



**Politecnico
di Torino**

Laurea Magistrale in Ingegneria Gestionale
A.a. 2023/2024
Sessione di Laurea 04/24

Intelligent Process Automation: Piano di Trasformazione Digitale

Relatore:
Tania Cerquitelli

Candidato:
Pietro Termitte

Indice

Lista di acronimi	3
Abstract	4
1. Introduzione	5
2. Intelligent Process Automation	6
2.1 Dimensioni e Previsioni	6
2.2 Tecnologie	8
2.3 Benefits	10
2.4 Rischi e Candidati ideali	12
2.5 Best Practices	13
2.6 Casi Studio e Applicazioni dell'IPA	13
3. Fasi del ciclo di vita di una soluzione di IPA	16
2.1 Definizione della metodologia di sviluppo software: Waterfall vs Agile	16
3.2 Definizione dei Requisiti	24
3.3 Pianificazione	25
3.4 Sviluppo	34
3.5 Test	37
3.6 Implementazione	39
3.7 Manutenzione	40
3.8 Monitoraggio	42
4. Process Mining: Teoria, Benefit e Sfide	44
4.1 Concetti Preliminari	44
4.2 Process Mining	50
4.3 Principi guida	54
4.4 Sfide	57
4.5 Benefits	58
5. Applicazione pratica del Process Mining: Caso studio con Celonis	61
5.1 Software Celonis	61
5.2 Variant Explorer	62
5.3 Process Explorer	66
5.4 Tabelle, grafici e KPI	68
5.5 Conformance Checking	69
6. Conclusioni	73
Bibliografia	75

Lista di acronimi

AI	Artificial intelligence
ASD	Adaptive Software Development
AWS	Amazon Web Services
BPM	Business Process Management
CD	Continuous Deployment
CI	Continuous Integration
GCP	Google Cloud Platform
IaC	Infrastructure as Code
ICR	Intelligent Character Recognition
IPA	Intelligent Process Automation
KPI	Key Performance Indicators
LLM	Large Language Model
ML	Machine Learning
NLP	Natural Language Processing
OBR	Optical Barcode Reader
OCR	Optical Character Recognition
OMR	Optical Mark Recognition
PM	Project Manager
RPA	Robotic Process Automation
TDD	Test Driven Development
UAT	User Acceptance Testing
VCS	Version Control System
WBS	Work Breakdown Structure

Abstract

Il lavoro di tesi esplora l'Intelligent Process Automation (IPA), un approccio emergente che combina l'intelligenza artificiale e altre tecnologie innovative per migliorare l'efficienza e l'efficacia dei processi aziendali. La parte iniziale dell'elaborato si concentra sugli aspetti principali dell'IPA investigando dallo stato attuale del mercato e delle tecnologie più influenti fino ad arrivare a come alcune di queste possono essere applicate in casi reali.

La tesi prosegue con un'analisi dettagliata delle fasi del ciclo di vita di una soluzione di IPA, partendo dalla definizione iniziale dei requisiti fino all'implementazione e monitoraggio del progetto. Inoltre, si pone particolare attenzione alla scelta tra due delle metodologie di sviluppo software più diffuse ed analizzate nel corso degli anni: Waterfall e Agile.

Nella parte successiva, la tesi si concentra su una ricerca approfondita di una delle tecnologie in crescita dell'IPA: il Process Mining. Si tratta di una tecnica di analisi dei dati che permette di scoprire, monitorare e migliorare i processi reali basandosi su dati oggettivi: i log degli eventi. Questi ultimi, insieme ad altri concetti fondamentali come il Business Process, forniscono una base solida per comprendere quando e come applicare il Process Mining. Inoltre, viene presentata un'applicazione pratica della tecnologia in questione grazie all'utilizzo del software Celonis, leader nel settore.

Questo lavoro rappresenta un contributo significativo alla letteratura sull'IPA in quanto fornisce una visione integrata e completa dell'Intelligent Process Automation. Infatti, viene messo in evidenza come le tecnologie di questa metodologia, le fasi del ciclo di vita per implementare un progetto e le tecniche come il Process Mining siano strettamente interconnesse. Attraverso questo lavoro di tesi, si è cercato di dimostrare come una comprensione approfondita di questi elementi possa fornire le basi per implementare con successo progetti di IPA, rivoluzionando così i processi aziendali per mezzo del suo potenziale. Per raggiungere questo risultato, è stata condotta un'analisi sistematica della letteratura prodotta negli anni, includendo casi studio, articoli scientifici e articoli di aziende specializzate nel settore.

1. Introduzione

In ogni epoca storica, il principale obiettivo delle aziende è sempre stato quello di minimizzare i costi, massimizzare i ricavi, ottenere prodotti di qualità e, nell'ottica migliore, realizzare gli obiettivi contemporaneamente.

Nell'era digitale, per raggiungere questo scopo, le aziende hanno bisogno di automatizzare le loro attività e processi aziendali, che molto spesso, oltre a portare i benefici citati precedentemente, aiutano a migliorare altri aspetti descritti più dettagliatamente nelle sezioni successive della tesi.

La struttura del presente lavoro di tesi è la seguente:

Il **secondo capitolo** si concentra sull'Intelligent Process Automation (IPA). In questa sezione, vengono descritte le dimensioni e le previsioni di mercato nell'ambito dell'IPA, le tecnologie chiave necessarie per implementare una soluzione, i benefici che l'IPA può portare alle organizzazioni, i rischi associati e i candidati ideali per l'implementazione. Inoltre, vengono esaminate alcune delle best practices da considerare per adottare l'IPA e infine vengono presentate delle applicazioni reali.

Il **terzo capitolo** è dedicato alle fasi del ciclo di vita di una soluzione di IPA, dalla definizione dei requisiti alla pianificazione, sviluppo, test, implementazione, manutenzione e monitoraggio. Si discutono anche delle metodologie di sviluppo software, in particolare confrontando l'approccio Waterfall con l'approccio Agile.

Il **quarto capitolo** è incentrato su una delle tecnologie dell'Intelligent Process Automation: il Process Mining. Durante il periodo di tirocinio, è stata condotta una ricerca su questo argomento. In questa sezione, si analizza nel dettaglio la tecnologia del Process Mining, discutendo di come può essere utilizzata per analizzare e migliorare i processi aziendali e quali sono le criticità.

Infine, nel **quinto capitolo** viene illustrato il funzionamento del software Celonis, leader nell'implementazione del Process Mining, e alcuni risultati che dimostrano l'utilità della tecnologia e del software.

Per concludere, nell'ultimo capitolo sono riportate le limitazioni e le implicazioni del lavoro di tesi, nonché possibili ricerche future.

2. Intelligent Process Automation

L'IPA, acronimo di Intelligent Process Automation (Automazione intelligente dei processi), è una strategia e un connubio di tecnologie perfettamente integrate. L'insieme di queste permette di ottimizzare e, dove possibile, automatizzare processi aziendali in modo da renderli autonomi.

Di conseguenza, le macchine possono svolgere compiti di diversa complessità e, grazie a strumenti come l'**Intelligenza Artificiale (AI)** e l'**Apprendimento Automatico (ML)**, possono effettuare delle scelte tramite i dati forniti in input e agli errori commessi conseguentemente alle azioni intraprese.

Nei paragrafi successivi, verranno esaminati più dettagliatamente gli strumenti che compongono l'IPA e i numerosi benefici che essi comportano. Inoltre, viene posta particolare attenzione sui candidati ideali che sono compatibili con questa innovazione tecnologica.

2.1 Dimensioni e Previsioni

L'Intelligent Process Automation è un ambito ancora relativamente poco esplorato, ma ha dimostrato di avere un elevato potenziale.

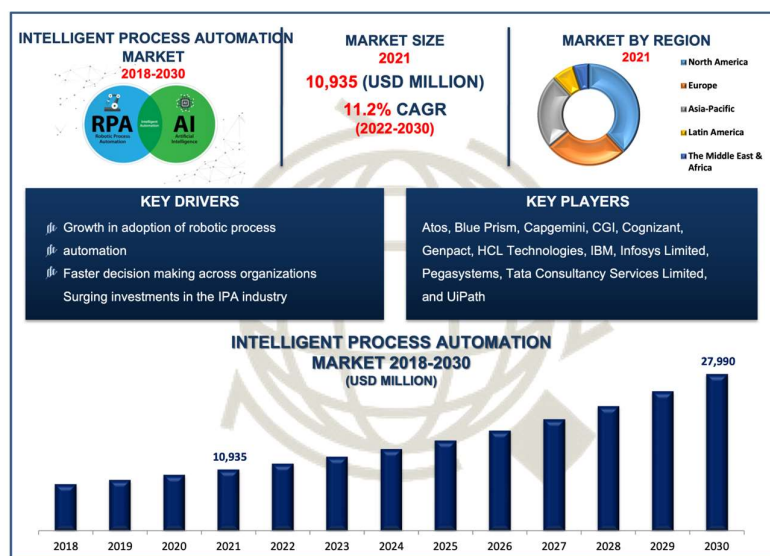


Figura 1: Intelligent Process Automation Market Size, Share, & Trends 2030 (Acumen)

Come si può osservare dalla Figura 1, il valore globale del mercato dell'IPA ha raggiunto i 10,935 miliardi di dollari nel 2021 e, anche se difficile da stimare in quanto l'IPA è mix di tecnologie, si presume (anche grazie alle evidenze delle tendenze passate) che entro il 2030

possa raggiungere un valore che si avvicina ai 28 miliardi con un tasso di crescita annuo composto (CAGR) dell'11,2% [1].

Nel grafico a donuts della Figura 1, si visualizza il mercato segmentato in 5 aree geografiche: Nord America, Europa, Asia-Pacifico, America Latina e Medio Oriente & Africa. Attualmente il nord America è l'area che ha generato il maggior fatturato nel mercato globale nell'ambito dell'IPA poiché è colei che sfrutta maggiormente l'intelligenza artificiale (AI), automazione robotica dei processi (RPA) e la gestione dei processi aziendali (BPMN). L'Europa detiene una notevole quota di mercato e si stima che continuerà a mantenere una presenza significativa. L'Area che subirà una crescita sensibilmente superiore rispetto le altre, è l'Asia-Pacifico, in quanto negli ultimi anni ha mostrato una crescente adozione dell'automazione, soprattutto grazie a Paesi in via di sviluppo come India e Cina.

Come è illustrato nella Figura 2, il mercato dell'IPA può essere segmentato su diverse applicazioni

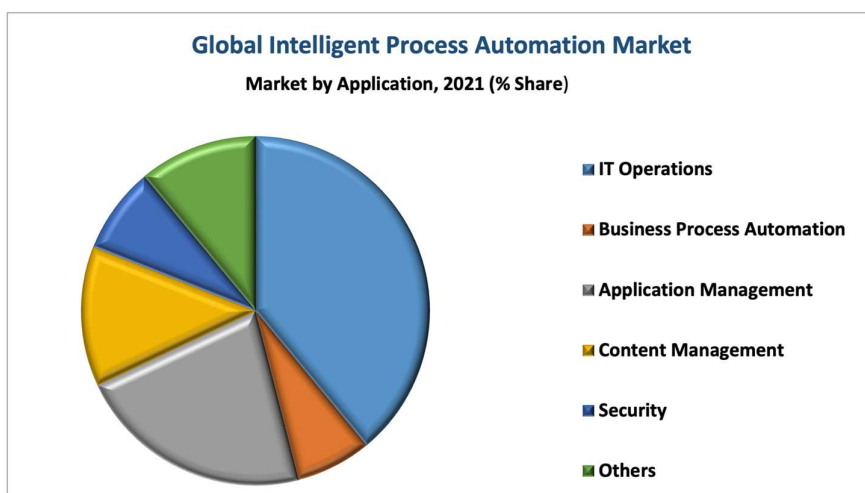


Figura 2: Market by Application (Acumen)

Nel 2021, il segmento delle operazioni IT ha dimostrato una particolare propensione nell'adozione dell'IPA rispetto agli altri segmenti. Questo può essere attribuito alla natura delle operazioni IT, che spesso includono attività e processi ripetitivi e basati su regole, rendendoli ideali per l'automazione.

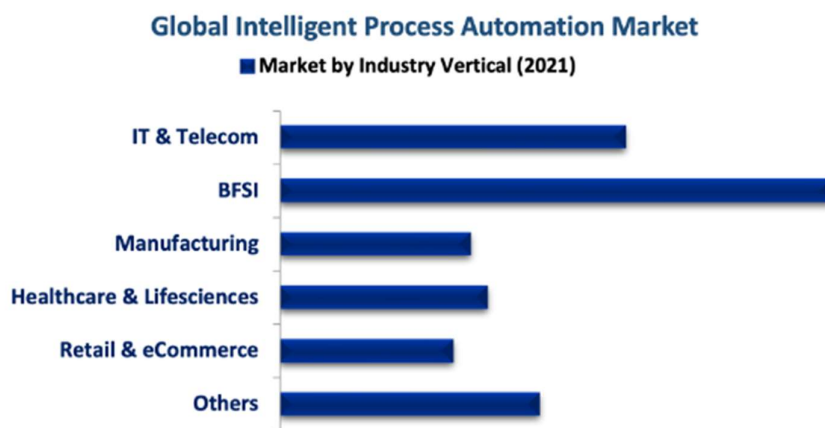


Figura 3: Market by Industry Vertical (Acumen)

L'industria BFSI, che comprende **Banche, servizi finanziari e assicurativi**, ha dimostrato di aver adottato al meglio delle soluzioni di IPA (Figura 3). Questo settore, nel 2021 ha generato ricavi significativi e portato una serie di benefici. Tra questi, si evidenziano un incremento della produttività, un miglioramento dell'efficienza, una riduzione dei costi e un'esperienza cliente notevolmente migliorata.

Il mercato IPA è segmentato in 2 principali modalità di distribuzione: locale e cloud [2]. La differenza tra i due è che la distribuzione in locale prevede l'installazione dell'hardware e del software presso la sede del cliente permettendo di ottenere il completo controllo su applicazioni, piattaforme, dati e sistemi. Nel caso in cui la distribuzione avvenisse in cloud, i fornitori dei servizi sono responsabili del trattamento e della sicurezza dei dati. Nel 2022 il segmento on-premise (e quindi locale) rappresentava la quota di mercato più elevata ma secondo il Gartner, spesso considerata come la più importante fonte di informazioni strategiche globali dell'IT, entro il 2026, il 75% delle organizzazioni adotterà come piattaforma sottostante un servizio cloud per implementare la propria trasformazione digitale [3].

2.2 Tecnologie

Come anticipato nell'introduzione di questo capitolo, esistono numerose tecnologie che combinate tra di loro compongono l'Intelligent Process Automation [4], [5], [6]. Le più importanti sono:

- **Intelligenza Artificiale (AI):** tecnologia che simula il comportamento della mente umana e che imita facoltà come la creatività, il ragionamento, l'apprendimento, la pianificazione. Proprio come il comportamento umano, l'AI mira a comprendere il

linguaggio naturale, imparare dai propri errori tramite ML e prendere decisioni basandosi sui dati che il sistema ha a disposizione.

- **Apprendimento Automatico (ML):** sotto-categoria dell'AI, nella quale i computer, tramite l'esperienza, apprendono dai dati e migliorano senza la necessità di essere programmati. Di conseguenza, maggiore è la mole di dati e maggiore è la loro precisione, tanto migliori risulteranno le applicazioni che si basano sul ML.
- **Robotic Process Automation (RPA):** robot software in grado di automatizzare flussi di lavoro e attività di back-office ad alta intensità di manodopera. Spesso si tratta di attività ripetitive eseguite da esseri umani come l'estrazione di dati, compilazioni di moduli, analisi di dati, processamento di documenti, ecc.. Solitamente sono attività che presentano set definiti di input, processi basati su delle regole ben precise e degli output misurabili.
- **Business Process Management (BPM):** approccio che consente di gestire i processi aziendali attraverso una combinazione di attività come l'analisi, la progettazione e il controllo dei processi. Il BPM permette di mappare il processo e individuare inefficienze dello stesso. Quando possibile, il BPM può essere integrato con altre tecnologie facenti parte dell'IPA, come la sopracitata RPA. Questo permette di automatizzare determinate attività che possono apportare migliorie al processo. Una volta implementate, il BPM offre delle funzionalità di monitoraggio e analisi avanzate. Queste includono la creazione di dashboard analitiche e la definizione di metriche, che consentono di monitorare il processo migliorato e indentificare ulteriori opportunità di ottimizzazione. Ciò comporta una migliore gestione delle risorse, maggiore collaborazione tra team e ottimizzazione dell'esperienza del cliente.
- **Process Mining:** tecnologia che offre un apporto cruciale in quanto permette all'utente di visualizzare i processi e in tempo reale analizzarne le criticità semplicemente analizzando gli "event logs". Come anticipato nell'introduzione, questa tecnica è stata oggetto di ricerca durante il periodo di tirocinio e vengono dedicati i capitoli 4 e 5 dai titoli "Process Mining: Teoria, Benefici e Sfide" e "Applicazione pratica del Process Mining: Caso studio con Celonis" più avanti nell'elaborato.
- **Strumenti low-code/no-code:** piattaforme che permettono lo sviluppo di applicazioni e soluzioni digitali mediante strumenti intuitivi, richiedendo un livello minimo o addirittura nullo di scrittura di codice.
- **Cognitive Computing:** tecnologia che mira ad imitare la personalità e il funzionamento del cervello umano utilizzando le stesse tecniche dell'intelligenza artificiale. Questa

tecnologia però, prova a simulare il processo di pensiero, la personalità e i sentimenti che sono utili al sistema nel processo di decision-making.

- **Intelligent Character Recognition (ICR) e Optical Character Recognition (OCR):** Entrambe le tecnologie hanno l'obiettivo di convertire i documenti in testo digitale leggibile da una macchina. La differenza tra i due è che l'OCR si limita a leggere, e quindi convertire, solo documenti che contengono caratteri "a stampa", cioè scritti al computer, come possono essere la patente e carta di identità, mentre l'ICR permette anche la digitalizzazione di documenti completamente o in parte scritti a mano da un essere umano.
- **Optical Mark Recognition (OMR):** tecnologia che aiuta ad estrarre informazioni da moduli cartacei come sondaggi, schede elettorali, candidature, moduli medici. In particolare, aiuta a riconoscere le spunte ed estrarre direttamente le informazioni correlate ad esse.
- **Optical Barcode Reader (OBR):** scanner ottico in grado di riconoscere codici a barre da dei documenti e decodificarli in dati.
- **Natural Language Processing (NLP):** tecnologia che utilizza tecniche come Machine Learning e Artificial Intelligence e serve a fornire alle macchine la capacità di interpretare il linguaggio umano e fornire agli utenti delle risposte pertinenti tramite l'utilizzo di chatbot.

Per adottare una soluzione di Intelligent Process Automation non è necessario utilizzare tutte le sopracitate tecnologie, ma anche solo alcune di esse. Combinandole nel tempo si applica un approccio di continuo miglioramento che consente un controllo più accurato delle attività e del flusso di lavoro.

2.3 Benefits

Come la maggior parte degli approcci innovativi, l'Intelligent Process Automation (IPA) apporta al mondo tecnologico una serie di benefici significativi. Questi vantaggi non si limitano solo all'ottimizzazione dei processi e alla riduzione dei costi, ma si estendono anche alla possibilità di liberare il personale per concentrarsi su compiti più strategici e gratificanti. L'IPA, infatti, consente alle aziende di automatizzare le attività ripetitive e a basso valore, permettendo ai dipendenti di dedicare il loro tempo a compiti più complessi che richiedono un pensiero critico. Inoltre, l'IPA può migliorare l'accuratezza dei processi aziendali riducendo gli errori

umani, aumentando così l'efficienza e la produttività [4]. Nel corso di questa sezione, si esplorano in dettaglio alcuni dei benefici più importanti:

- **Incremento della produttività e precisione:** i software robot sono più veloci, più accurati, possono lavorare 365 giorni all'anno senza interruzioni e soprattutto svolgono attività ripetitive che al personale che dovrebbe occuparsene possono risultare noiose e monotone, portando conseguentemente a una riduzione della produttività e un incremento degli errori. In questo modo il personale è più motivato e può svolgere compiti più complessi e stimolanti migliorando la qualità del lavoro e riducendo, in alcuni casi, la possibilità di errori umani fino al 75% [7].
- **Agilità e velocità:** l'IPA, oltre a velocizzare e migliorare i processi grazie a una rapida elaborazione di dati, che in alcuni casi ha portato ad una riduzione del tempo di elaborazione diretta tra il 50 e 60% [8], permette anche di integrare e distribuire le modifiche con agilità grazie ad approcci di continua integrazione e implementazione (CD/CI).
- **Aumenta la scalabilità e flessibilità:** l'architettura modulare e l'infrastruttura cloud di una soluzione di IPA, offrono alle aziende la flessibilità necessaria per soddisfare le esigenze del proprio settore che è in continua evoluzione. L'architettura modulare consente di sviluppare e distribuire diversi componenti di automazione, come bot RPA e modelli IA. In questo modo ogni componente può essere adattato a determinati processi o funzioni, offrendo la giusta flessibilità per scalare l'IPA in diverse unità aziendali. Le infrastrutture cloud, oltre a permettere di risparmiare notevolmente da ingenti investimenti iniziali, offre la possibilità di allocare le risorse in modo dinamico, in funzione della domanda e di conseguenza di quanto effettivamente si utilizza il servizio.
- **Incremento della soddisfazione del cliente:** Le interazioni che il cliente ha con il prodotto avvengono con più rapidità, efficienza e qualità portando a un miglioramento della soddisfazione e rappresentando un vantaggio competitivo per l'azienda.
- **Riduzione dei costi operativi e incremento del ROI:** Nonostante i robot possano comportare una spesa iniziale sostanziosa, sul lungo termine permettono di risparmiare notevolmente sui costi operativi e ottenere un ROI il più delle volte con valori percentuali a 3 cifre [8].

2.4 Rischi e Candidati ideali

Spesso le aziende hanno intenzione di approcciarsi al campo dell'Intelligenza Artificiale senza conoscere i requisiti necessari all'implementazione [9]. Sebbene l'Intelligent Process Automation stia guadagnando sempre più attenzione come un potente strumento per migliorare l'efficienza dei processi aziendali e nonostante sia una possibilità per tutte le aziende, la sua applicabilità e il grado di successo dipende da una serie di fattori come:

- **Prontezza tecnologica:** Per poter implementare un progetto di IPA, e quindi applicare l'automatizzazione dei processi aziendali, è necessario preparare l'organizzazione a livello tecnologico. Un'organizzazione dovrebbe redigere un piano strategico per adottare l'IPA e individuare le risorse necessarie per svilupparlo. È consigliabile iniziare dall'implementazione di RPA e l'azienda, per ottenere un risultato soddisfacente, deve individuare i flussi di lavoro o processi che sono altamente ripetitivi. Inoltre, deve pensare anche a come integrare il BPM e quindi capire come e quali processi e flussi di lavoro standardizzare e digitalizzare. Le organizzazioni, per poter procedere, devono disporre di processi e flussi di lavoro aziendali ben definiti e set di dati storici abbastanza grandi per l'addestramento. Queste tecnologie sono le principali abilitatrici per una corretta implementazione dell'IPA soprattutto in settori che presentano attività ripetitive e una grande mole di dati come quelli assicurativi, finanziari, sanità ecc.
- **Risorse e automazione in fase di esecuzione:** A differenza dell'RPA dove non sono previste risorse per la supervisione, è necessaria della competenza per supervisionare le prestazioni in modo da impostare correttamente i parametri iniziali dei motori di Intelligenza Artificiale e supportare un processo decisionale fluido. In questo modo si garantisce un'adozione sicura e di successo dell'IPA. Figure come esperti di dominio, dipendenti e manager hanno necessità di trasparenza dei processi, standardizzazione e AI flessibile e scalabile per poter suggerire delle decisioni di qualità.
- **Complessità dei processi:** non esiste un approccio di IPA compatibile a priori con i propri processi. È opportuno sottolineare che le attività aziendali possono risultare imprevedibili e i processi complessi. Di conseguenza, la reingegnerizzazione del BPM può richiedere esperti del settore, team di IPA interni e consulenti per fornire una soluzione adeguata.
- **Cultura aziendale:** Adottare l'IPA comporta un grande cambiamento dal punto di vista operativo e strategico. Per evitare che l'adozione fallisca, il top management deve avere una mentalità aperta e dunque essere predisposto al cambiamento, senza risultare troppo conservatore. Inoltre, è fondamentale che consideri l'AI come una tecnologia a supporto

del dipendente portando a una collaborazione uomo-robot e non come un'occasione per sostituire i dipendenti con l'automazione.

