Politecnico di Torino

Politecnico di Torino

Dipartimento di Ingegneria Gestionale e della Produzione

Corso di Laurea Magistrale in Ingegneria Gestionale – Percorso Informatico

A.a. 2022/2023

Sessione di Laurea Marzo 2023

# La Gestione delle Interazioni in Tempo Reale nel Marketing Relazionale e lo Sviluppo Agile di Soluzioni Aziendali su Piattaforme in Low-Code

**Relatore:**
Prof. CLAUDIO GIOVANNI DE MARTINI

**Candidato:**
MATTEO RIZZI

# Indice

Indice delle figure ..... I

Indice delle tabelle ..... IV

Indice dei grafici ..... V

Introduzione ..... 1

1. Real Time Interaction Management e Costumer Decision Hub ..... 4

1.1 Il modello di marketing tradizionale ..... 6

1.2 Il coinvolgimento dei clienti 1-a-1 nel marketing relazionale e il confronto con il modello tradizionale ..... 8

1.2.1 Segmentazione one-to-many vs approccio 1-to-1 ..... 9

1.2.2 Targeting su canali vs autorità decisionale centralizzata ..... 12

1.2.3 Campagne push pianificate vs modello always-on ..... 14

1.3 Il Customer Decision Hub di Pega ..... 15

1.3.1 Il Next-Best-Action Designer ..... 15

1.4 L'impatto economico totale dell'uso del CDH ..... 20

1.5 Un caso di studio: la Commonwealth Bank of Australia ..... 23

1.6 Confronto tra le piattaforme di *Real-Time Interaction Management* ..... 24

2. Piattaforme Low-Code e Pega Platform ..... 27

2.1 Il confronto con lo sviluppo tradizionale e i vantaggi e svantaggi del low-code ..... 28

2.2 Pega Platform ..... 30

2.3 L'impatto economico totale del Pega Platform per Low-Code ..... 32

2.4 Confronto tra le piattaforme di sviluppo Low-Code ..... 34

3. La metodologia Agile ..... 37

3.1 Il confronto tra la metodologia Agile e la metodologia Waterfall 38

3.2 I principi e il ciclo di vita dello sviluppo software in Agile ..... 40
3.3 Il framework Scrum ..... 43
3.3.1 I ruoli e gli eventi di Scrum ..... 43
3.3.2 Gli artefatti di Scrum ..... 46
4. Progetto ..... 48
4.1 Fase di analisi ..... 50
4.1.1 Analisi del problema: *Problem tree* ..... 50
4.1.2 Analisi degli obiettivi: *Solution tree* ..... 51
4.1.3 Analisi della strategia: *Strategy selection* ..... 52
4.2 Pega Platform e Scrum: identificazione ruoli del progetto ..... 53
4.3 Fase di pianificazione e Jira Software ..... 54
4.4 Realizzazione del progetto ..... 58
4.4.1 Lo stato dell'arte del progetto ..... 59
4.4.2 Aggiornamento *Cruscotto sottomissione RdA*: Inserimento dati richiedente ..... 60
4.4.3 Aggiornamento *Cruscotto sottomissione RdA*: Inserimento dati del bene richiesto ..... 64
4.4.4 Aggiornamento *Cruscotto sottomissione RdA*: Revisione richiesta ..... 68
4.4.5 Aggiornamento *Cruscotto sottomissione RdA*: Fase di approvazione ..... 71
4.4.6 *Cruscotto CIO*: Menu e Homepage ..... 76
5. Risultati del progetto ..... 79
6. Conclusioni ..... 86
Lista degli acronimi ..... 89
Bibliografia ..... 90
Ringraziamenti ..... 93

# Indice delle figure

**Figura 1:** HomePage del Costumer Decision Hub [8] ..... 5

**Figura 2:** Caratteristiche della NBA (da Karen Larrimer, capo Retail Bank e CCO PNC Bank) [6] ..... 10

**Figura 3:** Next-Best-Action e Next-Best-Offer [6] ..... 11

**Figura 4:** Strategia decentralizzata e strategia centralizzata a confronto [6] ..... 13

**Figura 5:** Interfaccia del Next-Best-Action Designer [7] ..... 15

**Figura 6:** Interfaccia della scheda *Taxonomy* [7] ..... 16

**Figura 7:** Interfaccia della scheda *Constraints* [7] ..... 17

**Figura 8:** Interfaccia della scheda *Engagement policies* [7] ..... 18

**Figura 9:** Interfaccia della scheda *Arbitration* [7] ..... 18

**Figura 10:** Interfaccia della scheda *Channels* [7] ..... 19

**Figura 11:** Come cambiano le azioni dopo l'uso delle regole del CDH [7] ..... 19

**Figura 12:** Confronto tra i fornitori di Real-Time Interaction Management [15] 24

**Figura 13:** Assegnazione punteggio ai fornitori di RTIM [15] ..... 25

**Figura 14:** Confronto tra i fornitori di LCDP [25] ..... 35

**Figura 15:** Modo con cui i metodi Waterfall e Agile creano valore e rischio associato [28] ..... 39

**Figura 16:** Schema del ciclo di sviluppo software in Agile [26] ..... 42

**Figura 17:** Ruoli, processi e responsabilità nel framework Scrum [28] ..... 44

**Figura 18:** Le cerimonie Scrum [33] ..... 46

**Figura 19:** Problem tree ..... 50

**Figura 20:** Solution tree ..... 51

**Figura 21:** Strategy selection ..... 52

**Figura 22:** Ruoli identificati da Pega [17] ..... 53

**Figura 23:** Esempio di elementi del Product Backlog su Jira Software ..... 55

**Figura 24:** Esempio di elementi dello Sprint Backlog su Jira Software (settimo Sprint) ..... 55

**Figura 25:** Esempio della sezione Board di Jira Software (settimo Sprint) ..... 56

**Figura 26:** Assegnatario e Story point estimate di una User Story ..... 57

**Figura 27:** Roadmap del progetto Portale CIO Blue Reply su Jira Software .....57
**Figura 28:** Case Life Cycle del Case Type *Richiesta RdA* a fine progetto .....58
**Figura 29:** Stato dell'arte del Case Life Cycle [36] .....59
**Figura 30:** Data Type *Manager* con alcuni dei suoi record .....60
**Figura 31:** Stage *Create RdA* per l'inserimento dati richiedente .....60
**Figura 32:** Section con i campi input per l'inserimento dati del richiedente .....61
**Figura 33:** Controllo sul campo *E-mail* .....62
**Figura 34:** Data transform *CheckManager* .....62
**Figura 35:** Condizioni per la disabilitazione della dropdown list *Nome Manager* .....63
**Figura 36:** Report definition per filtrare i manager con e-mail come parametro .63
**Figura 37:** Data transform *CheckManager* inserita nell'*Action* su Change dell'*E-mail* .....64
**Figura 38:** Stage *Raccolta informazioni* per l'inserimento dati del bene richiesto .....64
**Figura 39:** Stato del processo per lo Stage *Raccolta informazioni* .....65
**Figura 40:** Section con i campi per l'inserimento dati del bene richiesto .....65
**Figura 41:** Tabella dettagli ordine *Amazon* .....66
**Figura 42:** Tabella dettagli ordine *Esprinet* .....66
**Figura 43:** Tabella dettagli ordine *Altro fornitore* .....67
**Figura 44:** Data transform *GetProprietaTabellaGeneraliAmazon* .....67
**Figura 45:** Stage *Revisione Richiesta* .....68
**Figura 46:** Process flow del processo *Revisione Richiesta HW* .....69
**Figura 47:** Destinatari e-mail per richiesta approvazione del manager .....69
**Figura 48:** Corpo della mail in caso di tipo di richiesta *Hardware* .....70
**Figura 49:** Corpo della mail in caso di tipo di richiesta *Software/Altra Richiesta* .....70
**Figura 50:** Testo della modal dialog *Thank you page* .....71
**Figura 51:** Stage della fase di approvazione .....72
**Figura 52:** Section dello Step *Approvazione RdA* .....72
**Figura 53:** Routing dello Step *Approvazione RdA* .....73

**Figura 54:** Impostazione flow seguito dal pulsante *Conferma* nello Stage *Approvazione RdA* ..... 73
**Figura 55:** Data Type *Membro CIO FS* con i suoi record ..... 74
**Figura 56:** Invio della notifica al membro del team Blue CIO ..... 74
**Figura 57:** Process flow del processo *Approval Rejection* ..... 75
**Figura 58:** Corpo della mail di rifiuto con motivazione ..... 75
**Figura 59:** Corpo della mail di rifiuto senza motivazione ..... 75
**Figura 60:** Interfaccia della sezione *Channels* di App Studio ..... 76
**Figura 61:** Voce *Home* del menu inserita da App Studio ..... 76
**Figura 62:** Voci del menu inserite nella section *MainNavigation* in Dev Studio. 77
**Figura 63:** Item *Configurazione BU* ..... 77
**Figura 64:** Tabella con la lista delle richieste di acquisto ..... 78
**Figura 65:** Activity *DownloadExcel* ..... 78
**Figura 66:** Esempio di inserimento dati richiedente ..... 79
**Figura 67:** Esempio di inserimento dati richiedente con e-mail sbagliata ..... 80
**Figura 68:** Esempio inserimento dati del bene richiesto - Software ..... 80
**Figura 69:** Esempio inserimento dati del bene richiesto – Hardware ..... 81
**Figura 70:** Esempio revisione richiesta - Hardware ..... 81
**Figura 71:** Modal dialog della Thank you page ..... 82
**Figura 72:** Esempio di e-mail inviata al manager per l'approvazione - Hardware ..... 82
**Figura 73:** Esempio fase approvazione del manager ..... 83
**Figura 74:** Esempio fase approvazione del manager - Richiesta rifiutata ..... 83
**Figura 75:** Esempio mail di rifiuto inviata al richiedente ..... 83
**Figura 76:** Menu del portale *Cruscotto CIO* ..... 84
**Figura 77:** Interfaccia dell'homepage del *Cruscotto CIO* ..... 84
**Figura 78:** File excel generato dal pulsante *Download* dell'homepage ..... 84
**Figura 79:** Sprint Retrospective finale sul progetto ..... 87

# Indice delle tabelle

**Tabella 1:** Confronto tra campagne tradizionali e approccio 1-to-1 Customer Engagement [6] ..... 8

**Tabella 2:** Esempio di calcolo priorità per NBA [10] ..... 12

**Tabella 3:** Analisi del Cash Flow – TEI CDH [11] ..... 21

**Tabella 4:** Vantaggi dello sviluppo low-code rispetto allo sviluppo tradizionale [20] ..... 28

**Tabella 5:** Analisi del Cash Flow - TEI Pega Platform per low-code [24] ..... 33

**Tabella 6:** Confronto tra metodologia Waterfall e metodologia Agile [28] ..... 40

# Indice dei grafici

**Grafico 1:** Payback e PV dei profitti – TEI CDH [11] .....21
**Grafico 2:** Andamento del Cash Flow – TEI CDH [11].....22
**Grafico 3:** Payback e PV dei profitti - TEI Pega Platform per low-code [24].....32
**Grafico 4:** Andamento del Cash Flow - TEI Pega Platform per low-code [24] ...33
**Grafico 5:** Diffusione delle metodologie Agile [28].....38
**Grafico 6:** Esempio di Burndown Chart [28] .....47

# Introduzione

Negli ultimi anni è entrato nelle case delle persone il concetto di transizione digitale, amplificato soprattutto a partire dalla pandemia di Covid-19. Nei media nazionali questo argomento è legato in particolar modo al processo di digitalizzazione nella Pubblica Amministrazione; non è infatti un caso che nel sessantaseiesimo esecutivo - governo Conte II - e nel settantasettesimo esecutivo - Governo Draghi - della Repubblica Italiana sia stato istituito un Ministero dedicato, inizialmente denominato Ministero dell'Innovazione tecnologica e digitalizzazione, e successivamente cambiato in Ministero dell'Innovazione tecnologica e transizione digitale.

Il processo di digitalizzazione dei servizi, però, non coinvolge solo la Pubblica Amministrazione, ma è una sfida che riguarda, ormai da tempo, la maggior parte delle aziende. Una delle soluzioni che le organizzazioni adottano per accelerare questo processo è data dall'utilizzo delle *Low-Code Development Platforms* (LCDPs), che permettono di gestire in maniera efficace nuovi requisiti di business, tramite una visione semplificata dei sistemi, riducendo il time-to-market. Le piattaforme low code offrono un ambiente software e applicativo semplificato: invece di utilizzare la programmazione tramite codice, si usano interfacce grafiche e impostazioni user-friendly per creare soluzioni di business.

La *Forrester Research Inc.* descrive lo sviluppo low-code come prodotti e/o servizi cloud per lo sviluppo di applicazioni che impiegano tecniche visive e dichiarative invece della programmazione tradizionale tramite codice (*Low-code platforms employ visual, declarative techniques instead of traditional lines of programming*) [1], favorendo la collaborazione tra team aziendali e tecnologici grazie all'utilizzo di un linguaggio visivo comune. Degli studi condotti da *Gartner* hanno rivelato che il 76% delle aziende adottano low code per incrementare la produttività. Tra questi, il 67% desidera ridurre i tempi per guadagnare market share e il 63% desidera automatizzare i processi. [2] In questo elaborato verrà

approfondito il tema dello sviluppo di applicazioni a basso codice, anche in confronto allo sviluppo tradizionale.

Oltre al tema della digitalizzazione, un'altra sfida di questo periodo storico riguarda lo sviluppo di AI (*Artificial Intelligence*) e machine learning in moltissimi ambiti. L'esempio più recente che ha avuto una grossa risonanza nel grande pubblico generalista riguarda il rilascio della versione beta di *ChatGPT*, il prototipo di ChatBot realizzato da OpenAI che permette di interagire con un utente umano tramite conversazioni.

In questa tesi verrà approfondito il tema dell'intelligenza artificiale e del machine learning nell'ambito del marketing relazionale tramite la gestione delle interazioni in tempo reale (o *Real-Time Interaction Management* - RTIM), un motore decisionale istantaneo che permette di avere comunicazioni personalizzate con i clienti per creare esperienze ottimali e coerenti per ogni individuo, allineando tutti i canali. [3] Questa esigenza nasce in quanto le campagne promozionali, seppur rappresentino una parte significativa delle comunicazioni di un'organizzazione con i propri clienti, tramite la trasmissione di nuove offerte o proposte di servizi aggiunti, sono create per l'esigenza delle aziende e non dei clienti, non riuscendo dunque ad ottimizzare la soddisfazione di questi ultimi (che si riflette anche in risultati non ottimali per le aziende stesse).

Come esempi pratici di piattaforme per l'automazione di processi digitali e per la gestione delle interazioni in tempo reale verranno esaminati due software sviluppati da Pegasystems, facenti parte del portfolio di *Pega Infinity*: il *Customer Decision Hub* (CDH) e il *Pega Platform*. In questo elaborato la trattazione riguardante il Real-Time Interaction Management e il Customer Decision Hub sarà solamente teorica, e servirà come punto di partenza per ulteriori sviluppi per l'ente in cui ho realizzato lo stage di tesi in azienda, la società *Blue Reply Srl* (altresì *Blue Financial Services*) del gruppo Reply. La società sta cercando di sperimentare in questo ambito per successivi utilizzi aziendali, e la ricerca teorica potrà essere utile per una conseguente realizzazione pratica di una applicazione sulla gestione delle interazioni in tempo reale.

La trattazione si concluderà con la descrizione di un progetto che ho sviluppato durante il tirocinio di tesi in azienda con Blue Financial Services, che offre consulenza di soluzioni innovative di differenziazione basate su tecnologie open source e fornitori leader di mercato, tra cui Pega. Il progetto riguarda la realizzazione di una applicazione in low-code tramite l'utilizzo di Pega Platform. La piattaforma è stata sviluppata in cooperazione con un altro tesista e, per ottimizzare l'organizzazione del lavoro, è stato utilizzato un approccio Agile, con l'adozione del framework *Scrum*, che sarà argomento di trattazione della tesi.

Il risultato del progetto è stato il raggiungimento di una pietra miliare per la realizzazione dell'applicazione. Ha mostrato anche come sviluppatori non professionisti, grazie alla conoscenza di processi aziendali, alle competenze cross-funzionali acquisite durante il percorso di Laurea Magistrale in Ingegneria Gestionale, e ad una opportuna formazione, possono essere in grado di sviluppare una piattaforma funzionante e funzionale da un punto di vista grafico, anche in tempi brevi.

# 1. Real Time Interaction Management e Costumer Decision Hub

Il *Real Time Interaction Management* è definito da Forrester come una *tecnologia di marketing aziendale che fornisce esperienze, valore e utilità contestualmente rilevanti nel momento appropriato del ciclo di vita del cliente attraverso i punti di contatto del cliente preferiti.* Esso permette di prendere la decisione per la Next Best Experience (NBX) per i clienti attraverso vari canali, tenendo conto che i loro bisogni e comportamenti possono evolversi nel tempo, tramite l'utilizzo di interazioni in tempo reale, modellazione predittiva e machine learning. Si cerca di garantire risultati quanto più soddisfacenti possibile per i clienti, considerando degli obiettivi aziendali da raggiungere. Non si tratta, però, solo di acquisire e convertire nuovi clienti, ma anche di supportare i canali di auto-servizio digitale e quelli assistiti da operatori umani attraverso tutti i ruoli che hanno contatti con i clienti. [4] Questo concetto è alla base di ciò che è chiamato *Customer Relationship Management* (o CRM), che è *un approccio strategico che si occupa di creare un maggior valore per gli shareholders attraverso lo sviluppo di relazioni appropriate con i clienti chiave e i segmenti di clientela. Il CRM unisce il potenziale delle strategie di marketing relazionale e dell'IT per creare relazioni proficue e a lungo termine con i clienti e gli altri stakeholder chiave*. [5]

Implementare un sistema RTIM, in ogni caso, non è un *free lunch*, in quanto richiede significativi investimenti sia in tecnologia che in risorse umane, ma, per coloro che decidono di adottarlo, si prevede un sostanziale ROI a lungo termine, che può superare il mezzo miliardo di dollari. [4]

Per poter capire perché c'è stato un grande aumento di adozione di RTIM negli ultimi anni nell'ambito del marketing relazionale è necessario inquadrare il contesto in cui esso viene utilizzato. Il marketing tradizionale ha sempre puntato ad invadere i consumatori con comunicazioni, campagne strategiche e segmentazioni, retargeting, push su dispositivi mobili, mail personalizzate. I

risultati ottenuti ultimamente, però non sono più rilevanti come qualche anno fa e, siccome l'impatto delle iniziative si riduce sempre di più, per cercare di ottenere più risultati si punta ad aumentare le comunicazioni, coinvolgendo i clienti in più luoghi e modi. Questi sforzi, però, non portano benefici, ma porta i potenziali clienti a ignorare le promozioni, ottenendo una risposta sempre più bassa.

L'alternativa al marketing tradizionale è diventata possibile negli ultimi anni grazie ai miglioramenti nelle interazioni in tempo reale e nelle funzionalità IA, superando gli ostacoli tecnologici e permettendo un approccio più corretto, ovvero uno che permetta un marketing *customer-centric* e non *product-centric*. Ciò permetterebbe di aumentare il fatturato e profitto per l'azienda nel complesso e contestualmente aumentare la soddisfazione dei clienti, aumentando così potenzialmente il futuro fatturato. Per questo motivo ora il concetto di marketing relazionale con approccio al cliente 1-a-1 sta prendendo sempre più piede, e le uniche barriere rimanenti sono rappresentate dall'inerzia e dalla mentalità dovute a vecchie culture aziendali e processi. [6]

Pegasystems permette di implementare il Real Time Interaction Management tramite la piattaforma Customer Decision Hub. Al suo interno, il Next-Best-Action Designer guida la creazione di una strategia di Next-Best-Action e, tramite una interfaccia intuitiva, best practice comprovate e una sofisticata tecnologia decisionale di base, consente di offrire automaticamente esperienze cliente personalizzate attraverso i canali in entrata, in uscita e a pagamento. [7]

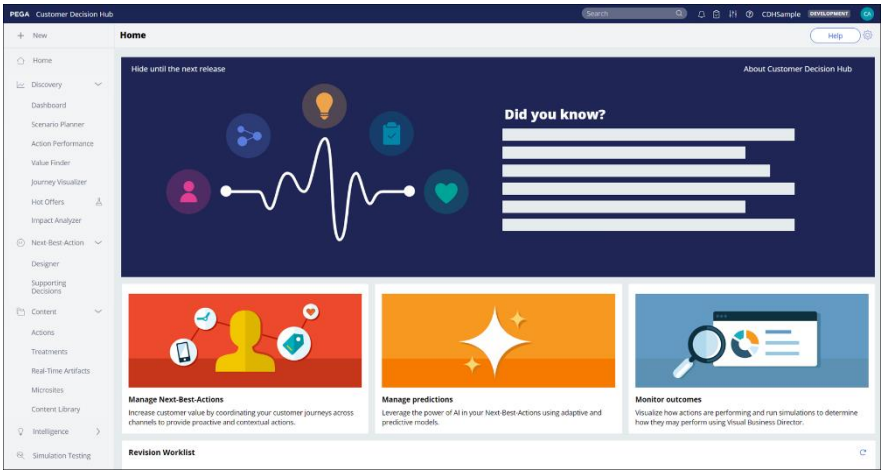


**Figura 1:** HomePage del Costumer Decision Hub [8]

## 1.1 Il modello di marketing tradizionale

Per riuscire ad analizzare meglio i limiti del marketing tradizionale e capire come e perché passare ad un approccio 1-to-1 con il consumatore, è necessario inquadrare più dettagliatamente il modello della campagna tradizionale. L'uso delle campagne si è diffuso specialmente negli anni '80 e '90 e sono uno strumento pensato per distribuire un obiettivo di fatturato aziendale complessivo su più linee di business e prodotti. Solitamente, a seguito della promozione di nuovi prodotti, una volta raggiunta una determinata soglia di vendite, l'obiettivo si considerava come raggiunto. Questo, però, ha portato le singole unità aziendali a competere per una fetta di mercato e a perdere comunicazione tra di loro, rendendo inefficiente e inefficace l'ottimizzazione dell'obiettivo finale. Le campagne, infatti, sono isolate le une dalle altre, e ognuna tenta di massimizzare un proprio obiettivo senza tener conto di tutte le altre. Questo approccio è definito come *inside-out* e *product-centric*, contrapposto ad uno *outside-in* e *customer-centric*. Ciò porta a perdere di vista lo scopo finale, ovvero l'aumento del fatturato dell'azienda nel suo complesso e la soddisfazione e fidelizzazione dei clienti, che impatta sui guadagni futuri.

Il processo che porta al rilascio di una nuova campagna promozionale si compone di diverse fasi, di seguito elencate:

1. **Definizione dell'obiettivo:** il primo step è quello di definire il/gli obiettivo/i della campagna, come ad esempio vendere almeno un tot di quantità di un prodotto.
2. **Definizione dell'audience:** fase in cui si costruiscono i segmenti e si identificano i destinatari della campagna.
3. **Design della creatività:** a questo punto occorre progettare la parte creativa della campagna, cercando di rendere il prodotto e la sua promozione accattivanti per il target di riferimento.
4. **Ottimizzazione su più campagne:** dato che è molto probabile che alcuni destinatari della campagna siano inclusi in più segmenti, li si dovrebbe

poter rimuove da alcune campagne, per incontrare alcuni criteri come numero di contatti o volumi necessari per la campagna.

5. **Esecuzione della campagna:** questo è lo step di lancio della campagna vero e proprio. Vengono individuati vari canali in uscita (es. mail, SMS, posta, ecc.) e viene talvolta configurata la visualizzazione dell'offerta su specifici canali in entrata.
6. **Analisi di risposta:** post lancio della campagna, gli analisti raccolgono tutti i dati della promozione durante tutta la durata della campagna e analizzano gli attributi dei contatti che hanno risposto o meno, in modo da migliorare la segmentazione.

