel settore assicurativo.

In conclusione, la metodologia Agile, se correttamente implementata e accompagnata da un'evoluzione culturale e organizzativa, rappresenta un fattore abilitante per l'innovazione, migliorando l'efficienza operativa, la qualità del servizio e la capacità di risposta al mercato delle compagnie assicurative.

## 5 Conclusioni

L'introduzione della metodologia Agile nel progetto assicurativo analizzato ha rappresentato un passaggio significativo verso un modo di lavorare più flessibile, collaborativo e orientato al valore. L'esperienza maturata nel corso del tirocinio presso Accenture ha permesso di osservare da vicino l'applicazione concreta dei principi agili in un contesto complesso, caratterizzato da elevata interdipendenza tra sistemi, numerosi stakeholder e una forte componente regolamentare.

Sin dalle prime fasi del progetto è emersa la necessità di conciliare la natura iterativa e adattiva del modello Agile con le esigenze di controllo e conformità proprie del settore assicurativo. L'approccio adottato ha consentito di mantenere un equilibrio tra flessibilità e rigore metodologico, favorendo un'evoluzione graduale ma costante del progetto.

L'analisi dei Key Performance Indicators (KPI) ha fornito evidenze oggettive dei miglioramenti ottenuti nel corso dei diversi sprint. I KPI quantitativi hanno mostrato un progressivo incremento dell'efficienza operativa, con un'elevata aderenza alle pianificazioni e una crescita nella qualità tecnica dei rilasci, testimoniata da un tasso di successo nei test sempre più alto. Allo stesso tempo, i KPI qualitativi hanno messo in luce un miglioramento nella collaborazione e nella comunicazione tra stakeholder, evidenziando un grado crescente di soddisfazione e di chiarezza sugli obiettivi comuni.

Questi risultati confermano come la metodologia Agile, se correttamente adattata al contesto, consenta di ottenere benefici concreti sia sul piano operativo sia su quello relazionale. Il lavoro incrementale, la definizione chiara dei ruoli e la condivisione costante delle informazioni hanno contribuito a ridurre la complessità e ad aumentare la capacità di risposta alle esigenze del business.

Naturalmente, l'esperienza ha evidenziato anche alcune criticità. In particolare, la gestione del cambiamento organizzativo richiede un impegno costante da parte della leadership e un percorso di crescita culturale diffuso. La metodologia Agile, infatti, non si limita a introdurre nuove pratiche operative, ma implica un cambiamento profondo nella mentalità aziendale, che deve essere sostenuto da una leadership attiva e da un sistema di governance capace di adattarsi ai contesti.

Nel complesso, il progetto ha raggiunto gli obiettivi prefissati, dimostrando che l'applicazione della metodologia Agile in ambito assicurativo è non solo possibile, ma auspicabile al fine di incrementare l'efficienza, migliorare la qualità del prodotto e favorire l'allineamento tra business e tecnologia.

È quindi fondamentale, per progetti futuri, investire nella formazione e nella sensibilizzazione di tutte le risorse coinvolte, così da promuovere una reale condivisione dei principi agili e un più rapido consolidamento dei benefici.

In conclusione, questa esperienza ha mostrato come la metodologia Agile, se applicata con consapevolezza, possa favorire una trasformazione reale del modo di lavorare, rendendo i processi più efficienti, le persone più coinvolte e i risultati più in linea con le esigenze del mercato. L'approccio iterativo, la collaborazione continua e la centralità del valore consegnato al cliente rappresentano oggi elementi imprescindibili per qualunque organizzazione che voglia affrontare con successo le sfide della complessità e dell'innovazione.

## 6 Riferimenti bibliografici

- [1] EduBP. (2025). *Waterfall vs Agile, quale metodologia scegliere e perché*.  
<https://www.edubp.it/waterfall-vs-agile/>
- [2] Wrzosek D. (2024). *Cos'è la metodologia waterfall e come funziona?* <https://flexi-project.com/it/cose-la-metodologia-waterfall-e-come-funziona/>
- [3] Laoyan S. (2025). *Tutto quello che c'è da sapere sulla metodologia waterfall per la gestione dei progetti*. Asana. <https://asana.com/it/resources/waterfall-project-management-methodology>
- [4] Atlassian. (n.d.). *Metodologia a cascata: una guida completa*.  
<https://www.atlassian.com/it/agile/project-management/waterfall-methodology>
- [5] SAEP (2025). *Metodologia Waterfall: la guida definitiva per la gestione dei progetti*.  
<https://www.saep-ict.it/magazine/metodologia-waterfall/#gref>
- [6] Špundak, M. (2014). *Mixed agile/traditional project management methodology – reality or illusion?* <https://doi.org/10.1016/j.sbspro.2014.03.105>
- [7] Nonaka I., Takeuchi H. (1986). *The new product development game*. Harvard Business Review <https://hbr.org/1986/01/the-new-new-product-development-game>
- [8] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. <https://agilemanifesto.org/iso/it/manifesto.html>
- [9] Cobb C.G. (2015). *The Project Managers Guide to Mastering Agile: Principles and Practices for an Adaptive Approach*
- [10] Agile Alliance (n.d.). *The 12 Principles behind the Agile Manifesto*.  
<https://agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/>
- [11] State of Agile (2023). *17<sup>th</sup> State of Agile Report*. [RE-SA-17th-Annual-State-Of-Agile-Report.pdf](https://www.stateofagile.com/2023/17th-Annual-State-Of-Agile-Report.pdf)
- [12] Berman, J. (2022). *Agile Frameworks*. Planview. [Framework Agili | Vista in pianta](#)