2.5 Best Practices

Come descritto in precedenza l'Intelligent Process Automation è una pratica che potenzialmente offre un vantaggio competitivo ma che può risultare difficile da implementare poiché richiede un piano e delle azioni precise e coordinate. Per sfruttare il potenziale dell'Automazione Intelligente e consentire all'organizzazione di implementare soluzioni efficaci è necessario seguire alcune buone pratiche [4]. Tra le più importanti ci sono:

- **Proof of Concept:** L'avvio di un progetto di IPA attraverso una PoC offre numerosi vantaggi. Tra questi, la possibilità di testare il prodotto dal punto di vista tecnico e operativo su una scala ridotta, minimizzando così il rischio di fallimento. Inoltre, una PoC consente di mettere in risalto i potenziali benefici dell'IPA, fornendo agli stakeholder aziendali una base per sostenere il progetto.
- **Aspettative corrette:** è di fondamentale importanza stabilire delle aspettative realistiche correlate ai potenziali benefici, al fine di prevenire la generazione di un clima di euforia ingiustificata.
- **Metodologia di consegna rapida:** è buona pratica implementare più set di automazione di piccole dimensioni e facilmente gestibili, piuttosto che un unico progetto grande e complesso. Questo approccio aumenta le probabilità di successo grazie all'individuazione e la risoluzione di problemi su piccola scala.
- **KPI e metriche:** i Key Performance Indicator (KPI) e le metriche sono indispensabili per misurare l'efficacia, per monitorare il progresso e anche per comunicare risultati agli stakeholder. In particolare, permettono di valutare se le singole attività, i processi e il progetto procedono verso l'obiettivo di business definito. Sono degli strumenti utili ad identificare le aree in cui sono necessari miglioramenti, contribuendo così all'ottimizzazione delle prestazioni.

2.6 Casi Studio e Applicazioni dell'IPA

L'Intelligent Process Automation può essere applicata in vari settori e con diverse applicazioni. Durante il mio periodo di tirocinio presso Data Reply s.r.l., ho avuto modo di assistere alla progettazione e realizzazione di un progetto di IPA per un cliente assicurativo. L'obiettivo era quello di creare un processo automatizzato che permettesse di **estrarre dati** da documenti

strutturati e non strutturati. Il processo consiste nel riconoscere e recuperare informazioni rilevanti da vari tipi di documenti, sia digitali che fisici. La distinzione principale tra documenti strutturati e non, è che i primi seguono un formato predefinito e previsto, come carte d'identità, fatture, moduli, constatazioni amichevoli mentre i secondi non hanno una struttura fissa o ricorrente, come e-mail, lettere e articoli. Basti immaginare che i dati anagrafici o il numero identificativo della carta d'identità sono inseriti sempre nello stesso luogo (Figura 4).

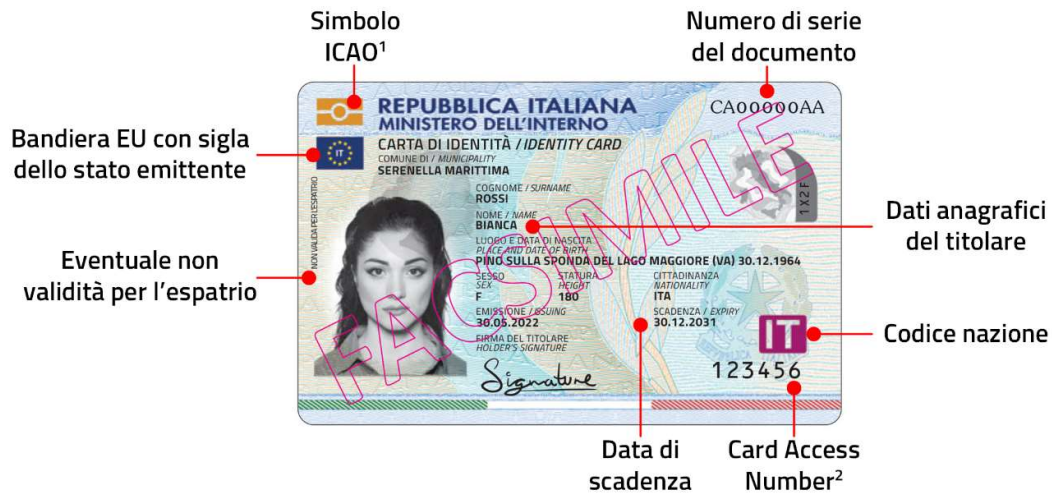


Figura 4: FacSimile Carta d'identità (cartaidentita.interno.gov)

Da quanto osservato, l'estrazione di dati da documenti strutturati risulta generalmente più semplice, più rapida e più accurata rispetto all'estrazione di dati da documenti non strutturati, in quanto i campi e le informazioni sono facilmente identificabili e standardizzabili. L'unica criticità in questo tipo di estrazioni risiede nella gestione di documenti di qualità variabile. Se la qualità di un documento risulta bassa, può infatti compromettere l'efficacia dell'estrazione. Inoltre, la struttura di tali documenti consente l'utilizzo di tecnologie come OCR, ICR e OMR, che, come spiegato nella Sezione 2.2, permettono di digitalizzare informazioni da documenti fisici. Al contrario, l'estrazione di dati da documenti non strutturati è più complicata, in quanto 2 e-mail provenienti da 2 fonti differenti possono contenere la stessa tipologia di informazioni ma scritte con uno stile diverso e presenti in parti del testo differenti. Di conseguenza, è necessario ricorrere a tecniche di Machine Learning che consentano di trasformare correttamente dati non strutturati in dati strutturati, al fine di facilitarne l'utilizzo e l'analisi. Queste tecniche consentono di risparmiare una notevole quantità di tempo, automatizzando attività ripetitive e monotone che potrebbero risultare noiose per il dipendente, compromettendo così la sua produttività. Inoltre, una volta automatizzate, vengono spesso eseguite con un grado di precisione superiore rispetto all'esecuzione manuale.

Un altro settore in cui l'IPA è molto sfruttata è quello della **supply chain e della logistica**. Grazie alle tecniche di forecasting, è possibile predire la domanda e l'inventario in modo da ridurre le spese di stoccaggio e allo stesso tempo permettere ai clienti di ricevere i prodotti richiesti. I sistemi di gestione dell'inventario basati sull'IPA, aiutano anche ad individuare strategie per piazzare sul mercato articoli "lenti" a cui i clienti non sono interessati probabilmente per via del prezzo o la poca visibilità. Inoltre, grazie a tecnologie come GPS e sensori, si può aumentare la visibilità dei prodotti durante l'intero ciclo della catena di fornitura o della logistica ma soprattutto durante la consegna negli ultimi chilometri. Grazie ad algoritmi di intelligenza artificiale è possibile ottimizzare i percorsi e i programmi di consegna rendendo così la supply chain più agile e sostenibile [9].

Nel settore **sanitario**, l'elaborazione del linguaggio naturale (NLP) è ampiamente utilizzata come tecnologia per raccogliere, analizzare e trattare i dati. Inoltre, l'interazione dei pazienti con i chatbot consente di accelerare i tempi delle visite di routine, liberando così il personale medico per dedicarsi a casi più urgenti. Anche nel settore farmaceutico, l'adozione dell'IPA può rappresentare una soluzione. Un esempio lampante è il vaccino sviluppato in tempi record per contrastare il Covid-19, grazie ad una rapidissima elaborazione di un grande volume di dati e alla riduzione degli errori di calcolo rispetto a quelli che si sarebbero verificati con un approccio manuale [10].

3. Fasi del ciclo di vita di una soluzione di IPA

Questo capitolo descrive le fasi fondamentali e necessarie per la progettazione, realizzazione ed esecuzione di un progetto di **Intelligent Process Automation (IPA)**.

È importante sottolineare che il campo dell'IPA è un territorio relativamente inesplorato, essendo una disciplina emergente nel panorama tecnologico, ed è caratterizzato da un costante stato di evoluzione, con nuove tecnologie e approcci emergenti. Di conseguenza, qualsiasi progetto di Intelligent Process Automation può subire modifiche nel corso del tempo. Pertanto, per mantenere un vantaggio competitivo nel mercato, è indispensabile adottare un approccio di miglioramento continuo e di costante adattamento ai cambiamenti che potrebbero occorrere durante l'esecuzione del progetto.

Nelle prossime sezioni di questo capitolo, sono presentate in modo dettagliato le fasi del ciclo di vita di un progetto IPA, dalla definizione dei requisiti alla pianificazione, implementazione, test e monitoraggio.

Ogni fase è discussa in termini di obiettivi e migliori pratiche in modo da fornire una linea guida per la realizzazione di un progetto IPA di successo ed evitare le problematiche più comuni in cui si potrebbe incorrere.

2.1 Definizione della metodologia di sviluppo software: Waterfall vs Agile

Prima di analizzare nel dettaglio ogni fase del ciclo di vita, è essenziale definire la **metodologia** di sviluppo software, poiché essa permette di definire dal punto di vista organizzativo le diverse fasi del progetto. Le metodologie di sviluppo software sono degli approcci sistematici che racchiudono un insieme di principi, pratiche e regole. Questi elementi sono pensati con il fine di guidare efficacemente il team nella realizzazione di un software, dalla fase iniziale di pianificazione fino alla fase finale di distribuzione.

Esistono numerose metodologie di sviluppo software, ciascuna con il proprio grado di rigidità, agilità e adattabilità. La decisione riguardo al modello da adottare dovrebbe tenere in considerazione vari fattori, come:

- **Natura del progetto:** la tipologia, la complessità, lo scopo e le specifiche tecniche del progetto, costituiscono degli elementi determinanti nella scelta della metodologia.
- **Aspettative del committente:** le aspettative del cliente, sia riguardo al risultato finale del progetto, che alle modalità di consegna (ad esempio se suddivisa in diversi deliverable), possono incidere significativamente sulla scelta. Inoltre, il grado di

coinvolgimento del cliente nel progetto, nonché quanto è disposto a partecipare attivamente, può influenzare particolarmente la metodologia da adottare. Il committente potrebbe desiderare un approccio che prevede frequenti aggiornamenti e revisioni, oppure uno che richiede meno interazioni.

- **Cultura aziendale:** le norme, i valori, le pratiche operative e il modo di pensare dei membri di una azienda rappresentano fattori chiave per il raggiungimento degli obiettivi aziendali. Ad esempio, l'apertura o la resistenza al cambiamento possono influenzare la scelta. Pertanto, è essenziale identificare la metodologia più adatta alla cultura aziendale specifica.
- **Tempistiche:** il tempo da dedicare alla realizzazione del progetto riveste un ruolo fondamentale. Le differenze sono notevoli nel caso in cui un progetto debba essere consegnato entro una data o periodo prestabilito (situazione che spesso si verifica quando i requisiti risultano ben definiti e gli obiettivi ritenuti realizzabili), rispetto a progetti innovativi e sperimentali, dove il periodo potrebbe non essere definito con precisione. Di conseguenza, per progetti con scadenze rigide e obiettivi chiari, potrebbe essere preferibile una metodologia più strutturata e prevedibile. Al contrario, progetti caratterizzati da un alto grado di incertezza sono compatibili con una metodologia flessibile che permetta iterazioni e modifiche.
- **Budget:** rappresenta le risorse finanziarie da dedicare al progetto. In funzione del budget a disposizione, la metodologia può essere differente.
- **Tolleranza al rischio:** un progetto può includere vari livelli di rischio dovuti a fattori come difficoltà tecniche, obiettivi poco chiari e non definiti correttamente, che possono portare a complicazioni, quali ritardi o costi aggiuntivi.

A seconda delle caratteristiche specifiche del progetto, è fondamentale riuscire ad indentificare la metodologia più adeguata e compatibile.

Nel corso degli ultimi decenni, sono state sviluppate svariate metodologie. Alcune di queste, come Lean e Kanban, pur avendo origini in contesti diversi dallo sviluppo software, sono state successivamente adattate ed utilizzate con efficacia anche in questo ambito.

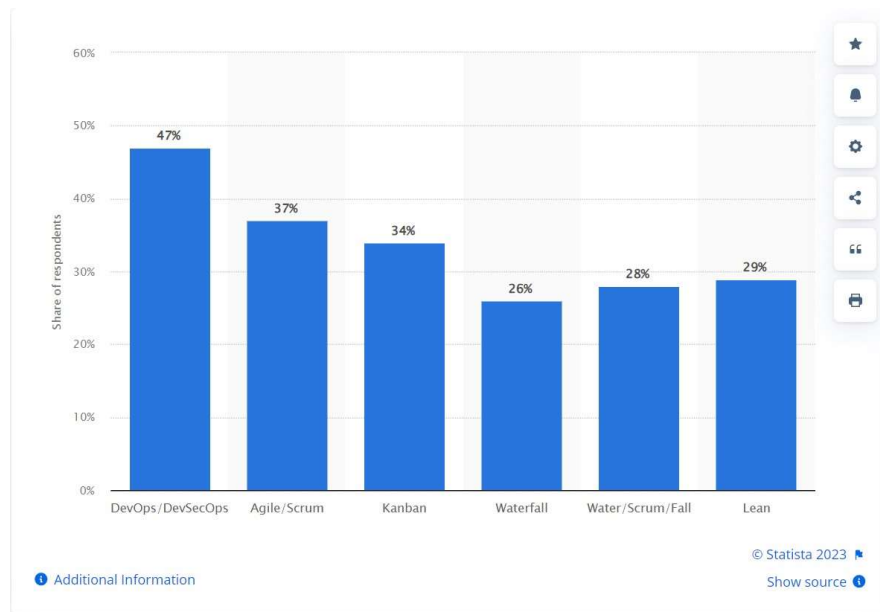


Figura 5: Distribuzione delle metodologie di sviluppo software (Statista)

Come illustrato in Figura 5, Statista, una nota piattaforma di divulgazione di statistiche, presenta dei dati molto interessanti riguardanti le metodologie di sviluppo software adottate dagli intervistati. Questi offrono un quadro significativo delle tendenze attuali nel campo di sviluppo software.

Un esempio è DevOps, una metodologia rappresentata da un connubio di pratiche, tecnologie e persone, concepita con l'obiettivo di velocizzare ed ottimizzare lo sviluppo del software. Il termine "DevOps" deriva dalla fusione dei termini inglesi "Development" (Sviluppo) e "Operation" (Operazioni).

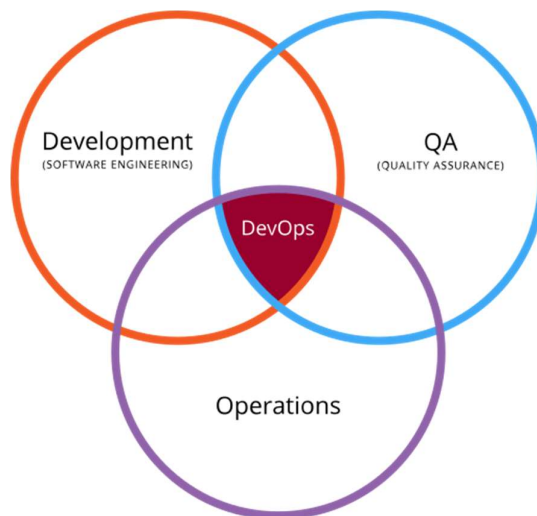


Figura 6: DevOps (Wikipedia)

È un approccio basato sulla collaborazione e integrazione delle risorse di diversi team come sviluppatori, addetti alle operation IT, controllo della qualità e della sicurezza. A differenza di altre metodologie, dove queste figure tendono ad operare in modo isolato, DevOps permette a questi ruoli di essere costantemente allineati.

Secondo i dati pubblicati da Statista, questa metodologia risulta particolarmente diffusa al giorno d'oggi. Questa tendenza è probabilmente dovuta ad una compatibilità con le esigenze di mercato attuale, che richiede una maggiore integrazione delle funzionalità e dei dati, la capacità di apprendimento continuo dei software, l'automazione del flusso di lavoro e feedback più immediati.

Tuttavia, l'analisi riportata in questo lavoro di tesi si concentra sulle metodologie Agile e Waterfall. Risultano gli approcci più antichi ed utilizzati nel corso degli anni, di conseguenza i più studiati e consolidati nel campo dello sviluppo del software. Essi rappresentano approcci distinti ma che entrambi perseguono il medesimo obiettivo: ottenere un prodotto di qualità.

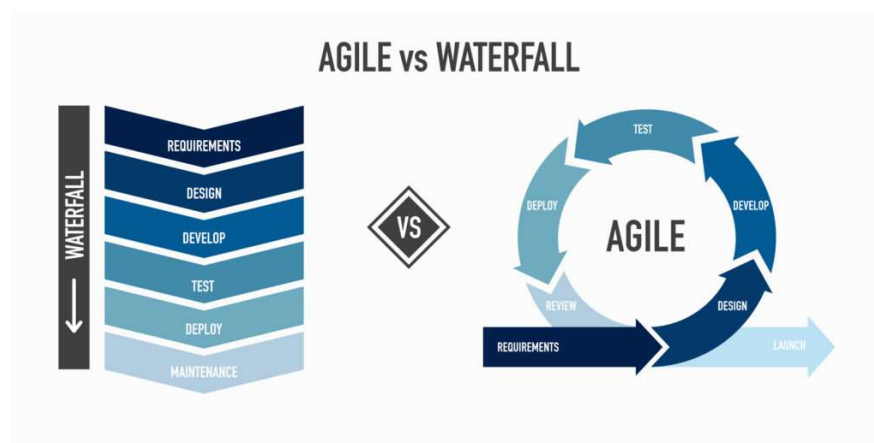


Figura 7: Agile vs Waterfall (ii)

Dal punto di vista strutturale, nonostante le differenze significative, condividono un punto in comune: entrambi adottano le stesse fasi di sviluppo del software, sebbene con approcci diversi (Figura 7).

Waterfall, conosciuto in italiano come “Metodo a Cascata”, è in ordine storico il più antico. È stato ideato dal Dr. Winston W. Royce ed introdotto per la prima volta tramite un articolo pubblicato nel 1970 dal titolo “Managing the Development of Large Software Systems”. È riconosciuto come il pioniere delle metodologie di sviluppo software, poiché ha contribuito alla definizione delle fasi-chiave fondamentali per il processo.

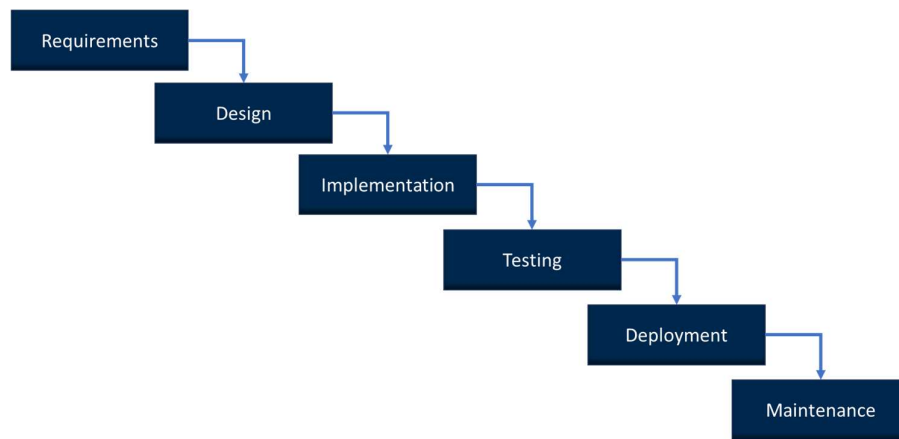


Figura 8: Fasi di sviluppo della metodologia Waterfall

In Figura 8 sono raffigurate le 6 fasi di sviluppo secondo la metodologia Waterfall: si può quindi notare il motivo per cui viene chiamata anche “*Liner Sequential Life Cycle Model*”. Essa, infatti, corrisponde ad una gestione “a cascata” delle attività, e l’avanzamento alla fase successiva richiede l’output della fase precedente. Espresso in altri termini, l’input della fase “*n*” corrisponde all’output della fase “*n-1*”.

Le fasi definite da questo approccio sono:

- **Definizione dei requisiti:** il primo passo consiste nel delineare i requisiti utili per portare a termine il progetto. Questi rappresentano la traduzione degli obiettivi in elementi tecnico-funzionali che definiscono le aspettative del progetto.
- **Progettazione:** una volta compresi i requisiti del progetto, figure chiave come il Project Manager e i tecnici collaborano al fine di pianificare le attività, allocare le risorse (sia umane che software/hardware), definire il budget e stabilire le misure di performance.
- **Implementazione:** dopo aver definito il quadro teorico, si procede con l’applicazione pratica. Si mettono in atto le conoscenze acquisite in passato, combinate alle tecnologie più appropriate, come le Intelligenze Artificiali o il Machine Learning.
- **Verifica:** il Team di sviluppo, successivamente alla creazione di un deliverable preliminare, sottopone il prodotto ad un team che ne testa la qualità. Quest’ultimo esamina a fondo l’output per verificare se i requisiti iniziali sono stati soddisfatti e se il prodotto funziona come previsto. Nel caso in cui emergano dei bug, il team di sviluppo interviene per risolverli.
- **Distribuzione:** Una volta corretti i bug e verificato il funzionamento del prodotto, il deliverable viene consegnato al cliente. In passato, questa fase prevedeva l’installazione del software, ma oggi, con l’avvento della tecnologia cloud, è sempre più comune

distribuire il prodotto finale su piattaforme come Amazon Web Services (AWS), Microsoft Azure e Google Cloud Platform (GCP).

- **Manutenzione:** Dopo la distribuzione del prodotto, potrebbero emergere nuovi bug segnalati dagli utenti. Il team di sviluppo deve essere quindi pronto ad intervenire per risolverli. Parallelamente, è importante considerare strategie finalizzate al miglioramento continuo e per eventuali aggiornamenti del prodotto.

Data la natura lineare e sequenziale di questo approccio, la metodologia è considerata rigida e la sua applicazione sta diventando sempre meno frequente. È ritenuta da molti obsoleta e applicabile solo a specifiche casistiche. Più in dettaglio, è necessario che il progetto da sviluppare presenti le seguenti caratteristiche:

- **Requisiti ben definiti e compresi:** il committente deve definire con chiarezza l'obiettivo da raggiungere ed essere in grado di farlo comprendere al team di sviluppo, che avrà bisogno di conoscere bene i requisiti richiesti per poter procedere nel modo corretto nello sviluppo del progetto.
- **Requisiti immutabili:** il committente non deve essere incline a richiedere cambiamenti significativi.
- **Coinvolgimento limitato del committente:** Il committente preferisce non essere coinvolto e ricevere direttamente il prodotto finale.

Nel caso in cui il progetto presenti le caratteristiche menzionate, la metodologia Waterfall potrebbe rivelarsi la scelta ottimale. Tale approccio permette di definire con precisione il perimetro del progetto e di seguire un percorso ben strutturato per raggiungere il prodotto finale desiderato. Adottando tale approccio, sin dal principio, si stabiliscono delle date di inizio e fine, si definiscono le consegne e si delinea il budget.

Frequentemente, all'avvio di un progetto, una delle ipotesi più importanti che non si verifica è quella dei requisiti immutabili o ben definiti. In tali circostanze, come precedentemente spiegato, la metodologia Waterfall risulta incompatibile a causa della sua rigidità. Applicarla comporterebbe un elevato grado di incertezza e una bassa probabilità di successo. Potrebbero verificarsi complicazioni come ritardi nel progetto, incremento dei costi, insoddisfazione del cliente e scarsa qualità del prodotto finale.

Questa mancanza di flessibilità ha portato all'esigenza di sviluppare un modello di sviluppo software meno strutturato e sequenziale. Di conseguenza, è nata la metodologia Agile, che offre una maggiore adattabilità e che permette di rispondere più velocemente ed efficacemente alle mutevoli esigenze di chi commissiona il progetto.

Il modello Agile nasce ufficialmente tra l'11 e il 13 febbraio del 2001, quando un gruppo di 17 pionieri della metodologia decide di incontrarsi e fondare il “Manifesto per lo sviluppo Agile del software” [11], documento che racchiude idee condivise sotto forma di 12 principi e 4 valori. Gli sviluppatori software che hanno partecipato a questo raduno erano figure di rilievo nel panorama dello sviluppo software, rappresentando approcci influenti e innovativi come Extreme Programming, SCRUM, Adaptive Software Development (ASD), Crystal e molti altri. Tali metodologie, anche se differenti sotto alcuni aspetti, inevitabilmente hanno apportato un contributo alla definizione di nuovi standard e pratiche nel settore, condividendo obiettivi comuni come l’ottimizzazione dell’efficienza, l’incremento della flessibilità e il miglioramento della qualità del processo di sviluppo software.

La metodologia Agile, pur condividendo le fasi fondamentali del modello Waterfall, le applica in modo differente. Il modello Waterfall segue un approccio lineare e sequenziale, in cui ogni fase deve essere completata prima di passare alla successiva. Come spiegato precedentemente, ciò implica che, una volta definiti i requisiti iniziali e avviato lo sviluppo, non è ammesso apportare modifiche poiché porterebbero a delle complicazioni. Al contrario, Agile è basata su un approccio iterativo e incrementale, e grazie alla sua flessibilità, consente di apportare modifiche ai requisiti o allo sviluppo durante l’intero ciclo di vita del progetto.