Uno step critico tra le varie fasi descritte è rappresentato dalla segmentazione, che può essere effettuata su più livelli, con, ad esempio, un primo livello più generico in cui si indicano informazioni anagrafiche, per poi dettagliare in seguito tramite criteri di idoneità all'offerta o propensione a rispondere a offerte simili. Questo modo di procedere, però, presenta dei limiti importanti. Innanzitutto, viene **compromessa la pertinenza**, in quanto il compito della segmentazione è quello di essere un filtro, che però mescola criteri di idoneità, compatibilità, urgenza, preferenza, il che porta a far venire meno precisione e trasparenza. Un altro limite della segmentazione è dato dal **lavoro non scalabile**: nel caso in cui si propensi per una micro-segmentazione, la riutilizzabilità, la trasparenza e la scalabilità verrebbero meno, poiché sarebbe difficile capire l'utilizzo di alcuni criteri, mentre una segmentazione troppo ampia non è efficace. Un ultimo problema, infine, è quello di eventuali **collisioni e conflitti**, in quanto accade spesso che i segmenti entrino in conflitto tra loro: può accadere che i clienti inclusi in un segmento sono inclusi anche in altri (o addirittura in campagne concorrenti). [6]

## 1.2 Il coinvolgimento dei clienti 1-a-1 nel marketing relazionale e il confronto con il modello tradizionale

Una volta analizzati i limiti del marketing tradizionale, occorre proporre l'alternativa a questo approccio, che è diventata possibile grazie all'evoluzione dell'intelligenza artificiale. Ciò che si tenta di fare, come già spiegato, è di fornire una migliore esperienza per il cliente, cercando di incrementare il suo benessere, e al contempo migliorare i KPI di tutti gli ambiti dell'azienda. Per perseguire questo obiettivo bisogna essere in grado di eseguire l'azione giusta nel canale giusto al momento giusto per ogni cliente, e ciò è possibile grazie al coinvolgimento dei clienti 1-a-1 (o *1-to-1 Customer Engagement*), che è alla base del marketing relazionale. Un approccio del genere consente alle aziende di avere delle conversazioni coerenti, contestuali e pertinenti con i singoli clienti attraverso qualsiasi canale, ma per poter fare ciò è necessario un unico cervello centralizzato che riesca a decidere e prevedere, tramite tutti i dati ottenuti – compresi quelli raccolti durante ogni iterazione col cliente, quale sia la migliore azione da proporre al cliente in un preciso momento. [7]

Per comprendere meglio la differenza tra i due approcci descritti, la seguente tabella mette a confronto le principali differenze tra il modello delle campagne tradizionali con il 1-to-1 Customer Engagement:

| Campagne tradizionali | 1-to-1 Customer Engagement |
|---|---|
| Selezione dell'offerta da promuovere | Selezione del cliente da coinvolgere |
| Definizione dell'audience con segmenti | Selezione della migliore offerta con l'IA decisionale |
| Pianificazione della comunicazione sui canali in uscita | Always-on sia in entrata che in uscita |
| Inizio e fine manuale della campagna | Offerte attivate/disattivate automaticamente |
| Successo determinato in base alla risposta dell'offerta | Successo determinato dal cambiamento del customer value |

**Tabella 1:** Confronto tra campagne tradizionali e approccio 1-to-1 Customer Engagement [6]

Come mostrato nella **Tabella 1**, l'approccio 1-to-1 ha come punto di inizio la selezione del cliente, invece che dell'offerta. L'analisi successiva, poi, produrrà l'offerta (in caso di Next-Best-Offer NBO), o più in generale l'azione (Next-Best-Action NBA), che potrebbe essere anche quella di non coinvolgere il cliente: questi verrà coinvolto solo quando c'è qualcosa di pertinente da dire nel giusto contesto, ovvero quando il cliente ha un'esigenza specifica. Inoltre, essendo l'approccio always-on, le offerte sono attivate o disattivate in base alle nuove informazioni in possesso. Infine, siccome si tenta di rivolgersi ai singoli clienti tenendo conto delle esigenze, delle preferenze e del contesto, si mira a ottimizzare i KPI di lungo termine come fatturato, profitto e *lifetime value* del cliente piuttosto che metriche di breve termine come il tasso di risposta alla campagna. [6]

### 1.2.1 Segmentazione one-to-many vs approccio 1-to-1

A differenza di una segmentazione one-to-many, che prevede la scelta di quali clienti includere per una campagna per una determinata offerta, un approccio 1-to-1 assegna una offerta, o azione, tra le varie possibili ad uno specifico cliente. Per riuscire a capire qual è l'azione migliore da compiere per uno specifico cliente, sono usate varie informazioni per calcolare, tramite l'AI, la NBA.

Il termine NBO o NBA è sempre più utilizzato per indicare una proposta personalizzata in base agli attributi e ai comportamenti del consumatore, al contesto d'acquisto, alle caratteristiche del prodotto o del servizio e alle strategie dell'organizzazione. Il più delle volte sono progettate per ispirare un acquisto, per fidelizzare o per entrambe le cose, e consistono in prodotti, servizi, informazioni e relazioni, servizi, informazioni e relazioni [9].

**Figura 2:** Caratteristiche della NBA (da Karen Larrimer, capo Retail Bank e CCO PNC Bank) [6]

L'approccio NBA guarda tutte le azioni potenziali, e non solo le migliori offerte di vendita. In molti casi, la vendita non è l'opzione migliore per un dato in cliente, in un dato momento e in un dato canale, ma potrebbero essere più pertinenti iniziative migliori in quel contesto, come un piano di fidelizzazione per un cliente con alte probabilità di rivolgersi alla concorrenza, oppure offrire assistenza proattiva ad un cliente in difficoltà. Per questo motivo, è più corretto parlare di approccio NBA (costumer experience) piuttosto che di approccio NBO (marketing), in quanto, al soddisfacimento di determinati criteri e condizioni, tiene conto non solo qual è l'offerta migliore, ovvero la Next-Best-Offer, ma anche fattori come fidelizzazione, assistenza, rischio, ecc.: la NBA, dunque, è un concetto più generale rispetto alla NBO, che invece corrisponde ad una delle opzioni che la NBA può prendere in considerazione. [6]

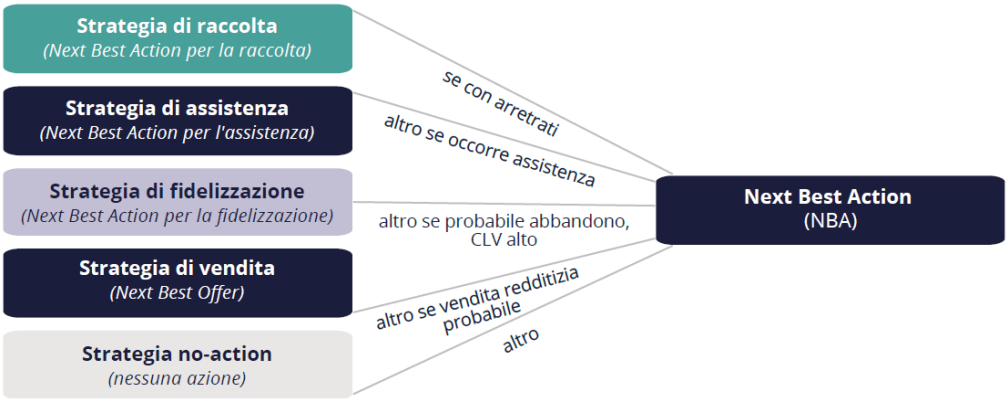


**Figura 3:** Next-Best-Action e Next-Best-Offer [6]

Per trovare la soluzione migliore, innanzitutto sono utilizzate le *business rules* per definire le condizioni in cui i clienti sono idonei, quando le azioni sono adatte e quando sono applicabili, per filtrare l'elenco delle azioni da considerare per un singolo cliente, per ottenere una migliore gestibilità.

Successivamente viene trovata la soluzione migliore, che solitamente è quella che aumenta maggiormente il Customer Life Value (CLV), calcolando le priorità di ogni azione tramite una formula che, nella versione più semplice, è: $Priorità = P * V * L$, dove:

- **P (Propensione):** è la propensione del cliente ad accettare una specifica azione, calcolata tramite un modello predittivo. È un numero compreso tra 0 ed 1 e, maggiore è la propensione del cliente a compiere una data azione, più vicino a 1 sarà il valore.
- **V (Valore):** è il valore che ha per l'azienda l'azione quando viene accettata dal cliente. Per calcolare questo valore possono essere utilizzate diverse metriche, dal calcolo del margine, al profitto monetario, al calcolo del CLV.
- **L (Leva):** è un valore opzionale che può essere utilizzato a propria discrezione dall'azienda se lo ritiene opportuno, magari per potenziare una azione con propensione bassa, ma che ha un interesse particolare per l'azienda. È un valore con una sua utilità, ma che se usato in maniera incontrollata può ridurre la pertinenza. [6]

Nella **Tabella 2** è presentato un esempio di calcolo di Next-Best-Action di varie azioni possibili per una azienda di servizi finanziari (che fruiscono molto di questo approccio). Come si può notare, l'azione *Well-being: COVID-19 payment holiday* è quella con un valore di priorità maggiore, ma è da sottolineare che è utilizzata una leva del 100%, che ha impattato significativamente sul risultato finale. [10]

| Potential actions | $P_{(Propensity)}$ | $V_{(Value)}$ | $L_{(Lever)}$ | Action ($) |
|---|---|---|---|---|
| **Offer:** Rewards card offer | 0.3% | $561 | 0% | 1.68 |
| **Offer:** Home equity loan | -- | -- | -- | -- |
| **Offer:** Mortgage loan offer | 0.1% | $834 | 0% | 0.83 |
| **Service:** Missing email address | 4.0% | $55 | 50% | 3.30 |
| **Service:** Travel notification | 3.0% | $29 | 20% | 1.04 |
| **Well-being:** COVID-19 payment holiday | 64.0% | $23 | 100% | 29.44 |
| **Retention:** "Manage your rewards" alert | 14.0% | $83 | 50% | 17.43 |
| **Retention:** Waive annual fee for one year | 5.0% | $48 | 0% | 2.40 |
| **Nurture:** Personal finance class | 3.0% | $135 | 20% | 4.86 |
| **Nurture:** New mobile app available | 21.0% | $20 | 20% | 5.04 |

**Tabella 2:** Esempio di calcolo priorità per NBA [10]

Utilizzare questo approccio permette anche di migliorare la scalabilità rispetto alle campagne di marketing tradizionali, in quanto serve solamente definire alcuni criteri per la nuova offerta e definire i valori per il calcolo della priorità.

### 1.2.2 Targeting su canali vs autorità decisionale centralizzata

Affinché le strategie NBA possano efficacemente aumentare il CLV occorre che siano utilizzate su tutti i canali, che siano *inbound* (in entrata), o *outbound* (in uscita). Per questo motivo è necessaria un'unica autorità decisionale centralizzata in grado di ottimizzare le comunicazioni su tutti i canali, appunto un *customer decision hub*. Questo aspetto è importante per diversi motivi, di seguito elencati:

- **Evitati i conflitti tra canali disconnessi:** nel caso del marketing tradizionale con targeting sui vari canali, accade che il canale, essendo

ogni decisione sconnessa dalle altre, proporrà una soluzione che collide con il resto della strategia. L'utilizzo di un'unica autorità decisionale, invece, permette il lavoro sinergico di tutti i canali, scegliendo la soluzione giusta per il cliente in quel dato contesto.

- **Ottimizzazione dei KPI a livello aziendale:** nel marketing tradizionale accade spesso che gli stessi clienti facciano parte di diverse campagne indipendenti l'una dall'altra, rendendo impossibile simulare l'impatto combinato delle varie campagne sull'intera azienda. Questo problema non si pone con una strategia centralizzata, in quanto i canali non sono indipendenti e al cliente viene proposta una soluzione alla volta.
- **Risolti i problemi di gestione dei cambiamenti:** la presenza di un'unica autorità decisionale permette di avere una gestione dei cambiamenti più efficiente a livello operativo e più informata. Inoltre, è anche più semplice e immediato imparare dalle risposte del cliente per ricalcolare la NBA e adattarsi al nuovo contesto. [6]

**Figura 4:** Strategia decentralizzata e strategia centralizzata a confronto [6]

### 1.2.3 Campagne push pianificate vs modello always-on

Un ultimo aspetto importante che differenzia l'approccio 1-to-1 Customer Engagement del marketing relazionale e le campagne di marketing tradizionali riguarda la tempistica di interazione con il cliente. Durante una campagna, i messaggi promozionali sono diffusi in un lasso di tempo stabilito dall'azienda, e non dal cliente. È dunque improbabile che quella finestra dedicata all'iniziativa riesca a catturare l'esigenza di un grande numero di clienti. A causa della pianificazione della campagna e di eventuali ritardi, inoltre, una operazione promozionale può partire con settimane di ritardo rispetto a quando si è manifestata l'esigenza del cliente, non riuscendo a soddisfare la domanda di consumatori sempre più abituati ad una gratificazione immediata, che non è in linea con le tempistiche delle campagne pubblicitarie.

Contrapposto a questo approccio, c'è quello di un coinvolgimento *always-on*, che invece non ha tempistiche prestabilite, ma l'azione si attiva quando è pertinente per il cliente. Questo comportamento, nel marketing tradizionale, funziona solamente nelle campagne di re-targeting, che riguardano il contesto in cui viene visitata una pagina di un prodotto, viene generato un cookie dal sito e poi quel prodotto verrà riproposto, tramite canali a pagamento, quando il cliente naviga in rete. Queste campagne, però, sono spesso utilizzate in modo eccessivo e senza precisione.

In un approccio always-on, invece, sono stabilite anche delle regole di tempistiche, come la frequenza dei contatti, le preferenze dei clienti, o dei limiti relativi ai canali, in modo da migliorare la precisione, non sovraccaricando il consumatore con messaggi promozionali. In questo modo, il cliente sarà contattato quando un'azione è pertinente e redditizia, ad esempio quando la propensione del cliente a compiere una data azione supera una certa soglia, senza dover aspettare l'inizio di una campagna. [6]

## 1.3 Il Customer Decision Hub di Pega

I vantaggi descritti in precedenza nell'usare un marketing relazionale con NBA sono visibili tramite l'utilizzo di apposite piattaforme. Il *Customer Decision Hub* di Pega ingloba tutte le caratteristiche di una metodologia di coinvolgimento del cliente 1 a 1, guidando decisioni in tempo reale attraverso una intelligenza artificiale *omnichannel*, che dunque funziona su tutti i canali esistenti, dal telefono, al web, alle chatbot, ecc. Il design del Next-Best-Action Designer è costruito in maniera intuitiva per chi fa business, non esclusivamente per gli sviluppatori, e sono combinate analisi predittive, utilizzando algoritmi che estraggono grandi set di dati, analisi adattive, con algoritmi di machine-learning che migliorano dopo ogni interazione, e regole di business che aiutano gli utenti a stabilire le priorità e arbitrare sulle decisioni. [7]

### 1.3.1 Il Next-Best-Action Designer

Il Next-Best-Action Designer è una interfaccia intuitiva che permette di offrire automaticamente esperienze personalizzate per il cliente, guidando verso la creazione di una NBA.

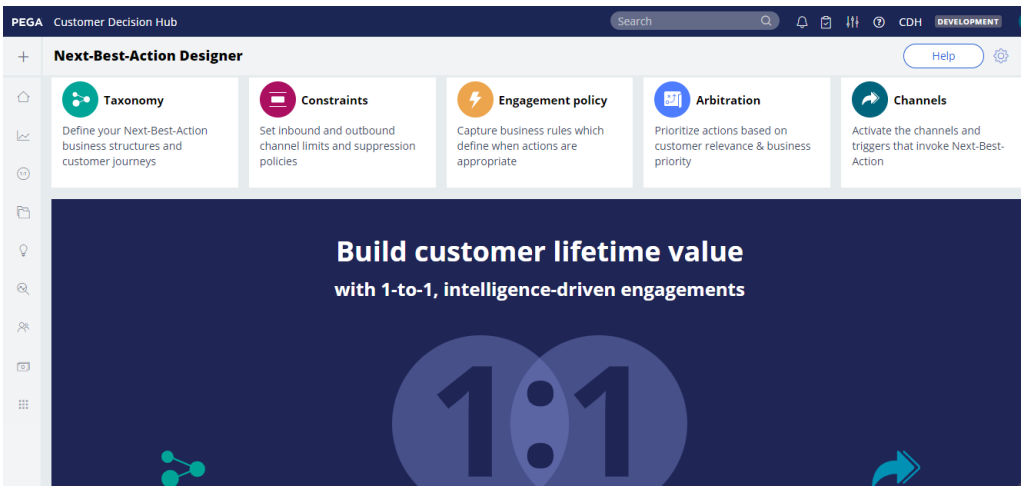


**Figura 5:** Interfaccia del Next-Best-Action Designer [7]

La **Figura 5** mostra l'interfaccia del Next-Best-Action Designer, che presenta cinque diverse schede nella parte superiore da completare per riuscire a definire delle NBA.

La scheda *Taxonomy* serve per definire la struttura di business dell'organizzazione. Come si può notare dalla **Figura 6**, è possibile riconoscere delle *business issues*, che identificano lo scopo delle azioni che vengono offerte ai clienti: *Acquire*, ad esempio, contiene azioni per acquisire nuovi clienti, mentre la business issue *Retain* contiene azioni per mantenere i clienti già acquisiti.

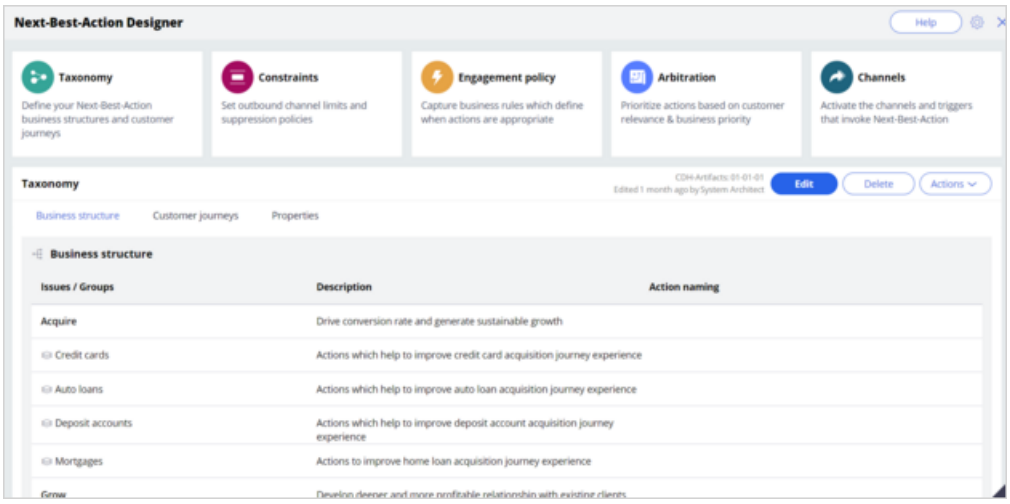


**Figura 6:** Interfaccia della scheda *Taxonomy* [7]

Tramite la scheda *Constraints*, invece, è possibile specificare dei vincoli nei contatti in uscita con il cliente, per evitare la sovraesposizione dei consumatori a un'azione specifica. Troppi contatti in poco tempo, infatti, possono rendere il cliente diffidente nei confronti di ulteriori azioni dell'azienda. È possibile, quindi, definire delle *contact policies* che prevengano questo problema, come, ad esempio, quello di non inviare ad un cliente più di due SMS promozionali a settimana, oppure di non mostrare un annuncio per tre settimane se il cliente lo ignora per quattro giorni nell'arco della settimana, e così via.

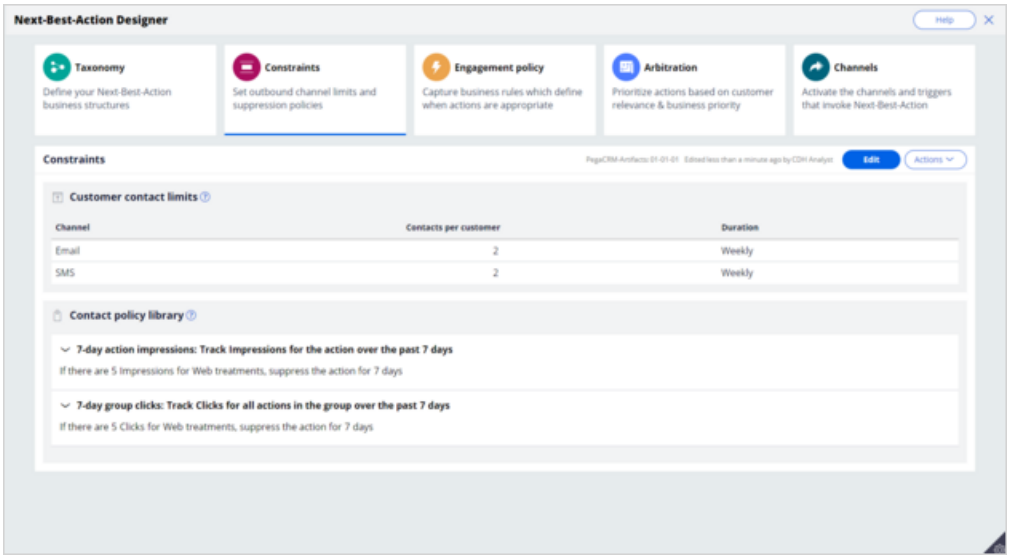


**Figura 7:** Interfaccia della scheda *Constraints* [7]

Nel tab *Engagement policies* è possibile definire quando e per quali clienti delle azioni o gruppi di azioni sono appropriate. Ci sono tre diversi tipi di *engagement policies*, oltre le *contact policies* appena descritte:

- **Eligibility** (ammissibilità): è un criterio che permette di capire se un cliente è idoneo o no per un'azione, e riguardano, ad esempio, criteri sull'età o sulla posizione geografica di un consumatore.
- **Applicability** (rilevanza): indica se un'azione sia rilevante per un cliente in un determinato momento. Ad esempio, una società finanziaria che vuole applicare uno sconto per una carta di credito non è rilevante per chi già possiede una carta.
- **Suitability** (idoneità): determina se una azione sia eticamente appropriata per determinati clienti. Ad esempio, per una società finanziaria sarebbe eticamente inappropriato offrire un prestito ad un cliente con un basso punteggio di credito, nonostante possa essere redditizio per la società.

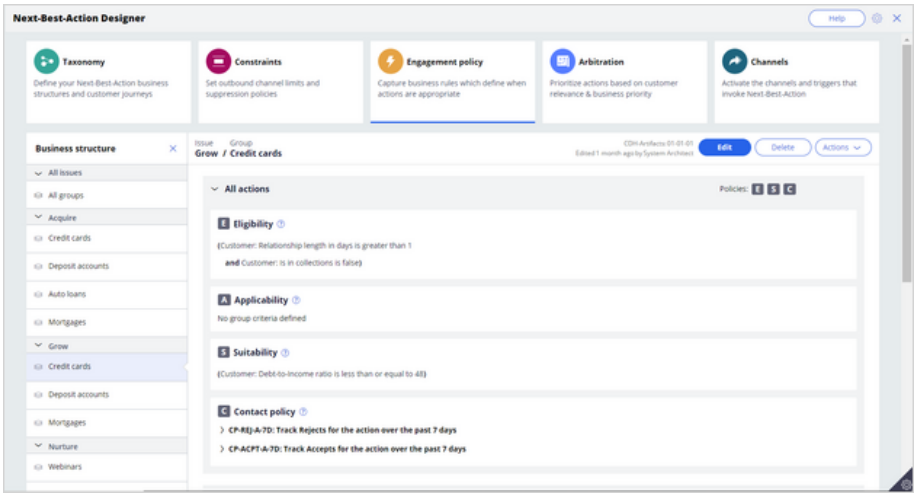


**Figura 8:** Interfaccia della scheda *Engagement policies* [7]

La scheda *Arbitration*, infine, permette di determinare i criteri con cui il CDH dà la priorità all'elenco delle azioni idonee e appropriate di ciascun gruppo. I criteri per calcolare le priorità delle azioni sono quelli descritti in precedenza, ovvero la propensione (*Propensity*), il valore (*Business Value*) e la leva (*Business levers*), con l'aggiunta di un quarto fattore, il peso del contesto (*Context weighting*), che consente di assegnare un peso ad un valore di contesto specifico per tutte le azioni all'interno di un gruppo.

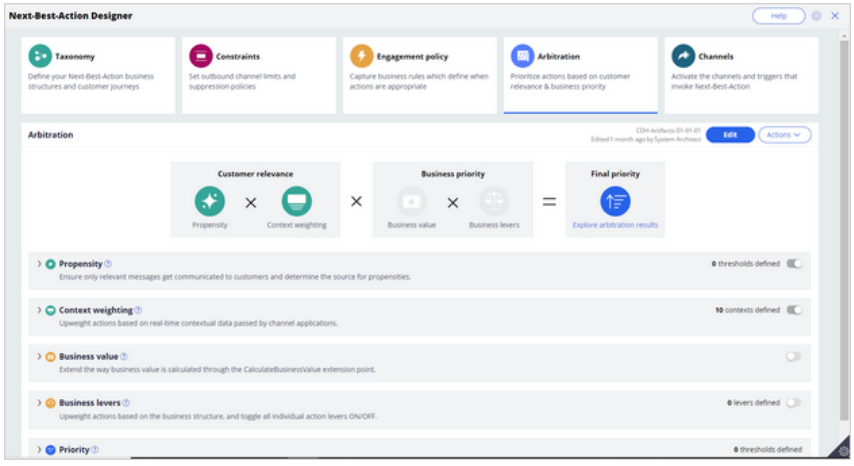


**Figura 9:** Interfaccia della scheda *Arbitration* [7]

L'ultima scheda è quella denominata *Channels*, che permette di attivare o disattivare i canali. In caso di canali disattivati, le NBA non verranno consegnate su tali canali.