- [13] Walee, N. A., Onisha, T. A., Akinola, A., van Deventer, G., & Chen, L. (2024). *Impact of Agile methodology in IT industries: A comparative study*. IEEE. <https://doi.org/10.1109/SoutheastCon52093.2024.10500231>
- [14] Schwaber, K., Sutherland, J. (2020). *Guida a Scrum - La guida definitiva a Scrum: le regole del gioco*. [2020-Scrum-Guide-Italian.pdf](#)
- [15] Terry, J. (2022). *What is Scrum?* Planview. <https://www.planview.com/resources/guide/what-is-scrum/>
- [16] Drumond C. (n.d.). *What is Scrum and how to get started*. Atlassian. <https://www.atlassian.com/agile/scrum>
- [17] Lynn, R. (2022). *Scrum Best Practices for Teams*. Planview. <https://www.planview.com/resources/guide/what-is-scrum/scrum-best-practices-teams/>
- [18] West, D. (n.d.). *Agile Scrum Role and responsibilities*. Atlassian. <https://www.atlassian.com/agile/scrum/roles>
- [19] Radigan, D. (n.d.). *An agile guide to scrum meetings*. Atlassian. <https://www.atlassian.com/agile/scrum/ceremonies>
- [20] Chandler, H. (n.d.). *Agile scrum artifacts*. Atlassian. <https://www.atlassian.com/agile/scrum/artifacts>
- [21] Kisielnicki, J., & Misiak, A. M. (2017). *Effectiveness of Agile compared to Waterfall implementation methods in IT projects: Analysis based on Business Intelligence projects*. *Foundations of Management*. <https://doi.org/10.1515/fman-2017-0021>
- [22] Bichard J., Mukhopadhyay A. (2022). *Insurance 2025 and beyond*. PWC. <https://www.pwc.com/gx/en/financial-services/fs-2025/pwc-insurance2025.pdf>
- [23] Majukumar S., Hernandez, D. (2020). *Un settore assicurativo agile e a prova di futuro grazie a una digital transformation completa*. IDC. <https://forms.workday.com/content/dam/web/it/documents/Enabling-Agile-Future-Ready-Insurance-Organizations-Through-Holistic-Digital-Transformation-IT.pdf>
- [24] Baiocchi F., Gorreri I., Iori T., Lado P. (2022). *L'assicurazione del futuro e i modelli abilitanti*. EY. <https://www.ey.com/content/dam/ey-unified-site/ey-com/it-it/events/documents/assicurazione-del-futuro-e-modelli-abilitanti.pdf>

- [25] Lorenz J.-T., Mahadevan D., Oncul B., Yenigün, M. (2020). *Scaling agility: A new operating model for insurers*. McKinsey & Company. <https://www.mckinsey.com/industries/financial-services/our-insights/scaling-agility-a-new-operating-model-for-insurers>
- [26] Agile Infoways LLC. (n.d.). *Case study: Digitalizing insurance operations from the ground up*. <https://www.agileinfoways.com/case-studies/insurance/>
- [27] PilotFish Technology, (n.d.). *Case study: Automated insurance underwriting & legacy system integration*. <https://www.pilotfishtechology.com/insurance-automated-underwriting-legacy-system-integration-case-study/>
- [28] Marks M., Härtl F. (2021). *Agile transformation in the European insurance sector*. The Transformation Alliance. [https://the-transformation-alliance.com/wp-content/uploads/2025/06/hz\\_agile\\_study\\_final.pdf](https://the-transformation-alliance.com/wp-content/uploads/2025/06/hz_agile_study_final.pdf)
- [29] Yudhokusuma K., Munanda P., Raharjo T. (2023). *Agile adoption challenges in insurance: a systematic literature and expert review*. [10.11591/cs.it.v4i3.pp268-278](https://doi.org/10.11591/cs.it.v4i3.pp268-278)
- [30] Accenture. (n.d.). *About Accenture: Global company information*. <https://www.accenture.com/it-it/about/company-index>
- [31] Accenture Italia. (n.d.). *Conosci Accenture in Italia*. <https://www.accenture.com/it-it/about/company/italy>
- [32] Accenture. (2025). *2025 Annual Report*. <https://www.accenture.com/us-en/about/company/annual-report>
- [33] Sales Silva F. (2025). *Pianificazione strategica: Oltre gli obiettivi, per risultati reali*.
- [34] Tableau. (2024). *SMART goals and KPI criteria: A guide to measurable success*. <https://www.tableau.com/it-it/learn/articles/smart-goals-criteria>