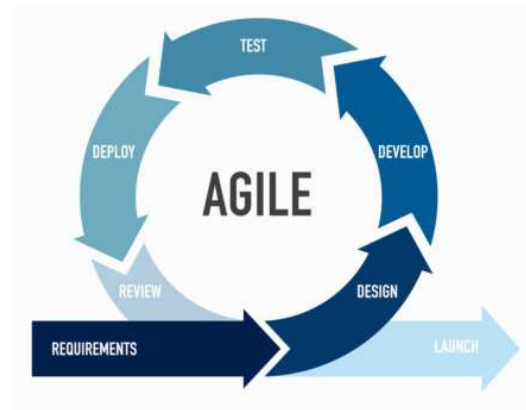


Figura 9: Agile Phases (iil)

Nel ciclo di vita di un progetto di sviluppo software dove si adotta la metodologia Agile, come illustrato in figura 9, il team di progetto inizia stabilendo una serie di requisiti che non sono considerati fissi. Successivamente, si procede attraverso le fasi di progettazione, sviluppo, test e distribuzione. Al termine di tali fasi, si raccolgono dei feedback che vengono sfruttati per stabilire nuovi requisiti da perseguire nella iterazione successiva. Questo processo iterativo viene ripetuto fino a quando non si ottiene il prodotto finale desiderato.

I 12 principi Agile enfatizzano gli aspetti prioritari che sono alla base del manifesto. Uno dei più importanti è la **soddisfazione del cliente**, la quale è possibile raggiungere tramite rilasci rapidi e continui del software. Questo meccanismo, unito ad una collaborazione quotidiana tra committenti e sviluppatori, consente di accogliere cambiamenti provenienti dal lato business e apportare le modifiche durante qualsiasi momento del ciclo di vita del progetto. In tal modo, si garantisce che al termine del progetto, il prodotto finale corrisponda alle aspettative del committente. Inoltre, secondo gli esperti della metodologia, è fondamentale fornire agli individui coinvolti sul progetto, un ambiente stimolante e una certa libertà organizzativa. Questo approccio favorisce la realizzazione di progettazioni, architetture e requisiti di qualità. La comunicazione diretta è considerata il metodo più efficiente ed efficace per scambiare informazioni con e all'interno del team, mentre gli spazi dedicati alla riflessione permettono di capire come migliorare la propria efficienza.

In sintesi, i principi descritti rappresentano le linee guida per conseguire uno sviluppo software di successo, ponendo al centro la soddisfazione del cliente, la flessibilità e l'efficienza del team di sviluppo.

È ampiamente riconosciuto che la metodologia Agile abbia numerosi vantaggi, tra cui una notevole flessibilità in termini di tempi, obiettivi, budget e risultati finali del progetto. Tuttavia, è importante sottolineare che l'applicazione di tale approccio potrebbe non essere l'opzione ideale per alcune iniziative. L'adattabilità che contraddistingue Agile implica che oltre ai risultati, anche tempistiche e budget finali potrebbero essere significativamente diversi da quanto inizialmente previsto. Inoltre, secondo uno dei quattro valori su cui si fonda il Manifesto Agile, si predilige un "software funzionante" piuttosto che una "documentazione esaustiva". Ciò significa che, sebbene Agile permetta di aumentare l'efficienza, potrebbe non essere adatto per progetti la cui documentazione dettagliata è essenziale (come, ad esempio, iniziative sanitarie e governative).

Nella tabella 1 sottostante sono riportate le principali differenze tra le due metodologie Agile e Waterfall [12], necessarie per poter decidere quale delle due si adatti meglio allo specifico progetto:

	Waterfall	Agile
Struttura	Sequenziale	Iterativa
Documentazione	Considerevole	Minimale

Idoneità	Predittiva	Adattiva
Natura	Orientata ai processi	Orientata alle persone
Efficienza	Bassa	Alta

Tabella 1: Principali differenze Waterfall vs Agile

Inoltre, i due approcci differiscono in modo speculare in termini di obiettivi, tempi e costi, come è illustrato nella seguente immagine:

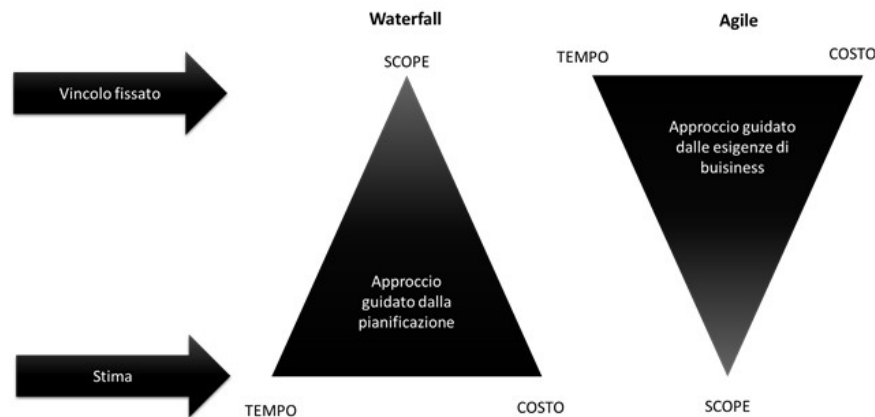


Figura 10: Differenze obiettivi, tempi e costi (Sinfo-one)

Il confronto visivo mette in evidenza i vincoli fissi, le stime e gli approcci associati alle metodologie in questione. I vincoli fissi delineano le priorità di ciascun modello, mentre le stime rappresentano le variabili che possono essere adattate durante l'esecuzione del progetto. Nel modello Waterfall, lo scopo del progetto è considerato un vincolo fisso, mentre tempo e costi sono stimati in funzione di tale scopo [13]. Questo approccio privilegia la pianificazione dettagliata, definendo, prima dell'implementazione del progetto, uno scopo chiaro. Al contrario, nell'approccio Agile, le tempistiche e i costi sono considerati vincoli fissi, mentre lo scopo del progetto è la variabile stimata. Questo permette una maggiore flessibilità consentendo al team di adattarsi alle mutevoli esigenze di business.

Da questa analisi, si evince che non esiste una metodologia considerata universalmente superiore alle altre. Piuttosto, la competenza risiede nel selezionare e implementare l'approccio che meglio si adatta alle esigenze specifiche del progetto, le esigenze del team e l'ambiente business.

3.2 Definizione dei Requisiti

La **definizione dei requisiti** rappresenta una delle fasi più importanti in qualsiasi progetto in quanto determina la direzione del progetto. Vengono identificate le attività necessarie allo

sviluppo del software dopo un'analisi approfondita del processo attuale (**As-Is**) e la successiva definizione del processo futuro (**To-Be**). Questo permette di stabilire un primo piano temporale.

Inoltre, durante questa fase, i requisiti business definiti dal committente vengono tradotti da un punto di vista tecnico, in modo che risultino comprensibili per il team di sviluppo. Questo comporta la stesura della documentazione funzionale e tecnica e la progettazione dell'architettura del sistema.

Nel caso in cui si affronti un problema mai analizzato prima, o si esplorino nuovi ambiti o tecnologie, potrebbe essere necessario condurre uno studio di fattibilità. In tal modo si assicura che il progetto sia realizzabile e che le risorse siano utilizzate in modo efficace.

Come sovente accade, ci sono aziende che desiderano sviluppare un progetto ma non dispongono internamente delle competenze o delle risorse (sia umane che infrastrutturali o software) necessarie ad implementarlo. Di conseguenza, se l'obiettivo è implementare il progetto nel breve periodo, queste aziende si rivolgono a consulenti esterni.

Le società di consulenza, a differenza delle risorse interne, hanno l'opportunità di lavorare su diversi progetti per vari clienti che, pur somigliandosi ad alto livello, presentano sempre delle peculiarità differenti. Nel corso delle loro attività, acquisiscono una vasta esperienza e si specializzano sia in ambiti di ampio respiro, che nella gestione di problemi più specifici.

Il project manager, assieme al suo team di esperti, deve sfruttare la propria esperienza e il proprio know-how per definire le attività, delimitare il perimetro del progetto, ma soprattutto interpretare le esigenze del cliente.

Questa fase è cruciale in quanto, se non è accurata o completa, si rischia di incorrere in sprechi significativi di tempo e denaro. Ciò accade poiché le risorse potrebbero essere impiegate per sviluppare funzionalità che non rispondono alle esigenze reali del cliente o che non apportano valore al progetto. Inoltre, nel caso in cui i requisiti non fossero definiti chiaramente, si corre il rischio di effettuare continue revisioni e modifiche alle attività. Questo non solo può causare ritardi nel progetto, traducendosi in una perdita economica, ma può anche generare frustrazione tra i dipendenti. Essi potrebbero trovarsi a dover implementare cambiamenti senza comprenderne il motivo, creando un ambiente di lavoro meno produttivo.

3.3 Pianificazione

La fase di **pianificazione** rappresenta un pilastro fondamentale nel ciclo di vita di una soluzione di Intelligent Process Automation. Essa è caratterizzata da una serie di sotto-attività distinte ma

interconnesse, che combinate delineano un quadro coerente e completo. Il team del progetto sviluppa un piano dettagliato, delineando gli step necessari per l'ottenimento del prodotto finale. Questo viene realizzato tramite due modalità di collaborazione: una interna, che coinvolge sviluppatori e ruoli funzionali come il Project Manager, e una esterna, che include gli stakeholder, tra cui i committenti. Si procede quindi all'identificazione delle attività, delle risorse umane necessarie, alla stima del budget e alla definizione degli indicatori di performance (KPI).

Durante la definizione delle attività, si procede all'individuazione dei compiti indispensabili per il conseguimento degli obiettivi, i quali sono stati precedentemente stabiliti nella fase di definizione dei requisiti descritta nella Sezione 3.2.

È molto comune che la complessità di un progetto possa risultare elevata. In tal caso, è buona pratica spaccettare le macro-attività, tramite una scomposizione gerarchica, in compiti più semplici. Questo approccio è rappresentato dalla **Work Breakdown Structure (WBS)**, uno strumento fondamentale che consente al Project Manager di organizzare o governare il progetto efficacemente. Inoltre, è vantaggiosa anche per il team, poiché fornisce una chiara visione del flusso di lavoro. La Figura 11 presenta un esempio di WBS adatto per progetti di sviluppo software. In blu sono indicate le fasi del ciclo di vita, a ciascuno dei quali si associano diverse macro-attività, ad esempio la definizione dei ruoli funzionali all'interno della fase di sviluppo. A loro volta, alcune macro-attività più complesse sono suddivise in micro-attività più specifiche.

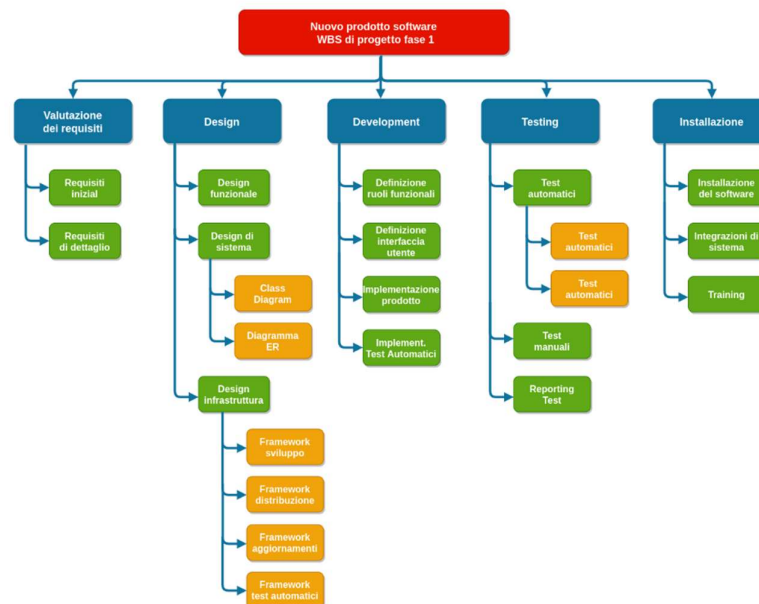


Figura 11: Work Breakdown Structure sviluppo software (Tt-ppm)

Nel contesto dell'Intelligent Process Automation, ad esempio, una delle principali macro-attività consiste nella progettazione e realizzazione dell'architettura, mentre, parallelamente, una micro-attività può concentrarsi sullo sviluppo di uno script dedicato all'automazione di compiti specifici come l'estrazione di dati da documenti. Prima di poter procedere con la creazione dello script, è fondamentale disporre delle "true label", ovvero delle etichette che rappresentano i dati in maniera corretta e che sono fondamentali per l'addestramento di modelli di machine learning. Qualora le true label non siano fornite dal committente, si commissiona alle risorse del team di progetto questa micro-attività, che per natura deve essere svolta antecedentemente allo sviluppo dello script.

Questi task sono oggetto di un'analisi approfondita che consente di quantificare gli sforzi richiesti per la loro realizzazione. Tali sforzi vengono successivamente convertiti in giorni lavorativi. Di conseguenza, ogni attività viene distribuita lungo un orizzonte temporale prestabilito, e si definiscono le date di inizio e di conclusione. Il Project Manager collabora con il team di sviluppatori per stimare il tempo necessario per l'esecuzione di ciascun compito e, sulla base dell'esperienza acquisita in progetti precedenti, determina anche il livello di "contingency" da applicare. Il termine "contingency", di origine inglese, è utilizzato per indicare un margine di tempo aggiuntivo che viene considerato al fine di compensare eventuali imprevisti o ritardi.

Uno strumento diffusissimo che consente di mappare le attività lungo un asse temporale è il **diagramma di Gantt**. Le attività sono rappresentate da rettangoli disposti in sequenza. A ciascun rettangolo è associata una serie di attributi, tra cui nome del compito e le date di inizio e di fine.

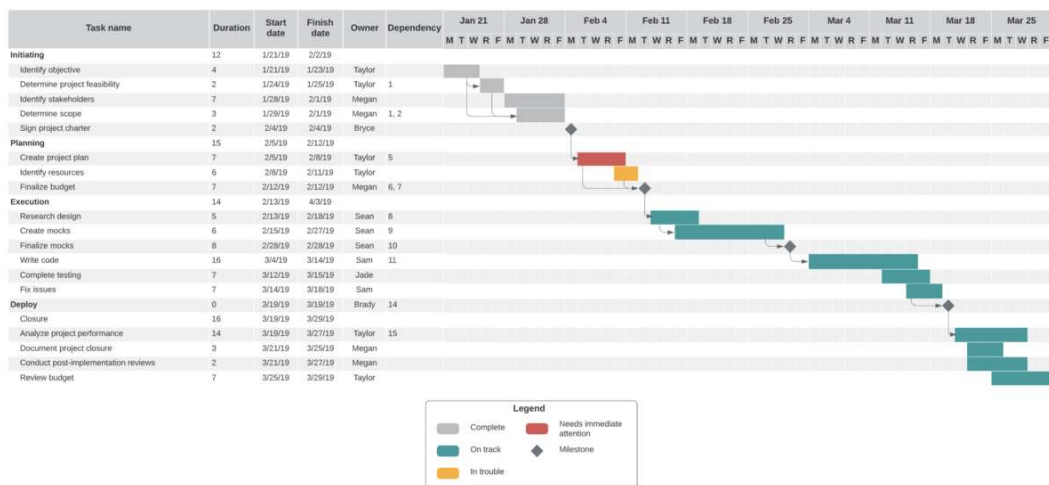


Figura 12: Diagramma di Gantt (Lucidchart)

Nonostante sia uno strumento introdotto nel 1917 dall'ingegnere statunitense Henry Laurence Gantt, la sua rilevanza e importanza rimane attuale. Nel corso degli anni, sono stati sviluppati sia software nati con finalità differenti e che successivamente hanno integrato il diagramma di Gantt, sia software concepiti con l'intento di sviluppare il diagramma, ma che hanno introdotto nuove funzionalità, pur mantenendo inalterato l'obiettivo principale: la pianificazione e il monitoraggio delle attività. Ad esempio, alcuni dei software più comuni sono Jira, Norton, MSProject, i quali utilizzano delle funzionalità innovative (come visibile in Figura 12), tra cui le dipendenze tra le attività, le milestone e l'assegnazione delle risorse alle attività. Più nello specifico:

- Le **dipendenze tra le attività** sono fondamentali per definire la sequenza logica in cui le stesse devono essere eseguite, garantendo che ogni compito sia completato prima di procedere con il successivo.
- Le **milestone** rappresentano degli obiettivi intermedi utili a monitorare l'avanzamento del progetto. Sono dei punti specifici solitamente rappresentati tramite dei rombi ed utilizzati per diversi scopi. Ad esempio, possono essere utilizzati per fornire informazioni ad alto livello, come date o eventi chiave del progetto, agli stakeholder, oppure per motivare il team.
- Infine, l'**assegnazione delle risorse alle attività** è un elemento chiave per garantire che ogni attività disponga delle risorse necessarie per la sua realizzazione, evitando così di sottostimarle o sovrastimarle.

L'abilità di un Project Manager risiede anche nella capacità di distinguere tra le attività che sono interconnesse e quelle che possono essere eseguite in parallelo. Questa distinzione facilita l'identificazione delle risorse da assegnare e consente di ottimizzare l'allocazione delle stesse. In tale modo si minimizzano i tempi morti, si massimizza l'efficienza ma soprattutto si tutela il team da un possibile sovraccarico di lavoro.

In sintesi, l'importanza del diagramma di Gantt nella fase di pianificazione di un progetto di Intelligent Process Automation è indiscutibile. Questo strumento consente di identificare le singole attività, quantificare e individuare le risorse da associare, stabilire le dipendenze tra le attività stesse e definire delle tempistiche. Fornisce una visione sia di alto livello che dettagliata, utile non solo per il team di progetto, ma anche per gli stakeholder. Se il diagramma è stato progettato ed organizzato correttamente, i benefici che ne derivano possono facilitare anche altre attività, come la pianificazione del budget. Tuttavia, per garantire una buona riuscita, è essenziale che il diagramma sia organizzato in modo chiaro e intuitivo; in caso contrario, potrebbe risultare complicato da interpretare, portando a possibili incomprensioni. Inoltre, un

aggiornamento costante del diagramma consente di monitorare lo stato di avanzamento del progetto, fornendo un quadro realistico del lavoro svolto e di quello ancora da realizzare. Di conseguenza, il diagramma di Gantt non è solo uno strumento di pianificazione, ma anche di monitoraggio.

Come appena spiegato, software come il diagramma di Gantt aiutano a identificare le risorse necessarie. Tuttavia, il project manager, conoscendo le risorse del team e la situazione attuale, deve allocarle in modo efficace ed efficiente. Di conseguenza vanno considerati diversi fattori, tra cui:

- **Ruoli e responsabilità** sono essenziali per una corretta scelta. Ci sono risorse che durante il corso della loro vita hanno studiato e appreso delle competenze specifiche. Ad esempio, delle risorse possono essersi specializzate per sviluppare l'architettura mentre altre nelle analisi sui dati.
- **Stato attuale delle risorse** aiuta a capire quanto limitate esse siano. Quando si definiscono le risorse necessarie, inizialmente si risolvono i problemi di project scheduling con risorse illimitate. In questo caso si assume che le risorse siano disponibili per sviluppare le attività come e quando richiesto. Però, situazioni del genere nella realtà sono inverosimili, quindi quelle risorse potrebbero non essere da subito disponibili poiché impegnate in altri progetti oppure non presenti. Di conseguenza, potrebbero verificarsi delle problematiche decisionali dovute a delle incompatibilità tra delle attività e le risorse.

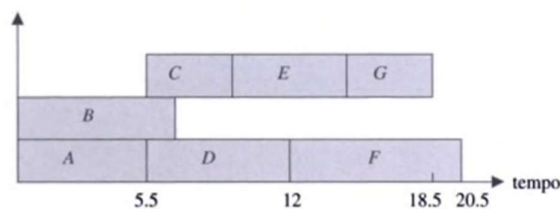


Figura 13: Diagramma di Gantt

Supponendo che le attività A e B (Figura 13) possano essere svolte da una sola risorsa che è considerata l'unica a possedere le competenze si parla di "conflitto" tra le attività. In questo caso il Project Manager deve procedere in base all'obiettivo che si vuole raggiungere.

- **Costo** delle risorse che può essere diretto oppure stimato nel caso in cui serva assumerne delle nuove.
- **Luogo geografico** delle risorse inteso sia se all'interno del Paese da cui si opera e sia se lavora in maniera telematica o fisica.

- **Calendario delle risorse** è importante per schedulare le attività in base ai giorni lavorativi, festivi e feriali.

L'allocazione ideale è legata alla funzione obiettivo stabilita dall'organizzazione. In riferimento alla Figura 13, qualora l'obiettivo fosse minimizzare i tempi legati al completamento del progetto, una delle possibili soluzioni consiste nell'ingaggiare temporaneamente una risorsa aggiuntiva che possa essere impiegata nel compito B. Di conseguenza, ipotizzando che i 2 compiti richiedano competenze equivalenti, è necessario che la nuova risorsa disponga di abilità simili a quelle dell'individuo incaricato del compito A. In tal modo si consente la parallelizzazione delle attività, favorendo il conseguimento dell'obiettivo di minimizzazione dei tempi. Naturalmente, ciò implicherebbe un incremento dei costi, che può rappresentare un vincolo; di conseguenza, il Project Manager si trova davanti a dover risolvere dei problemi di ottimizzazione vincolata.

In parallelo alla pianificazione delle attività e delle risorse, un ulteriore compito del Project Manager è quello di definire il budget. La stima dei costi per un progetto di sviluppo del software può essere fatta tramite diverse metodologie. Come precedentemente menzionato, viene stimato lo sforzo (spesso chiamato con il termine inglese "effort") necessario al completamento di ciascuna attività. L'effort è espresso con le unità di misura ore/persona o giorni/persona. Tali metriche permettono di tradurre il lavoro stimato in termini economici, consentendo così la definizione del budget. Le tecniche utilizzate differiscono le une dalle altre poiché misura l'effort in maniera differente.

Tra le più utilizzate ci sono:

- **Stima per analogia** (Analogy Based): si basa sul riutilizzo di esperienze pregresse come attività o progetti che sono analoghi o simili a quelli che bisogna stimare.

Se per esempio un'attività richiede 5 giorni lavorativi e il costo giornaliero associato allo sviluppatore risulta essere pari a 200 €, il costo totale dell'attività sarà pari a:

$$\text{Costo totale} = N^{\circ} \text{giorni} \times \text{CostoMedioGiornaliero} = 5 \times 200 \text{ €} = 1000 \text{ €}$$

- **Giudizio degli esperti** (Expert Judgment): si raggiunge un accordo comune comparando e discutendo stime di diversi esperti effettuate in maniera indipendente. Di

conseguenza, essendo legata a persone esterne, la stima dipende da fattori soggettivi e soprattutto dal grado di comprensione del problema da parte dell'esperto.

- **Stima tramite misure** (Metrics Based): si utilizza come metrica l'Effort, espressa con la stessa unità di misura ma che, a differenza della stima per analogia, si calcola in maniera differente:

$$Effort = Size_{sw} \times ProductivityNorm_{team}$$

Il primo fattore rappresenta la dimensione del software da dover realizzare, mentre il secondo la produttività del team. Esistono differenti misure che possono essere utilizzate per stimare la dimensione del software e una di queste consiste nell'utilizzare la tecnica dei Function Points. Il calcolo dei Function Points prevede la somma di 5 elementi pesati per dei pesi che dipendono dalla complessità di ciascun elemento:

$$FP = EI + EO + ILF + ELF + EQ$$

Dove:

- External Input (EI) = attività elementari di input
- External Output (EO) = attività elementari di output
- Internal Logical File (ILF) = i file interni all'applicazione
- External Interface File (EIF) = I file esterni all'applicazione
- External Inquiry (EQ) = attività elementari di interrogazione

Lo scopo di questa metodologia è quello di quantificare solamente le funzionalità che il software offre all'utente in termini di dati e processi significativi. Di conseguenza, è una tecnica utilizzata più lato committente per remunerare gli sforzi effettivi richiesti per la realizzazione dei requisiti.

Queste sono solo alcune delle tecniche possibili per stimare delle attività e di conseguenza il budget da investire per la realizzazione dell'intero progetto. A seconda del caso specifico e nell'ottica di raggiungere la massima efficienza, è necessario adottare la metodologia più idonea alle circostanze. Ad esempio, se i requisiti sono chiari, definiti e poco variabili, la stima per analogia è quella più adatta. In caso contrario, ovvero con requisiti variabili o poco chiari, la scelta ideale può ricadere nel giudizio di esperti. Attività, tempistiche, budget e approccio da utilizzare, vengono discusse tra committente e team di sviluppo con l'obiettivo di raggiungere una soluzione congeniale per tutti.