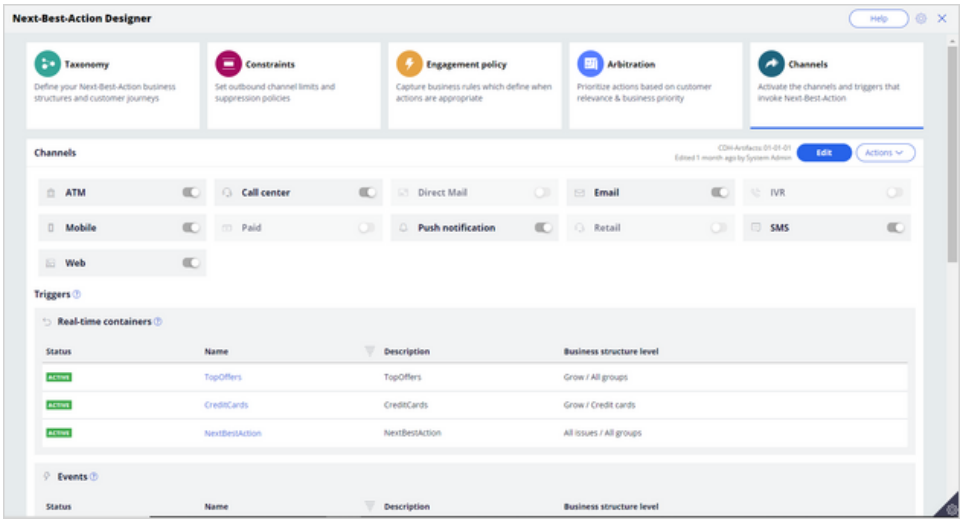


**Figura 10:** Interfaccia della scheda *Channels* [7]

Una volta definiti i parametri di tutte le schede, la situazione sarà come quella nella **Figura 11**: del numero iniziale di azioni disponibili, solamente poche saranno quelle di cui verrà stabilita la priorità attraverso l'algoritmo utilizzato dall'AI.

**Figura 11:** Come cambiano le azioni dopo l'uso delle regole del CDH [7]

## 1.4 L'impatto economico totale dell'uso del CDH

Nel 2020 Pega ha commissionato all'azienda Forrester la conduzione di uno studio sull'impatto economico totale (*Total Economic Impact*, TEI) sull'uso del Customer Decision Hub, e di calcolare il relativo Return On Investment (ROI). L'obiettivo dello studio è stato quello di fornire un *framework* per valutare l'impatto finanziario in termini di benefici, costi e rischi associati all'investimento del CDH su alcune aziende con fino a cinque anni di esperienza con l'uso della piattaforma e sono stati calcolati benefici e costi sulla base dei tre anni precedenti.

Forrester ha intervistato quattro diverse aziende appartenenti a diversi settori (due di servizi finanziari, una di assicurazioni e una di telecomunicazioni), che cercavano soluzioni per sopperire a dei problemi che sono incorsi nel servire i clienti. Le organizzazioni hanno registrato un declino nei tassi di accettazione dei programmi di marketing per colpa di messaggi non rilevanti, iniziative di marketing isolate che hanno portato a messaggi incoerenti ed esperienza del cliente scadente, e dati dei clienti sottoutilizzati che ha portato a mancate opportunità di affari. Attraverso l'utilizzo del CDH di Pega, le aziende sono riuscite a consegnare messaggi rilevanti e consistenti ai consumatori, a rendere le interazioni degli agenti con i clienti attraverso il miglior supporto possibile in ogni momento, e a creare una vista unificata del cliente per consegnare le decisioni.

I grandi vantaggi economici derivati dall'adozione di questa tecnologia sono stati calcolati da Forrester e sono riassunti nel **Grafico 1** e nella **Tabella 3**. Il *Present Value* (PV) del totale dei profitti è pari a 180 milioni di dollari, che, a fronte di un PV dei costi totali pari a 30,6 milioni di dollari, permette di ottenere un *Net Present Value* (NPV) pari a 149,8 milioni di dollari. Anche il valore del *Payback* è molto positivo, essendo inferiore ai 6 mesi, così come è largamente positivo il ROI, che ammonta al 489%.

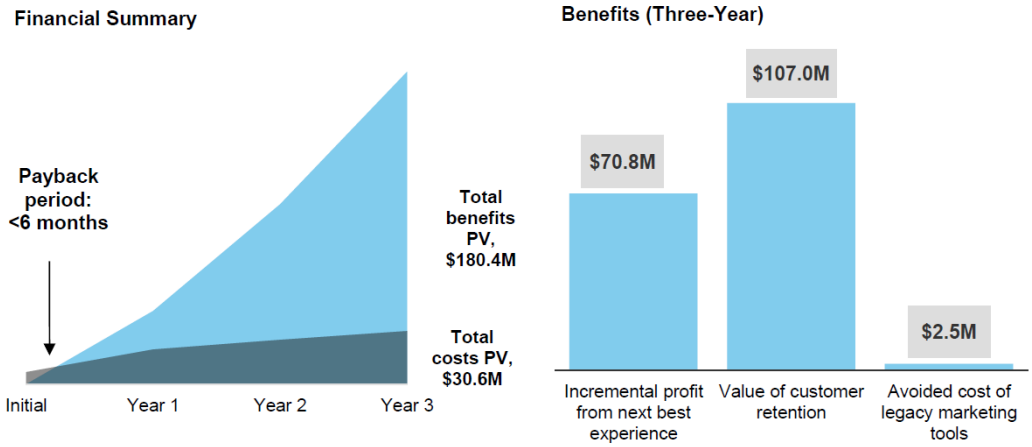


**Grafico 1:** Payback e PV dei profitti – TEI CDH [11]

**Cash Flow Table (Risk-Adjusted)**

| | INITIAL | YEAR 1 | YEAR 2 | YEAR 3 | TOTAL | PRESENT VALUE |
|---|---|---|---|---|---|---|
| Total costs | ($5,913,380) | ($13,563,960) | ($7,665,160) | ($8,045,160) | ($35,187,660) | ($30,623,543) |
| Total benefits | $0 | $46,440,000 | $74,493,000 | $101,926,238 | $222,859,238 | $180,361,336 |
| Net benefits | ($5,913,380) | $32,876,040 | $66,827,840 | $93,881,078 | $187,671,578 | $149,737,793 |
| ROI | | | | | | 489% |
| Payback period | | | | | | <6 months |

**Tabella 3:** Analisi del Cash Flow – TEI CDH [11]

I profitti derivati dall'adozione del CDH sono riconducibili innanzitutto alla migliore esperienza offerta al cliente, proponendogli la NBX secondo l'approccio 1 a 1. Una migliore visione delle intenzioni dei clienti, inoltre, ha permesso di ridurre il tasso di abbandono, aiutando le aziende ad evitare perdite per oltre 578 milioni di dollari in tre anni. Prima del Pega CDH, infatti, le organizzazioni non potevano conoscere le intenzioni dei clienti fino al momento della risoluzione, e i dipendenti del servizio clienti non avevano modo di sapere quale offerta avrebbe potuto permettere al cliente di non applicare la risoluzione. Pega fornisce l'AI per identificare preventivamente i clienti a rischio di abbandono e fornisce la NBX che potrebbe aiutare a mantenere i clienti, risolvendo le loro difficoltà o proponendo delle offerte più interessanti. C'è, infine, il beneficio legato ad aver evitato costi di licenze per vari tool di marketing incentrati su canali, grazie all'utilizzo di un'unica piattaforma per tutti i canali.

I costi da sostenere, dall'altra parte, sono legati ai costi della licenza Pega, ai costi di implementazione, che includono costi legati al training e al bisogno di

figure professionali che potessero istruire i dipendenti all'uso della piattaforma, e costi di gestione continua, che riguardano i test, la gestione dei progetti, la configurazione e i miglioramenti dei tool.

Il grafico sottostante, oltre a segnalare profitti e costi, mostra l'andamento del profitto netto cumulato, che sembra essere ancora in crescita.

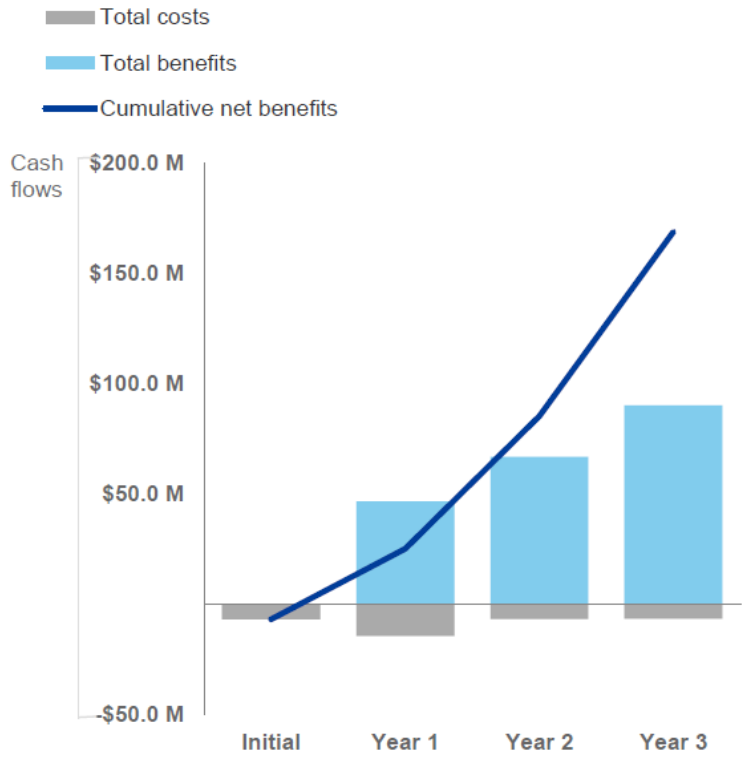


**Grafico 2:** Andamento del Cash Flow – TEI CDH [11]

Oltre a benefici quantificati economicamente, ci sono stati anche dei benefici non quantificabili, come una migliore esperienza del cliente, un servizio clienti in grado di fornire un supporto rilevante e pertinente e la possibilità per la divisione di marketing di sviluppare un'unica strategia, con un allineamento dei KPI. Infine, è stata valutata la flessibilità della piattaforma, intesa come il beneficio che può derivare da ulteriore investimento delle aziende. Ulteriori opportunità di business possono riguardare l'uso dell'intelligenza di Pega per anticipare i bisogni dei clienti, oppure opportunità di migrare al cloud in modo da consentire aggiornamenti più rapidi, oppure l'integrazione di canali aggiuntivi, magari non ancora presenti. [11]

## 1.5 Un caso di studio: la Commonwealth Bank of Australia

La Commonwealth Bank of Australia (CBA) è un istituto di credito australiano ed è considerata una delle quattro maggiori banche australiane. La loro strategia, nonché priorità, è quella di fornire prodotti e servizi reinventati e le migliori esperienze digitali e tecnologiche. [12] In particolare, il Chief Analytics Officer di CBA, Andrew McMullan, ha inquadrato il problema del membro del personale dei call center che *cercava di servire il cliente e pensava a ciò che il cliente gli stava chiedendo di fare, ma passava anche da una schermata all'altra per vedere se c'era qualcosa che poteva vedere che avrebbe permesso loro di avere una conversazione utile con il cliente di fronte a loro*. [13]

Per migliorare la customer experience in questo contesto, la CBA si è affidata a Pegasystems per implementare un sistema che permetta di intrattenere conversazioni più proattive e basate sui bisogni con i clienti, utilizzando l'AI per generare le Next Best Conversations. [14] Il *Costumer Engagement Engine* (CEE) fornito prende oltre 35 milioni di decisioni ogni giorno [12], su diciotto canali differenti, generando una Next Best Conversation in meno di 160 millisecondi. Si stima che il 95% delle potenziali offerte inviate sono state prese in considerazione, mentre la soddisfazione dei clienti è significativamente aumentata. [14]

La CBA ora sta cercando di sfruttare il CEE per diversi scopi. Ad esempio, in seguito alle emergenze nazionali australiane del Covid-19 e degli incendi boschivi, la CBA è riuscita a sfruttare la CEE per creare un modello meteorologico di dati provenienti da diverse fonti ufficiali in grado di fornire ai clienti colpiti da disastri naturali soluzioni di supporto personalizzate, in linea con le esigenze del cliente, come il rinvio di un prestito o di un mutuo, o la riduzione di tassi e commissioni. Inoltre, nel 2021 la CBA è stata una delle prime banche a sviluppare un modello AI in grado di identificare in modo proattivo i casi di abusi nei propri sistemi di home banking. [12]

## 1.6 Confronto tra le piattaforme di *Real-Time Interaction Management*

Un ulteriore report di Forrester mette a confronto i principali fornitori di Real-Time Interaction Management. La valutazione della società si basa su 28 criteri e ha individuato le 14 aziende fornitrici più significative. Dall'analisi è risultato che a maggio 2022 tra le priorità delle aziende B2C per i successivi 12 mesi erano migliorare il ROI o l'efficacia del marketing in generale (per il 25% dei casi), migliorare l'allineamento del marketing e la collaborazione con altri dipartimenti (come Customer Experience, servizio clienti ed e-commerce) (24%) e concentrarsi sul coinvolgimento del cliente post-vendita (23%).

L'analisi e il confronto tra le varie piattaforme hanno permesso di individuare quattro categorie di fornitori: i *Leaders*, i forti interpreti (*Strong Performers*), i contendenti (*Contenders*) e gli sfidanti (*Challengers*).

**Figura 12:** Confronto tra i fornitori di Real-Time Interaction Management [15]

| | Forrester's weighting | ActionIQ | Adobe | CSG | HCL Software | Microsoft | NGDATA | Pegasystems |
|---|---|---|---|---|---|---|---|---|
| **Current offering** | 50% | 1.77 | 2.87 | 2.79 | 2.41 | 2.26 | 1.80 | 4.82 |
| Customer recognition | 10% | 4.20 | 3.80 | 2.20 | 3.00 | 1.80 | 1.00 | 5.00 |
| Contextual understanding | 10% | 1.00 | 3.80 | 3.40 | 1.00 | 3.00 | 1.80 | 3.80 |
| Decision arbitration | 30% | 1.70 | 2.70 | 3.50 | 1.50 | 2.00 | 1.80 | 5.00 |
| Experience orchestration | 20% | 1.60 | 2.60 | 2.00 | 3.30 | 2.30 | 1.30 | 5.00 |
| Measurement and optimization | 15% | 1.00 | 3.00 | 3.00 | 2.60 | 3.00 | 2.20 | 5.00 |
| Integration and usability | 15% | 1.80 | 2.20 | 2.20 | 3.40 | 1.80 | 2.60 | 4.60 |
| **Strategy** | 50% | 1.90 | 2.60 | 1.70 | 2.00 | 3.60 | 1.00 | 4.70 |
| Product vision | 20% | 1.00 | 3.00 | 3.00 | 3.00 | 3.00 | 1.00 | 5.00 |
| Innovation roadmap | 20% | 1.00 | 1.00 | 1.00 | 1.00 | 3.00 | 1.00 | 5.00 |
| Market approach | 15% | 1.00 | 3.00 | 3.00 | 3.00 | 3.00 | 1.00 | 5.00 |
| Performance | 15% | 5.00 | 1.00 | 1.00 | 1.00 | 3.00 | 1.00 | 3.00 |
| Supporting products and services | 15% | 3.00 | 5.00 | 1.00 | 1.00 | 5.00 | 1.00 | 5.00 |
| Partner ecosystem | 15% | 1.00 | 3.00 | 1.00 | 3.00 | 5.00 | 1.00 | 5.00 |
| **Market presence** | 50% | 1.00 | 3.50 | 2.00 | 3.00 | 3.50 | 1.00 | 5.00 |
| Revenue | 50% | 1.00 | 4.00 | 2.00 | 3.00 | 4.00 | 1.00 | 5.00 |
| Customers | 50% | 1.00 | 3.00 | 2.00 | 3.00 | 3.00 | 1.00 | 5.00 |

| | Forrester's weighting | Precisely* | Qualtrics | Redpoint Global | Salesforce | SAS | Thunderhead | Treasur |
|---|---|---|---|---|---|---|---|---|
| **Current offering** | 50% | 3.51 | 2.87 | 1.65 | 3.34 | 4.62 | 3.93 | 1.86 |
| Customer recognition | 10% | 2.20 | 2.20 | 3.00 | 3.80 | 4.20 | 3.00 | 1.00 |
| Contextual understanding | 10% | 3.80 | 4.20 | 1.00 | 3.80 | 3.80 | 4.20 | 1.80 |
| Decision arbitration | 30% | 3.50 | 3.50 | 1.50 | 4.20 | 5.00 | 3.50 | 1.80 |
| Experience orchestration | 20% | 3.60 | 1.70 | 1.30 | 1.80 | 4.10 | 4.50 | 2.50 |
| Measurement and optimization | 15% | 3.00 | 3.00 | 1.80 | 3.80 | 5.00 | 4.20 | 1.00 |
| Integration and usability | 15% | 4.60 | 2.60 | 1.80 | 2.60 | 5.00 | 4.20 | 2.60 |
| **Strategy** | 50% | 2.00 | 4.00 | 2.30 | 3.30 | 4.00 | 2.80 | 1.30 |
| Product vision | 20% | 3.00 | 5.00 | 1.00 | 3.00 | 3.00 | 5.00 | 1.00 |
| Innovation roadmap | 20% | 1.00 | 3.00 | 3.00 | 3.00 | 5.00 | 3.00 | 1.00 |
| Market approach | 15% | 1.00 | 3.00 | 3.00 | 3.00 | 5.00 | 3.00 | 1.00 |
| Performance | 15% | 1.00 | 5.00 | 1.00 | 5.00 | 3.00 | 1.00 | 3.00 |
| Supporting products and services | 15% | 3.00 | 3.00 | 3.00 | 3.00 | 5.00 | 1.00 | 1.00 |
| Partner ecosystem | 15% | 3.00 | 5.00 | 3.00 | 3.00 | 3.00 | 3.00 | 1.00 |
| **Market presence** | 50% | 3.50 | 5.00 | 1.50 | 3.50 | 3.50 | 1.50 | 2.50 |
| Revenue | 50% | 3.00 | 5.00 | 1.00 | 4.00 | 4.00 | 1.00 | 3.00 |
| Customers | 50% | 4.00 | 5.00 | 2.00 | 3.00 | 3.00 | 2.00 | 2.00 |

**Figura 13:** Assegnazione punteggio ai fornitori di RTIM [15]

La **Figura 12** e la **Figura 13** mostrano rispettivamente la categoria di appartenenza delle aziende considerate e i punteggi assegnati per ogni criterio. Si può notare come, sia a livello di offerta corrente, sia a livello strategico, sia a livello di presenza sul mercato, Pegasystems sia l'azienda con il punteggio più alto e la vera leader del mercato. In seguito, verranno presentate le caratteristiche di alcune delle aziende prese in esame da Forrester:

1. **Pegasystems (Leader):** è il fornitore che definisce lo standard di riferimento per implementazioni aziendali sofisticate, grazie all'allineamento del RTIM con l'automazione dei processi digitali ed un ampio portafoglio di servizi, che consente di ottimizzare la CX durante tutto il ciclo di vita del cliente. Come già analizzato, Pega permette alle aziende di puntare ad un valore di 150 milioni di dollari di NPV per ogni 10 milioni di clienti. Tra i punti a favore del CDH di Pega è la combinazione tra la modellazione analitica all'avanguardia con la logica di business, per fornire un motore decisionale NBX molto robusto, riuscendo ad allineare la rilevanza del cliente con la strategia di business.
2. **SAS (Leader):** è l'unico fornitore leader insieme a Pegasystems e riesce a coniugare la sua potenza analitica con il *decisioning* in tempo reale. Riesce a differenziarsi perché permette ai clienti di mettere a punto modelli analitici personalizzati per garantire risultati di successo allineati con le

operazioni di business. Le capacità analitiche avanzate permettono di prendere decisioni NBX anche per casi d'uso operativi come gestione dei rischi e frodi. Inoltre, i clienti di SAS lodano il personale e la tecnologia di SAS, e come unico difetto viene riscontrata una esperienza utente non di facile utilizzo per le aziende.

3. **Qualtrics (Strong Performer):** la soluzione di Qualtrics colpisce con la sua visione per come la gestione dell'esperienza *empathy-driven* possa evolversi per generare un vantaggio competitivo sostenibile. La roadmap a breve termine si concentra sull'integrazione e sul potenziamento della sua soluzione combinata. Deve, però, riuscire a fornire tecnologie e servizi in grado di allineare le metriche di CX con gli obiettivi di business, come ricavi, profitti e CLV.
4. **Salesforce (Strong Performer):** secondo Forrester, Salesforce può puntare al successo se riuscirà a portare l'innovazione dell'azienda nel portafoglio di strategie orientate all'integrazione, con più piani per produrre i visionari principi dell'azienda di etica e fiducia. Presenta una solida ottimizzazione dell'esperienza e permette alle aziende di misurare l'impatto delle proprie attività di marketing, ma deve perfezionare la flessibilità della gestione dei dati aziendali e le integrazioni con i suoi cloud di marketing e servizio per avere una esperienza più fluida tramite dispositivi mobili e contact center.
5. **Microsoft (Strong Performer):** grazie a Microsoft Dynamics, anche la nota azienda ha iniziato ad avere un suo ruolo nel RTIM. Essendo ancora una piattaforma nuova, ha ampi margini di crescita, e può sfruttare le funzionalità del suo ampio portafoglio CX e i modelli di dati specifici per il settore, ma i concorrenti offrono strumenti di analisi e dati più sofisticati.
6. **Adobe (Strong Performer):** secondo Forrester Adobe offre una soluzione promettente, ma deve accelerare l'adozione. Per sostenere la transizione verso la sua Adobe Experience Platform può fare affidamento al suo grande peso in termini di esperienza digitale, ma i suoi strumenti analitici e decisionali sono in ritardo rispetto a quelli dei leader affermati, così come migliorabile la UX. [15]

# 2. Piattaforme Low-Code e Pega Platform

Lo sviluppo di software in low-code è una concreta alternativa allo sviluppo tradizionale, in quanto offre un modo semplice e accessibile di creare applicazioni senza richiedere una estesa scrittura di codice. Ciò rende accessibile la creazione di software non solo agli sviluppatori professionisti, ma anche ad utenti non tecnici, chiamati *Citizen Developers*. La diffusione del low-code deriva dalla potenziale riduzione di costi e tempi richiesti per lo sviluppo di software, risultando un'ottima soluzione per le aziende che si stanno approcciando alla transizione digitale. [16] Lo sviluppo di piattaforme in low-code, come anticipato nell'introduzione, è possibile tramite delle tecniche visive e dichiarative e, man mano che si espande il modello visivo, il codice viene creato automaticamente grazie agli strumenti delle LCDP. Tra le varie funzionalità dell'interfaccia utente ci sono il trascinamento della selezione, i flussi di processo e strumenti visivi. Questo permette di colmare il divario tra gli stakeholder aziendali e gli sviluppatori professionisti, riuscendo ad ottenere una migliore collaborazione, in quanto si riesce a concentrare sulla logica aziendale piuttosto che sul codice. [17]

Oltre alle piattaforme di sviluppo low-code, è possibile uno sviluppo di applicazioni in no-code. Le applicazioni in no-code consentono a chiunque nell'organizzazione di creare programmi software senza alcuna conoscenza di codifica, ed è un'ottima soluzione per le aziende che hanno bisogno di utilizzare talenti non tecnici per progetti di applicazioni web e mobili. Se questa soluzione può essere un grosso vantaggio per le aziende, talvolta non è possibile attuarla, in quanto i modelli rigidi non permettono una grossa personalizzazione e flessibilità delle app. [16]

Pegasystems offre una LCDP tramite Pega Platform, che permette la gestione e lo sviluppo delle applicazioni tramite quattro workspace (*App Studio*, *Dev Studio*, *Admin Studio* e *Prediction Studio*), che consentono ai membri del team di concentrarsi sulle attività che sono in linea con le loro competenze. [18]

## 2.1 Il confronto con lo sviluppo tradizionale e i vantaggi e svantaggi del low-code

Secondo uno studio condotto da Gartner, entro il 2024 il 65% delle applicazioni saranno sviluppate con sistemi low-code, e si prospetta un volume d'affari generato di 187 miliardi di dollari dal mercato low-code/no-code entro il 2030. [19] Per comprendere meglio il perché del successo di questa metodologia di sviluppo possono essere evidenziati i vantaggi rispetto allo sviluppo tradizionale tramite codice, che sono riassunti nella **Tabella 4**.

| Caratteristiche | Sviluppo tradizionale | Sviluppo low-code |
|---|---|---|
| **Time to market** | Codice manuale<br>Rilascio lento<br>Lento a cambiare | Progettazione drag and drop<br>Componenti precostruiti<br>Rilascio veloce |
| **Ambito** | Creare intere applicazioni contemporaneamente<br>Grandi progetti | Costruire piccole soluzioni indipendenti una alla volta<br>Framework per multiple piccole soluzioni |
| **Sviluppo** | Tempi di consegna lenti<br>Non allineato alle esigenze aziendali | Sviluppo fino a cinque volte più veloce<br>Test efficienti<br>Incrementa il ROI |
| **Manutenzione** | Costoso da supportare<br>Richiede uno sviluppo aggiuntivo | Facile/rapido da aggiornare o estendere<br>Eccellente per la prototipazione |
| **Integrazione** | Tempi e investimenti onerosi<br>Richiede sviluppatori e documentazione<br>Soggetto a ritardi nei test | Connettori precostruiti<br>Debug in tempo reale<br>Crea servizi web/API senza codice |
| **Distribuzione** | Lenta e complessa<br>Multipli step che richiedono risorse di sviluppo | Implementazione con un solo tocco<br>Distribuzione su più ambienti<br>Cloud o in sede |

**Tabella 4:** Vantaggi dello sviluppo low-code rispetto allo sviluppo tradizionale [20]

Come già accennato, il grande vantaggio dello sviluppo low-code riguarda il time to market: grazie agli elementi precostruiti permette un notevole risparmio di tempo rispetto allo sviluppo tradizionale e al suo codice manuale. Una ricerca

condotta da RedHat nel 2018 ha stimato che il tempo di sviluppo di applicazioni in low-code possa essere ridotto potenzialmente del 50-90% rispetto ad uno sviluppo tramite codice.[21] La possibilità di costruire piccole soluzioni indipendenti, invece, permette di poter lavorare con una metodologia Agile più semplicemente e aiuta nello sviluppo più veloce delle applicazioni. Come già scritto in precedenza, inoltre, l'intuibilità dell'interfaccia di sviluppo permette un maggiore allineamento con le esigenze aziendali e una migliore collaborazione tra l'area tecnica e non tecnica aziendale. È anche possibile una migliore manutenzione grazie alle interfacce grafiche intuitive, rispetto allo sviluppo tradizionale, che, invece, richiede che si debba decodificare il codice scritto, non sempre semplice da attuare, così come è più semplice e meno costosa l'integrazione, poiché permette la creazione di *Application Programming Interface* (API) e servizi web senza codice.

Le LCDP, però, presentano anche degli svantaggi, che, tuttavia, sono in via di miglioramento negli ultimi tempi. Uno di questi riguarda la maggiore inflessibilità rispetto alle applicazioni sviluppate tramite codifica tradizionale, poiché gli elementi costitutivi sono pre-implementati: lo sviluppo di funzionalità più complicate potrebbe essere più difficoltoso nelle piattaforme low-code. Ciò, però, dipende anche dalla piattaforma utilizzata e dal livello di personalizzazione offerto. Un altro problema comune è la limitata scalabilità, che, però, non si riscontra in tutte le piattaforme: con Pega, infatti, è possibile sviluppare applicazioni molto robuste su molti canali. È in via di miglioramento anche un altro problema importante, quello legato alla sicurezza. Siccome le organizzazioni dipendono dai loro fornitori di piattaforme low-code, i loro dati potrebbero essere soggetti a violazioni dei dati poiché la sicurezza dei dati e il codice sorgente non sono completamente controllati dalle organizzazioni. [22] Per questo motivo alcuni fornitori di piattaforme low-code si stanno adattando agli standard di sicurezza, come ISO/IEC 27001:2013. [23] Un ultimo limite, infine, riguarda il *vendor lock-in*, ovvero la dipendenza di un cliente da un fornitore per i suoi prodotti e servizi. Passare ad un altro fornitore, infatti, potrebbe essere molto costoso, poiché la portabilità di prodotti e servizi non è possibile da una piattaforma ad un'altra. [22]

## 2.2 Pega Platform

L'ambiente low-code per la gestione dei casi e lo sviluppo di applicazioni, fornito da Pegasystems, è tra le opzioni disponibili di Pega Platform. Questo strumento è stato progettato per permettere alle aziende di creare, distribuire e migliorare le proprie applicazioni strategiche, ovvero quelle applicazioni che generano una differenziazione e una innovazione per l'impresa.

L'architettura della piattaforma si compone di diversi strati che lavorano insieme per fornire una soluzione completa e scalabile per la gestione dei processi aziendali, utilizzando tecnologie moderne e flessibili per garantire una personalizzazione completa in base alle esigenze del cliente.

In seguito, verranno presentate le principali funzionalità e aree di prodotto di Pega Platform:

- **Sviluppo di applicazioni a basso codice:** per la realizzazione di applicazioni in low-code Pega Platform utilizza un approccio completamente basato su modelli, o a basso codice, per lo sviluppo di applicazioni, insieme a un'architettura di runtime dinamica. Sono disponibili quattro diverse aree di lavoro per lo sviluppo e la gestione delle applicazioni, permettendo ai membri del team di concentrarsi sulle attività che rispettano la loro competenza:
    - *App Studio*: è la principale area di lavoro a basso codice, dove sia gli sviluppatori di business che quelli tecnici possono definire i casi di applicazione, l'esperienza utente e i modelli di dati senza scrivere codice e comunicare tra loro.
    - *Dev Studio*: questa area di lavoro consente funzioni di sviluppo più avanzate, come la configurazione della mappatura di integrazione o dei controlli di accesso. Qui è possibile creare set di regole e altri componenti che saranno disponibili per l'uso in App Studio.
    - *Admin Studio*: fornisce agli amministratori gli strumenti per gestire le applicazioni, i dati, gli utenti, ecc.

- *Prediction Studio*: è un'area destinata a data scientists e analisti per definire e affinare i modelli di previsione e di apprendimento automatico.

- **Gestione delle decisioni:** Pega Platform offre una serie di funzioni di gestione delle decisioni (*decisioning*), tra cui strategie di decisione, modelli adattivi e predittivi, ed elaborazione del linguaggio naturale, che possono aiutare a prendere le decisioni sulle azioni più rilevanti da presentare ai clienti durante la gestione delle comunicazioni in entrata e in uscita o fornire suggerimenti in tempo reale ai rappresentanti del servizio clienti. Il livello di decision-making supporta anche set di dati e flussi di dati per la gestione e l'elaborazione dei dati analitici essenziali per la decisione. Le funzioni di decisione sono alla base di Pega Customer Decision Hub, di cui è stato discusso nel precedente capitolo.
- **Gestione dei casi:** Un caso è un'unità di lavoro finalizzata a raggiungere un obiettivo aziendale e cattura il ciclo di vita del lavoro dall'inizio alla risoluzione. La gestione dei casi (*case management*) garantisce un'esperienza unificata, capacità, analisi e contenuti cross-platform in tutti i canali di interazione. Un utente può iniziare un'interazione in un canale e passare a un altro a metà processo. Ad esempio, un cliente può iniziare la selezione di un prodotto su un dispositivo mobile e quindi completarlo sul desktop o tramite una chiamata telefonica al call center. La piattaforma Pega mantiene il contesto delle interazioni su tutti questi canali e dispositivi per il caso sia per il cliente che per il rappresentante del servizio clienti.
- **Integrazione dei dati:** Le applicazioni aziendali devono comunicare con altri sistemi e servizi all'interno e all'esterno dell'organizzazione. L'integrazione di dati offerta da Pega consente alle applicazioni di integrarsi perfettamente in un ecosistema aziendale e può essere utilizzata per integrarsi direttamente con i sistemi esistenti o possono utilizzare l'infrastruttura di bus di servizi aziendali (ESB) o di architettura orientata ai servizi (SOA). [18]

## 2.3 L'impatto economico totale del Pega Platform per Low-Code

Similmente a quanto già visto per il Customer Decision Hub, l'azienda Forrester ha condotto uno studio sull'impatto economico totale di Pega Platform per low-code, anche in questo caso per valutarne il relativo ROI. Le modalità di analisi di questo studio sono le stesse viste per il CDH in precedenza, con alcune aziende intervistate, in questo caso cinque, analizzando i risultati ottenuti nei tre anni precedenti.

Prima di investire nel low-code di Pega, alcune organizzazioni non impiegavano personale con le necessarie competenze per lo sviluppo a basso codice, per questo hanno dovuto esternalizzare il lavoro di sviluppo a costi più elevati. Con l'adozione di Pega, invece, le organizzazioni intervistate hanno riallocato il personale dell'*Information Technology* (IT) e si sono allontanate dagli sviluppatori di terze parti, riducendo i costi di uno sviluppo personalizzato, accelerando i processi di sviluppo e semplificando le architetture tecniche.

Anche questa analisi evidenzia numeri molto incoraggianti in termini economici, come emerge anche dal **Grafico 3** e dalla **Tabella 5**. Il *Present Value* del totale dei profitti è pari a 13,9 milioni di dollari, che, a fronte di un PV dei costi totali pari a 2 milioni di dollari, consente di ottenere un *Net Present Value* pari a 11,9 milioni di dollari. I valori del *Payback*, equivalente ad un periodo minore di 3 mesi, e del ROI, pari al 598%, sono molto positivi ed in linea con quanto atteso.

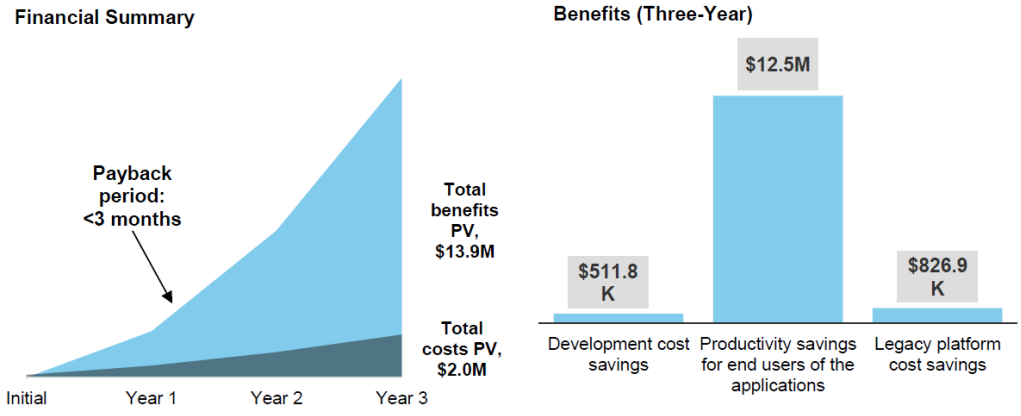


**Grafico 3:** Payback e PV dei profitti - TEI Pega Platform per low-code [24]

**Cash Flow Analysis (risk-adjusted estimates)**

| | INITIAL | YEAR 1 | YEAR 2 | YEAR 3 | TOTAL | PRESENT VALUE |
|---|---|---|---|---|---|---|
| Total costs | ($117,490) | ($473,400) | ($754,521) | ($1,084,113) | ($2,429,524) | ($1,985,935) |
| Total benefits | $0 | $2,349,050 | $5,654,167 | $9,382,450 | $17,385,667 | $13,857,539 |
| Net benefits | ($117,490) | $1,875,650 | $4,899,646 | $8,298,337 | $14,956,143 | $11,871,604 |
| ROI | | | | | | 598% |
| Payback period (months) | | | | | | <3 |

**Tabella 5:** Analisi del Cash Flow - TEI Pega Platform per low-code [24]

I profitti derivanti dall'adozione del low-code di Pega Platform sono dati in primo luogo dai risparmi dei costi di sviluppo dalla redistribuzione delle risorse da terze parti agli IT interni e ai Citizen Developers, riducendo la spesa di servizi di consulenza del 60% in tre anni. Inoltre, l'automatizzazione dei processi ha permesso un aumento della produttività stimata pari a 12,5 milioni di dollari, con un totale di 20 applicazioni sviluppate a supporto di 3.350 utenti. Come per il CDH, infine, c'è il risparmio derivante dai risparmi dei costi di licenze. I costi, invece, sono riconducibili alla licenza annuale di Pega, all'implementazione e gestione della piattaforma e alla formazione delle risorse di sviluppo per il personale IT e aziendale.

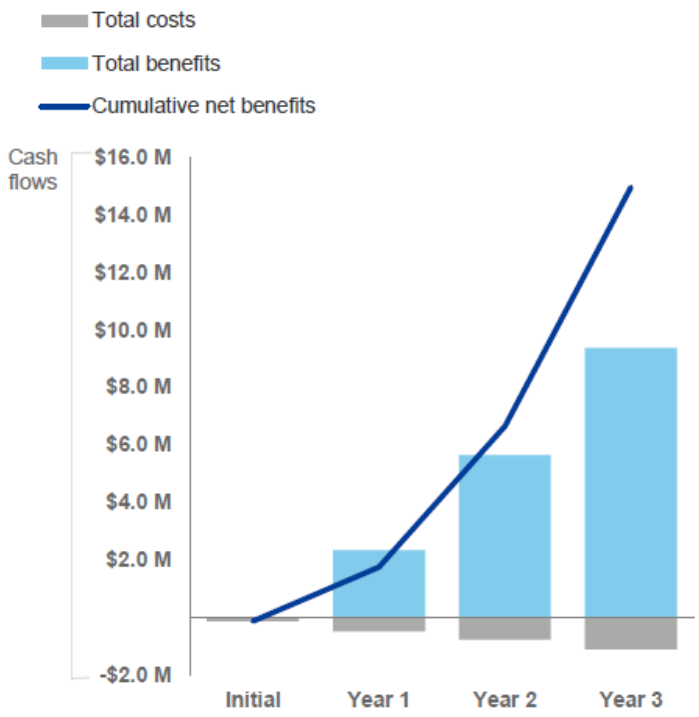


**Grafico 4:** Andamento del Cash Flow - TEI Pega Platform per low-code [24]

Anche in questa analisi sono stati valutati benefici non quantificabili portati da Pega Platform. Innanzitutto, è stata riscontrata una maggiore produttività dell'utente finale quando le applicazioni sviluppate includono funzionalità aggiuntive di Pega, come l'automazione robotica dei processi (*Robotic Process Automation* – RPA), che ha fornito la possibilità di sviluppare bot per supportare le funzioni di vendita. Inoltre, è migliorata l'esperienza utente con una più semplice manutenzione ed un supporto di applicazioni personalizzate, oltre una migliore collaborazione tra l'IT e l'azienda, ed una migliore qualità del prodotto grazie ad un approccio allo sviluppo più funzionale. Infine, è da registrare una maggiore soddisfazione dei dipendenti, sia per le risorse di sviluppo IT che per gli utenti aziendali. [24]

## 2.4 Confronto tra le piattaforme di sviluppo Low-Code

Uno studio condotto da Gartner ha messo a confronto diverse piattaforme di sviluppo low-code tramite il *Magic Quadrant*. Per effettuare l'analisi sono considerati più criteri, raggruppati sotto le due dimensioni di **capacità di esecuzione**, che comprende criteri di esecuzione/prezzo del prodotto e delle vendite, e di **completezza della visione**, che presenta criteri che enfatizzano la strategia offerta/prodotto e i criteri di innovazione. Gartner aggiorna costantemente questo studio, e le organizzazioni presenti nel Magic Quadrant possono anch'esse variare. I criteri per l'inclusione nei Magic Quadrant sono revisionati e adeguati in base ai cambiamenti del mercato e i fornitori possono essere aggiunti o rimossi. Per essere ammessi, i fornitori LCDP devono soddisfare specifici criteri di mercato, tecnici e di business. I criteri di mercato e tecnici includono una strategia di commercializzazione, il supporto per gli sviluppatori professionisti e specifiche capacità tecniche. I criteri di business includono la dimensione, la crescita e la presenza internazionale. I fornitori possono essere esclusi per motivi specifici, come l'esigenza di componenti di terze parti con licenza o la mancanza di offerte commerciali di supporto aziendale.

**Figura 14:** Confronto tra i fornitori di LCDP [25]

Come nell'analisi di Forrester per le piattaforme di RTIM, anche in questo caso sono state individuate quattro categorie di fornitori: i *Leaders*, i visionari (*Visionaries*), gli sfidanti (*Challengers*) e gli attori di nicchia (*Niche Players*). La piattaforma low-code di Pegasystems si colloca tra i visionari; dunque, fornisce prodotti innovativi e riescono ad anticipare le esigenze emergenti, presentando un forte potenziale per influenzare la direzione del mercato, ma i suoi prodotti e la presenza sul mercato non sono ancora completi o consolidati da sfidare i leader. In seguito, verranno presentate le caratteristiche della piattaforma di Pegasystems, confrontata con alcuni fornitori leader del settore secondo Gartner:

1. **Pegasystems (Visionary):** Pegasystems offre tecnologie di automazione a basso codice attraverso la sua piattaforma Pega Infinity, che include l'ottimizzazione dei processi, il CRM, l'RPA e le API di esperienza digitale. I clienti di Pega sono principalmente grandi aziende nei settori bancario, finanziario, assicurativo, delle telecomunicazioni e della sanità. I suoi punti di forza risiedono nello sviluppo a basso codice per l'automazione dei processi e dei flussi di lavoro, nell'esperienza utente e

nell'architettura della sua piattaforma cloud-native. Tuttavia, il suo modello di pricing e la complessità della piattaforma possono rappresentare un ostacolo per alcuni clienti e le sue strategie di marketing e di vendita non sono altrettanto forti come quelle dei suoi concorrenti in questo settore.

2. **Mendix (Leader):** Mendix, filiale di Siemens, è un leader nel mercato del low-code development, con la sua piattaforma Mendix Platform e il supporto combinato sia per Citizen Developers che professionisti. Si differenzia in particolare per la disponibilità di diverse opzioni di distribuzione, l'investimento in innovazioni come l'*Internet of Things* (IoT) e la presenza di funzionalità di governance e sicurezza. Tuttavia, Mendix deve migliorare la propria strategia di marketing e la sua presenza geografica al di fuori di Nord America ed Europa.
3. **OutSystems (Leader):** è un leader nel Magic Quadrant con la sua piattaforma di sviluppo di applicazioni low-code che offre avanzate capacità come lo sviluppo aumentato dall'AI, il testing dell'applicazione e la governance. Il suo programma di successo del cliente fornisce una formazione intensiva, una guida esperta e partnership strategiche per offrire un'ottima esperienza del cliente. Tuttavia, la sua dipendenza dai partner per tutti gli aspetti delle operazioni commerciali e la mancanza di una focalizzazione distintiva sulle industrie o sui settori aziendali sono identificate come potenziali sfide.
4. **Microsoft (Leader):** l'offerta LCDP di Microsoft è Microsoft Power Apps, che comprende vari servizi. Microsoft Power Apps si rivolge sia ai Citizen Developers che agli sviluppatori professionisti e fornisce un'ampia gamma di strumenti code-first. Microsoft Power Apps ha la base di utenti più ampia di qualsiasi altro LCDP, grazie al numero di utenti aziendali di Microsoft 365 e Dynamics. I punti di forza di Microsoft Power Apps sono il modello di business, la strategia di prodotto e l'estensibilità della piattaforma, mentre i punti deboli sono la complessità dei prezzi e delle licenze, la distribuzione mobile e l'integrazione e le API. [25]

# 3. La metodologia Agile

La crescita e l'espansione dell'industria dell'Information Technology ha portato alla necessità di migliorare i metodi di consegna dei software e perfezionare il flusso di lavoro nel settore dello sviluppo degli applicativi. È importante raggiungere un livello di standardizzazione a livello organizzativo per allineare il processo di consegna dei prodotti software alla qualità desiderata e per misurare le prestazioni. Il metodo più utilizzato fino a qualche anno fa è il cosiddetto *Waterfall* (cascata), che però non è quello ottimale in questo campo, che è caratterizzato da un ambiente guidato dal cambiamento costante, rendendo più efficiente l'adozione di un approccio Agile. [26] Mentre la metodologia Waterfall prevede che il prodotto venga consegnato solo al termine del progetto, la metodologia Agile prevede il rilascio costante di piccoli incrementi di lavoro entro determinati intervalli di tempo.

Il responsabile di un progetto, il cosiddetto *project manager*, deve capire quale metodologia di sviluppo di applicazioni è quella più adatta, considerando, nella decisione, fattori come il tempo, il budget e le competenze del team di sviluppo. È da sottolineare, inoltre, l'importanza di strumenti online come servizi di condivisione di file e applicazioni per videoconferenze per la comunicazione e la coordinazione tra i vari membri del gruppo di lavoro. [27]

Esistono diversi framework agili per lo sviluppo di software, come Scrum, Kanban ed Extreme Programming. [26] Quello che sarà presentato in questa tesi è il framework Scrum, che è quello che è stato utilizzato durante lo sviluppo del progetto in azienda ed è anche quello più diffuso, come si può notare nel **Grafico 5**.

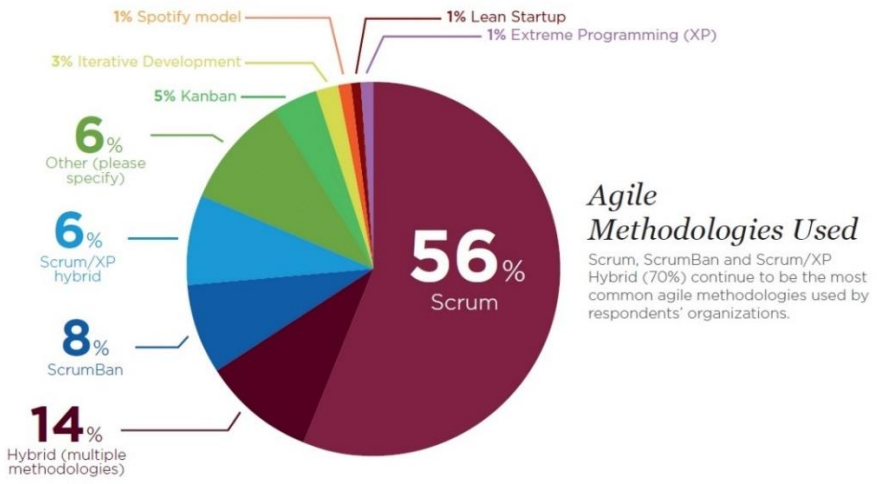


**Grafico 5:** Diffusione delle metodologie Agile [28]

## 3.1 Il confronto tra la metodologia Agile e la metodologia Waterfall

Negli ultimi vent'anni i metodi Agile di implementazione del software sono diventati tre volte più diffusi di quelli Waterfall e hanno ottenuto il triplo dei risultati positivi. Tuttavia, il Waterfall non è superato e le metodologie alternative devono essere ben comprese per fare la scelta giusta per un progetto software. Il modello a cascata coinvolge diverse competenze e prevede un'accettazione e un'approvazione formale alla fine di ogni fase, mentre la metodologia Agile divide l'ambito del prodotto in incrementi più piccoli, consegnati alla fine di ogni iterazione (*sprint*). I team Agile devono possedere tutte le competenze necessarie per realizzare un prodotto funzionante al termine di ogni sprint. [26]

Il modello Waterfall è il primo modello di processo introdotto e presenta un ciclo di vita lineare e sequenziale, semplice da capire e da utilizzare. Ogni fase del processo di sviluppo inizia solo se la fase precedente è stata completata e non vi sono sovrapposizioni tra le varie fasi. Questo approccio è ampiamente utilizzato nell'ingegneria del software per garantire il successo del progetto. L'intero processo di sviluppo del software è suddiviso in fasi distinte e il risultato di una

fase funge da input per la fase successiva. Il modello Waterfall comprende cinque fasi: analisi, progettazione, implementazione, test e manutenzione. [27]

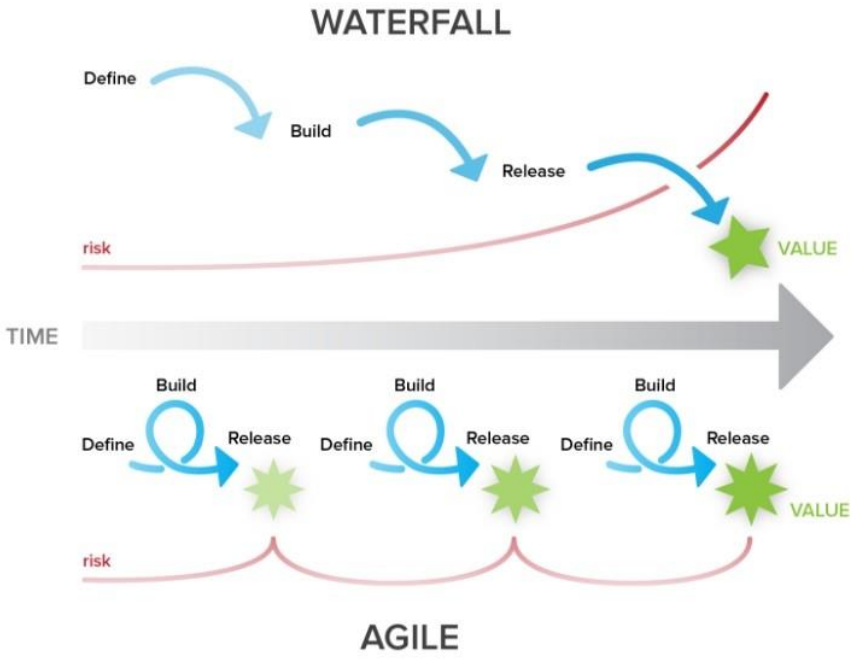


**Figura 15:** Modo con cui i metodi Waterfall e Agile creano valore e rischio associato [28]

Il metodo Agile, invece, è un modo di affrontare e avere successo in un ambiente incerto e mutevole, in cui il processo di sviluppo è allineato alle mutevoli esigenze aziendali. Si riferisce a un gruppo di metodologie di sviluppo del software basate sullo sviluppo iterativo, sulla collaborazione e su frequenti ispezioni e adattamenti. Agile promuove il lavoro di squadra, l'auto-organizzazione e la consegna rapida di software di alta qualità allineato alle esigenze dei clienti e agli obiettivi aziendali. Utilizza brevi iterazioni da 1 a 4 settimane e un processo di feedback frequente per fornire un prodotto funzionante in modo incrementale. [27]

La **Tabella 6** mette a confronto le principali caratteristiche delle due metodologie:

| Approccio Waterfall | Approccio Agile |
|---|---|
| Lunga fase di definizione delle specifiche | Fase di specifica continua |
| Elevato impegno per la documentazione | Solo la documentazione necessaria |
| Lunga durata del rilascio | Sviluppo con cicli brevi |
| Le modifiche dovrebbero essere evitate | I cambiamenti sono previsti |
| Poca comunicazione e collaborazione | Alta comunicazione e collaborazione |
| L'obiettivo è determinato | La scadenza è determinata |
| Rischi alti al termine del progetto | Rischi bassi al termine del progetto |

**Tabella 6:** Confronto tra metodologia Waterfall e metodologia Agile [28]

La differenza principale tra i due approcci riguarda l'elevata flessibilità della metodologia Agile, in cui i cambiamenti sono previsti nel tempo, rispetto a quella Waterfall, dove invece le modifiche dovrebbero essere evitate perché generano costi elevati. Tutte le altre caratteristiche aiutano ad accentuare questa diversa flessibilità. La fase di definizione delle specifiche nel metodo a cascata è lunga e viene fatta una volta sola ad inizio progetto, mentre nell'approccio Agile viene definito un quadro generale all'inizio del progetto, e poi per ogni incremento vengono dettagliate le specifiche. Con il metodo Waterfall, inoltre, il valore per il cliente è creato solo a fine progetto, a differenza dell'approccio Agile in cui viene creato valore incrementale alla fine di ogni interazione. Infine, la metodologia Agile ha un'alta comunicazione e collaborazione sia tra i membri del team che con i clienti, e, per di più, non è l'obiettivo ad essere determinato, bensì la scadenza e si cerca di creare quanto più valore possibile fino alla scadenza del progetto. [28]

## 3.2 I principi e il ciclo di vita dello sviluppo software in Agile

La metodologia Agile è stata creata per rispondere alla necessità di avere una metodologia più leggera e semplice per accelerare il processo di rilascio del software. Questo approccio comprende una serie di metodi di sviluppo del software che sono emersi negli anni 2000 e si basano su principi comuni, che

derivano direttamente o indirettamente dal "Manifesto per lo sviluppo agile del software" (conosciuto anche come "Manifesto Agile") pubblicato nel 2001. Il Manifesto Agile fissa come obiettivo quello di garantire la piena soddisfazione del cliente, e l'adozione delle metodologie agili può portare ad una riduzione dei costi e dei tempi di sviluppo del software, con conseguente aumento della qualità.