Come descritto nelle sezioni successive dell'elaborato, è essenziale che gli output di progetto siano sottoposti a delle fasi di convalidazione e test prima della consegna definitiva al cliente. Tuttavia, tali consegne devono rispettare standard di qualità prestabiliti, spesso definiti in accordo con il cliente nella fase di definizione dei requisiti. Inoltre, gli standard non devono

solamente riferirsi alla qualità del prodotto finale, ma si estendono anche ai processi e alle attività necessari al conseguimento. I deliverable generati da questi processi e attività, se non conformi agli standard di qualità desiderati, potrebbero ostacolare il raggiungimento degli obiettivi stabiliti in termini di qualità dell'output finale. Quest'ultima rappresenta l'accuratezza con cui i risultati soddisfano i requisiti concordati; per quantificarla, è necessario stabilire metriche specifiche che possano essere monitorate. Pertanto, il Project Manager, oltre a definire attività, risorse e budget è anche tenuto ad elaborare un piano di gestione della qualità del progetto, che tra le varie attività, includa la definizione di misure di performance.

Nell'ambito del miglioramento continuo, emerge spesso una delle più famose citazioni del fisico e ingegnere inglese William Thomson, meglio conosciuto come Lord Kelvin, e noto per le numerose scoperte in diversi ambiti scientifici: "Se non si può misurare qualcosa, non si può migliorare". Pertanto, seguendo questo approccio, diventa indispensabile individuare e definire degli **indicatori chiave di performance (KPI)**.

È necessario sottolineare che la progettazione dei KPI è un processo critico e non esiste un approccio puramente scientifico. Piuttosto, esistono alcune linee guida pratiche che possono aiutare a progettare efficacemente i KPI. Un KPI ben strutturato dovrebbe essere **SMART** (Specific, Measurable, Achievable, Relevant, Time-bound) ovvero:



Figura 14: KPI SMART (fieldcheck)

- **Specifico:** Un KPI troppo generico diventa difficile da misurare e da raggiungere, perché si rischia di generare fraintendimenti e di commettere errori. Ad esempio, "soddisfazione del cliente" è troppo generico e potrebbe generare risultati diversi; è necessario specificare la modalità di misurazione, che deve essere univoca e oggettiva per tutti.
- **Misurabile:** l'indicatore deve essere quantificabile e misurabile, in modo da permettere di capire a che punto ci si trova nel raggiungimento dello stesso. Ad esempio, Potrebbe trattarsi di un sondaggio sull'indice di gradimento condotto all'uscita di un negozio.
- **Ottenibile:** Ogni KPI dovrebbe avere un obiettivo associato che sia realistico. Se non è possibile migliorare su un KPI, non è un buon KPI.

- **Rilevante:** Un KPI dovrebbe essere rilevante in relazione alla strategia dell'azienda. Ad esempio, se l'azienda pone il suo focus sulla riduzione del tempo di attesa alla cassa, allora il tempo di attesa alla cassa potrebbe essere un buon KPI.
- **Time-bound:** Un KPI dovrebbe essere legato al tempo, misurato in un determinato momento e relativo a un determinato periodo di tempo. Un KPI ha significato se è conosciuta la dimensione temporale con cui è definito. Inoltre, questo permette di costruire una serie storica.

È molto importante prestare attenzione alla scelta dei KPI in base alle esigenze dell'azienda e alla sua strategia: se si scelgono troppi KPI o questi sono in contraddizione tra di loro, perderebbero di significato e l'azienda ne risentirebbe nella sua efficienza, oltre che nella motivazione dei suoi dipendenti.

Questi indicatori rappresentano la pietra miliare per la misurazione e l'incremento dell'efficienza in un'azienda. Sono utili soprattutto per le fasi successive di analisi e monitoraggio delle attività; pertanto, risultano uno strumento che permette di migliorare i processi aziendali in modo da raggiungere gli obiettivi strategici dell'organizzazione. La sfida risiede nello standardizzare, ottimizzare, implementare e generalizzare misure come qualità e flessibilità; non sempre ottimizzare misure come tempi e costi produce risultati aziendali migliori [14].

Nell'ambito dell'Intelligent Process Automation, i KPI sono utilizzati principalmente per misurare l'efficienza dei processi automatizzati. Ovviamente l'implementazione di tecnologie e tecniche di automazione, porta ad influenzare anche dimensioni più di alto livello come quelle di Strategia e Leadership. Esistono 4 tipi di KPIs [15]:

- **Financial KPIs**, ovvero il monitoraggio mensile dei benefit raggiunti dall'utilizzo dei robot;
- **Virtual Workforce KPIs**, monitorano l'utilizzo complessivo e la potenziale ottimizzazione apportata dai robot;
- **Employees KPIs**, misurano il livello di performance del team (il loro sviluppo, il training di cui necessita, ecc.);
- **Automated Processes KPIs**, tengono traccia dell'efficienza dei processi automatizzati.

Questo elaborato si concentra sulle tecnologie e processi riguardanti l'IPA, che portano ad automatizzare i processi. Di conseguenza, è opportuno porre il focus sulla quarta tipologia di KPI, ovvero quelli riguardanti l'automazione dei processi. Essi si concentrano principalmente su 2 aree: aumento dell'efficienza e riduzione del tempo. Degli esempi rilevanti sono:

- **Precisione dei dati:** Calcola la percentuale di errori di elaborazione commessi dal robot in relazione al volume totale delle transazioni elaborate.
- **Tempo medio di risposta:** Misura il tempo medio necessario al completamento di un processo automatizzato.
- **Durata Media per Risolvere Errori:** Una volta segnalato l'errore, misura il tempo medio necessario per risolverlo.
- **Riduzione dello Sforzo Umano:** Quantifica il tempo risparmiato, in termini di minuti/ore, grazie all'automazione dei processi.

Ad esempio, l'Head of Smart Process Automation and Innovation di Generali Italia, in un'intervista del 2019, afferma che l'automazione dei processi nel loro ambito assicurativo, in alcuni casi ha ridotto il tempo di processo anche più dell'80%. In particolare, la liquidazione dei sinistri salute è passata da 8 giorni a 1 giorno, mentre lo smistamento della posta interna da 24-48 ore a 5 minuti. Anche in termini di efficienza di processo, la riduzione dell'effort medio risulta essere tra il 20 e il 40% per ogni processo [16].

3.4 Sviluppo

Durante la fase di sviluppo vengono finalmente eseguite le attività identificate e pianificate nella fase precedente. Ovviamente le attività e le tecnologie da utilizzare possono variare in base all'obiettivo del progetto. Alcuni degli approcci allo sviluppo più utilizzati sono:

- **Tecniche di automazione dei flussi di lavoro**, tra cui:
 - **Version controlling**, gli sviluppatori utilizzano dei Version Control System (VCS), ovvero strumenti di controllo delle versioni per tracciare e gestire le modifiche del codice [17]. In particolare, permettono di mantenere un registro di ogni modifica del codice e condividerlo con i membri del team. Nel caso venissero commessi degli errori, gli sviluppatori possono confrontare le diverse versioni del codice in modo da individuare gli errori e correggerli o ripristinare direttamente il codice con una versione precedente. Questi sistemi facilitano la collaborazione all'interno del team di sviluppo, permettendo una condivisione del codice più rapida e riducendo al minimo possibili complicazioni. Tra i vari sistemi di controllo delle versioni, Git è uno dei più utilizzati. A differenza di altri sistemi come CVS o Subversion, Git è considerato anche un repository, in grado di contenere la cronologia completa di tutte le modifiche apportate al codice.

- **Pipeline CI/CD**, insieme di procedure volte a migliorare la distribuzione e integrazione delle nuove versioni del software rendendole più rapide e affidabili garantendo la coerenza del processo. Tramite le pipeline Continuous Integration(CI)/Continuous Deployment (CD) è possibile, in maniera continua e immediata, apportare modifiche al codice, integrarle con quello precedente e distribuirle nell'ambiente in cui si sta sviluppando [18].
 - **Test automatici**, gli sviluppatori implementano test automatici per verificare la correttezza del codice durante il ciclo di sviluppo. Esistono diverse tipologie di test che è possibile automatizzare e tra i più comuni ci sono: test unitari, test di integrazione e test di regressione. Rispettivamente, garantiscono il funzionamento delle singole parti del codice, assicurano che le unità collaborino correttamente quando sono collegate tra loro e assicurano che la riparazione o gli aggiornamenti del codice non interrompano le funzionalità di componenti che prima funzionavano correttamente [19].
- **Apprendimento automatico**: in primo luogo, i Data Scientist procedono verificando che i dati siano rappresentativi del dominio di interesse. Inoltre, capiscono se gli stessi risultano essere strutturati o non in quanto la qualità del modello di Machine Learning è fortemente influenzata dalla struttura dei dati. Anche nel caso in cui i dati siano strutturati, gli sviluppatori si devono assicurare che righe rappresentino le osservazioni e le colonne le variabili. Di conseguenza, è buona pratica utilizzare delle tecniche per riordinare il set di dati. Alcune di queste includono il Melting, lo String-splitting e il Casting. Inoltre, gli sviluppatori devono assicurarsi che i dati siano sufficienti e conformi, svolgendo attività come trasformazione degli stessi, standardizzazione, gestione degli outlier e dei valori mancanti, riduzione della dimensione e selezione delle variabili più rilevanti per il modello. In base alla struttura dei dati, alla natura del problema da risolvere e al tempo disponibile per l'addestramento, lo sviluppatore individua il modello più adatto. Successivamente, si seleziona un set di dati per l'addestramento (train) e una volta che l'algoritmo è stato addestrato, vengono utilizzate tecniche di validazione e test in modo da valutare le performance del modello sul set di dati di test. Tra le misure più comuni che gli sviluppatori utilizzano per valutare le prestazioni del modello includono la **Precision**, la **Recall** e l'**Accuracy**. In base agli obiettivi del committente, lo sviluppatore deve essere abile ad ottimizzare queste metriche in modo da migliorare le prestazioni del modello [20].
- **Integrazione di LLM**: questi modelli possono fungere da Copilot agli sviluppatori, nonché degli strumenti che li assistono nell'esecuzione di una serie di attività. Alcuni

esempi includono la generazione di ipotesi, l'analisi di dati, la codifica e l'analisi tematica, l'elaborazione del linguaggio naturale, la triangolazione dei dati, la revisione e la sintesi della letteratura, la scrittura e il debug del codice. Inoltre, gli LLM possono anche risultare un requisito del cliente da integrare nel prodotto finale. Ad esempio, possono essere utilizzati per creare ChatBot che supportano l'utente nell'esecuzione di determinati compiti o nella risoluzione di dubbi, come ad esempio la procedura necessaria per un reso. Questo tipo di soluzioni migliorano molto la user experience dell'utente. Tendenzialmente, esistono 2 tipi di ChatBot, quelli basati su delle regole (Rule Based ChatBot) e quelli basati sull'intelligenza artificiale (AI). Per i primi, lo sviluppatore deve progettare il flusso di lavoro identificando le possibili domande che l'utente farà quando interagirà con il Bot, sviluppare il Bot utilizzando linguaggi di programmazione specifici o framework di sviluppo di bot e infine testarlo per assicurarsi che funzioni come previsto. I Bot basati su AI prevedono attività come la selezione, l'integrazione e l'addestramento del modello di linguaggio. Questo potrebbe richiedere l'utilizzo di API (Application Programming Interface) specifiche fornite dal produttore del modello di linguaggio [21], [22].

- **Implementazione dell'architettura:** in questo ambito, esistono 2 figure considerate fondamentali, il Cloud Architect e il Cloud Engineer. Il Cloud Architect è colui che progetta delle soluzioni orientate al futuro, compatibili con le esigenze dei clienti e seleziona i servizi necessari all'implementazione. Definendo l'architettura del sistema si delineano i componenti del sistema e il modo in cui essi interagiscono tra di loro. In altre parole, gestisce il flusso di dati. Spesso, in accordo con il business, sceglie il fornitore di servizi che più è adatto al caso specifico e alle competenze tecniche del team. Tra i provider più comuni ci sono Amazon Web Services (AWS), Google Cloud Platform (GCP), Azure. Ha il compito di comunicare i designs delle sue soluzioni ai membri del team, in particolare al Cloud Engineer che è colui che implementa la soluzione pensata dal Cloud Architect. Il Cloud Engineer è responsabile della creazione di funzioni e applicazioni interconnesse, della configurazione di processi, della protezione dei dati aziendali del cloud attraverso misure di sicurezza e infine della formazione di colleghi e stakeholder per garantire un utilizzo efficace e sicuro della tecnologia cloud.

3.5 Test

Durante la fase di **test** si eseguono determinate verifiche per assicurare che il prodotto finale sia conforme ai requisiti stabiliti e, di conseguenza, soddisfi le aspettative del committente. Per eseguire questi test, si ricorre a particolari ambienti di non produzione [23], che frequentemente coincidono con quelli di sviluppo. Sebbene sia consigliabile utilizzare ambienti distinti per una gestione ottimale delle versioni, talvolta vengono combinati a causa di limitazioni legate all'hardware, al tempo o ad altre risorse. È sicuramente conveniente separare gli ambienti di non produzione da quelli di produzione, che presentano differenze sia logiche che fisiche. Gli ambienti di test, facenti parte degli ambienti di non produzione, vengono condivisi tra gli sviluppatori e i committenti in modo tale che si possano effettuare delle verifiche sul funzionamento.

È necessario sottolineare che esistono molte tipologie di test che possono essere applicate anche durante altre fasi del ciclo di vita di un progetto di Intelligent Process Automation. Ad esempio, nella fase precedente di sviluppo viene descritta dell'importanza per gli sviluppatori di effettuare dei test automatici come quelli di unità che sono indispensabili per verificare che ciò che si sta implementando sia corretto dal punto di vista strutturale e logico.

Tuttavia, si identificano due macrocategorie di software testing nelle quali i test possono essere classificati:

- **Black box testing:** tecnica utilizzata per verificare le funzionalità del prodotto. Bisogna immaginarla come una scatola chiusa dove il tester non può accedere a contenuti tecnici come il codice sorgente.

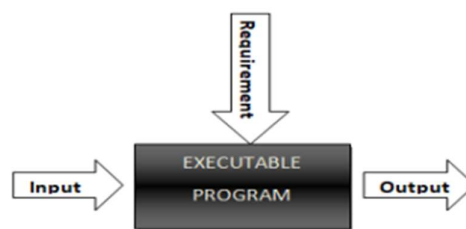


Figura 15: Black Box Testing (A Comparative Study of Black Box Testing and White Box Testing)

Il software tester ha a disposizione solo le informazioni da inserire nella scatola nera, nonché i dati di input, e utilizzando la propria conoscenza dei requisiti funzionali del progetto, verifica che l'output della scatola nera sia corretto (Figura 15). In caso contrario deve segnalare le anomalie al team di sviluppo che procederà a risolvere il problema. Questa tecnica è chiamata anche "Specification Based Testing" o "Behaviour Testing" proprio perché si verifica se il comportamento del prodotto è coerente con quello che si

era prestabilito. Frequentemente, l'individuo che conduce i test seguendo questa metodologia, è un membro funzionale lato committente, e soprattutto, una figura distinta dallo sviluppatore. Questo accade per prevenire il “developer-bias”, cioè la tendenza dello sviluppatore ad influenzare inconsciamente i risultati dei test.

Tra i test più comuni ci sono:

- **System Testing:** si valuta la conformità del sistema ai requisiti corrispondenti testando il sistema in diverse situazioni. Questo processo include la valutazione delle prestazioni del sistema, misurando caratteristiche come la velocità, la scalabilità, la stabilità e l'affidabilità del prodotto. Inoltre, si effettuano test di carico e di stress per assicurare l'integrità del sistema sotto condizioni di carico estreme e variabili.
 - **Acceptance Testing:** spesso denominato User Acceptance Testing (UAT), coinvolge gli utenti finali nel processo di verifica del software per assicurarsi che funzioni come previsto. Questo tipo di test, se superato con successo, segna il passaggio alla fase di implementazione in quanto si considera il software pronto per essere utilizzato in un contesto reale. In altre parole, l'UAT serve a confermare che il software risponde adeguatamente alle esigenze degli utenti.
- **White box testing:** tecnica utilizzata per testare la struttura dell'applicazione. In particolare, si verificano le logiche interne e la struttura del codice. In questo caso, bisogna interpretarla come se fosse una scatola aperta che il tester può esaminare più nel dettaglio.

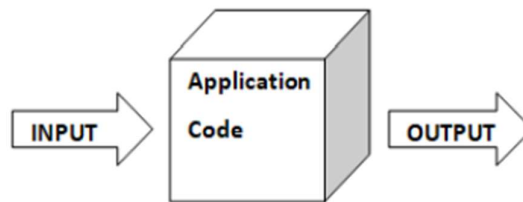


Figura 16: White Box Testing (A Comparative Study of Black Box Testing and White Box Testing)

Infatti, questa tecnica è anche chiamata “Code-Based Testing” o “Structural Testing” e a differenza del Black Box Testing, per adottarla sono richieste competenze tecniche come scrivere codice e script, familiarità con il linguaggio di programmazione in uso ed informazioni dettagliate dell'applicazione. Generalmente, chi conduce questi test è la figura che ha condotto lo sviluppo e che quindi ha scritto il codice.

- Lo **Unit Testing** è un test appartenente alla categoria del White Box Testing. Infatti, è un test basato sul codice e viene adottato per testare ogni singola unità separatamente. Viene effettuato generalmente per piccole unità di codice che

possono essere singole funzioni, metodi o classi nella programmazione ad oggetti.

I test soprariportati appartengono quindi a una delle due macrocategorie descritte, ovvero la Black Box Testing e la White Box Testing. Tuttavia, esistono dei test particolari che possono ricadere in entrambe le categorie come si può evincere dalla Figura 17 in corrispondenza della colonna **Opacity** [24], [25].

Testing Type	Opacity	Specification	Who will do this testing?	General Scope
Unit	White Box Testing	Low-Level Design Actual Code structure	Generally Programmers who write code they test	For small unit of code generally no larger than a class
Integration	White & Black Box Testing	Low and High Level Design	Generally Programmers who write code they test	For multiple classes
Functional	Black Box Testing	High Level Design	Independent Testers will Test	For Entire product
System	Black Box Testing	Requirements Analysis phase	Independent Testers will Test	For Entire product in representative environments
Acceptance	Black Box Testing	Requirements Analysis Phase	Customers Side	Entire product in customer's environment
Beta	Black Box Testing	Client Adhoc Request	Customers Side	Entire product in customer's environment
Regression	Black & White Box Testing	Changed Documentation High-Level Design	Generally Programmers or independent Testers	This can be for any of the above

Figura 17: Tipologia di Test e caratteristiche (Black Box and White Box Testing Techniques - A Literature Review)

3.6 Implementazione

La fase dell'**implementazione** è una delle più importanti di un progetto di Intelligent Process Automation in quanto comporta la distribuzione del prodotto finale al cliente. È un processo che include diverse attività correlate tra loro tra cui la configurazione del software, l'installazione del software nell'ambiente di esecuzione e l'attivazione del software. Il primo passo per l'implementazione è distribuire il prodotto in un ambiente di produzione. Successivamente, è necessario monitorare le prestazioni del prodotto e lo stato dell'ambiente di produzione in modo da individuare potenziali problemi relativi all'ambiente software, alle

prestazioni del database oppure errori di tipo HTTP. Tramite metriche e smoke test (test preliminari che verificano le funzionalità di base di un programma, fornendo un feedback rapido su aspetti critici ed elementari del software), è possibile capire se l'implementazione è avvenuta con successo o meno. In caso contrario è buona pratica eseguire dei rollback automatizzati che sono necessari per passare automaticamente alla versione precedente all'implementazione. Un metodo efficace per far ciò è quello di utilizzare dei Track logs, ovvero dei "registri di traccia" che contengono informazioni sul funzionamento del software. L'utilizzo di questi registri è essenziale per capire come il software viene eseguito sui componenti dell'infrastruttura, investigare sugli errori e identificare possibili minacce di sicurezza [26].

Al fine di distribuire il software efficacemente, è possibile adottare alcune best practices, tra cui:

- Adottare l'approccio di **Continuous Deployment (CD)**, ovvero un processo in cui le modifiche del software sono automaticamente testate e distribuite negli ambienti di produzione. Questa pratica fornisce numerosi vantaggi come l'aumento dell'efficienza e della velocità, la riduzione dell'errore umano, maggiore scalabilità e riduzione dei costi.
- Adottare il **Serverless Computing**. È un modello di cloud computing in cui i fornitori di servizi cloud come AWS, GCP e Azure, gestiscono l'allocazione e la manutenzione di risorse di calcolo, liberando lo sviluppatore dalla gestione dell'infrastruttura, e quindi dalla gestione dei server.
- Adottare l'**Infrastructure as Code (IaC)**, una pratica che permette di utilizzare un linguaggio di codifica descrittivo per automatizzare la creazione e configurazione dell'infrastruttura IT. In tal modo, gli sviluppatori, ogniqualvolta hanno necessità di sviluppare, eseguire test o implementare, non devono gestire manualmente server, sistemi operativi, connessioni al database o storage dei dati [27].

3.7 Manutenzione

Dopo aver rilasciato il prodotto software e dopo che gli utenti hanno iniziato ad utilizzarlo, inizia la fase di manutenzione. È molto raro che dei prodotti una volta implementati non necessitino di aggiustamenti, in quanto possono sorgere ancora piccoli bug da correggere, possono essere richieste nuove caratteristiche tecniche, di mercato, norme di legge ecc. oppure viene richiesto di migliorare caratteristiche del software come le performance, la robustezza

ecc. [28], [29] Queste modifiche vengono integrate dallo sviluppatore ed è possibile classificarle in 4 categorie:

- **Manutenzione correttiva:** lo sviluppatore riceve delle segnalazioni di bug o errori, spesso individuati durante la fase di test, che possono essere classificati in:
 - **Errori di Design**
 - **Errori Logici**
 - **Errori di codice**
- Dopo aver verificato che effettivamente risultano degli errori, lo sviluppatore procede al rilascio di **Patch**, nonché parti di codice che correggono i bug e gli errori precedentemente sfiggiti.
- **Manutenzione adattiva:** lo sviluppatore effettua dei cambiamenti a causa di una mutazione dell'ambiente circostante. Degli esempi possono essere cambiamenti dovuti alla legislazione o a nuove piattaforme.
- **Manutenzione evolutiva:** lo sviluppatore interviene modificando o arricchendo delle funzioni del programma a causa di nuove necessità operative richieste dall'utente al fine di ottimizzare le prestazioni.
- **Manutenzione preventiva:** lo sviluppatore utilizza pratiche come apprendimento automatico e analisi di dati e serve per affrontare piccole problematiche in modo da prevenire difficoltà più gravi che potrebbero accorrere nel futuro.

Il pensiero comune di manutenzione si riflette nella definizione di manutenzione correttiva, e quindi orientata a risolvere diverse tipologie di errore, ma come ho spiegato nel paragrafo precedente, esistono altre tipologie che servono a implementare migliorie sul prodotto. In Figura 18 è illustrato un possibile raggruppamento delle tecniche di manutenzione secondo le caratteristiche appena descritte.

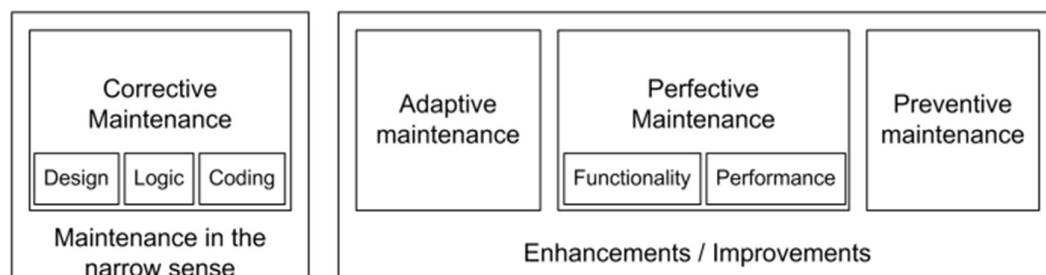


Figura 18: Tipologie di Manutenzione (Software-Maintenance Lecture Notes)

In Figura 19 sono presenti dati interessanti che indicano come circa l'80% dello sforzo per la manutenzione sia dedicato alla manutenzione adattiva ed evolutiva, mentre il 20 % alla manutenzione correttiva.