Il Manifesto Agile contiene 12 principi:

1. La nostra massima priorità è soddisfare il cliente rilasciando software di valore, fin da subito e in maniera continua.
2. Accogliamo i cambiamenti nei requisiti, anche a stadi avanzati dello sviluppo. I processi agili sfruttano il cambiamento a favore del vantaggio competitivo del cliente.
3. Consegniamo frequentemente software funzionante, con cadenza variabile da un paio di settimane a un paio di mesi, preferendo i periodi brevi.
4. Committenti e sviluppatori devono lavorare insieme quotidianamente per tutta la durata del progetto.
5. Fondiamo i progetti su individui motivati. Diamo loro l'ambiente e il supporto di cui hanno bisogno e confidiamo nella loro capacità di portare il lavoro a termine.
6. Una conversazione faccia a faccia è il modo più efficiente e più efficace per comunicare con il team ed all'interno del team.
7. Il software funzionante è il principale metro di misura di progresso.
8. I processi agili promuovono uno sviluppo sostenibile. Gli sponsor, gli sviluppatori e gli utenti dovrebbero essere in grado di mantenere indefinitamente un ritmo costante.
9. La continua attenzione all'eccellenza tecnica e alla buona progettazione esalta l'agilità.
10. La semplicità - l'arte di massimizzare la quantità di lavoro non svolto - è essenziale.
11. Le architetture, i requisiti e la progettazione migliori emergono da team che si auto-organizzano.

12. A intervalli regolari il team riflette su come diventare più efficace, dopodiché regola e adatta il proprio comportamento di conseguenza. [29]

A partire da questi principi, sono stati poi sviluppati tutti i framework Agile presenti oggi e, sebbene ogni organizzazione IT possa avere i propri processi e le proprie procedure, il ciclo di vita del progetto con metodologia Agile consiste generalmente in fasi simili. La prima fase è quella di avvio del progetto, che viene gestita da un team diverso da quello che realizzerà il progetto. Dopo aver ricevuto una richiesta di proposta, si riunisce un piccolo gruppo di esperti per raccogliere i requisiti, stimare l'impegno e negoziare un contratto. Questa fase può durare dalle due alle quattro settimane. Una volta firmato il contratto, viene assegnato un project manager e inizia la fase di pianificazione. La prima iterazione è in genere di portata ridotta e serve a impostare il progetto, a perfezionare i requisiti e a costruire il backlog. Dopo aver completato questi compiti, può iniziare il lavoro di sviluppo. Il team di sviluppo del prodotto deve avere tutte le competenze necessarie per consegnare il primo incremento del prodotto.

I cicli *Define*, *Build*, *Review* e *Release* continueranno fino alla consegna di tutti gli incrementi di prodotto e al completamento dell'ambito del progetto. Il progetto entra quindi nella fase di chiusura, in cui ci si concentra sull'accettazione formale e sulla riassegnazione dei membri del team ad altri progetti. [26]

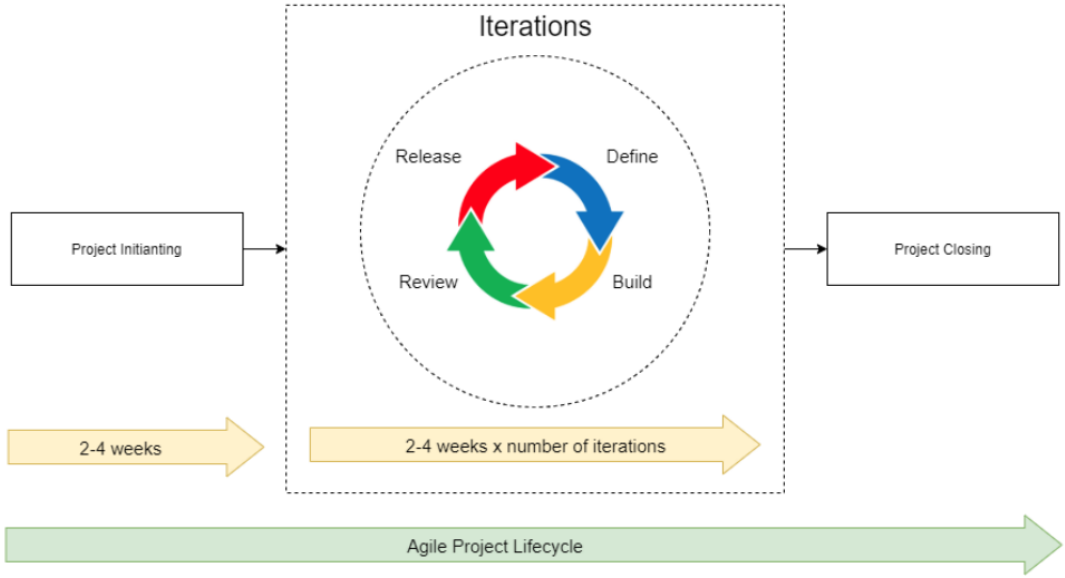


**Figura 16:** Schema del ciclo di sviluppo software in Agile [26]

## 3.3 Il framework Scrum

Lo Scrum è uno dei numerosi framework di sviluppo Agile, il più famoso e più utilizzato come visto nell'introduzione di questo capitolo, sviluppato da Ken Schwaber e Jeff Sutherland per affrontare problemi complessi e rilasciare prodotti di alta qualità. I due hanno sviluppato Scrum nei primi anni Novanta e, nel 2010, hanno scritto la prima versione della Guida Scrum. [30] Scrum è un metodo leggero, facile da comprendere ma richiede un cambio di mentalità da parte dell'azienda per poterlo padroneggiare.

La metodologia Scrum, in un contesto basilare, ha dimostrato diversi principi utili per lo sviluppo di un prodotto, come ad esempio la creazione di team con membri effettivi e con competenze trasversali, la gestione autonoma del team, l'adozione di cicli iterativi brevi per il feedback completo e il contenimento del costo in caso di cambiamenti. L'applicazione di tali principi aumenta l'agilità e il feedback, permette una stima anticipata del ROI e riduce i rischi. I pilastri empirici su cui si fonda Scrum sono: trasparenza, ispezione e adattamento. [30]

I componenti di Scrum che saranno oggetto di trattazione nei prossimi paragrafi sono i ruoli e gli eventi di Scrum e i suoi artefatti.

### 3.3.1 I ruoli e gli eventi di Scrum

L'elemento fondamentale per l'adozione del framework Scrum è dato dallo Scrum Team, un gruppo cross-funzionale con diversi ruoli. I ruoli del framework sono tre: il *Product Owner*, il *Team di Sviluppo* (*Development Team*) e lo *Scrum Master*. La **Figura 17** mostra i ruoli di Scrum con le proprie competenze all'interno del framework: il colore verde riguarda le responsabilità del Product Owner, il colore blu del Team di Sviluppo, mentre il colore arancione indica lo Scrum Master.

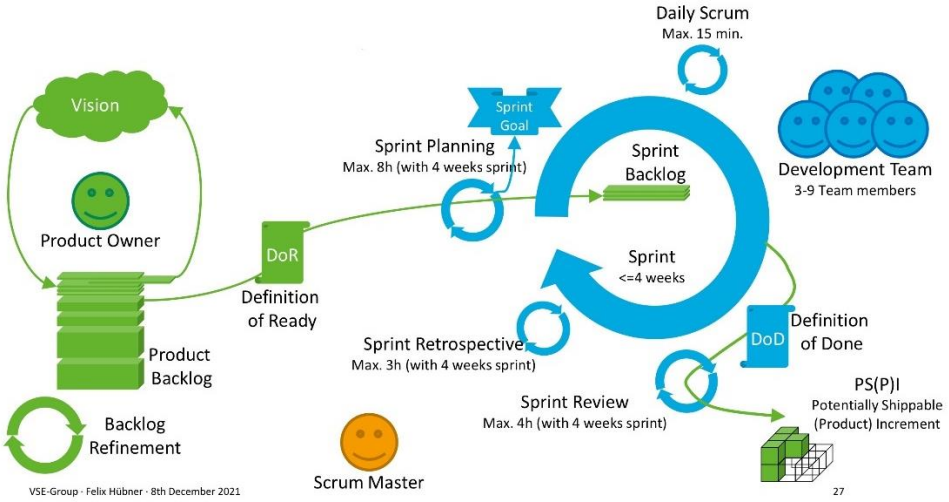


**Figura 17:** Ruoli, processi e responsabilità nel framework Scrum [28]

Il Product Owner ha il compito di massimizzare il ritorno sugli investimenti, identificare le caratteristiche del prodotto e decidere cosa andrà in cima alla lista di priorità per il successivo Sprint. È responsabile della scelta delle caratteristiche di prodotto che portano il maggior valore (*Product Backlog*), che può essere influenzata dal desiderio di soddisfare i clienti, gli obiettivi strategici e altri fattori. In Scrum, c'è una sola persona che agisce come Product Owner ed è responsabile per il valore del lavoro.

Il Team di Sviluppo, invece, è un gruppo di sviluppo cross-funzionale e auto-organizzato che costruisce il prodotto indicato dal Product Owner. Non ci sono titoli specialistici fissati e ogni membro ha competenze di primo, secondo e terzo livello. Il Team fornisce idee al Product Owner su come rendere il prodotto migliore e si dedica al 100% al prodotto durante lo Sprint, evitando il multitasking su più prodotti o progetti. I gruppi di prodotto con molte persone sono organizzati in più team, ciascuno focalizzato su diverse funzionalità del prodotto, con uno stretto coordinamento dei loro sforzi.

Il ruolo di Scrum Master, infine, consiste nell'assicurarsi che il Team di Sviluppo segua i valori, le pratiche e le regole di Scrum, e può essere svolto da una persona all'interno del team stesso. Il suo obiettivo è aiutare il gruppo di sviluppo a organizzarsi in modo agile e dinamico, promuovendo l'auto-

organizzazione e la creatività. Inoltre, si occupa di rimuovere eventuali ostacoli che potrebbero impedire il successo del progetto, lavorando per garantire la massima produttività del gruppo. [31]

La **Figura 17** mostra anche gli eventi previsti da Scrum, ognuno con durata ben definita, che prendono il nome di *Cerimonie.* In seguito, verranno descritti i principali eventi del framework:

- **Sprint:** gli Sprint sono eventi per creare valore e includono attività come Sprint Planning, Daily Scrum, Sprint Review e Sprint Retrospective. Durante gli Sprint, non ci sono modifiche che potrebbero minare lo Sprint Goal, la qualità non diminuisce, il Product Backlog viene perfezionato e l'ambito di applicazione può essere chiarito. Gli Sprint favoriscono la prevedibilità e l'ispezione regolare dei progressi verso il Product Goal. Ogni Sprint può essere considerato un breve progetto. [30]
- **Sprint Planning:** Durante lo Sprint Planning viene pianificato il lavoro da eseguire durante lo sprint ed è diviso in due fasi. Durante la prima fase, l'intero gruppo Scrum decide gli obiettivi e le funzionalità del prossimo Sprint, mentre nella seconda fase, lo Scrum Master e il Team di Sviluppo stabiliscono i dettagli tecnici su come verrà implementato l'incremento durante il successivo Sprint. L'evento prevede la selezione del lavoro da fare, la preparazione dello Sprint Backlog e l'identificazione della maggior parte del lavoro che è probabile sia realizzato durante l'attuale Sprint. [32]
- **Daily Scrum**: Il Daily Scrum è un evento di 15 minuti che coinvolge i Developer dello Scrum Team, e che si tiene alla stessa ora e nello stesso luogo ogni giorno lavorativo dello Sprint. Il suo scopo è ispezionare l'avanzamento verso lo Sprint Goal e adattare lo Sprint Backlog, concentrando il lavoro sui problemi da risolvere, migliorando le comunicazioni e promuovendo un rapido processo decisionale.
- **Sprint Review**: Lo Sprint Review è un evento che si tiene alla fine di ogni Sprint per ispezionare i risultati del lavoro svolto dallo Scrum Team e determinare eventuali adattamenti futuri. Durante l'evento, il team presenta

il lavoro svolto agli stakeholder chiave e discute l'avanzamento verso il Product Goal.

- **Sprint Retrospective**: Lo Sprint Retrospective è un evento nello sviluppo di software Agile Scrum che si svolge alla fine di ogni sprint. Lo scopo di questo evento è quello di ispezionare l'andamento del precedente Sprint, identificare ciò che è andato bene, ciò che ha causato problemi e cosa può essere fatto per migliorare la qualità e l'efficacia del processo. Durante lo Sprint Retrospective, lo Scrum Team identifica i cambiamenti più utili e i miglioramenti più impattanti per migliorare l'efficacia del processo. [30]

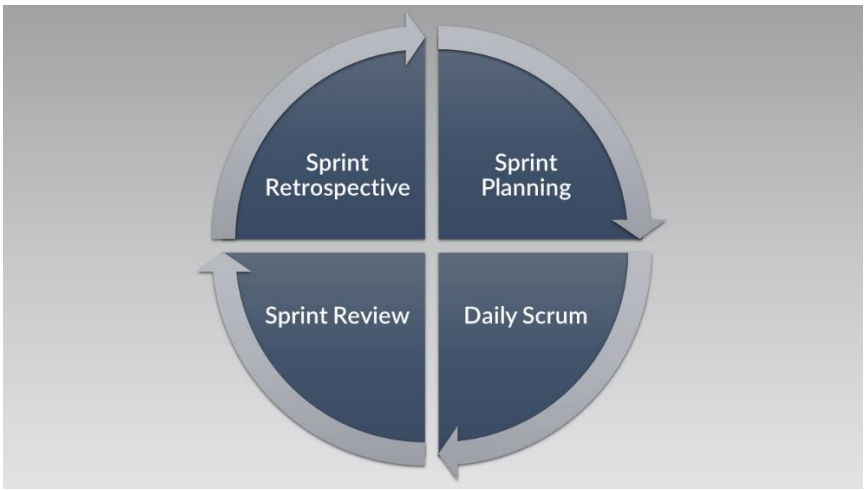


**Figura 18:** Le cerimonie Scrum [33]

### 3.3.2 Gli artefatti di Scrum

Gli artefatti di Scrum servono per massimizzare la trasparenza delle informazioni e comprensione per tutti i soggetti coinvolti. Ogni artefatto è associato ad un impegno per garantire che fornisca le informazioni necessarie per misurare l'avanzamento del progetto. Di seguito si parlerà dei principali artefatti di Scrum:

- **Product Backlog:** è un elenco ordinato di ciò che serve per migliorare il prodotto, redatto dal Product Owner, ed è l'unica fonte di lavoro per lo Scrum Team. Il Refinement del Product Backlog è un'attività continuativa

per definire ulteriormente gli elementi in modo più preciso e dettagliato. Gli sviluppatori sono responsabili della stima della dimensione degli elementi, ma il Product Owner può influenzarli aiutandoli a comprendere meglio e a cercare soluzioni di compromesso.

- **Sprint Backlog:** è un piano per consegnare l'incremento durante lo Sprint. È composto dallo Sprint Goal, dagli elementi del Product Backlog selezionati per lo Sprint e da un piano attuabile. È un piano elaborato dagli sviluppatori per raggiungere l'obiettivo dello Sprint e viene aggiornato man mano che si approfondisce l'apprendimento. Inoltre, dovrebbe avere abbastanza dettagli per consentire ai Developer di ispezionare i propri progressi durante il Daily Scrum. [30]
- **Burndown Chart:** è uno strumento usato per monitorare l'avanzamento del lavoro in uno Sprint, tramite un grafico che rappresenta i giorni rimanenti sull'asse X e il lavoro ancora da fare sull'asse Y. La linea ideale di progresso viene tracciata giorno per giorno in base alla velocità di sviluppo, mentre la linea reale rappresenta il lavoro effettivamente svolto. Se la linea reale si trova sopra la linea ideale, significa che il progetto è in ritardo e devono essere prese azioni correttive. Il grafico deve essere trasparente e comprensibile anche per coloro che non sanno leggere i dati. [32]

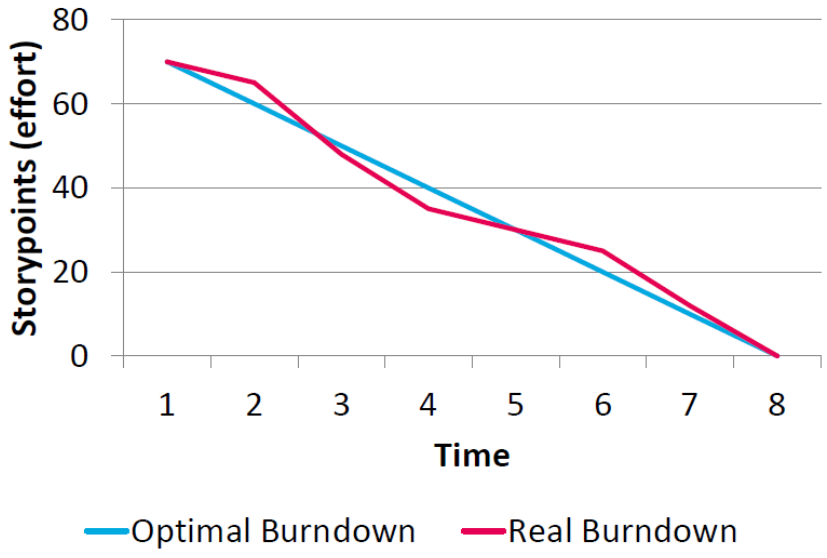


**Grafico 6:** Esempio di Burndown Chart [28]

# 4. Progetto

Il progetto che sarà descritto in questo capitolo riguarda lo sviluppo di una applicazione in low-code tramite Pega Platform, con metodologia Agile Scrum. Si tratta, dunque, di una applicazione pratica di ciò che è stato discusso nei due precedenti capitoli. Il progetto è stato realizzato durante il tirocinio di tesi che ho svolto con la società Blue Reply Srl, e in particolare con la Business Unit *Products & Claims*.

Il team *Blue CIO* (Chief Information Officer) riceve e gestisce le *richieste di acquisto* (abbreviato in RdA) di hardware e software per i propri dipendenti, permettendo la personalizzazione della postazione del lavoratore e favorendo l'*on-boarding* dei nuovi ingressi. Al momento, questo processo è gestito tramite posta elettronica: i richiedenti inviano e-mail ai membri della squadra Blue CIO con le proprie richieste, che però sono più o meno strutturate, e molte volte non contengono le informazioni necessarie e utili per il processamento dell'acquisto desiderato.

Blue Financial Services è una realtà in crescita. Inizialmente la richiesta di acquisto era una esigenza di pochi, ma si è poi allargata a macchia d'olio; quindi, se in principio questo metodo era sostenibile e gestibile, presto è diventato obsoleto e non più tollerabile. Un primo miglioramento al processo è stata l'aggiunta di un nuovo membro nel team, dato che precedentemente solo una persona era addetta alla gestione delle RdA, ma ciò ha portato anche ad un successivo problema: quella della divisione della gestione delle code di lavoro.

Inoltre, quando arriva una e-mail con le informazioni per una RdA, i membri del team Blue CIO popolano a mano un foglio Excel inserendo le informazioni necessarie, un modo sicuramente non efficiente di gestire le richieste.

Per questo motivo è stato deciso di implementare una applicazione che potesse rendere il processo di invio di richieste di acquisto formale, standardizzato e strutturato, che riuscisse a gestire il processamento delle code di lavoro, e che

aiutasse i membri del team Blue CIO a gestire le richieste pervenute e accettate dai manager delle varie *Business Unit* (BU).

Lo sviluppo della piattaforma non è partito completamente da zero, ma alcune funzionalità erano state implementate in un precedente lavoro di tesi in azienda. Dunque, a partire da questa base, è stato raffinato il lavoro precedentemente svolto ed è stata portata avanti la realizzazione dell'applicazione.

Durante lo svolgimento del progetto sono stati utilizzati vari tool e programmi, che sono stati fondamentali per la coordinazione, la cooperazione e l'implementazione dell'applicazione. In seguito, saranno citati i principali software utilizzati:

- **Microsoft Teams:** Blue Financial Services ha provvisto un'utenza provvisoria con accesso a Microsoft Teams, piattaforma fondamentale per le comunicazioni tra i membri del team e per effettuare le Cerimonie Scrum, come lo Sprint Planning, il Daily Scrum, lo Sprint Review e lo Sprint Retrospective.
- **Jira Software:** è un software sviluppato da Atlassian per la gestione del progetto con metodologia Agile. Nel corso di questo capitolo ci sarà una parte dedicata su come è stato utilizzato questo software nel progetto.
- **Pega Academy:** è la piattaforma utilizzata per la formazione sull'utilizzo di Pega Platform nel contesto dello sviluppo di applicazioni a basso codice. Il corso prevede parti teoriche (*Module*) alternate ad esercizi pratici (*Challenge*) ed ha avuto una durata di quarantaquattro ore e trenta minuti.
- **Pega Platform:** Pega Platform è stata la piattaforma utilizzata per sviluppare l'applicazione in low-code per la gestione delle richieste di acquisto.

## 4.1 Fase di analisi

La fase di analisi, comprendente l'analisi del problema, l'analisi degli obiettivi e l'analisi della strategia, è necessaria per inquadrare un problema e tutti i sotto-problemi che ne derivano, per capire come trasformarli in opportunità e su quali soluzioni porre particolarmente l'attenzione.

### 4.1.1 Analisi del problema: *Problem tree*

Il metodo del *problem tree* viene utilizzato per analizzare i problemi e si basa sulla creazione di un elenco di tutti i sotto-problemi del problema principale e delle relative relazioni causa-effetto. Il problem tree riflette gli aspetti negativi della situazione attuale, identifica le cause e mette in relazione gli effetti. [34]

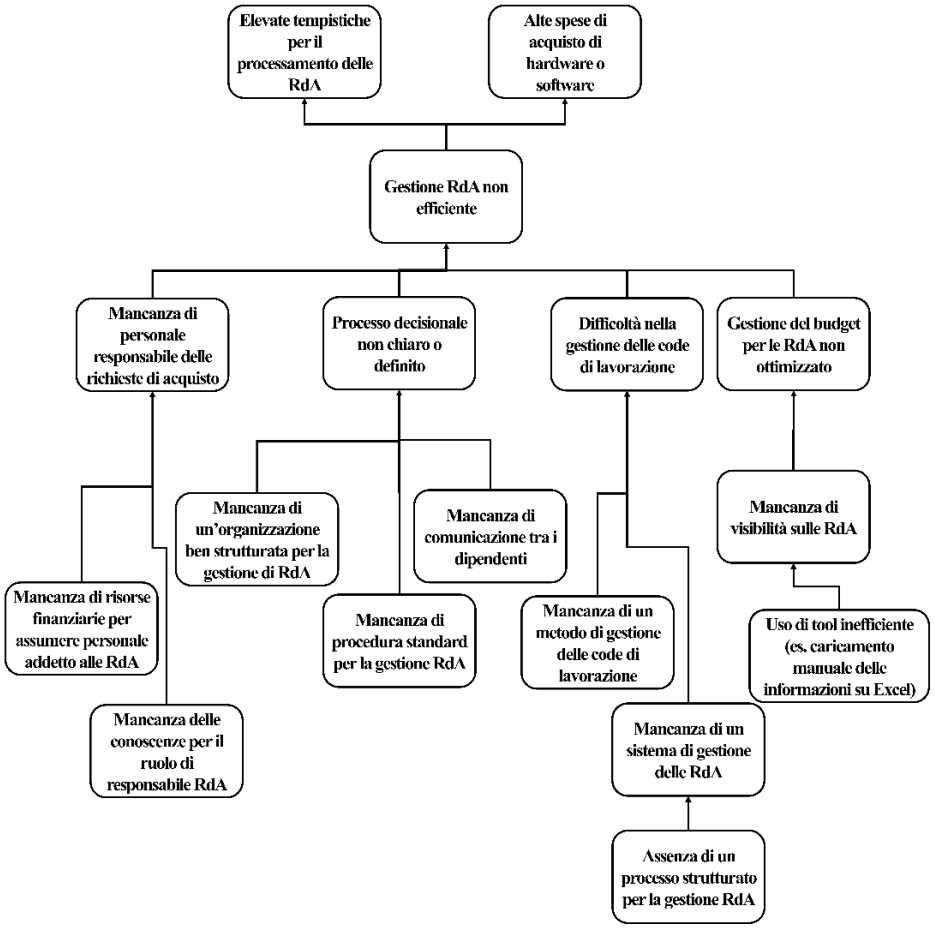


**Figura 19:** Problem tree

### 4.1.2 Analisi degli obiettivi: Solution tree

Il *solution tree* segue la stessa struttura gerarchica dell'albero dei problemi, ma rappresenta una versione positiva del problema. I punti deboli e le problematiche diventano punti di forza e opportunità. I problemi sono riformulati in situazioni positive che siano desiderabili e fattibili e le relazioni causa-effetto sono trasformate in connessioni mezzo-fine. [34]

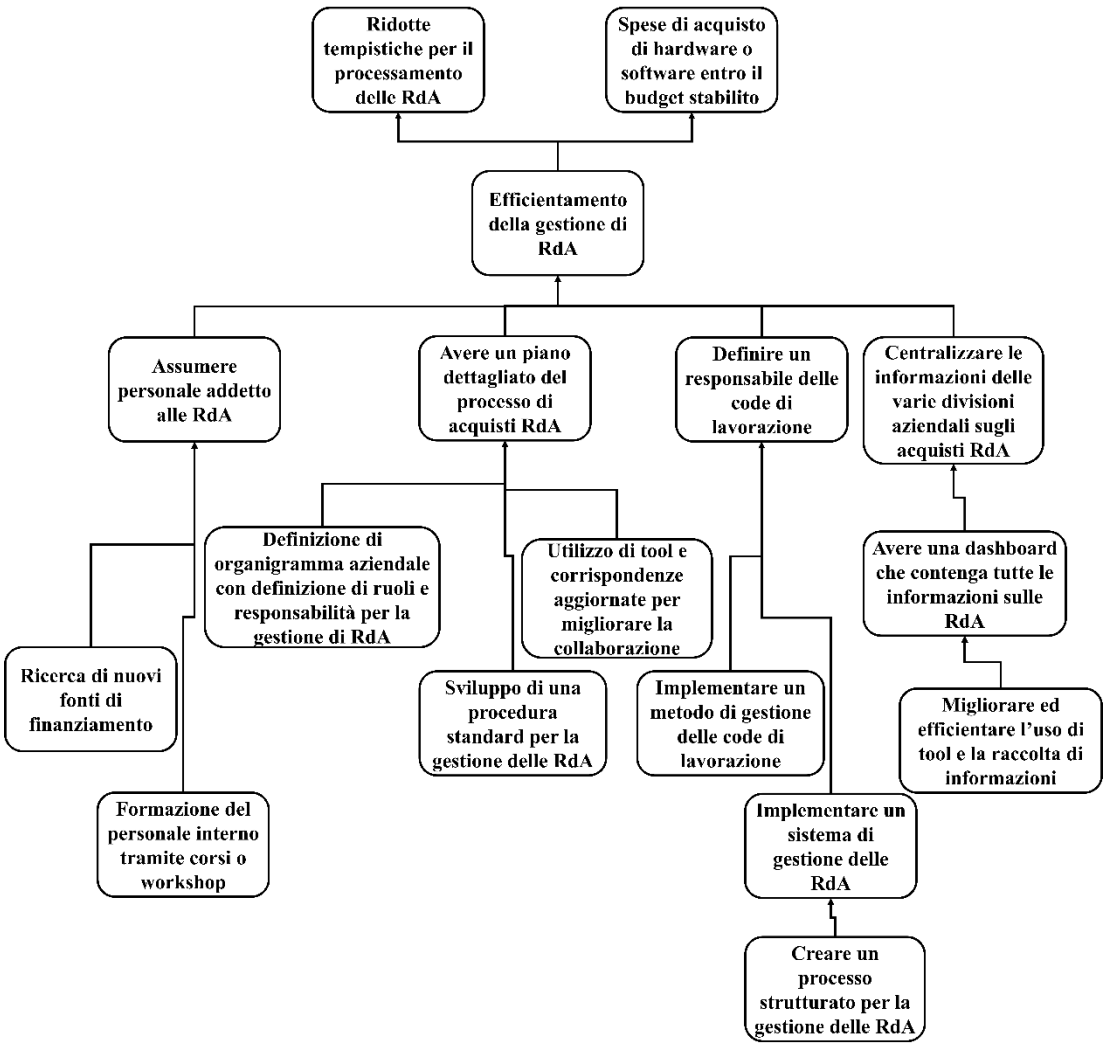


**Figura 20:** Solution tree

### 4.1.3 Analisi della strategia: *Strategy selection*

Dopo aver identificato il problema, le sue cause e le soluzioni previste, si sono individuate le aree principali che richiedono particolare attenzione utilizzando il metodo dello *strategy selection*, che rappresenta la strategia selezionata per raggiungere gli obiettivi desiderati. Si ottiene eliminando i rami dell'albero delle soluzioni che non sono rilevanti per il progetto e che possono essere trascurati. [34]

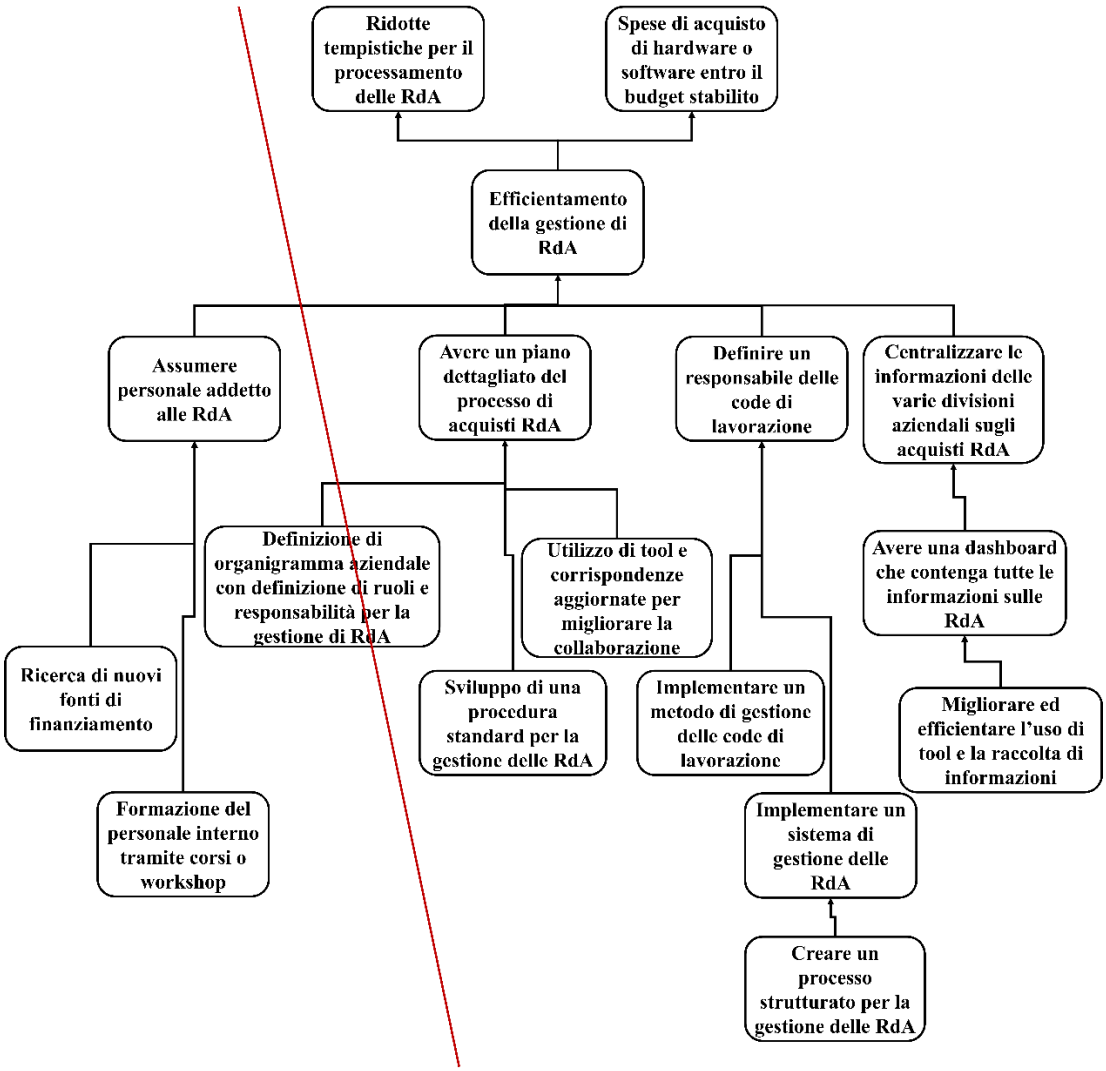


**Figura 21:** Strategy selection

## 4.2 Pega Platform e Scrum: identificazione ruoli del progetto

Il progetto è stato svolto, come già accennato, seguendo il framework Scrum Agile, che prevede dei ruoli ben definiti. È stato essenziale, dunque, definire correttamente gli attori coinvolti in tutto il processo. Per fare ciò, è stata di aiuto anche l'Academy di Pega, che ha fornito, oltre ai ruoli di Scrum già discussi, altre funzioni che potrebbero prendere parte nel processo di sviluppo.

**Figura 22:** Ruoli identificati da Pega [17]

Per questo progetto sono state coinvolte nove persone; in seguito, saranno descritti i ruoli e la suddivisione dei partecipanti:

- **Citizen Developers:** sono utenti aziendali non tecnici che partecipano attivamente allo sviluppo delle applicazioni. [17] In questo progetto sono coloro che hanno realizzato la piattaforma e che nel framework Scrum sono identificati come Team di Sviluppo. Questo ruolo è stato ricoperto dal sottoscritto e da un altro tesista, Roberto Cutaia.
- **Product Owner:** come già visto nel capitolo dedicato alla metodologia Agile, possiede il backlog del prodotto e dà priorità alle voci del backlog. È anche responsabile della creazione dei criteri di accettazione. [17] Il ruolo del Product Owner è stato svolto in alternanza dai due membri del team Blue CIO.

- **Scrum Master:** come già visto in precedenza, è responsabile di promuovere e sostenere l'uso di Scrum all'interno del team, nonché di garantire che tutti i membri del team abbiano una chiara comprensione della teoria e delle pratiche di Scrum. Inoltre, facilita gli eventi Scrum, se necessario, per garantire che il team operi in modo efficiente ed efficace. [17] Il ruolo è stato ricoperto, in alternanza, da coloro che si sono occupati anche della parte organizzativa del tirocinio in azienda.
- **System Administrator and IT resources:** forniscono competenze in base alle necessità, che possono includere aree quali l'infrastruttura, la sicurezza o l'integrazione. [17] In questo caso questo ruolo è stato ricoperto da due esperti del System Architect di Pega, che hanno fornito e gestito la macchina virtuale in cui sviluppare il progetto e supporto tecnico per l'implementazione della piattaforma.
- **Project Delivery Leader:** è responsabile di fornire indicazioni generali sulla pianificazione e la realizzazione dei progetti. [17] Il ruolo è stato svolto dal manager della Business Unit.

## 4.3 Fase di pianificazione e Jira Software

Per la pianificazione del lavoro in Scrum è stato di grande aiuto il tool *Jira Software*, facente parte dei prodotti di Jira sviluppati dalla *public company* australiana Atlassian. Prima della data del *kick-off* del progetto il Product Owner ha definito i requisiti di business, creando e caricando su Jira Software le *User Stories* (o storie), ovvero gli elementi che popolano il Product Backlog.

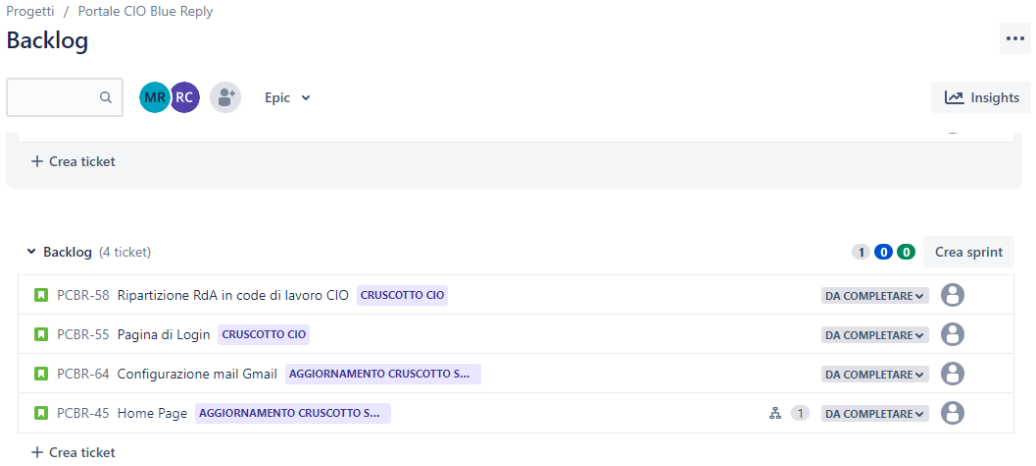


**Figura 23:** Esempio di elementi del Product Backlog su Jira Software

A questo punto ha avuto inizio la realizzazione dell'applicazione. In totale sono stati effettuati sette Sprint, pianificati su Jira stesso, della durata di una settimana ciascuno. All'inizio di ogni Sprint è stato effettuato lo Sprint Planning, in cui si sono scelte anche le User Stories inserire nel corrente Sprint Backlog (vedi **Figura 24** per un esempio di elementi inseriti in uno Sprint Backlog). Ogni giorno i due Citizen Developer, tramite Microsoft Teams, hanno organizzato il Daily Scrum per la revisione dello Sprint Backlog, per l'aggiornamento dell'avanzamento dello Sprint e per la risoluzione di eventuali problemi. Al termine di ogni Sprint, infine, come da prassi, sono stati effettuati lo Sprint Review e lo Sprint Retrospective.

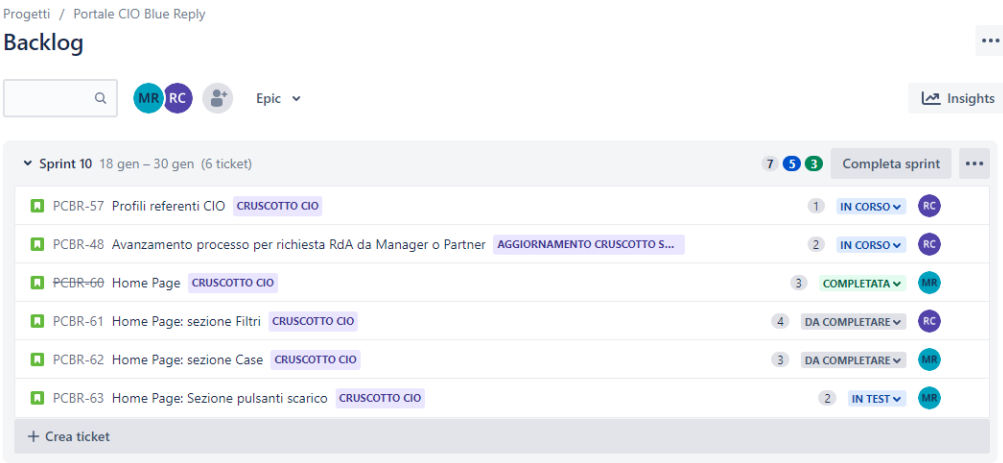


**Figura 24:** Esempio di elementi dello Sprint Backlog su Jira Software (settimo Sprint)

Jira Software ha offerto anche altre funzionalità per l'organizzazione e la gestione del progetto. Tramite la sezione *Board* è stato possibile controllare lo stato di avanzamento delle User Stories nel corso dello Sprint: il Team di Sviluppo ha costantemente aggiornato lo stato delle storie tramite questa sezione. I quattro stati disponibili sono *da completare*, *in corso*, *in test* e *completato*.

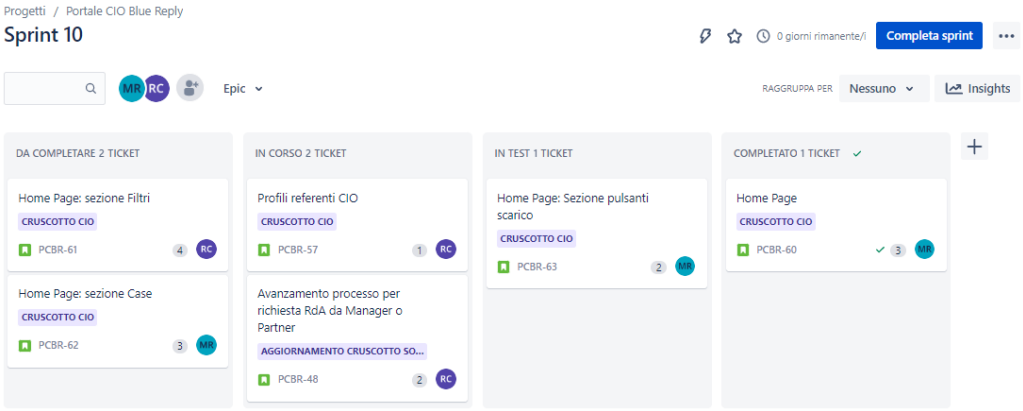


**Figura 25:** Esempio della sezione Board di Jira Software (settimo Sprint)

Per ogni User Story, inoltre, è stato possibile assegnare un punteggio che indicasse la complessità dell'incarico della storia in termini di difficoltà di implementazione e di lunghezza della richiesta, tramite la voce *Story point estimate*. Si è deciso di assegnare punteggi da 1 a 5 in ordine crescente di complessità. Il software ha anche consentito di assegnare una storia ad uno dei membri del Team di Sviluppo, tramite il campo *Assegnatario*. La divisione delle User Stories tra i Citizen Developer non è stata comunque vincolante, e in caso di necessità ognuno ha dato il proprio supporto anche alle storie non assegnategli.

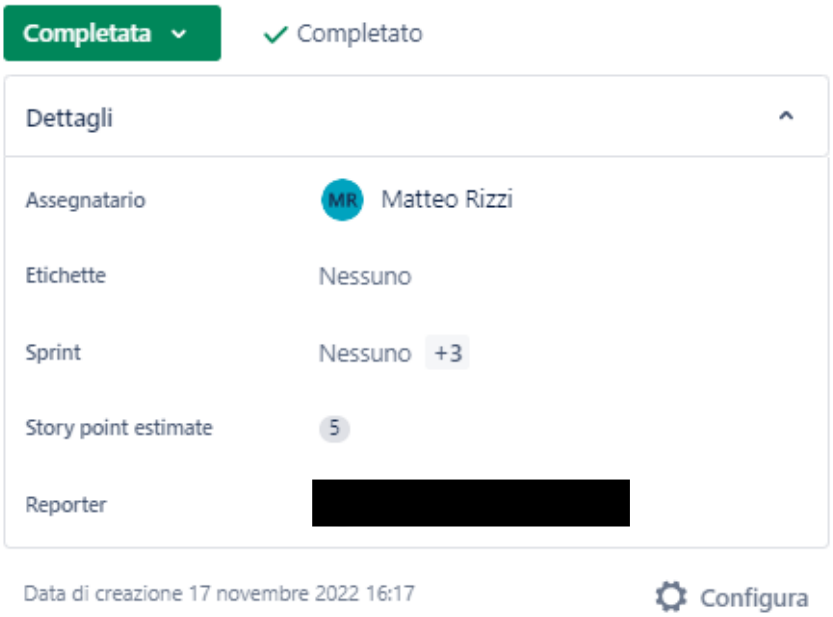


**Figura 26:** Assegnatario e Story point estimate di una User Story

L'ultima funzionalità di Jira Software utilizzata per il progetto è la *Roadmap*, che presenta una visione olistica del progetto, dal suo principio (con le storie facenti parte della sezione *Cruscotto Sottomissione RdA*, del precedente lavoro di tesi), fino al termine del progetto di tesi corrente (sezioni *Aggiornamento Cruscotto Sottomissione RdA* e *Cruscotto CIO*).

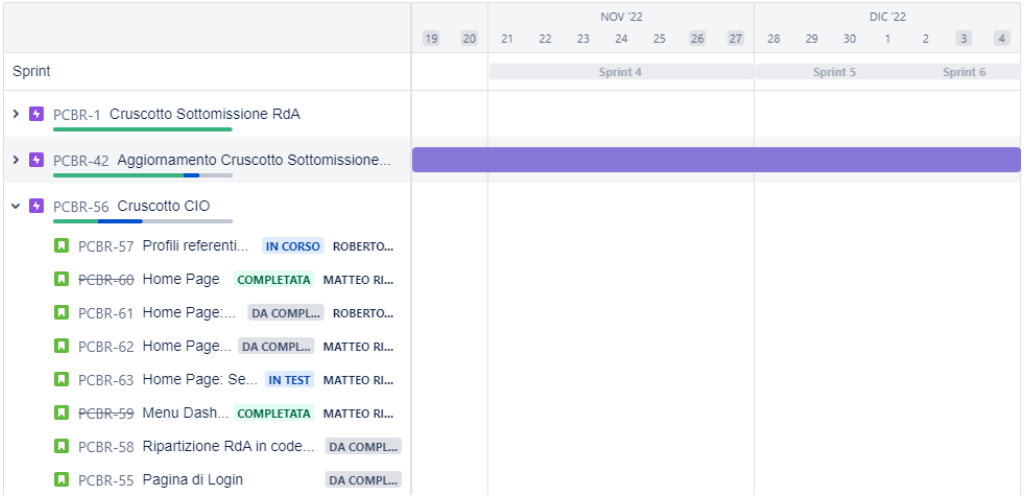


**Figura 27:** Roadmap del progetto Portale CIO Blue Reply su Jira Software

## 4.4 Realizzazione del progetto

L'adozione di una metodologia Agile nel progetto è stata facilitata grazie all'approccio in micro-processi (*Microjourney*) del *Case Designer* di Pega Platform. Forrester definisce un processo (*Journey*) come una "serie di interazioni tra un cliente ed una organizzazione che si verificano mentre il cliente persegue un obiettivo specifico" [35]. Poiché un processo può essere lungo e complesso, Pega Platform permette di dividerlo in più micro-processi, ognuno dei quali rappresenta una parte del percorso complessivo del cliente che raggiunge un sottoinsieme dei risultati aziendali richiesti. [35]

La **Figura 28** mostra il processo riguardate il *Case Type* **Richiesta RdA** al termine del progetto e i suoi relativi micro-processi, che saranno analizzati più nel dettaglio nei successivi paragrafi.