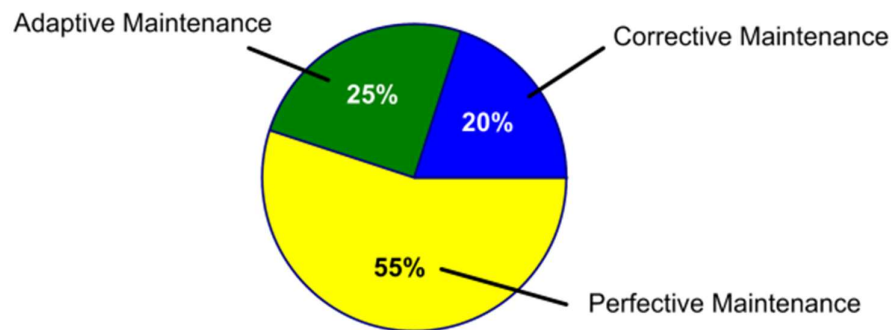


Figura 19: Grafico a torta Tipologie di Manutenzione (Software-Maintenance Lecture Notes)

Le migliori pratiche su cui concentrarsi per ottenere una manutenzione efficace sono:

- Sviluppare un **piano di manutenzione** che definisca come verrà mantenuto nel tempo il software;
- Redigere una **documentazione accurata** sull'architettura del sistema, sul codice sorgente, sui test e sulle procedure di distribuzione;
- Monitorare e valutare regolarmente le **prestazioni** del sistema analizzando parametri come il tempo di attività del sistema e l'utilizzo di risorse;
- Mappare il **cambiamento del comportamento** del sistema richiesto dall'utente;
- **Comprendere al meglio il software** in modo tale da apportare modifiche accurate ed evitando così di introdurre effetti inattesi;
- Utilizzare un approccio di **Test-Driven Development (TDD)**, cioè lo sviluppo di test automatici antecedente allo sviluppo del codice sorgente;
- Riconvalidare la modifica [30].

3.8 Monitoraggio

In aggiunta alle fasi del ciclo di vita descritte nelle sezioni precedenti, la fase di monitoraggio è uno stadio fondamentale per la corretta riuscita del progetto. Non si tratta della fase finale del progetto, quanto più di una serie di attività da soddisfare parallelamente alle altre fasi del ciclo di vita.

Essa, infatti, garantisce che il progetto sia in linea con gli obiettivi prefissati e con i tempi e il budget stabiliti. Il Project Manager è il responsabile principale di questa serie di attività.

Alcuni esempi sono:

1. **Identificazione dei processi:** Il PM può aiutare a identificare i processi che possono beneficiare dell'automazione e a valutare l'impatto potenziale sull'organizzazione.
2. **Progettazione del processo:** Il PM può supervisionare la progettazione del processo, assicurando che sia in linea con gli obiettivi del progetto e le esigenze dell'organizzazione.
3. **Sviluppo:** Durante la fase di sviluppo, il PM può monitorare il progresso e assicurarsi che il lavoro sia completato secondo i piani.
4. **Test:** Il PM può coordinare la fase di test, assicurando che tutti gli aspetti dell'automazione siano adeguatamente testati prima dell'implementazione.
5. **Implementazione:** Durante l'implementazione, il PM può monitorare l'introduzione dell'automazione nell'ambiente di produzione e gestire eventuali problemi che possono sorgere.
6. **Manutenzione e miglioramento continuo:** Anche dopo l'implementazione, il PM continua a monitorare l'efficacia dell'automazione e a coordinare eventuali aggiornamenti o miglioramenti necessari.

In ogni fase, il PM dovrebbe anche comunicare regolarmente con tutte le parti interessate, fornendo aggiornamenti sullo stato del progetto e raccogliendo feedback per miglioramenti futuri.

4. Process Mining: Teoria, Benefit e Sfide

In questo capitolo viene esplorata la tecnologia del Process Mining che, come anticipato nella sezione 2.2, è una tecnologia che può essere impiegata in concomitanza ad altre per guidare lo sviluppo di un progetto di Intelligent Process Automation. In particolare, si esplorano in dettaglio i concetti preliminari fondamentali per comprendere il Process Mining, si delineano le tre modalità di utilizzo, si discutono alcuni dei principi guida da seguire, le possibili sfide che possono emergere e i benefici derivanti dall'adozione di questa tecnologia. Inoltre, viene illustrato l'applicazione pratica di un software leader nel campo del Process Mining: Celonis. Questo capitolo fornisce quindi una panoramica completa e approfondita del Process Mining e del suo ruolo nell'ambito dell'Intelligent Process Automation.

4.1 Concetti Preliminari

Prima di spiegare cos'è il Process Mining è fondamentale stabilire una solida base di comprensione. Questa sezione introduce alcuni concetti chiave che sono essenziali per capire come e perché si utilizza il Process Mining. Questi concetti, che includono il Business Process, il BPMN, la Rete di Petri, gli Event Log e il Data Model, sono componenti che giocano un ruolo cruciale nella comprensione e implementazione del Process Mining.

Per adottare il Process Mining, è essenziale avere una chiara idea di cosa sia un processo. La definizione di **Business Process** secondo Mathias Weske, ideatore del libro "Business Process Management, Concept, Languages, Architectures", è la seguente:

“Un processo aziendale consiste in un insieme di attività che vengono svolte in coordinamento in un ambiente organizzativo e tecnico. Queste attività realizzano congiuntamente un obiettivo aziendale.”[31]

L'obiettivo aziendale è inteso come un output del processo aziendale, che può essere un servizio o un prodotto. Un esempio di processo aziendale può essere l'ordine e la consegna di una pizza.

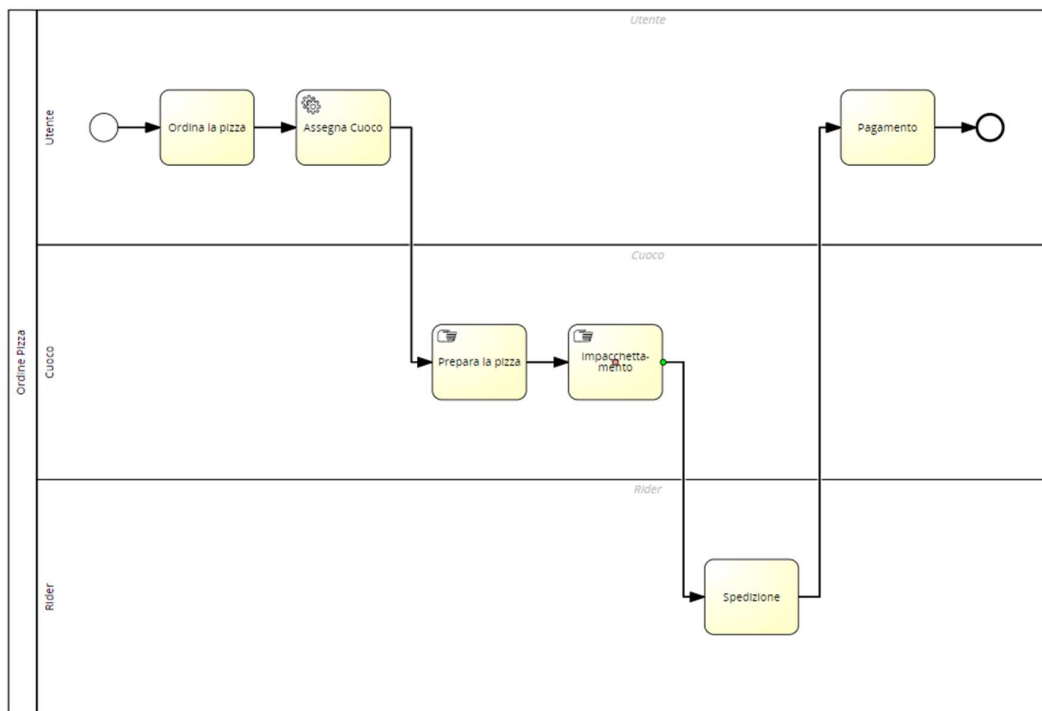


Figura 20: Esempio di processo illustrato tramite il BPMN (Signavio)

Le attività più comuni di uno scenario ideale sono quelle illustrate nella Figura 20, ovvero l'utente ordina una pizza da un'applicazione o un sito web, l'ordine viene assegnato automaticamente a un cuoco che successivamente prepara la pizza. Poi viene impacchettata, spedita e l'utente finale paga all'arrivo. Oltre al fatto che l'ordine cronologico delle attività potrebbe essere differente (ad esempio, il pagamento può essere effettuato al momento dell'ordine), il flusso potrebbe non seguire questo scenario ideale. Per esempio, il cuoco si può confondere nel comporre la pizza, oppure l'addetto alla spedizione può sbagliare indirizzo, oppure il pagamento può fallire per qualche motivo.

Quindi manualmente, sarebbe possibile rappresentare graficamente questo flusso tramite un modello di **Business Process Model and Notation** (BPMN). Esso permette principalmente di comprendere e descrivere le operazioni o le azioni svolte sui dati da parte del sistema informativo, in seguito alle interazioni con gli utenti, ma anche di includere attività che avvengono esternamente al sistema informativo (come, ad esempio, "impacchettamento della pizza") ma che comunque lascino traccia nel sistema. Il BPMN, dunque, fornisce in maniera visuale e immediata come le informazioni viaggino all'interno del sistema, rendendole comprensibili oltre che al team di sviluppo, anche ad altri attori come gli stakeholder. Gli elementi principali di questo diagramma sono:



Figura 21: Costrutti principali di BPMN

- **Eventi (Event)**: rappresentano la connessione tra il processo stesso e l'esterno. Essi si distinguono in eventi d'inizio, di fine e intermedi.
- **Attività o compiti (Activity/Task)**: ogni azione è rappresentata da un rettangolo con vertici arrotondati, i quali contengono la descrizione sintetica dell'azione. È possibile classificare le attività in azioni manuali (per es. "prepara la pizza"), azioni gestite dal sistema informativo con l'ausilio degli utenti (per es. "ordina la pizza") e azioni svolte dal sistema informativo in completa autonomia (per es. "assegna cuoco").
- **Flussi (Flow)**: frecce che collegano due elementi consecutivi, definendo l'ordine con cui vengono eseguiti gli elementi che descrivono i passi del processo.
- **Portali (Gateway)**: rombi contrassegnati da un simbolo che descrive la regola da seguire per instradare l'esecuzione dei passi. I portali più comuni sono quello esclusivo e parallelo. Il primo indica la scelta esclusiva di uno tra i flussi entranti o uscenti da esso e si rappresenta con una "X" all'interno del rombo, mentre il secondo indica la partenza di più flussi indipendenti tra loro e si rappresenta con un "+". Il primo rappresenta delle condizioni di OR e il secondo quelle di AND [32].

Tutti questi elementi sono modellati all'interno di Pool, graficamente rappresentato da un rettangolo con un comparto sulla sinistra che include il nome del partecipante o servizio. Nell'esempio soprastante, è possibile individuare 3 pool appartenenti all'utente che ordina la pizza, il cuoco che la prepara e il rider che la spedisce.

Un altro modello molto utilizzato in sostituzione al BPMN è la **Rete di Petri** [33]. Entrambi condividono lo stesso concetto di base, ma la loro rappresentazione differisce leggermente. Infatti, sono modelli interscambiabili, il che significa che qualsiasi processo modellato tramite uno può essere rappresentato anche con l'altro. Nella Rete di Petri, le attività vengono rappresentate attraverso i quadrati, formalmente noti come "**transizioni**", mentre i "**posti**", che rappresentano le condizioni o gli stati del sistema, sono rappresentati da cerchi. Nel BPMN, i Gateway più comuni sono utilizzati per rappresentare sia le condizioni di XOR che quelle di AND, mentre nel caso della Rete di Petri, se da una transizione (quadrato) partono o arrivano più frecce, vuol dire che si verificano condizioni di AND-split e AND-join. Invece, le

condizioni di OR funzionano in modo simile a quelle del BPMN. In Figura 22 un esempio concettuale che illustra l'utilizzo dei vari componenti.

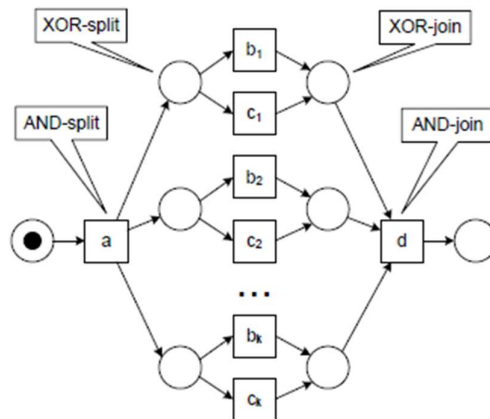


Figura 22: Costrutti principali della Rete di Petri

Tuttavia, rappresentare questi flussi manualmente potrebbe comportare dei problemi in quanto si potrebbero commettere errori o sviste, portando ad un modello inesatto o incompleto. Il Process Mining rappresenta una soluzione a questo problema, in quanto fornisce una rappresentazione oggettiva del flusso tramite gli Event Logs.

Ma cosa rappresentano esattamente gli Event Logs? Per rispondere a questa domanda, è fondamentale comprendere le basi su cui si fondano.

Ogni volta che svolgiamo un'attività tramite un sistema informativo, lasciamo delle tracce sotto forma di dati chiamate “**Digital Footprints**”, ovvero impronte digitali. Queste tracce contengono le seguenti informazioni:

- **Case ID**: un identificatore univoco correlato a delle attività e dei timestamp che serve ad individuare l'istanza, ovvero una singola esecuzione del processo. Ad esempio, il numero di ordine è un ottimo identificatore per distinguere anche 2 ordini contenenti gli stessi prodotti ma effettuati in momenti differenti.
- **Activity Description**: informazioni riguardanti la natura dell'attività che servono a far capire che azioni svolge l'utente tramite il sistema informativo, come quelle mostrate nella Figura 20 precedente.
- **TimeStamp**: informazioni relative al momento in cui l'attività è stata svolta, quindi, data e ora in cui è avvenuta.

La combinazione di questi 3 dati è chiamata Digital Footprints e fornisce informazioni su cosa, quando e da chi è stata svolta una determinata attività. Sono le minime informazioni necessarie per implementare il Process Mining, che sfrutta in modo oggettivo per generare

automaticamente il flusso di processo. Tutte queste Digital Footprints vengono raccolte in un registro noto come **Event log**. Più informazioni vengono registrate, più utili diventano per estrarre conoscenza.

Event Log		
Order Number	Activity	Timestamp
135	Selection of Item in the Menu	28.02.2020 18:37:00
135	Payment	28.02.2020 18:46:00
135	Preparation of the Meal	28.02.2020 18:48:00
135	Pick up by Deliverer	28.02.2020 19:06:00
135	Arrival	28.02.2020 19:26:00
136	Selection of Item in the Menu	29.02.2020 14:26:00
136	Payment	29.02.2020 14:34:00
136	Preparation of the Meal	29.02.2020 14:45:00
136	Pick up by Deliverer	29.02.2020 15:02:00
136	Arrival	29.02.2020 15:46:00

Figura 23: Event Log (Celonis)

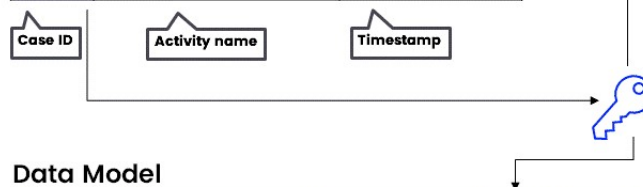
Come descritto in precedenza, la tupla di dati visualizzata nella Figura 23, costituisce l'insieme minimo di informazioni necessarie per l'implementazione del Process Mining. Tuttavia, nei sistemi informativi, è possibile registrare ulteriori dettagli nella stessa tabella degli Event Log o in tabelle separate. Nel caso in cui tali informazioni siano presenti in tabelle distinte, è possibile correlare i dati attraverso a un Case ID comune. Questo permette la creazione di un **data model**, quindi un Event Log più esteso.

Event Log

Order No.	Activity	Time
10001	Create purchase order	01-01-2009, 8:35 am
10001	Print and send purchase order	03-01-2009, 12:13 am
10001	Goods receipt	07-01-2009, 07:01 am
10001	Scan invoice	09-01-2009, 2:00 pm
10001	Book invoice	10-01-2009, 10:30 am
10002	Create purchase requisition	02-02-2009, 1:17 pm
10002	Create purchase order	04-02-2009, 9:15 am
10002	Print and send purchase order	07-02-2009, 4:41 pm
10002	Goods receipt	27-02-2009, 6:53 am
10002	Scan invoice	28-02-2009, 1:00 pm
10002	Book invoice	13-03-2009, 11:59 am
10003	Scan invoice	13-04-2009, 10:00 am
10003	Create purchase order	17-04-2009, 3:47 pm
10003	Print and send purchase order	17-04-2009, 5:30 pm
10003	Goods receipt	27-04-2009, 4:23 pm
10003	Book invoice	30-04-2009, 8:50 am

Case Table

Order No.	Net Order Value	Vendor	Company Code
10001	5337.98	Unisono SE	100
10002	250.30	Piccolo Ltd.	600
10003	12.17	Poly AG	100



Data Model

Order No.	Activity	Time	Net Order Value	Vendor	Company Code
10001	Create purchase order	01-01-2009, 8:35 am	5337.98	Unisono SE	100
10001	Print and send purchase order	03-01-2009, 12:13 am	5337.98	Unisono SE	100
10001	Goods receipt	07-01-2009, 07:01 am	5337.98	Unisono SE	100
10001	Scan invoice	09-01-2009, 2:00 pm	5337.98	Unisono SE	100
10001	Book invoice	10-01-2009, 10:30 am	5337.98	Unisono SE	100
10002	Create purchase requisition	02-02-2009, 1:17 pm	250.30	Piccolo Ltd.	600
10002	Create purchase order	04-02-2009, 9:15 am	250.30	Piccolo Ltd.	600
10002	Print and send purchase order	07-02-2009, 4:41 pm	250.30	Piccolo Ltd.	600
10002	Goods receipt	27-02-2009, 6:53 am	250.30	Piccolo Ltd.	600
10002	Scan invoice	28-02-2009, 1:00 pm	250.30	Piccolo Ltd.	600
10002	Book invoice	13-03-2009, 11:59 am	250.30	Piccolo Ltd.	600
10003	Scan invoice	13-04-2009, 10:00 am	12.17	Poly AG	100
10003	Create purchase order	17-04-2009, 3:47 pm	12.17	Poly AG	100
10003	Print and send purchase order	17-04-2009, 5:30 pm	12.17	Poly AG	100
10003	Goods receipt	27-04-2009, 4:23 pm	12.17	Poly AG	100
10003	Book invoice	30-04-2009, 8:50 am	12.17	Poly AG	100

Figura 24: Data Model (Celonis)

La Figura 24 mostra la creazione di un Data Model attraverso la correlazione delle chiavi; in questo specifico caso, il Case ID in comune alle due tabelle corrisponde al numero di ordine. Inoltre, questa immagine evidenzia anche alcuni dati che possono risultare utili per le analisi come il nome del venditore o il valore netto dell'ordine. Altri esempi di queste informazioni aggiuntive possono includere la risorsa che ha eseguito una specifica attività, il codice univoco dell'azienda che sta vendendo un determinato prodotto, il costo associato a quella specifica attività, e così via. Queste informazioni supplementari arricchiscono l'Event Log, rendendolo uno strumento ancora più efficace per l'analisi dei processi.

Avendo stabilito una solida comprensione dei concetti preliminari, tra cui Business Process, BPMN, Rete di Petri, Event Log e Data Model, è ora possibile procedere con una discussione più approfondita sul Process Mining.

4.2 Process Mining

Il Process Mining, oltre ad essere considerato come una componente fondamentale dell'IPA, lo si può vedere come un sottoinsieme della Business Intelligence. La Business Intelligence non è altro che un insieme di pratiche, tecnologie e strategie che si utilizzano per ricavare conoscenza ulteriore dai dati. Infatti, la tecnologia spiegata in questa sezione è utile sia come strumento per guidare l'automazione dei processi e sia per ottenere informazioni dettagliate che aiutano il business nel decision making.

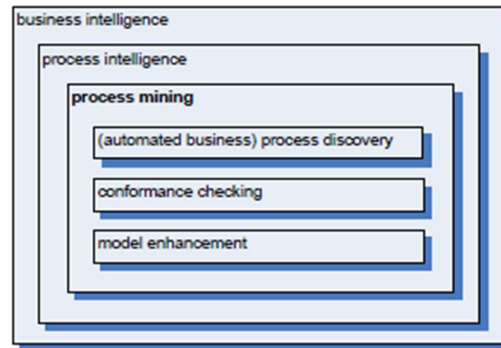


Figura 25: Business Intelligence & Process Mining (Process Mining Manifesto)

Per aziende di grandi dimensioni, con centinaia di uffici e migliaia di dipendenti, tracciare i propri processi aziendali può risultare complicato. Tradizionalmente, le organizzazioni dovevano basare la mappatura dei propri flussi di lavoro su assunzioni che potevano portare a una visione errata o inaccurata del processo. Inoltre, questa attività, oltre a portare inevitabilmente ad errori dovuti alla natura manuale del compito, comporta una serie di inefficienze come aumento di costi e spreco di tempo.

Negli ultimi 20 anni, è stata scoperta e implementata questa nuova tecnologia principalmente finalizzata alla scoperta, monitoraggio e miglioramento di processi aziendali in maniera automatizzata e attraverso l'utilizzo di dati oggettivi presenti nei sistemi informativi: gli “**Event Logs**”.

Come si può evincere dalla Figura 26, il Process Mining è possibile suddividerlo in tre tipologie che rispecchiano la definizione di scoperta, monitoraggio e miglioramento dei propri processi. Questo perché il Process Mining prevede tre tipi di utilizzo:

- **Process Discovery:** tramite l'utilizzo del registro degli eventi in input, viene generato in maniera automatica un modello di processo come modelli BPMN o rete di Petri.
- **Conformance Checking:** un modello di processo già preesistente e gli event log vengono usati come input in questa tipologia di utilizzo. Si confrontano in modo da

verificare se ciò che accade nella realtà (cioè, le attività presenti negli event log) è conforme al modello e viceversa. L'output consiste in delle informazioni diagnostiche che sottolineano le differenze.

- **Enhancement:** gli input di questo tipo di Process Mining sono uguali a quelli precedenti. Tuttavia, l'enhancement propone come obiettivo quello di cambiare o estendere il modello preesistente utilizzando informazioni aggiuntive come timestamp in modo da estrarre conoscenza per individuare colli di bottiglia, tempi di produttività ecc.. Di conseguenza, il prodotto finale è comunque il modello dato in input con la differenza di essere stato migliorato o esteso.

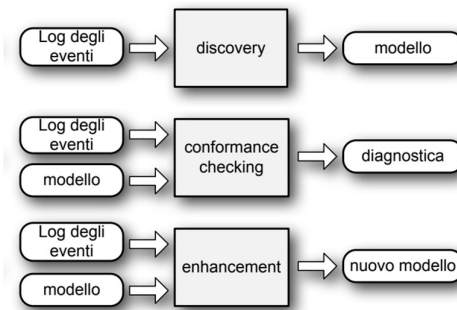


Figura 26: I 3 tipi di Process Mining (Process Mining Manifesto)

In Figura 26 una sintesi delle 3 tipologie di Process Mining spiegati in termini di input e output.

Sebbene il process mining non riguardi solamente l'estrazione di un modello di processo a partire da un log, sicuramente è l'aspetto più complicato e interessante. Di conseguenza, si è preferito analizzare un esempio di **Process Discovery** partendo proprio da un Event Log. La Figura 27 mostra una serie di attività e altre informazioni importanti presenti in un Registro degli Eventi, che riguardano la gestione delle richieste di risarcimento in una compagnia aerea. Tali richieste possono derivare da diverse circostanze, come ritardi o cancellazioni di voli.

identificatore del case	id. dello evento	proprietà				costo	...
		timestamp	attività	risorsa			
1	35654423	30-12-2010:11.02	registra richiesta	Pete	50	...	
	35654424	31-12-2010:10.06	esamina interamente	Sue	400	...	
	35654425	05-01-2011:15.12	controlla biglietto	Mike	100	...	
	35654426	06-01-2011:11.18	decidi	Sara	200	...	
	35654427	07-01-2011:14.24	rigetta richiesta	Pete	200	...	
2	35654483	30-12-2010:11.32	registra richiesta	Mike	50	...	
	35654485	30-12-2010:12.12	controlla biglietto	Mike	100	...	
	35654487	30-12-2010:14.16	esamina parzialmente	Pete	400	...	
	35654488	05-01-2011:11.22	decidi	Sara	200	...	
	35654489	08-01-2011:12.05	paga compensazione	Ellen	200	...	
3	35654521	30-12-2010:14.32	registra richiesta	Pete	50	...	
	35654522	30-12-2010:15.06	esamina parzialmente	Mike	400	...	
	35654524	30-12-2010:16.34	controlla biglietto	Ellen	100	...	
	35654525	06-01-2011:09.18	decidi	Sara	200	...	
	35654526	06-01-2011:12.18	reinizializza richiesta	Sara	200	...	
	35654527	06-01-2011:13.06	esamina interamente	Sean	400	...	
	35654530	08-01-2011:11.43	controlla biglietto	Pete	100	...	
	35654531	09-01-2011:09.55	decidi	Sara	200	...	
	35654533	15-01-2011:10.45	paga compensazione	Ellen	200	...	
4	35654641	06-01-2011:15.02	registra richiesta	Pete	50	...	
	35654643	07-01-2011:12.06	controlla biglietto	Mike	100	...	
	35654644	08-01-2011:14.43	esamina interamente	Sean	400	...	
	35654645	09-01-2011:12.02	decidi	Sara	200	...	
	35654647	12-01-2011:15.44	rigetta richiesta	Ellen	200	...	
5	35654871	06-01-2011:15.02	registra richiesta	Mike	50	...	
	35654873	06-01-2011:16.06	esamina parzialmente	Ellen	400	...	
	35654874	07-01-2011:16.22	controlla biglietto	Mike	100	...	
	35654875	07-01-2011:16.52	decidi	Sara	200	...	
	35654877	16-01-2011:11.47	paga compensazione	Mike	200	...	
6	35654711	06-01-2011:09.02	registra richiesta	Ellen	50	...	
	35654712	07-01-2011:10.16	esamina parzialmente	Mike	400	...	
	35654714	08-01-2011:11.22	controlla biglietto	Pete	100	...	
	35654715	10-01-2011:13.28	decidi	Sara	200	...	
	35654716	11-01-2011:16.18	reinizializza richiesta	Sara	200	...	
	35654718	14-01-2011:14.33	controlla biglietto	Ellen	100	...	
	35654719	16-01-2011:15.50	esamina parzialmente	Mike	400	...	
	35654720	19-01-2011:11.18	decidi	Sara	200	...	
	35654721	20-01-2011:12.48	reinizializza richiesta	Sara	200	...	
	35654722	21-01-2011:09.06	esamina interamente	Sue	400	...	
	35654724	21-01-2011:11.34	controlla biglietto	Pete	100	...	
	35654725	23-01-2011:13.12	decidi	Sara	200	...	
	35654726	24-01-2011:14.56	rigetta richiesta	Mike	200	...	

Figura 27: Event Log completa (Process Mining Data Science in Action)

È possibile osservare che ogni attività appartenente alla propria istanza di processo è raggruppata per un **Case ID** (identificatore del case) e che ad ogni riga corrisponde un'azione svolta da una certa risorsa. Ad esempio, Pete, alle 11:02 del 30-12-2010, ha provato a registrare una richiesta. Inoltre, in questo caso, ogni attività è caratterizzata da un proprio **Event ID** (id dell'evento) necessario a distinguere la stessa tipologia di attività svolta in momenti e casi differenti. Infatti, anche nel Case ID = 2 come prima attività viene registrata una richiesta, tuttavia, nel primo caso l'Event ID è 35654423 e nel secondo 35654483. È interessante notare che le attività che si svolgono nel Case ID = 1 sono differenti dal Case ID = 2, che a loro volta sono differenti dal Case ID = 3 e così via. Potenzialmente, potrebbero esserci molti percorsi che differiscono per alcune attività e altri che risultano essere identici. Come spiegato in precedenza, analizzarli manualmente potrebbe essere troppo dispendioso in termini di tempo e si potrebbero commettere degli errori o inesattezze. Gli algoritmi di Process Mining hanno come sfida quella di generalizzare il comportamento del sistema rappresentato dai log evitando i fenomeni di “overfitting” e “underfitting”, presentando quindi un equilibrio tra i due. Il primo fenomeno porta a rappresentare il modello con un livello di dettaglio troppo elevato, mentre il secondo un livello troppo generale [33].

Quindi solitamente, come primo passo si prendono in input gli event log e tramite delle tecniche di **discovery** si produce come output un modello di processo (spesso in formato BPMN o rete di Petri). Ad esempio, dei possibili output che si possono ottenere utilizzando come input l'Event Log in Figura 27, sono rappresentati nella Figura 28 e Figura 29.

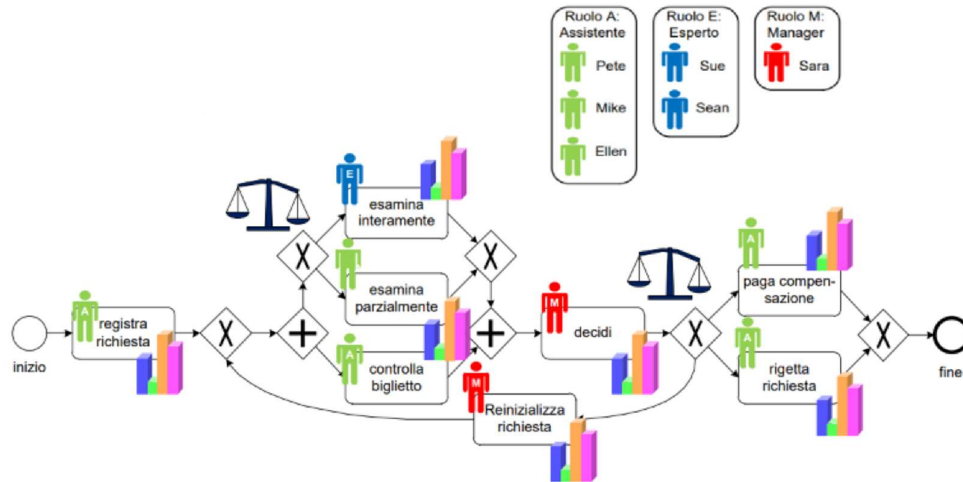


Figura 28: BPMN gestione richieste di risarcimento (Process Mining Manifesto)

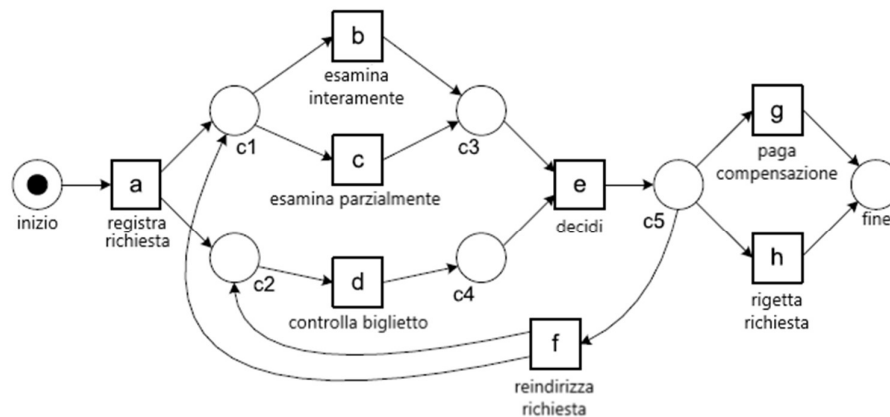


Figura 29: Rete di Petri gestione richieste di risarcimento (Process Mining Data Science in Action)

Questo processo di discovery tramite i log può aiutare ad individuare i ruoli di un'organizzazione, che a loro volta possono essere utilizzati per trovare una correlazione tra gli individui e le attività. Inoltre, è possibile estrarre informazioni di performance come, ad esempio, il tempo che passa tra l'esecuzione di 2 attività. È anche possibile estrarre delle regole di decisione permettendo di capire sulla base di quale criterio, ad esempio, l'assistente rigetta la richiesta.

4.3 Principi guida

Come accade per tutte le tecnologie innovative, è possibile che applicarle in scenari reali possa diventare complicato a causa di alcuni errori che si possono commettere. Questo porterebbe a non sfruttare il potenziale del Process Mining e a non produrre risultati significativi che il business si aspettava. Di conseguenza, seguire i principi guida qui elencati potrebbe essere utile per una corretta applicazione del Process Mining [34]. I principi sono i seguenti:

- **Qualità del Registro degli eventi:** si presume che gli eventi registrati negli Event Log siano accurati e affidabili. Tuttavia, la qualità dei Registri di Log può variare, influenzando l'applicabilità e l'efficacia del Process Mining. Secondo i principi stabiliti nel Manifesto, la qualità degli Event Log è suddivisa in 5 livelli. La Figura 30 illustra i 5 livelli di log e fornisce esempi di sistemi con la quale il Process Mining tendenzialmente è più compatibile. Idealmente, il Process Mining dovrebbe essere applicato su log di qualità compresa tra 3 e 5 stelle. Tuttavia, è possibile applicare il Process Mining a log di bassa qualità (1 e 2 stelle) ma i risultati ottenuti potrebbero non essere affidabili. Pertanto, al fine di ottimizzare l'utilizzo di questa tecnologia, le organizzazioni dovrebbero mirare a tracciare gli Event Log con l'obiettivo di raggiungere il più alto livello di qualità possibile. Non è strettamente necessario che gli eventi siano memorizzati in file di log dedicati; possono anche essere memorizzati all'interno di tabelle relazionali, log di messaggi, archivi di posta elettronica e altre sorgenti informative. La qualità di un registro degli eventi dipende dalla sua **completezza** e **attendibilità**: deve contenere attività che si sono effettivamente realizzate e che risultino pertinenti per la descrizione del proprio processo. Inoltre, ogni evento dovrebbe avere un significato chiaro con una **semantica** ben definita e i dati associati ad esso dovrebbero essere accessibili, nel rispetto delle politiche di privacy e sicurezza.

Livello	Descrizione	Esempi
★★★★★	Livello più elevato: l'event log è di qualità eccellente (ovvero affidabile e completo) e gli eventi sono ben definiti. Gli eventi vengono registrati con un approccio automatico, sistematico, attendibile e sicuro. Considerazioni legate a privacy e sicurezza sono affrontate adeguatamente. Inoltre, gli eventi registrati (e tutti i loro attributi) hanno una semantica chiara. Questo implica l'esistenza di una o più ontologie, a cui gli eventi e gli attributi si riferiscono.	Log di sistemi BPM annotati semanticamente.
★★★★	Gli eventi vengono registrati automaticamente e in modo sistematico ed attendibile, ovvero il log è affidabile e completo. A differenza dei sistemi che operano al livello ★★★, concetti quali istanza di processo (case) e attività sono supportati in maniera esplicita.	Gli event log dei classici sistemi di BPM e workflow.
★★★	Gli eventi vengono tracciati automaticamente, ma non viene seguito nessun approccio sistematico. A differenza dei log a livello★★, c'è un certo grado di garanzia che gli eventi registrati corrispondano alla realtà (in altre parole, l'event log è affidabile ma non necessariamente completo). Si considerino, a scopo illustrativo, gli eventi generati da un sistema ERP. Anche se questi eventi devono essere estratti da diverse tabelle, si può assumere che l'informazione ottenuta sia corretta (per esempio è sicuro che un pagamento registrato dal sistema ERP esista e viceversa).	Tabelle nei sistemi ERP, event log nei sistemi CRM, log delle transazioni di sistemi basati sullo scambio di messaggi, event log di sistemi high-tech, ecc.
★★	Gli eventi vengono registrati automaticamente come prodotto secondario di un sistema informativo. La copertura rispetto alla completezza è variabile, nel senso che non viene seguito un approccio sistematico per decidere quali eventi vengono tracciati. Inoltre, è possibile aggirare il sistema informativo. Di conseguenza, alcuni eventi potrebbero mancare o non essere tracciati correttamente.	Event log prodotti da sistemi di gestione documentale e di gestione dei prodotti, log degli errori nell'ambito di sistemi integrati, fogli di lavoro in ambito ingegneristico, ecc.
★	Livello più basso: l'event log è di bassa qualità. Gli eventi tracciati possono non collimare con la realtà, e può accadere che alcuni eventi siano assenti nel log. I log che contengono eventi registrati manualmente presentano tipicamente queste caratteristiche.	Tracce lasciate nei documenti cartacei che circolano all'interno di un'organizzazione (come note a margine e annotazioni), record medici cartacei, ecc.

Figura 30: Qualità degli Event Log (Process Mining Manifesto)

- **Estrazione dei log basata su domande:** è necessario porre delle domande per individuare le tabelle che sono di interesse per lo sviluppo del modello, e che quindi contengano le attività significative. Potenzialmente, in un sistema SAP, potrebbero essere presenti numerose tabelle ed è importante individuare quelle rilevanti per l'estrazione dei dati, e che quindi si riferiscano alla tipologia di Case che si vuole analizzare. Ad esempio, consideriamo la gestione degli ordini dei clienti di Amazon. I clienti, tramite un solo ordine possono comprare più articoli, che però potenzialmente possono appartenere a linee di ordine differenti. Quindi un ordine di un cliente può comportare diverse spedizioni a causa di prodotti dislocati in magazzini differenti. Quindi, ci sarebbe una relazione 1 a N tra il singolo ordine del cliente e i diversi prodotti ordinati (1 ordine contiene N prodotti) e una relazione 1 a N tra una singola spedizione e i diversi prodotti ordinati. Di conseguenza, c'è una relazione N a N tra gli ordini dei clienti e le spedizioni. È necessario quindi porsi delle domande per capire quale processo si vuole modellare; si può decidere di rappresentare il ciclo di vita dei singoli ordini, delle singole linee d'ordine (quindi dei singoli prodotti) e delle singole spedizioni.
- **Corrispondenza eventi ed elementi del modello:** è fondamentale ricordarsi che il Process Mining non è solo uno strumento utile a scoprire il proprio processo tramite gli event log, ma è anche una tecnica necessaria per capire se il proprio modello rispetta la realtà. Di conseguenza, tramite la tecnica del "replay", è possibile verificare se c'è

corrispondenza tra gli eventi registrati negli Event Log e il modello. Ad esempio, è possibile che alcuni eventi che accadono nella realtà (quindi presenti nei registri) non siano presenti nel modello e dunque si rileverebbe una discrepanza. Inoltre, è possibile arricchire il modello con informazioni aggiuntive grazie ai timestamp che, durante il replay, ti permettono di individuare l'andamento temporale delle attività e quindi arricchire il modello con colli di bottiglia o tempi di attesa stimati.

- **Modelli di processo come cartine geografiche:** è interessante porre il focus su un'analogia tra modelli di processo e mappe geografiche. Esistono differenti tipologie di cartine geografiche, come cartine stradali, cartine escursionistiche per sentieri da trekking o mountain bike, mappe di percorsi ciclabili, cartine militari e così via. Sulla base del proprio obiettivo, l'utente sceglie la mappa più appropriata. L'idea di creare una mappa unica che incorpori le informazioni delle varie tipologie di cartine sarebbe impensabile, poiché renderebbe la mappa troppo complessa e difficile da interpretare. Questo perché ogni tipo di mappa è ottimizzato per fornire un certo tipo di informazioni, e l'aggiunta di troppi dettagli potrebbe oscurare le informazioni cruciali rendendo meno utile la mappa. Allo stesso modo, quando si crea un modello di processo, è importante che esso sia costruito in funzione dell'obiettivo che ha l'utente che deve utilizzarlo. Se ad esempio l'utente è un manager, allora potrebbe essere più interessato a un processo incentrato sui costi, mentre se fosse un analista probabilmente sarebbe più interessato a individuare le deviazioni del flusso da quello classico.
- **Dinamicità del Process Mining:** Adottare il Process Mining un'unica volta, con l'aspettativa che il modello generato sia definitivo, rappresenta un approccio errato all'implementazione della tecnologia. Considerando la natura dinamica dei processi, l'uso del Process Mining dovrebbe essere un processo continuo, permettendo agli utenti di esplorare i processi e visualizzare istantaneamente informazioni sulle mappe di processo, fornendo così una visione in tempo reale degli eventi. Questo modo di procedere permette di ottenere informazioni utili per intraprendere azioni su diverse scale temporali (minuti, ore, settimane ecc..). Infatti, un sondaggio condotto da Deloitte (Figura 31), una delle più grandi aziende che offre servizi di consulenza e revisione, rivela che circa il 50% degli intervistati utilizza il Process Mining quotidianamente per ottenere le informazioni necessarie al miglioramento dei processi [35].

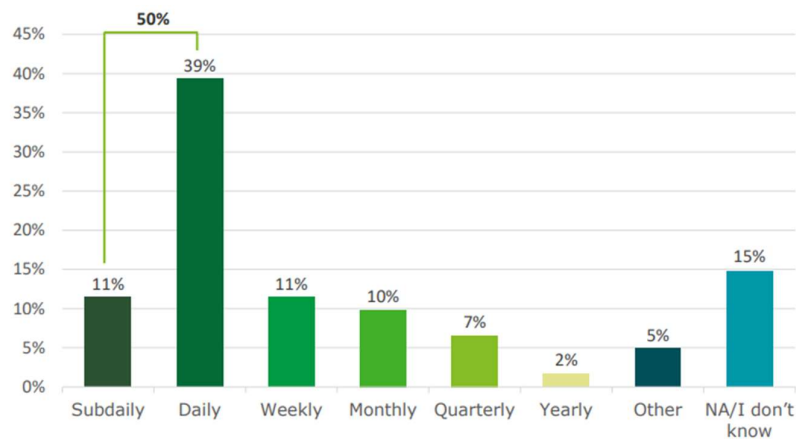


Figura 31: Frequenza di utilizzo del Process Mining (Global Process Mining Survey insights)

4.4 Sfide

Sebbene il Process Mining possa conferire un notevole vantaggio competitivo, essendo una tecnologia ancora in fase di diffusione, può presentare alcuni ostacoli. Pertanto, prima di implementare questa tecnologia, è fondamentale riconoscere e superare alcune sfide per garantire un'applicazione efficace del Process Mining [34]. Le sfide più comuni includono:

- **Distribuzione dei dati:** nei sistemi informativi, i dati necessari per applicare il Process Mining potrebbero essere presenti, ma dislocati in sorgenti eterogenee. Come delineato nella sezione 4.1 dei concetti preliminari, è possibile collegare tabelle differenti attraverso chiavi comuni in modo da generare dei Data Model. Tuttavia, gli identificatori potrebbero non avere la stessa denominazione. Di conseguenza, la sfida risiede nell'individuare e mappare correttamente questi identificatori.
- **Dati incompleti:** i dati presenti nei sistemi informativi potrebbero risultare incompleti. In particolare, gli eventi registrati non sempre sono associati ad un'istanza di processo, rendendo l'interpretazione dei dati più complessa. Inoltre, potrebbero essere assenti anche riferimenti temporali, come i timestamp, inducendo ad ulteriori complicazioni. Quando non sono disponibili, si dovrebbe ricorrere alla tecnica dell'interpolazione dei timestamp, in modo da stimarli.
- **Livello di granularità:** il livello di granularità degli eventi può variare notevolmente, aumentando così il livello di complessità. Ad esempio, in un event log di un sistema informativo di un'azienda di produzione, gli eventi potrebbero riferirsi ad operazioni semplici (come l'assemblaggio di un componente), o a procedure più complesse (come la produzione di un intero prodotto). Anche il livello di granularità dei timestamp potrebbe avere differenze significative. Ad esempio, l'assemblaggio potrebbe avere un

timestamp molto preciso, registrando il tempo con precisione al secondo (es. 15-02-2024; 14:37:20) mentre la produzione di un intero prodotto potrebbe richiedere un livello di precisione giornaliero (es. 15-02-2024).

- **Interfacce e risultati User-Friendly:** la creazione delle interfacce e i risultati ottenuti tramite i software di Process Mining, deve risultare intuitivi anche per gli utenti che non hanno familiarità diretta con il software. In questo modo, gli utenti possono comprendere rapidamente come utilizzare il software ma anche come interpretare e applicare le informazioni scoperte da altri. Questo permette di ridurre errori di interpretazione dei dati che potrebbero portare a conclusioni sbagliate.

4.5 Benefits

Nonostante nelle sezioni precedenti si sia discusso di principi guida utili per l'applicazione ottimale del Process Mining e potenziali sfide che possono insorgere, i benefici derivanti da questa tecnica sono molteplici e sempre più evidenti per le organizzazioni. Secondo il sondaggio condotto da Deloitte, è possibile individuare sei benefici di spicco. In Figura 32 sono riportate le percentuali delle risposte degli intervistati in relazione ai 6 benefici individuati [36].

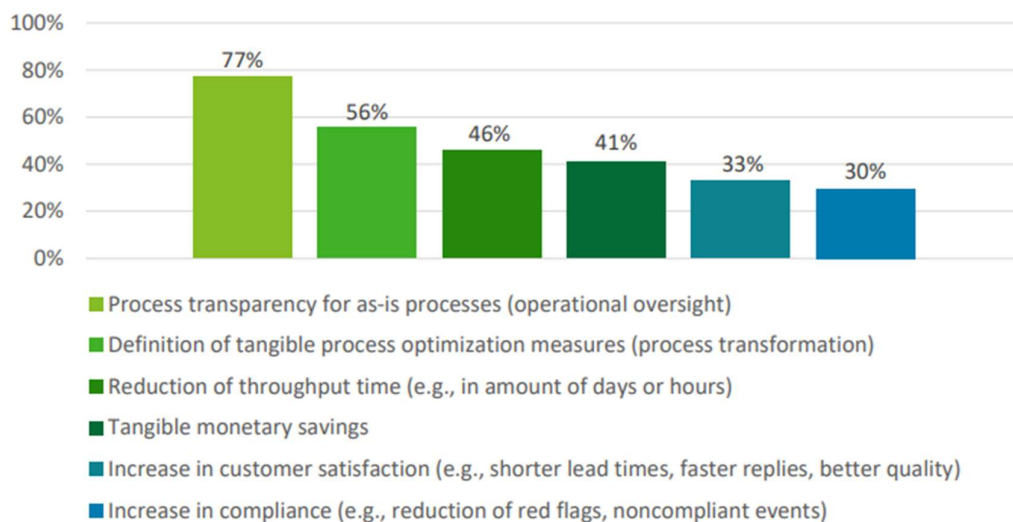


Figura 32: (Global Process Mining Survey insights)

Una significativa percentuale degli intervistati, il 77 %, ha identificato come principale vantaggio del Process Mining, la capacità di fornire una **trasparenza del processo**. Tramite l'analisi dei dati reali provenienti dagli Event Log, è possibile ottenere una vista chiara e attuale del processo di business così com'è, permettendo agli stakeholder di comprendere il processo attuale (As-Is) e di individuare inefficienze, deviazioni dal flusso principale e di prendere

decisioni guidate dai dati. In particolare, facilita l'individuazione di sprechi di risorse, attività duplicate o processi che non aderiscono alle procedure standard.

Più della metà degli intervistati ha riconosciuto come secondo vantaggio chiave del Process Mining il supporto che esso apporta nella **trasformazione del processo**. Questa tecnica fornisce alle organizzazioni gli strumenti necessari per affinare i loro modelli di processo confrontando il processo attuale (As-Is) con quello ideale ed evidenziando le aree che necessitano miglioramenti. In questo modo si semplifica lo sviluppo di strategie per la trasformazione dei processi. Tra le strategie adottabili vi sono:

- La **ristrutturazione dei flussi** per eliminare complessità e inefficienze.
- L'**individuazione dei compiti manuali** in modo automatizzarli e aumentare la velocità e l'accuratezza delle operazioni.
- L'**implementazione di nuove politiche e procedure** per garantire conformità e una migliore efficienza generale.

Il terzo beneficio, riscontrato dal 46% del campione, è la **riduzione del tempo totale** necessario per il completamento del processo, dal suo inizio alla sua conclusione. Questo tempo, spesso definito con il termine inglese "throughput time", indica quanto tempo impiega un processo, come la fornitura di un servizio o il processamento di un ordine. I processi aziendali spesso possono essere soggetti a colli di bottiglia, che inevitabilmente rallentano il processo incrementando il throughput time. A volte, questi colli di bottiglia possono risultare tanto intrinseci da risultare invisibili. Il Process Mining in alcuni casi può aiutare ad identificare i colli di bottiglia permettendo così di decidere come ottimizzare, quindi allocando più risorse, semplificando i passaggi o automatizzando attività.

Un beneficio che il 40% degli intervistati ha riscontrato con l'implementazione del Process Mining, è una **riduzione tangibile dei costi**. Questa tecnologia ha permesso alle aziende di minimizzare lo spreco e nel ridurre lo sforzo manuale, identificando le inefficienze e le deviazioni dal processo standard. Ciò ha portato a un risparmio significativo attraverso l'allocazione di risorse più efficiente. Inoltre, il Process Mining ha semplificato l'identificazione di opportunità di automazione, permettendo di individuare compiti ripetitivi, che richiedono molto tempo o sono soggetti a numerosi errori umani. Questo, oltre a tradursi in un risparmio di tempo, spesso si traduce anche in risparmio di costi legati al lavoro.

Oltre all'ottimizzazione interna, il Process Mining gioca un ruolo decisivo anche lato cliente. Al giorno d'oggi è estremamente importante dedicarsi alla **soddisfazione del cliente** e per 1/3 degli intervistati questa tecnologia ha contribuito ad incrementarla. Monitorando i propri

processi tramite Process Mining, le aziende possono identificare più rapidamente i problemi che possono sorgere e intervenire prontamente per risolverli. Ciò aumenta la velocità nel reagire ai problemi riducendo il tempo di reazione. Inoltre, il Process Mining può contribuire a ridurre il lead time, ovvero il tempo necessario per soddisfare la richiesta di un cliente, attraverso una riduzione del throughput time e l'ottimizzazione dei processi.