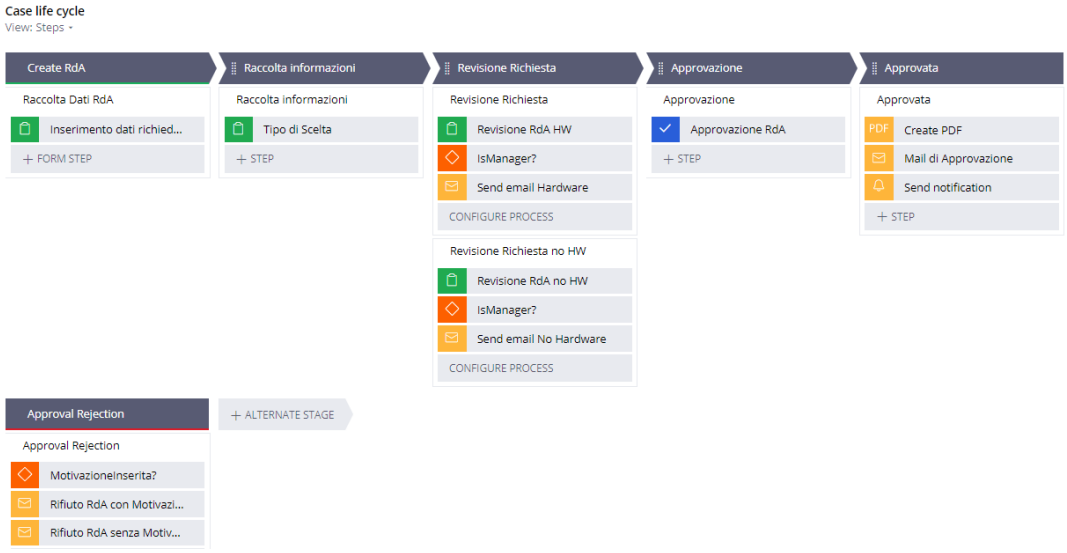


**Figura 28:** Case Life Cycle del Case Type *Richiesta RdA* a fine progetto

Ogni parte del processo è suddiviso su vari livelli gerarchici. Il livello più alto è rappresentato dallo *Stage*, che indica in quale punto del flusso si è giunti. All'interno di esso sono presenti i *Process*, che racchiudono il flusso di alcuni *Step* dello Stage corrente. Gli Step possono essere di varia natura: possono raccogliere delle informazioni dagli utenti (caratterizzati da un'icona di colore verde), oppure possono essere Step di approvazione/rifiuto (icona blu), o ancora

possono essere degli Step automatici, come un invio mail o di notifica (icona gialla).

### 4.4.1 Lo stato dell'arte del progetto

Al momento dell'avvio del progetto, era stato già impostato un prototipo della piattaforma nella sezione del *Cruscotto sottomissione RdA*. L'obiettivo del precedente lavoro di tesi è stato quello di impostare il flusso per l'invio della richiesta di acquisto e testare quanto complicati e dettagliati potessero essere i requisiti di business. Per i vari step sono state implementate alcune funzionalità base.

**Figura 29:** Stato dell'arte del Case Life Cycle [36]

Sono stati realizzati, inoltre, il portale *User Portal* e le *Persone* che possono interagire con il portale. Sono stati definiti anche alcuni *Data Type*, ovvero un insieme di campi che definiscono un oggetto, che sono stati necessari per la raccolta e la gestione delle informazioni. Alcuni Data Type creati sono *Richiedente*, *Sede*, *Fornitore*, *Business Unit* e *Manager*. Sono stati caricati, infine, dei *Record* associati ai Data Type che richiedessero un dataset già fornito.

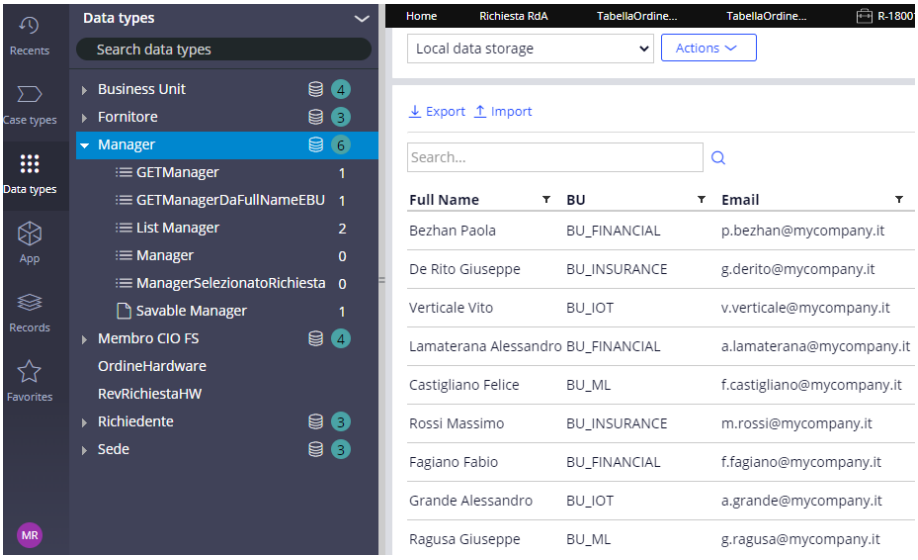


**Figura 30:** Data Type *Manager* con alcuni dei suoi record

### 4.4.2 Aggiornamento *Cruscotto sottomissione RdA*: Inserimento dati richiedente

La maggior parte degli Sprint sono serviti per modificare e raffinare ciò che era stato svolto in precedenza, nonché ad implementare ciò che mancava per la realizzazione del cruscotto per la sottomissione di RdA. In generale, il lavoro precedente è stato quasi completamente rivisto, mantenendo le funzionalità di base.

La prima funzione che è stata realizzata è quella relativa all'inserimento dei dati del richiedente, riferito allo Stage *Create RdA*. È stata richiesta la possibilità di inserire tutti i dati necessari in una sola pagina di navigazione.

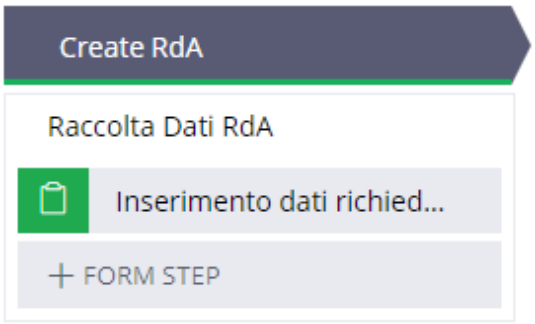


**Figura 31:** Stage *Create RdA* per l'inserimento dati richiedente

Lo Stage comprende un unico Step di raccolta di informazioni del richiedente. Le informazioni da inserire sono *Nome*, *Cognome* ed *E-mail* del richiedente, raccolti tramite dei campi testuali di input, e la *Business Unit* di riferimento e il *Nome Manager* a cui inoltrare la richiesta, entrambi selezionabili tramite delle *dropdown list*, che erano state popolate precedentemente.

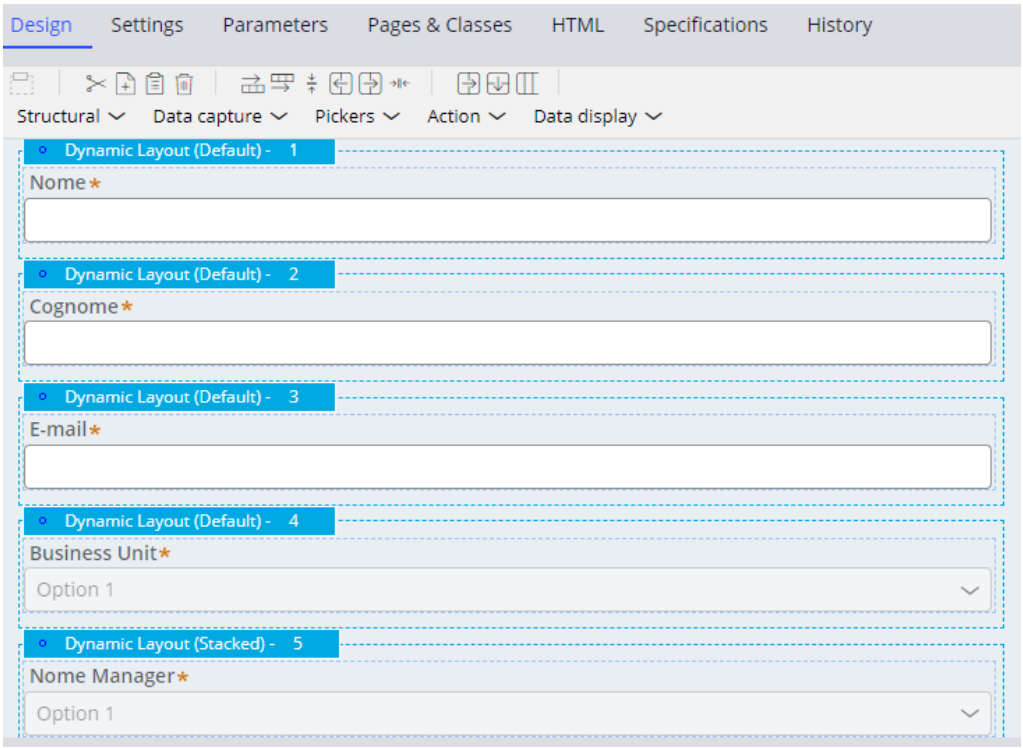


**Figura 32:** Section con i campi input per l'inserimento dati del richiedente

Per alcuni campi di input è stato richiesto di implementare qualche controllo o funzione specifica. Nel campo *E-mail* del richiedente, ad esempio, è stato inserito il controllo sull'indirizzo di posta elettronica, che deve necessariamente terminare per *@mycompany.it*. In caso contrario, l'applicazione non permette di continuare con il processo, mostrando il messaggio di errore "*Per favore inserisci una mail che abbia come dominio: @mycompany.it*" (vedi **Figura 33**). Altri controlli che sono stati inclusi riguardano il campo *Business Unit*, che deve essere abilitato solo dopo l'inserimento della e-mail, e il campo *Nome Manager*, la cui abilitazione deve avvenire solo una volta selezionata una BU, o se la BU inserita non è *BU_FINANCIAL*.

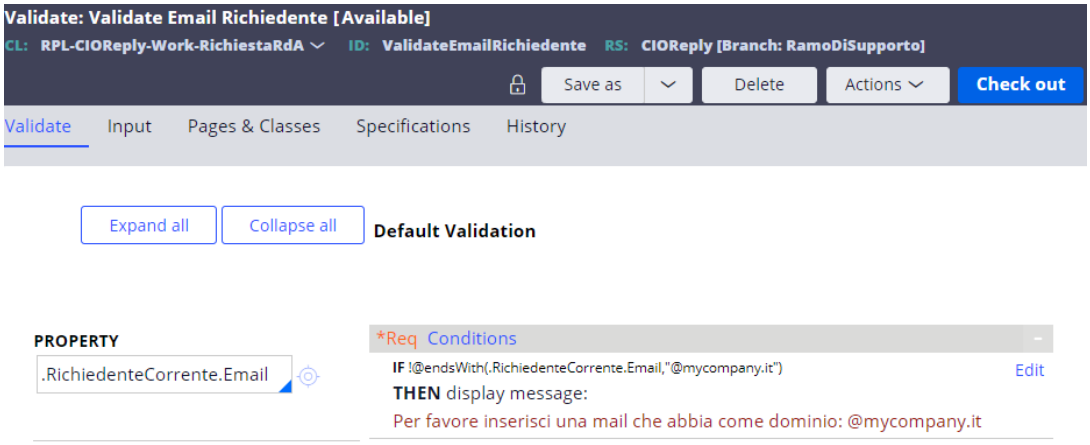


**Figura 33:** Controllo sul campo *E-mail*

In seguito, è stato gestito il caso in cui l'e-mail inserita dal richiedente coincida con l'indirizzo di posta elettronica di uno dei manager presenti nel database. Al riguardo, dopo aver compilato il campo *E-mail*, la dropdown list *Business Unit* deve essere pre-valorizzata con la BU associata al manager in questione, mentre la dropdown list *Nome Manager* deve risultare disabilitata. Per poter fare ciò, è stata utilizzata una *data transform*, uno strumento che serve per manipolare i dati.

| | Action | Target | Relation | Source |
|---|---|---|---|---|
| • 1 | Set | TmpPageManager | equal to | D_GETManager[Email: Param.email |
| • 2 | When | TmpPageManager.pxResultCount>0 | | |
| • 2.1 | Comment | L'email corrisponde a quella di un manager | | |
| • 2.2 | Set | Primary.BusinessUnit | equal to | TmpPageManager.pxResults(1).BU |
| • 2.3 | Set | Primary.disableManager | equal to | "true" |
| • 3 | Otherwise | | | |
| • 3.1 | Set | Primary.disableManager | equal to | "false" |
| • 3.2 | Set | Primary.BusinessUnit | equal to | "" |

**Figura 34:** Data transform *CheckManager*

La **Figura 34** mostra il funzionamento della data transform appena menzionata. Prima di tutto viene creata *TmpPageManager*, una temporanea *data page*, ovvero una pagina che preleva le informazioni da una fonte dati specificata e le conserva nella memoria cache. *TmpPageManager* contiene i dati di tutti i manager presenti nel database che hanno lo stesso indirizzo e-mail inserito nel campo *E-mail* del richiedente, i quali sono stati prelevati dalla data page esistente *D_GETManager*. Se questa pagina di dati temporanea ha un numero di record

maggiore di zero (condizione *When*), vuol dire che effettivamente l'indirizzo di posta inserito in input coincide con quello di un manager presente in database. Dunque, a questo punto, al campo *Business Unit* viene assegnato il valore della BU associato a quel manager (vedi step 2.2 della **Figura 34**). Successivamente viene impostato il valore della *Propoerty .disableManager* a *true*. Una Property è un elemento che fornisce nome e caratteristiche per i dati di un oggetto. La proprietà *.disableManager*, poi, è stata inserita nelle condizioni per disabilitare il campo *Nome Manager*, come è possibile notare nella **Figura 35**.

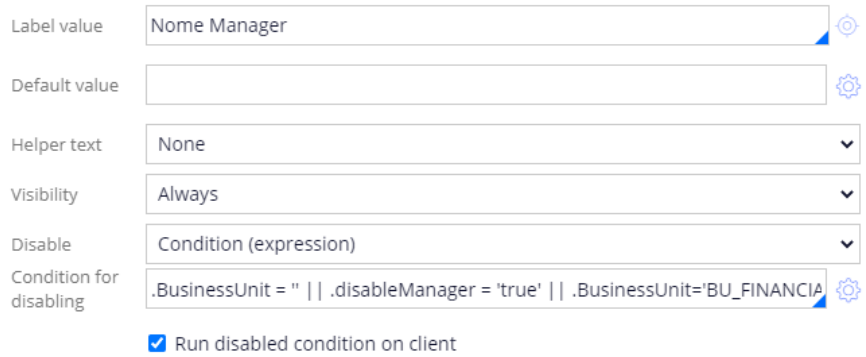


**Figura 35:** Condizioni per la disabilitazione della dropdown list *Nome Manager*

Nello step 1 della **Figura 34**, la data page di origine utilizza dei parametri in input (*D_GETManager[Email: Param.email]*). Tali parametri sono stati ricavati grazie ad una *report definition*, la quale consente di acquisire i dati restituiti da una query. Nella figura sottostante è presente la report definition a cui fa riferimento la data transform descritta precedentemente. La condizione F1 specifica che i manager sono filtrati in base all'e-mail passata come parametro.

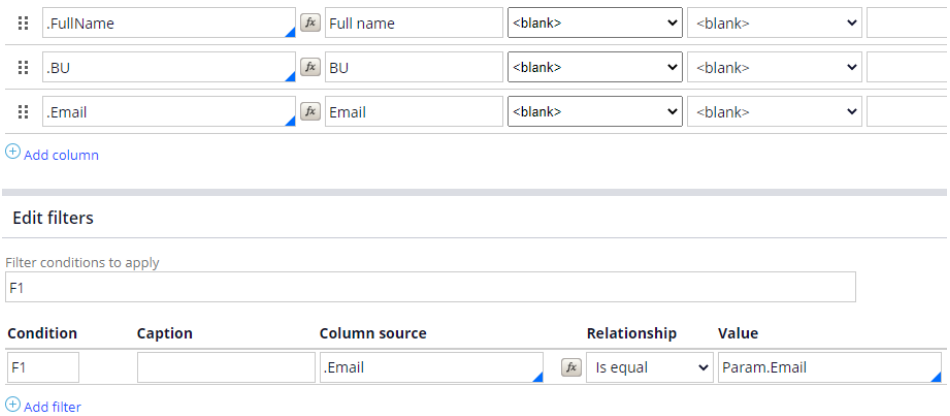


**Figura 36:** Report definition per filtrare i manager con e-mail come parametro

La data transform, poi, è stata inserita come *Action* in caso di *Change* sul campo *E-mail*. Ciò vuol dire che ogni volta che viene cambiato l'e-mail del richiedente è lanciata la data transform.

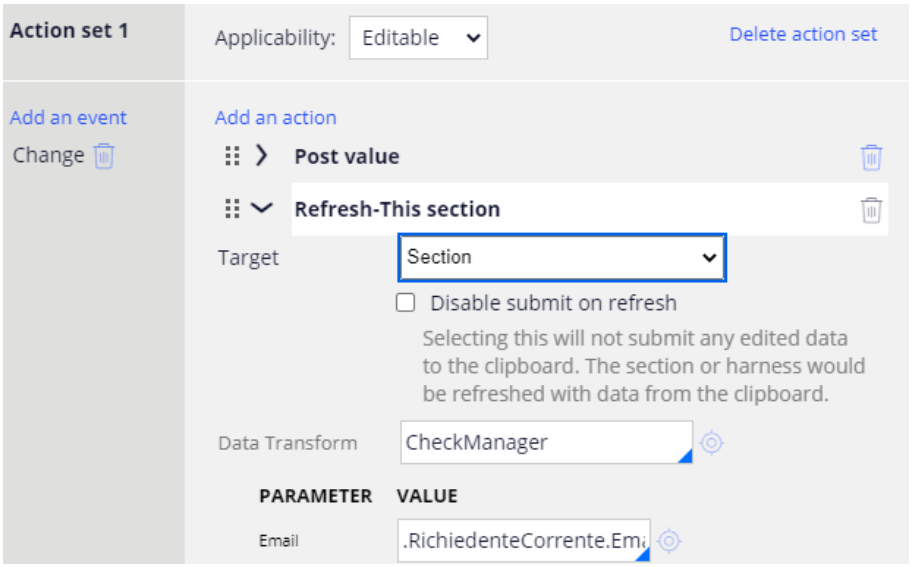


**Figura 37:** Data transform *CheckManager* inserita nell'*Action* su *Change* dell'E-mail

### 4.4.3 Aggiornamento *Cruscotto sottomissione RdA*: Inserimento dati del bene richiesto

La seconda fase del processo RdA riguarda l'inserimento dei dati del bene richiesto, implementato nello Stage *Raccolta informazioni*, comprendente l'unico Step *Tipo di Scelta*, che presenta tutte le informazioni necessarie da compilare.

**Figura 38:** Stage *Raccolta informazioni* per l'inserimento dati del bene richiesto

Per prima cosa, è stato modificato lo stato del processo, che da *New* ora è *Inserimento prodotti*. In questo modo, nella gestione delle varie richieste, i membri CIO possono essere al corrente dello stato delle richieste.

**Figura 39:** Stato del processo per lo Stage *Raccolta informazioni*

Questa pagina di navigazione consente di inserire informazioni relative al tipo di bene richiesto. Innanzitutto, è possibile selezionare il *Tipo di richiesta* tramite un menù a tendina, con opzioni come *Hardware*, *Software* o *Altra richiesta*. Dopodiché, a seconda della scelta sul tipo di richiesta, sono inseribili diversi dati, come la scelta della destinazione (con i valori già inseriti nella fase precedente di progetto), la scelta del fornitore, la tabella di dettagli dell'ordine e la scelta del destinatario in caso di richiesta *Hardware*, mentre è presente una *text area* per inserire manualmente i dettagli della richiesta qualora il tipo di richiesta sia *Software* o *Altra richiesta*.

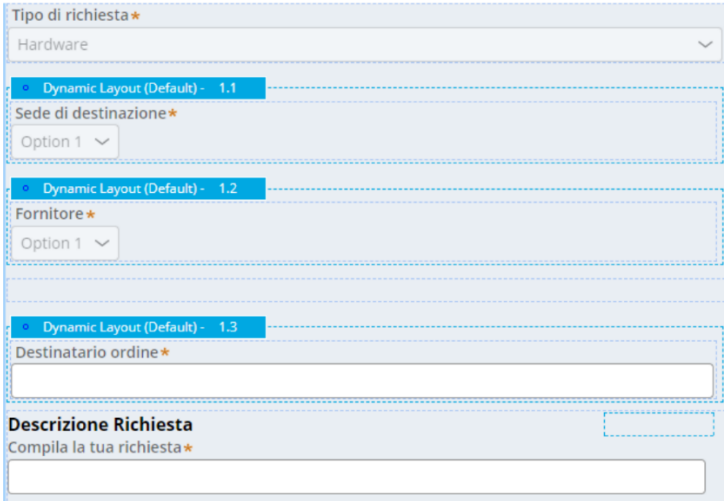


**Figura 40:** Section con i campi per l'inserimento dati del bene richiesto

Anche in questo caso sono state inserite delle condizioni per alcuni campi. Ad esempio, i campi caratterizzanti il tipo di richiesta *Hardware* sono stati resi visibili solamente se è stato selezionato *Hardware* nella dropdown list *Tipo di richiesta*. La stessa condizione vale anche per i campi di *Software* e *Altra richiesta*.

Per quanto riguarda la tabella con i dettagli dell'ordine per le richieste hardware, sono state creati tre diversi tipi di tabelle, in base ai diversi fornitori selezionabili (*Amazon*, *Esprinet* o *Altro fornitore*). Ognuna di queste tabelle ha diverse colonne, in base alle informazioni necessarie per ogni fornitore.

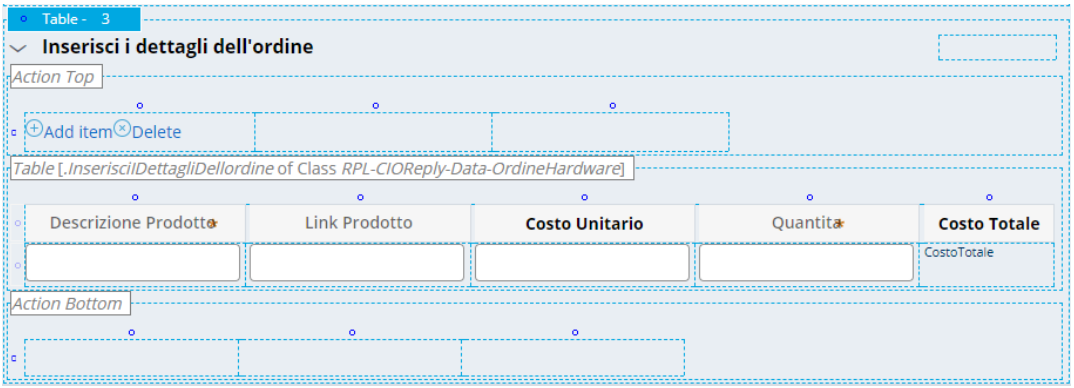


**Figura 41:** Tabella dettagli ordine *Amazon*

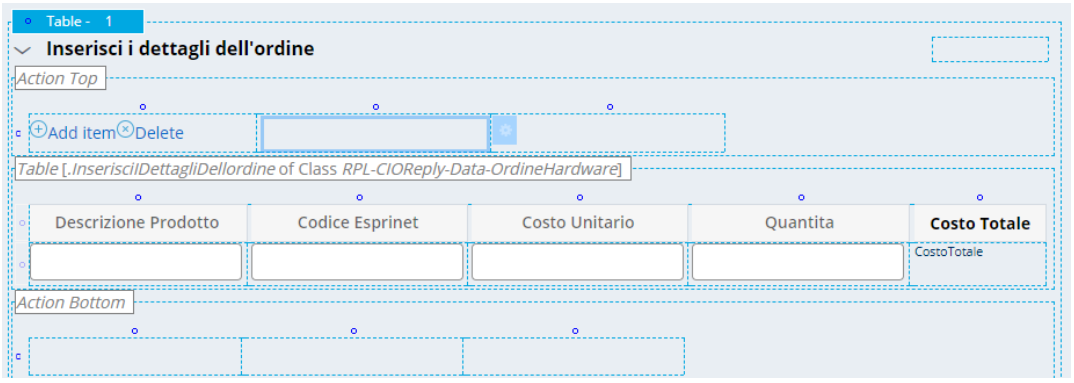


**Figura 42:** Tabella dettagli ordine *Esprinet*

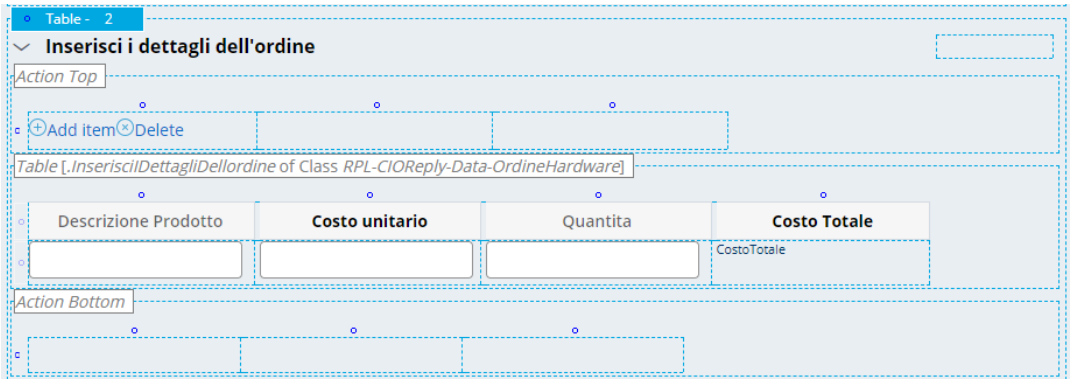


**Figura 43:** Tabella dettagli ordine *Altro fornitore*

La pagina di navigazione si aggiorna con il tipo di tabella desiderato in base alla scelta del fornitore. Le altre due tabelle, se non selezionate, non devono essere visibili e, inoltre, i campi della tabella devono essere obbligatoriamente compilati solamente per quella del fornitore scelto. Queste funzionalità sono state implementate andando a specificare opportune condizioni per tutte le tabelle. Infine, per poter utilizzare negli step successivi una stessa Property per i campi comuni delle tabelle (ad esempio *descrizione prodotto*), ogni volta che è cambiato qualche campo di input di una tabella, viene lanciata una data transform. Ogni cella della tabella ha Property diverse per ciascun fornitore, e la data transform consente di salvare i dati inseriti nella proprietà specifica della tabella nella proprietà comune da utilizzare nelle fasi successive.