Infine, per il 30% del campione, il Process Mining è stato utilizzato anche per ottenere una **maggiore conformità**. In un mondo in cui le organizzazioni devono aderire a determinati standard e garantire una certa conformità legislativa, il Process Mining può risultare una soluzione anche da questa prospettiva. Questa tecnologia può aiutare le aziende a monitorare e confrontare determinati parametri rispetto ai requisiti, offrendo una visione trasparente dei propri processi e segnalando situazioni potenzialmente rischiose. Ciò aiuta le organizzazioni a evitare potenziali multe e danni reputazionali e potrebbe essere utile anche per promuovere una cultura di trasparenza.

Questi sono tra i benefici più comuni individuati che le aziende hanno identificato con l'uso del Process Mining. Tuttavia, essendo una tecnologia emergente, potrebbero emergere ulteriori vantaggi del Process Mining in futuro.

5. Applicazione pratica del Process Mining: Caso studio con Celonis

Questo capitolo è dedicato all'analisi e alla discussione dell'applicazione pratica del Process Mining, con un particolare focus sul software Celonis. L'obiettivo principale consiste nel dimostrare come il Process Mining, partendo da una semplice tabella di dati, possa essere utilizzato per ottenere risultati significativi e utili per l'analisi dei processi aziendali. In particolare, verranno esaminati due approcci fondamentali offerti dal software Celonis: l'**Exploratory Approach** e il **Confirmatory Approach**. Attraverso l'uso di questi due approcci, sarà possibile comprendere come il Process Mining possa trasformare una semplice tabella di dati in una fonte preziosa di insight per l'ottimizzazione dei processi.

5.1 Software Celonis



Figura 33: Quadrante competitor Process Mining (Gartner)

Il software Celonis, leader riconosciuto nel campo del Process Mining con una quota di mercato stimata superiore al 60% (Kerremans et al., 2023), è stato impiegato per testare questa tecnologia. L'analisi si è focalizzata sugli strumenti chiave offerti dal software, che rispecchiano le caratteristiche teoriche discusse nel capitolo precedente, tra cui il Variant

Explorer, il Process Explorer e il Conformance Checking. Tutti questi strumenti possono essere utilizzati per creare Dashboard interattive che potenzialmente possono essere arricchite dalla creazione di grafici, tabelle e KPI personalizzabili.

Per esplorare le potenzialità di questi strumenti, è stato utilizzato un dataset che racchiude informazioni relative a un processo di vendita di pizza d'asporto. Questi Event Log, arricchiti da dati supplementari oltre a quelli minimi richiesti per applicare il Process Mining, coprono una vasta gamma di scenari, sia ideali che non. Tra le attività desiderate, ad esempio, rientrano la vendita tramite app, sito web e direttamente in loco, mentre le chiamate ai clienti sono esempi di attività indesiderate.

5.2 Variant Explorer

Il **Variant Explorer** di Celonis è uno strumento progettato per l'analisi esplorativa dei propri processi aziendali. Inizialmente, fornisce una visione della variante più comune del processo, ma offre la possibilità di identificare e confrontare diverse varianti. Queste rappresentano i casi che si sono verificati nella realtà con una specifica sequenza di attività e una certa frequenza. Anche se la variante ideale non corrisponde alla più comune, è possibile comunque individuarla e selezionarla dal menù sulla destra, come illustrato in Figura 34. Inoltre, il software offre l'opzione di selezione multipla, che consente all'utente di selezionare più percorsi, sia simili che differenti, per comprendere immediatamente le differenze.

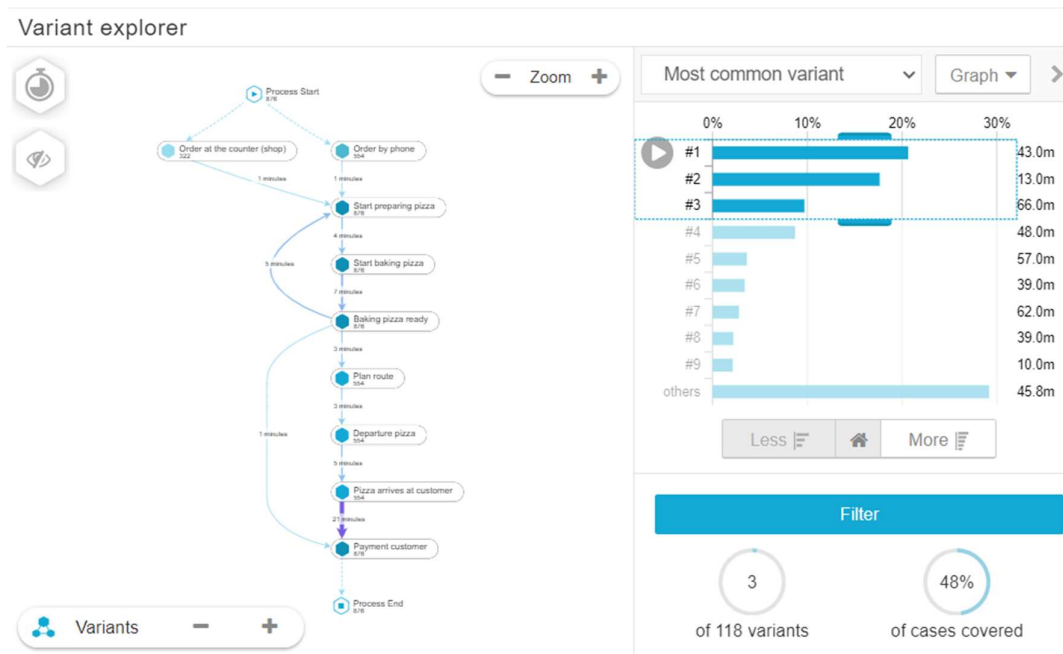


Figura 34: Variant Explorer: 3 varianti più comuni del processo (Celonis)

Nell'esempio illustrato in Figura 34, si osserva la selezione multipla dei tre percorsi più frequenti, che rappresentano il 48% dei casi totali, ma contemporaneamente costituiscono solo 3 varianti su 118. Le prime due varianti differiscono tra di loro in base al metodo di ordinazione: in un caso l'ordine viene effettuato tramite telefono, nell'altro direttamente in negozio. Come si può evincere dalle metriche a destra delle varianti, che indicano il tempo medio impiegato per completare l'ordine, la seconda variante (pizza ordinata in negozio) impiega in media 13 minuti per completare il processo, mentre la prima 43, poiché è compresa anche la fase di spedizione. La terza casistica invece impiega 66 minuti, un tempo significativamente superiore rispetto alle due varianti precedenti, causato da un arco secondario che, partendo dall'attività "Baking pizza ready", ritorna all'attività "Start preparing pizza". Questo suggerisce che il pizzaiolo potrebbe aver bruciato o composto in modo errato la pizza, costringendosi a prepararne un'altra e incrementando il tempo di attesa del cliente e dell'intero processo. Da questa breve analisi, si può giungere alla conclusione che le prime due varianti sono sicuramente quelle ideali che la pizzeria desidererebbe, mentre la terza, portando ad un incremento dei tempi, potrebbe comportare una riduzione della soddisfazione del cliente e un aumento dei costi.

Analizzando più nel dettaglio questo caso (Figura 35), è possibile notare che esiste un tempo medio di 3 minuti che intercorre tra le attività "Baking pizza ready" e "Plan Route", e un ulteriore ritardo di 3 minuti tra "Plan Route" e "Departure pizza".

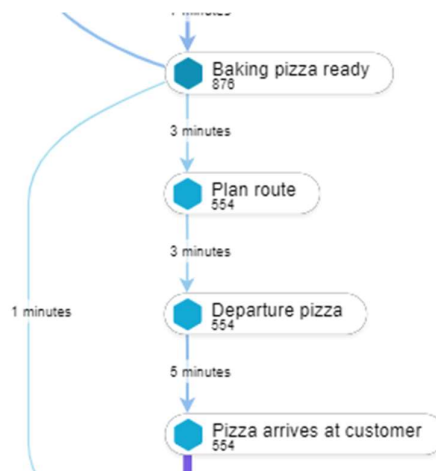


Figura 35: Variant Explorer: Baking pizza ready, Plan Route e Departure pizza (Celonis)

Questo suggerisce la possibilità di ottimizzare il processo per ridurre questi ritardi. Ad esempio, si potrebbe considerare l'opzione di pianificare il percorso di consegna durante la cottura della pizza. Questo porterebbe potenzialmente a risparmiare un tempo medio di 6 minuti sull'intero processo, poiché una volta terminata la cottura della pizza, risulterebbe già impostato il percorso

di consegna. Questa azione di pianificazione del percorso è inevitabile nel momento in cui l'ordine viene effettuato telefonicamente, ma si verifica anche nel caso in cui l'ordine avviene tramite sito web. Pertanto, un ulteriore miglioramento potrebbe essere l'automazione di questa attività. Evitando di svolgere manualmente la pianificazione del percorso, si potrebbe risparmiare tempo portando non solo ad un'ottimizzazione del processo, ma anche la soddisfazione del cliente.

Se invece si procedesse con la selezione di tutte le varianti, l'analisi risulterebbe complicata a causa dell'eccessivo numero di connessioni tra varie attività, rendendo la visione dei processi confusa. Tuttavia, effettuando uno zoom sulle varie attività, è possibile individuare alcune che inizialmente erano sfuggite e che possono essere utili al fine delle analisi. Il risultato di questa operazione è visibile in Figura 36.

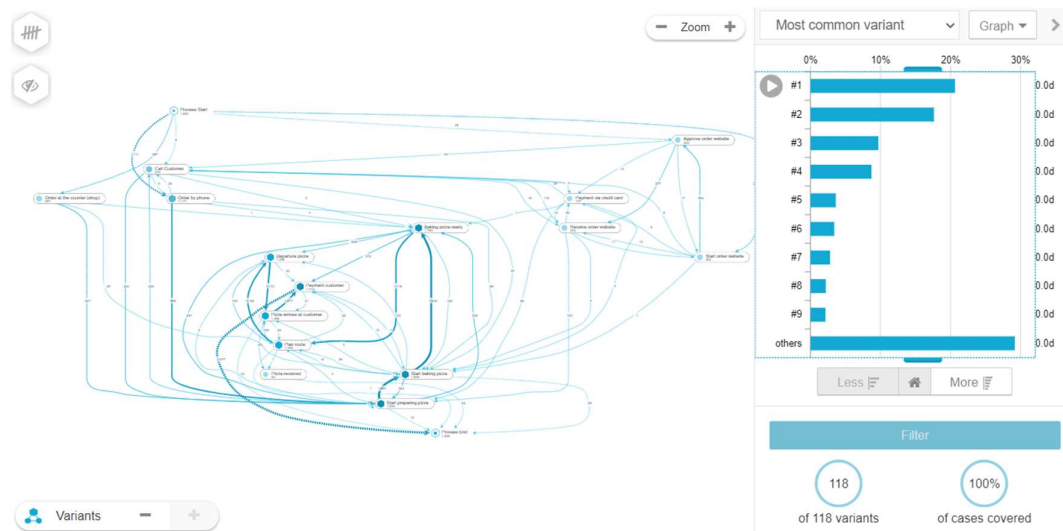


Figura 36: Variant Explorer: Visualizzazione 118 varianti (Celonis)

Il Variant Explorer di Celonis offre anche la possibilità di selezionare specifiche connessioni o attività e filtrarle, decidendo se includerle o escluderle, in base al tipo di percorso che si desidera analizzare.

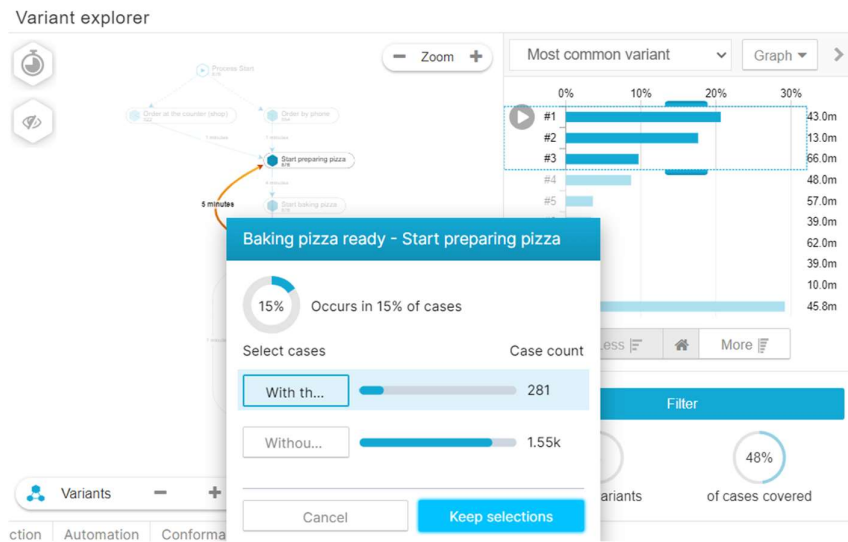


Figura 37: Variant Explorer: Inclusione connessione Baking pizza ready - Start preparing pizza (Celonis)

Ad esempio, come mostrato in Figura 37, si potrebbe decidere di includere la connessione precedentemente osservata, che collega l’attività “Baking pizza ready” con “Start preparing pizza”. Questo suggerisce che l’attività di preparazione della pizza per lo stesso ordine è stata eseguita due volte. Filtrando sull’inclusione di questa connessione, si può proseguire l’analisi con tutti i Case che comprendono questa connessione, che in questo esempio rappresentano il 15% del totale. Questo metodo facilita l’individuazione di altre varianti che contengono la stessa connessione, permettendo di capire in quali altri casi si verifica questa attività indesiderata.

Inoltre, questo tool permette di impostare rapidamente e intuitivamente uno tra i KPI mostrati in Figura 38. Ciò consente di comprendere le prestazioni del processo in ogni sua fase e non solo a livello generale.

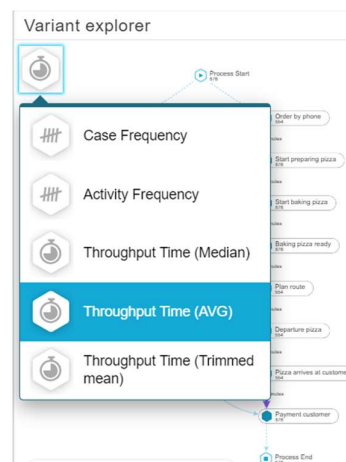


Figura 38: Variant Explorer: Scelta KPI (Celonis)

In questo modo si fornisce la possibilità di scegliere se visualizzare il tempo medio che intercorre tra le attività, la mediana, oppure la frequenza con cui avvengono i case o le attività. La differenza tra gli ultimi due KPI risiede nel fatto che una stessa attività può essere ripetuta più volte all'interno di un Case, come nell'esempio precedente in cui il pizzaiolo ha dovuto preparare nuovamente la pizza, svolgendo di fatto 2 volte la stessa attività.

Quindi con il tool di Variant Explorer, si possono ottenere in maniera rapida molti insight e una visione trasparente e oggettiva di quali sono i flussi del proprio processo.

5.3 Process Explorer

Nonostante le funzionalità del tool precedente, potrebbe sfuggire qualche attività che potrebbe rilevante per l'analisi. Per questo motivo, Celonis ha sviluppato un altro strumento: il **Process Explorer**. Questo consente di regolare facilmente il numero di attività o connessioni da visualizzare.

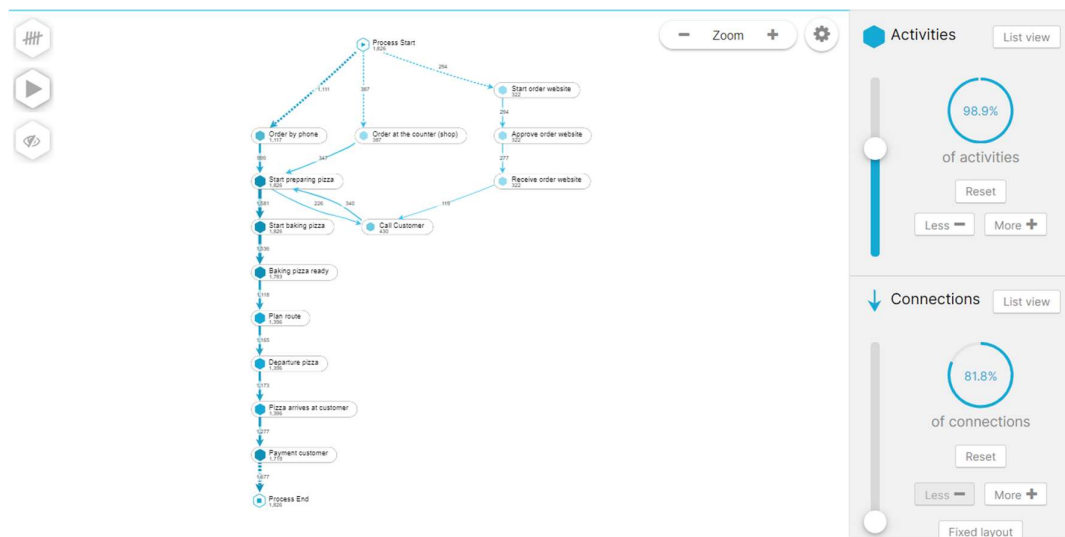


Figura 39: Process Explorer (Celonis)

Regolando le barre posizionate sulla destra della Figura 39, è possibile aumentare o diminuire la visualizzazione delle attività e delle connessioni. Inoltre, vengono mostrate le percentuali di attività e connessioni visualizzate rispetto a quelle potenziali. In questo caso, nonostante la visualizzazione sia impostata per includere la quasi totalità delle attività (98,9%), la rappresentazione risulta più snella rispetto a quella della Figura 36, che includeva un numero di attività quasi identico. La differenza è attribuibile al livello minimo di connessioni impostato, necessarie per garantire un flusso continuo e prevenire la presenza di attività non collegate con le altre. Rispetto al Variant Explorer, l'obiettivo di questo strumento è fornire una visione più

chiara delle diverse attività e permettere un focus più mirato sulle singole attività. Infatti, selezionando un'attività di interesse, è possibile visualizzare le attività correlate, sia precedenti che successive.

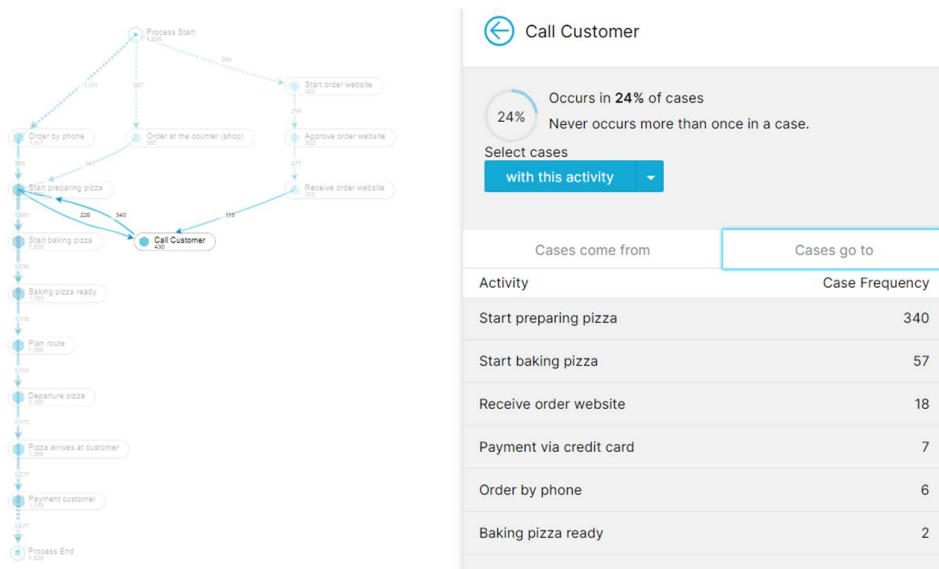


Figura 40: Process Explorer: Attività che precedono o succedono "Call Customer" (Celonis)

Ad esempio, l'attività denominata "Call Customer" precede altre attività come "Start preparing pizza", "Start baking pizza" o "Receive order website" (Figura 40). È interessante osservare che la pizzeria tende a contattare il cliente telefonicamente in maniera piuttosto frequente (24% delle volte) e il più delle volte accade prima di preparare la pizza, probabilmente a causa dello smarrimento delle ordinazioni o dell'esaurimento di alcuni ingredienti, costringendo così il cliente a modificare la sua scelta. Anche questa analisi dimostra come è possibile investigare ed individuare attività indesiderate che inducono all'incremento di tempi, costi e recensioni negative dei clienti.

Filtrando invece solo sugli ordini effettuati attraverso il sito web (come illustrato in Figura 41), risulta particolarmente rilevante l'aumento della frequenza delle chiamate effettuate dalla pizzeria verso i clienti. Questo fenomeno, che vede un incremento significativo rispetto allo scenario precedente (con 172 casi su 322, pari a circa il 53%), potrebbe essere attribuito a due fattori principali. Il primo riguarda la problematica precedentemente citata e relativa alla disponibilità degli ingredienti. Il secondo, invece, potrebbe essere legato a potenziali problemi con il modulo di ordinazione online, che potrebbe non essere compilato in modo adeguato dal cliente o potrebbe non essere strutturato in modo da raccogliere tutte le informazioni rilevanti per la pizzeria. Considerata la marcata differenza tra la percentuale di chiamate effettuate nel

caso generale (24%) e quella nel caso degli ordini online (53%), è probabilmente che l'aumento dell'attività indesiderata sia principalmente dovuto a problemi con il modulo di ordinazione.

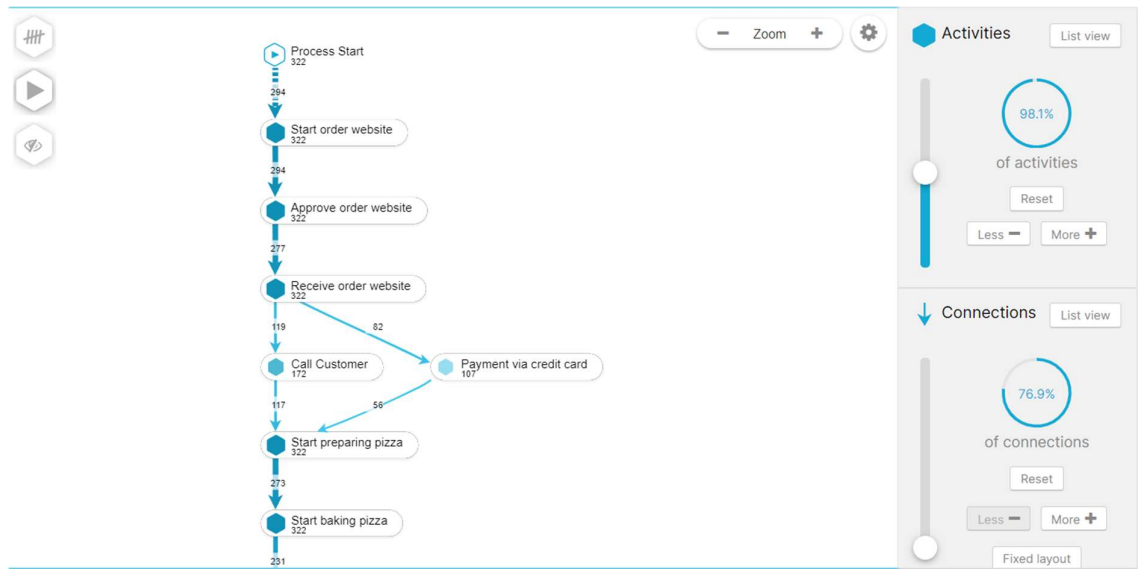


Figura 41: Process Explorer: Applicazione filtro su "Start Order website" (Celonis)

5.4 Tabelle, grafici e KPI

Inoltre, è possibile combinare l'uso di questi tool con dei grafici, tabelle o singoli KPI che si adattano istantaneamente ai filtri applicati e che possono essere generati in base alle analisi che si desidera condurre. Questo permette di creare delle dashboard personalizzate in base ai vari scenari.

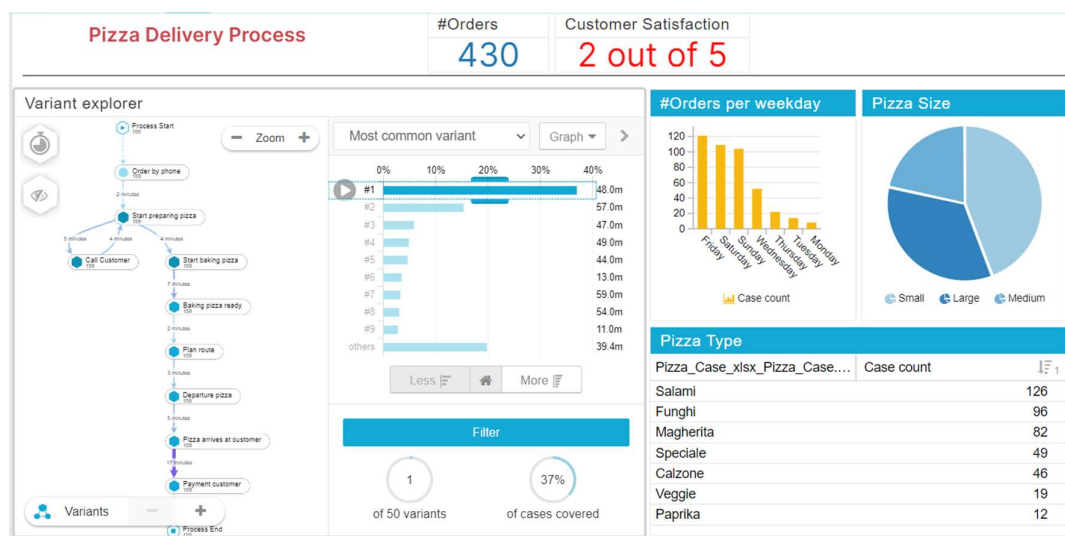


Figura 42: Dashboard (Celonis)

Ad esempio, come illustrato nella Figura 42, filtrando tutti i casi in cui la pizzeria ha contattato il cliente, si può dedurre dal grafico “#Orders per weekday” che la maggioranza degli ordini (334 su 430) avviene nel fine settimana, tra venerdì e domenica. Questo potrebbe indicare che ci sia un certo grado di confusione in quei giorni, probabilmente a causa di una domanda elevata. Se il problema risiedesse nello smarrimento di ordinazioni a causa dal caos generato dall’intensità delle richieste, una possibile soluzione potrebbe essere quella di allocare più risorse per prevenire inconvenienti. Tuttavia, questa situazione potrebbe verificarsi solo se gli ordini vengono effettuati telefonicamente (che rappresenta solo il 60% dei casi, ovvero 258 su 430), ma in realtà i clienti ordinano anche tramite il sito web, rendendo di fatto nulla l’ipotesi iniziale. Un’altra ipotetica motivazione, come menzionato in precedenza, potrebbe essere l’esaurimento di alcuni prodotti. Per questo motivo, è stata generata la tabella “Pizza Type” per esaminare quali tipologie di pizza, e quindi quali ingredienti, siano i più richiesti. Dalla Figura 41 si evince che le pizze con il salame e con i funghi sono le più richieste e di conseguenza quelle che possono indurre a criticità. Inoltre, se necessario, l’analisi potrebbe essere ulteriormente approfondita, concentrandosi esclusivamente sui tre giorni del fine settimana.

5.5 Conformance Checking

L’ultimo tool testato è il **Conformance Checking**. Come suggerito dal nome e sulla base di ciò che è stato descritto nella Sezione 4.2 dell’elaborato, questo tool permette di verificare quanto i casi reali presenti negli event log siano conformi al modello ideale. In questo modo, l’organizzazione ha la possibilità di comprendere quanto sia distante dal suo obiettivo e di indagare sulle cause.

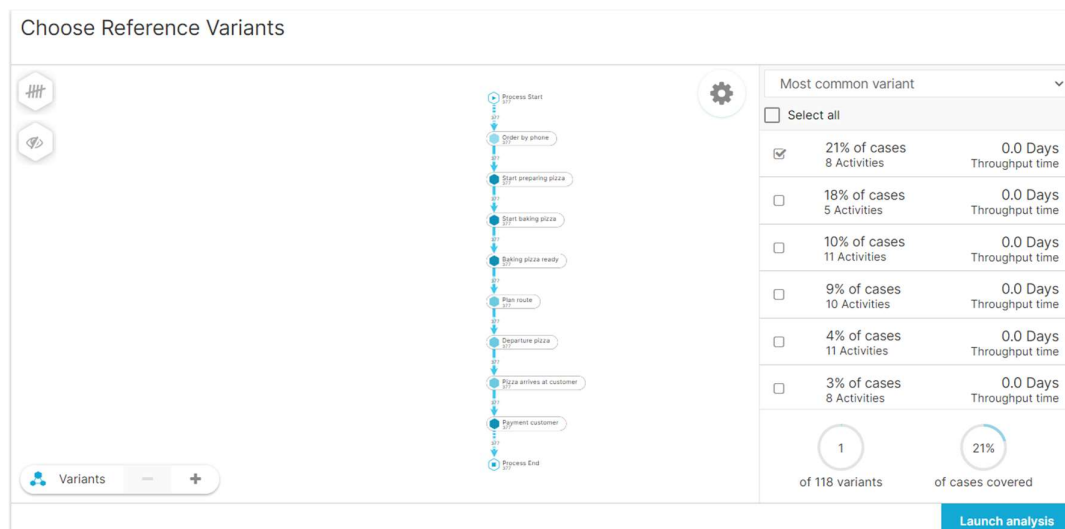


Figura 43: Conformance Checking: Selezione varianti ideali (Celonis)

Nella Figura 43 è mostrata una schermata iniziale dello strumento in questione. Esso permette di selezionare i percorsi, e quindi le attività e connessioni che vengono considerate ideali. In questo modo, può generare un flusso di processo BPMN (descritto nella Sezione 4.2) che corrisponde al modello ideale.

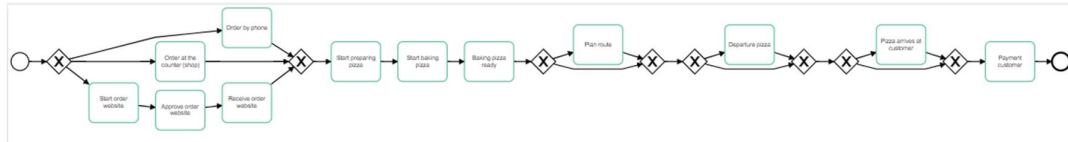


Figura 44: Conformance Checking: BPMN (Celonis)

Per generare il modello BPMN in Figura 44, sono state selezionate 3 opzioni della Figura 43, che rappresentano i percorsi standard di ordinazione e spedizione attraverso il telefono, il sito web o direttamente dal negozio fisico.

Una volta confermato il modello, si accede a una schermata (Figura 45) che fornisce informazioni sulla percentuale e sul numero di casi reali conformi al modello ideale, sulla quantità di violazioni (ovvero deviazioni rispetto al percorso ideale), e un grafico che mostra dati storici sulla conformità dei processi.

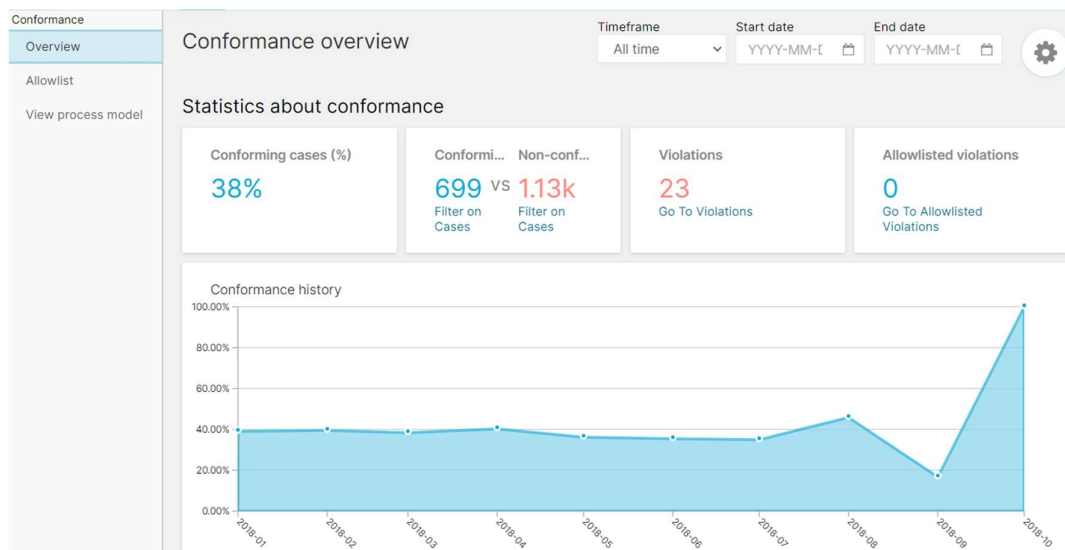


Figura 45: Conformance Checking: KPI e grafici violazioni (Celonis)

La stessa pagina presenta anche KPI standard come il throughput time e gli step per case, forniti automaticamente dal software. Tuttavia, è comunque possibile aggiungere KPI personalizzati in base alle proprie necessità. Una caratteristica particolarmente interessante è la visualizzazione delle deviazioni dal processo ideale (Figura 46), che suggerisce i motivi per cui

il processo non si svolge come desiderato. I criteri con cui il software individua le violazioni sono:

- **Attività non conformi** a quelle indicate nel processo ideale;
- **Ordine delle attività non conforme** a quello indicato nel processo ideale.

Per ogni violazione, viene indicato quanto KPI si discosta da quello ideale. Ad esempio, la prima violazione riguarda l'attività indesiderata "Call Customer", che era già stata individuata tramite il tool Process Explorer. In media, questa attività aumenta il throughput time di 17 minuti e gli step di 3,4.

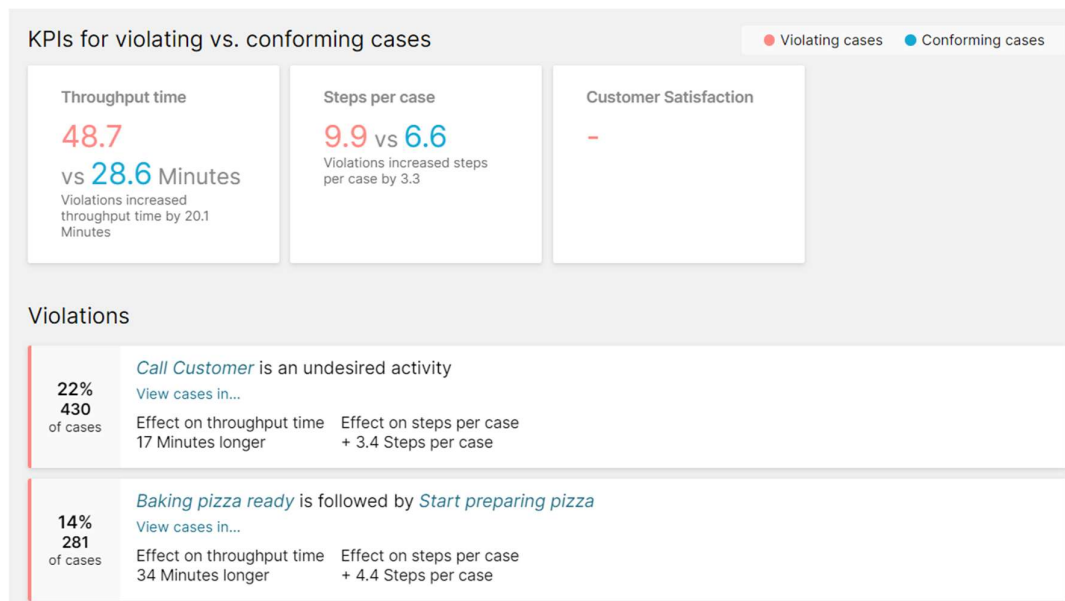


Figura 46: Conformance Checking: Violazioni (Celonis)

Selezionando la violazione che si desidera analizzare più approfonditamente, è possibile ottenere ulteriori informazioni grazie ai dati che indicano a cosa essa è correlata. Ad esempio, in Figura 47, è possibile notare che la maggior parte delle chiamate provengono da studenti e adulti e sono collegate ad ingredienti come salame e funghi. Quest'ultima correlazione conferma una delle ipotesi precedentemente menzionate nella Sezione 5.4.

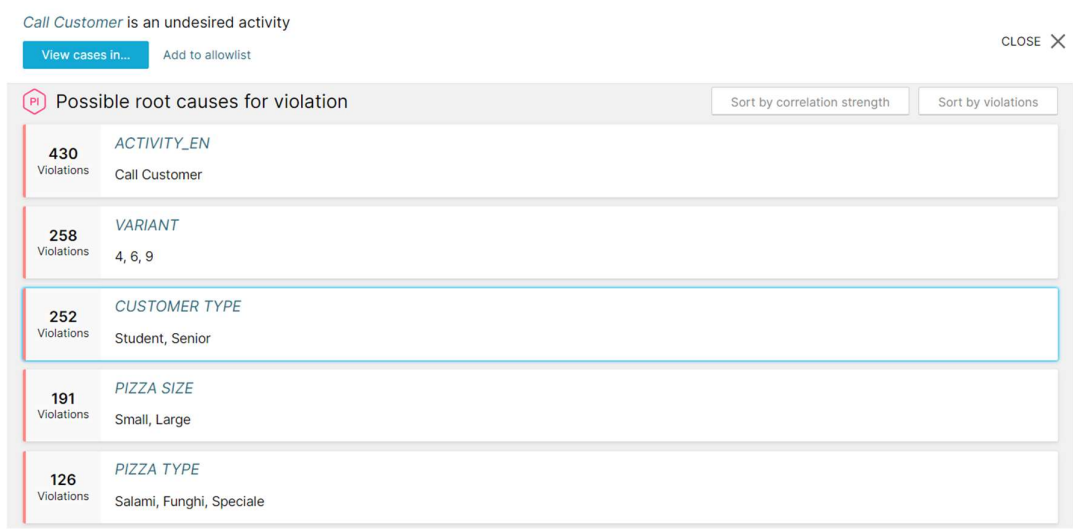


Figura 47: Conformance Checking: Possibili cause delle violazioni (Celonis)

È possibile ottenere queste informazioni anche su un sottoinsieme di dati su cui si desidera concentrare l'analisi, semplicemente includendo o escludendo determinate attività, connessioni e altri metadati.

6. Conclusioni

Il fulcro di questo lavoro è stato illustrare come l'applicazione dell'Intelligent Process Automation possa essere considerata una pratica efficace, capace di offrire numerosi vantaggi alle organizzazioni. Questa ricerca ha fornito le basi per comprendere la natura di un progetto di IPA e le circostanze appropriate per la sua applicazione. Un'analisi delle metodologie di sviluppo comunemente più utilizzate per realizzare un progetto software ha permesso di evidenziare i vantaggi e gli svantaggi di ciascuna, i casi in cui dovrebbero essere utilizzate e le migliori pratiche per una corretta implementazione. Inoltre, sono state analizzate in dettaglio le fasi del ciclo di vita di un progetto, offrendo una visione ampia delle tecniche più utilizzate per progettare, sviluppare, testare e implementare un progetto di IPA.

In seguito, è stata esplorata una tecnologia emergente, il Process Mining, che sta crescendo rapidamente. I risultati ottenuti hanno rivelato come questa possa offrire numerosi vantaggi e sia fortemente interconnessa con l'IPA. Attraverso l'utilizzo di Celonis e il caso studio sulla preparazione e spedizione della pizza, è stato possibile verificare i benefici teorici riscontrati dal questionario condotto sul Process Mining. Infatti, si è potuto constatare che questa tecnologia offre una **visione trasparente del proprio processo**, evidenziando tutti i flussi (dal più al meno frequente), permettendo di confrontare quelli ideali e non, e mettendo in risalto quelle attività o l'ordine con cui vengono eseguite, che non sono considerate ideali.

L'analisi condotta ha rivelato la presenza di attività duplicate, come la preparazione della pizza, e attività ricorrenti che non dovrebbero far parte del flusso desiderato dal cliente, come le chiamate effettuate da parte della pizzeria verso il cliente stesso. Si è quindi dimostrato che il Process Mining offre una concreta possibilità di **ristrutturare i propri flussi** attraverso l'identificazione di attività ripetitive e manuali che inevitabilmente portano a un incremento dei tempi di esecuzione. Alcune di queste attività (come la pianificazione del percorso, il pagamento online) potrebbero essere automatizzate o svolte in un altro momento (ad esempio, mentre si cuoce la pizza) in modo da evitare un collo di bottiglia.

L'attività indesiderata della chiamata al cliente ha messo in luce un problema nel flusso generale del processo che però viene amplificato nel momento in cui si effettuano ordini online. Questo suggerisce la possibilità di apportare modifiche al processo per evitare di chiedere manualmente informazioni che porterebbero ad un incremento del tempo di esecuzione. Inoltre, Celonis, individuando correlazioni tra i dati e le attività indesiderate, suggerisce le possibili cause offrendo così la possibilità di confermarle o meno. Questi suggerimenti, oltre a portare a una

potenziale modifica dei propri flussi, portano inevitabilmente a un **risparmio di tempo e di costi** e un **aumento della soddisfazione del cliente**.

Questa analisi dimostra quindi come il Process Mining sia perfettamente allineato con l'Intelligent Process Automation (IPA). Oltre ad offrire una migliore comprensione di come vengono eseguiti i processi all'interno della propria organizzazione, il Process Mining offre delle opportunità per individuare le aree in cui vengono svolte attività manuali e ripetitive, rendendo possibile l'introduzione dell'automazione per migliorare l'efficienza e ridurre i costi. Una volta identificate queste aree e apportate le modifiche, è anche possibile monitorarle e misurare l'efficacia dell'automazione tramite l'utilizzo di KPI appositamente realizzati.

Bibliografia

- [1] Acumenresearchandconsulting, 'Intelligent Process Automation Market Size, Share, Analysis Report By Component, By Technology , By Deployment Type, By Organization Size, By Application, By Industry Vertical, And Region Forecast, 2022 - 2030'. Accessed: Mar. 26, 2024. [Online]. Available: <https://www.acumenresearchandconsulting.com/intelligent-process-automation-market>
- [2] Precedence Research, 'Intelligent Process Automation Market - Global Industry Analysis, Size, Share, Growth, Trends, Regional Outlook, and Forecast 2023-2032'. Accessed: Mar. 26, 2024. [Online]. Available: <https://www.precedenceresearch.com/intelligent-process-automation-market#:~:text=The%20global%20intelligent%20process%20automation%20market%20size%20is%20expected%20to,14.6%25%20between%202023%20and%202032>
- [3] Gartner, 'Gartner Forecasts Worldwide Public Cloud End-User Spending to Reach Nearly \$600 Billion in 2023'. Accessed: Mar. 26, 2024. [Online]. Available: <https://www.gartner.com/en/newsroom/press-releases/2023-04-19-gartner-forecasts-worldwide-public-cloud-end-user-spending-to-reach-nearly-600-billion-in-2023>
- [4] S. Mohanty and S. Vyas, 'Intelligent Process Automation = RPA + AI', in *How to Compete in the Age of Artificial Intelligence*, Apress, 2018, pp. 125–141. doi: 10.1007/978-1-4842-3808-0_5.
- [5] A. S. Ray, C. Tornbohm, M. Kerremans, and D. Miers, 'Move Beyond RPA to Deliver Hyperautomation'.
- [6] P. S. Kholiya, A. Kapoor, M. Rana, and M. Bhushan, 'Intelligent Process Automation: The Future of Digital Transformation', in *Proceedings of the 2021 10th International Conference on System Modeling and Advancement in Research Trends, SMART 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 185–190. doi: 10.1109/SMART52563.2021.9676222.
- [7] Thoughtful, 'What Are the Benefits of Intelligent Process Automation?' Accessed: Mar. 26, 2024. [Online]. Available: <https://www.thoughtful.ai/blog/what-are-the-benefits-of-intelligent-process-automation>
- [8] F. Berruti, G. Nixon, G. Taglioni, and R. Whiteman, 'Intelligent process automation: The engine at the core of the next-generation operating model'.
- [9] K. K. H. Ng, C. H. Chen, C. K. M. Lee, J. (Roger) Jiao, and Z. X. Yang, 'A systematic literature review on intelligent automation: Aligning concepts from theory, practice, and future perspectives', *Advanced Engineering Informatics*, vol. 47, Jan. 2021, doi: 10.1016/j.aei.2021.101246.
- [10] Paola Cozzi, 'Il passaggio dalla Robotic Processing Automation all'Intelligent Automation: vantaggi, applicazioni e sfide aperte'. Accessed: Mar. 26, 2024. [Online]. Available: <https://tech4future.info/convergenza-automazione-intelligenza-artificiale/>
- [11] Kent Beck *et al.*, 'Manifesto per lo Sviluppo Agile di Software'.
- [12] UKEssays, 'Agile methodology'. Accessed: Mar. 26, 2024. [Online]. Available: <https://www.ukessays.com/essays/information-systems/agile-methodology.php#citethis>

- [13] Mirko Menecali, 'Waterfall vs agile, quali differenze e quale metodologia usare'. Accessed: Mar. 26, 2024. [Online]. Available: <https://blog.sinfo-one.it/waterfall-e-agile-due-approcci-contrapposti-al-project-management>
- [14] ZapTest, 'Qual è il ciclo di vita della RPA?'
- [15] S. ANAGNOSTE, 'Setting Up a Robotic Process Automation Center of Excellence', *Management Dynamics in the Knowledge Economy*, vol. 6, no. 2, pp. 307–322, 2013, doi: 10.25019/MDKE/6.2.07.
- [16] Expert.ai, 'Generali Customer Interview'. Accessed: Mar. 26, 2024. [Online]. Available: <https://www.expert.ai/it/testimonials/generali-customer-interview-2/>
- [17] N. N. Zolkifli, A. Ngah, and A. Deraman, 'Version Control System: A Review', in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 408–415. doi: 10.1016/j.procs.2018.08.191.
- [18] Aayush Agarwal, Subhash Gupta, and Tanupriya Choudhury, 'Continuous and Integrated Software Development using DevOps', *International Conference on Advances in Computing and Communication Engineering*, 2018.
- [19] ZapTest, 'Che cos'è l'automazione dei test? Una guida semplice e senza gergo'.
- [20] B. Wujek, P. Hall, and F. Güneş, 'Best Practices for Machine Learning Applications'.
- [21] R. Kumar and M. M. Ali, 'A Review on Chatbot Design and Implementation Techniques', *International Research Journal of Engineering and Technology*, 2008, [Online]. Available: www.irjet.net
- [22] J. Kantor, 'Best practices for implementing ChatGPT, large language models, and artificial intelligence in qualitative and survey-based research', *JAAD International*, vol. 14. Elsevier B.V., pp. 22–23, Mar. 01, 2024. doi: 10.1016/j.jdin.2023.10.001.
- [23] Harold Campos, Saisang Cai, Kent Sharkey, Duncan Mackenzie, and Mandi Ohlinger, 'Pianificazione degli ambienti di sviluppo, test, gestione temporanea e produzione'. Accessed: Mar. 26, 2024. [Online]. Available: <https://learn.microsoft.com/it-it/biztalk/technical-guides/planning-the-development-testing-staging-and-production-environments>
- [24] S. Nidhra, 'Black Box and White Box Testing Techniques - A Literature Review', *International Journal of Embedded Systems and Applications*, vol. 2, no. 2, pp. 29–50, Jun. 2012, doi: 10.5121/ijesa.2012.2204.
- [25] A. Verma, A. Khatana, and S. Chaudhary, 'A Comparative Study of Black Box Testing and White Box Testing', *Article in International Journal of Computer Sciences and Engineering*, 2017, doi: 10.26438/ijcse/v5i12.301304.
- [26] Codefresh.io, 'Software Deployment in 2024: Checklist, Strategies & Tips'. Accessed: Mar. 26, 2024. [Online]. Available: <https://codefresh.io/learn/software-deployment/>
- [27] Hardik Thakker, 'Software Deployment Best Practices in 2023'. Accessed: Mar. 26, 2024. [Online]. Available: <https://medium.com/@hardikthakker/software-deployment-best-practices-4a0c0e0af65b>

- [28] Wotawa Franz and Hofer Birgit, 'Software-Maintenance Lecture Notes', 2009. [Online]. Available: <http://www.ist.tugraz.at/>
- [29] G. Destri and C. Lombardi, 'I PROCESSI DI SVILUPPO SOFTWARE: LA STORIA', 2016.
- [30] Hardik Shah, 'Best Practices for Effective Software Maintenance'. Accessed: Mar. 26, 2024. [Online]. Available: <https://hardiks.medium.com/best-practices-for-effective-software-maintenance-9493934470b5>
- [31] Mathias Weske, *Business Process Management Concepts, Languages, Architectures*. 2007.
- [32] L. Ardito, F. Corno, and M. Torchiano, 'Sistemi Informativi Aziendali Appunti per il corso', 2022.
- [33] W. Van der Aalst, *Process mining: Data science in action*. Springer Berlin Heidelberg, 2016. doi: 10.1007/978-3-662-49851-4.
- [34] W. Van Der Aalst *et al.*, 'Process Mining Manifesto', 2012.
- [35] Deloitte, 'Global Process Mining Survey insights', 2023.
- [36] Lari Numminen, 'Top 6 benefits of Process Mining (new 2024 research)'. Accessed: Mar. 27, 2024. [Online]. Available: <https://www.workfellow.ai/it/learn/benefits-of-process-mining>