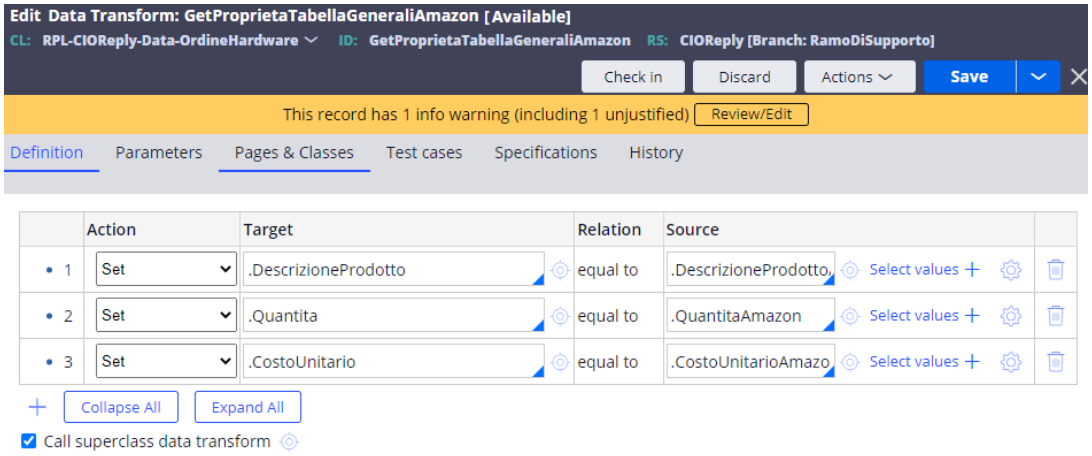


**Figura 44:** Data transform *GetProprietaTabellaGeneraliAmazon*

### 4.4.4 Aggiornamento *Cruscotto sottomissione RdA*: Revisione richiesta

Dopo aver completato l'inserimento di tutti i dati richiesti, il processo passa alla fase di revisione della richiesta, in cui è possibile controllare i dati inseriti negli step precedenti e, eventualmente, confermarli.

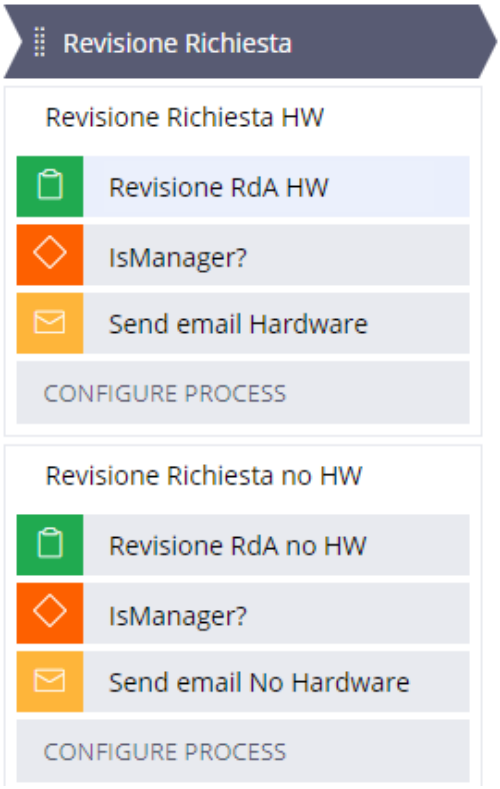


**Figura 45:** Stage *Revisione Richiesta*

La **Figura 45** mostra che lo stage *Revisione Richiesta* presenta al suo interno due processi diversi, a seconda che il *Tipo di richiesta* selezionato nello Stage precedente fosse *Hardware* oppure *Software/Altra Richiesta* (No Hardware).

All'interno di ogni processo sono presenti tre step. Il primo contiene il riepilogo di tutte le informazioni inserite negli step precedenti; il secondo è un blocco decisionale (*Decision*) che permette di saltare lo step successivo in caso in cui la richiesta è stata inserita da un manager (vedi **Figura 46**); il terzo step è lo

step automatico che invia una mail al manager selezionato nel primo Stage notificando l'avvenuta richiesta di RdA.

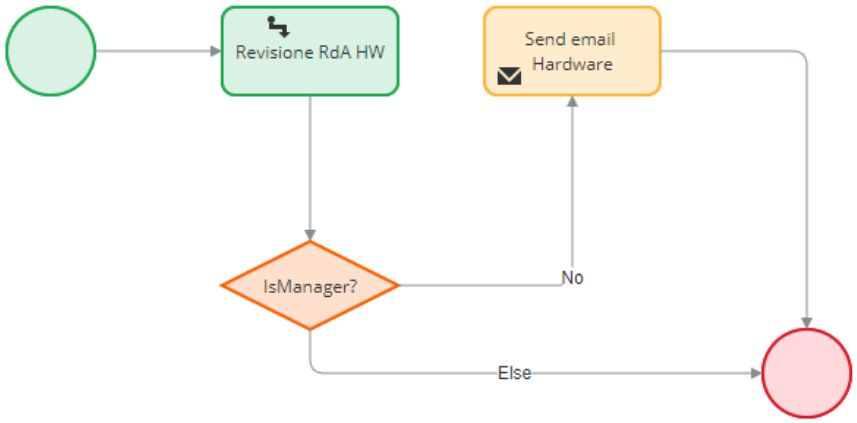


**Figura 46:** Process flow del processo *Revisione Richiesta HW*

A seconda che la richiesta riguardi un prodotto hardware o meno, quando sono confermati i dati inseriti, sono inviate due diverse e-mail, che hanno come destinatari il manager, il richiedente stesso e la lista di distribuzione di Blue CIO.

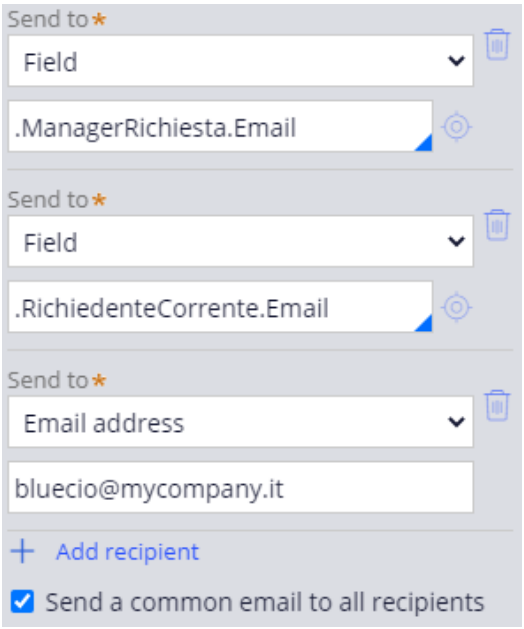


**Figura 47:** Destinatari e-mail per richiesta approvazione del manager

La **Figura 48** mostra il corpo della mail in caso di richiesta hardware. I campi inseriti tra "<< >>" sono le proprietà in cui sono state salvate le informazioni precedentemente raccolte.

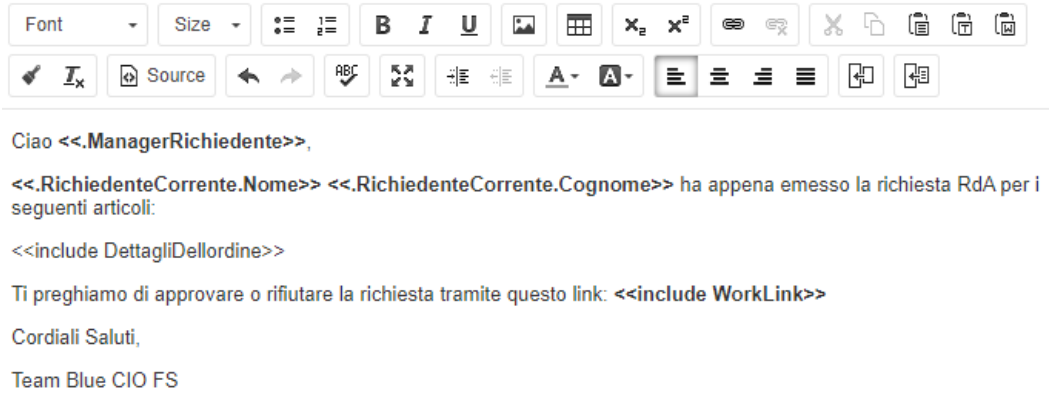


**Figura 48:** Corpo della mail in caso di tipo di richiesta *Hardware*

In particolare, <<*include DettagliDellOrdine*>> inserisce la tabella con i prodotti richiesti nel corpo della mail, mentre <<*include WorkLink*>> include un link che rimanda alla fase di approvazione.

Similmente accade anche per la mail in caso in cui il tipo di richiesta scelto sia *Software/Altra Richiesta*. L'unica differenza, in questo caso, è la presenza della proprietà <<*.CompilaLaTuaRichiesta*>>, che sostituisce la *rule* <<*include DettagliDellOrdine*>>.

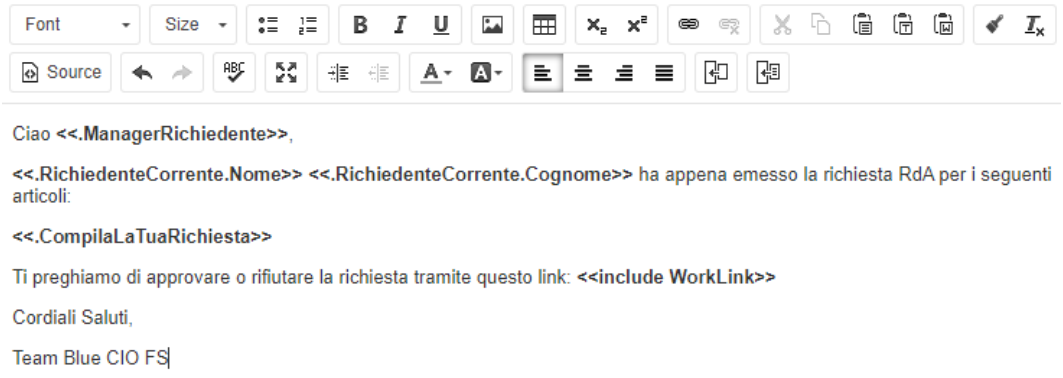


**Figura 49:** Corpo della mail in caso di tipo di richiesta *Software/Altra Richiesta*

Infine, dopo aver confermato i dati, viene mostrata a schermo una *Thank you page* tramite una *Modal Dialog* con due diversi messaggi a seconda che il richiedente sia un manager o meno. Questa funzionalità era già stata integrata, ma senza la distinzione tra richiedente manager o non manager e, inoltre, la modal dialog è stata spostata in uno step diverso rispetto alla precedente implementazione.

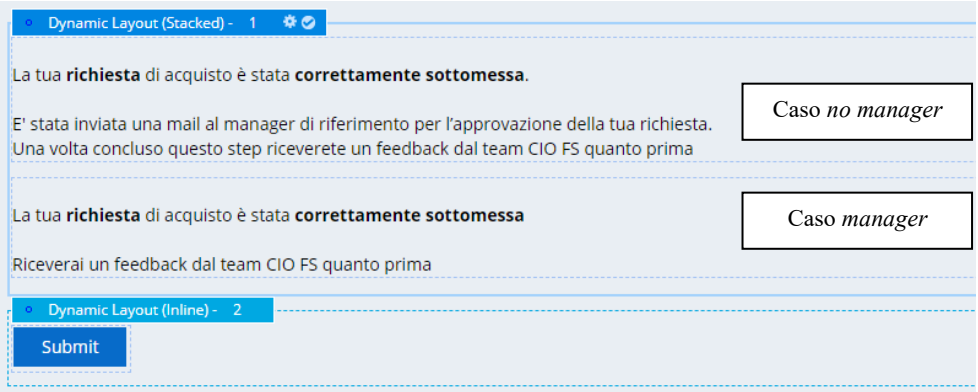


**Figura 50:** Testo della modal dialog *Thank you page*

### 4.4.5 Aggiornamento *Cruscotto sottomissione RdA*: Fase di approvazione

L'ultima fase del processo riguardante la sottomissione di RdA è quella di approvazione da parte del manager. Questa fase è composta da tre Stage. Il primo è lo Stage *Approvazione*, comprendente lo step *Approvazione RdA*, tramite cui il manager selezionato nella prima fase del processo è incaricato di approvare o rifiutare la richiesta di acquisto; il secondo è lo Stage *Approvata*, che è raggiunto nel caso i cui il manager approva la richiesta, ed è composto da tre Step automatici di creazione di un PDF di riepilogo (già implementato precedentemente), di invio mail di approvazione al richiedente e di notifica di approvazione al membro Blue CIO a cui è stata assegnata la gestione della RdA; l'ultimo Stage è *Approval Rejection*, quello raggiunto in caso di rifiuto della richiesta, in cui è inviata una mail di rifiuto al richiedente.

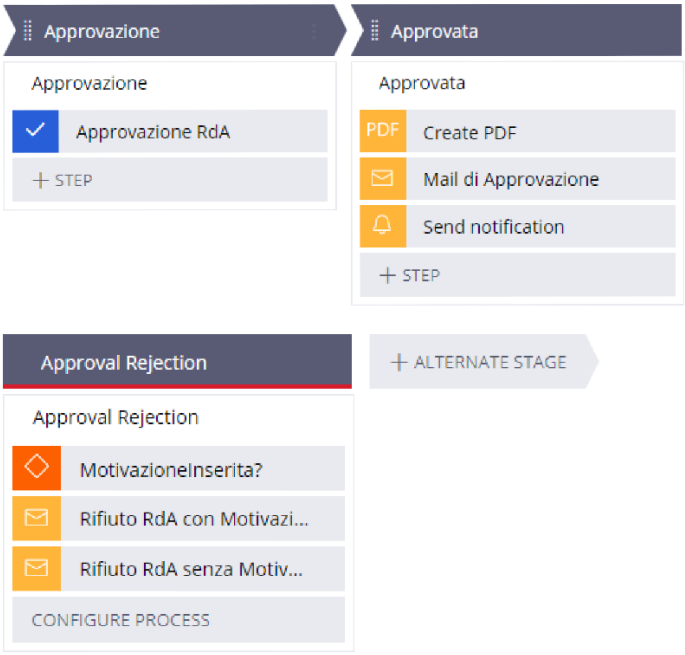


**Figura 51:** Stage della fase di approvazione

Nello Stage di approvazione è stata creata una *Section* in cui sono presenti un *radio button*, tramite cui è possibile scegliere se approvare o rifiutare la richiesta, ed un pulsante di conferma per convalidare la scelta effettuata. È presente, inoltre, una text area *Motivazioni*, in cui è possibile inserire opzionalmente le motivazioni per cui si è deciso di rifiutare la RdA. È stata implementata anche la condizione di visibilità di questa text area, impostata come visibile solamente quando viene selezionato *Rifiutato* nella scelta del radio button.

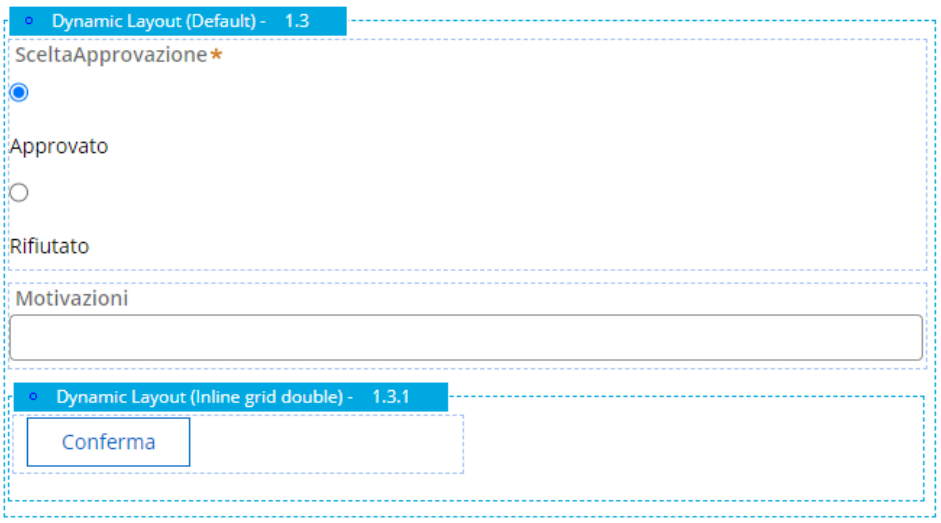


**Figura 52:** Section dello Step *Approvazione RdA*

Lo Step *Approvazione RdA* dovrebbe essere accessibile e messo nella coda di lavoro solo del manager a cui è stata assegnata l'approvazione della richiesta. Per implementare ciò è stato modificato il Data Type *Manager*, con l'inserimento di un ulteriore campo *Username*. In seguito, è stato inserito nello Step in questione un *routing* verso lo username del manager selezionato nello stage di inserimento dati richiedente, attraverso la proprietà *.ManagerRichiesta.Username*.

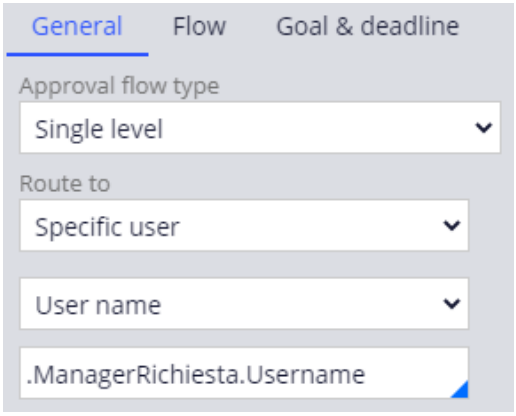


**Figura 53:** Routing dello Step *Approvazione RdA*

Il pulsante *Conferma* dello step di approvazione permette di procedere con lo Stage *Approvata* in caso di approvazione da parte del manager, altrimenti si conclude con lo Stage *Approval Rejection*. La **Figura 54** mostra come è stato impostato il pulsante di conferma per la decisione del flow da seguire.

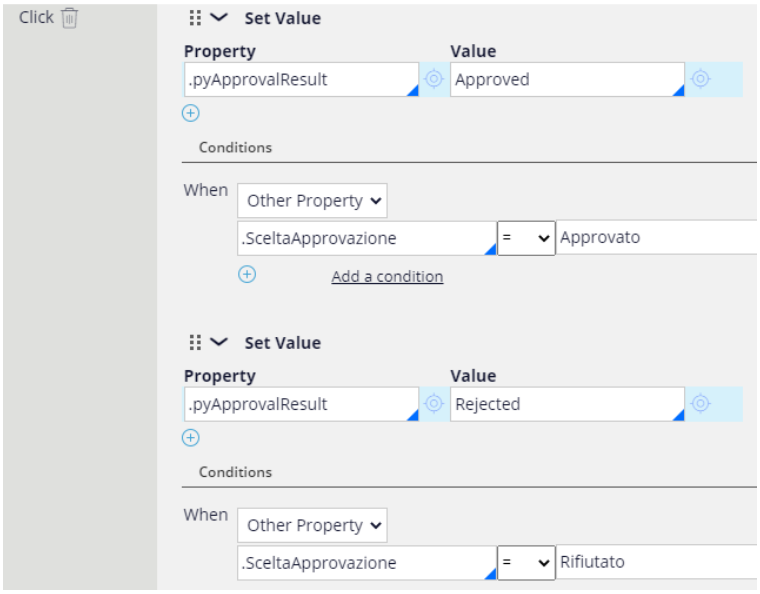


**Figura 54:** Impostazione flow seguito dal pulsante *Conferma* nello Stage *Approvazione RdA*

Nello Stage *Approvata* è stato inserito uno Step automatico *Send Notification*. Per gestire al meglio la coda di lavoro da parte dei membri del team Blue CIO, ogni membro della squadra deve essere associato ad una specifica BU. Per questo motivo, è stato creato un Data Type *Membro CIO FS* e i relativi record, oltre ad aggiornare il Data Type *Business Unit*, associando ad ognuno BU la mail di uno dei membri del team. Tramite una data transform inserita nello Stage *Create RdA*, poi, ogni volta che viene selezionata una BU differente, sono salvate le informazioni del membro CIO associato nella proprietà *.MembroCIOassociato*.

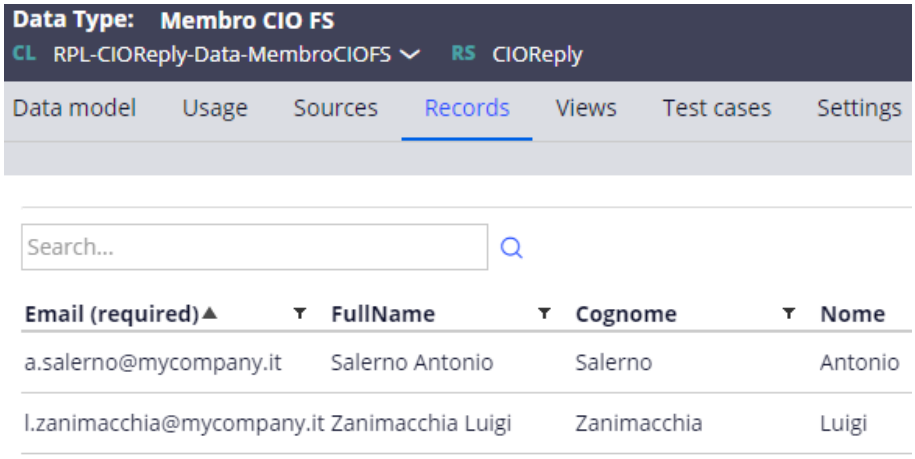


**Figura 55:** Data Type *Membro CIO FS* con i suoi record

Lo Step *Send Notification* nello Stage *Approvata*, in seguito, invia una notifica al Membro CIO associato alla BU selezionata all'inizio della richiesta, notificando l'avvenuta approvazione da parte del manager.

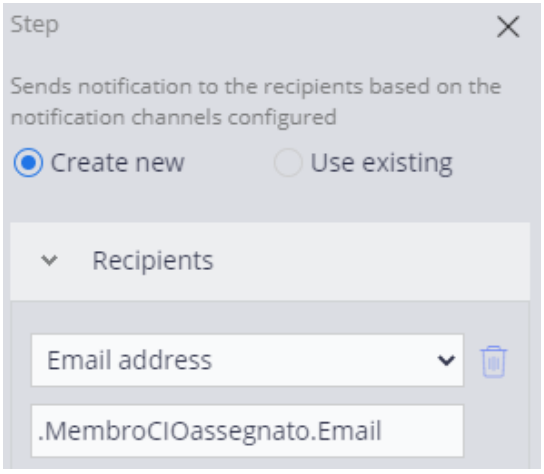


**Figura 56:** Invio della notifica al membro del team Blue CIO

Lo Stage *Approval Rejection*, infine, contiene un blocco di decisione che si dirama in due diversi tipi di mail di rifiuto. Se viene inserita una motivazione per il rifiuto, verrà inviata una mail con questa motivazione. Se invece non viene inserita alcuna motivazione, verrà inviata la mail dello step *Rifiuto RdA senza Motivazione*.

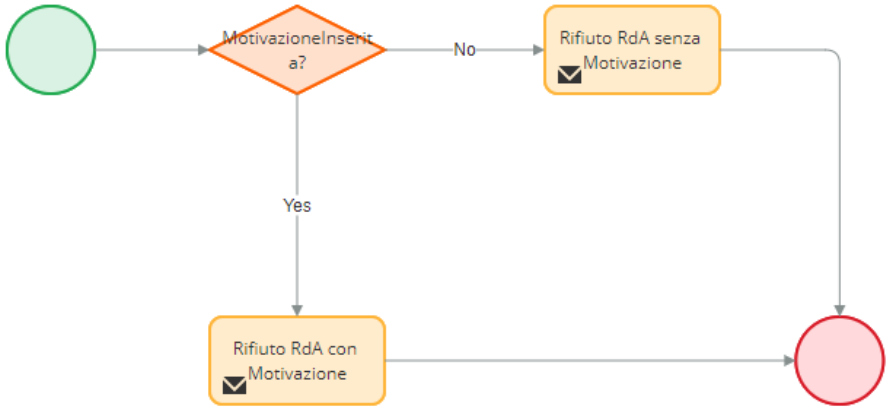


**Figura 57:** Process flow del processo *Approval Rejection*

La differenza principale tra le due mail, come è possibile notare dalla **Figura 56** e dalla **Figura 57** è data dalla presenza della Property *.MotivazioneRifiuto* nel caso in cui è stata inserita una motivazione del rifiuto.

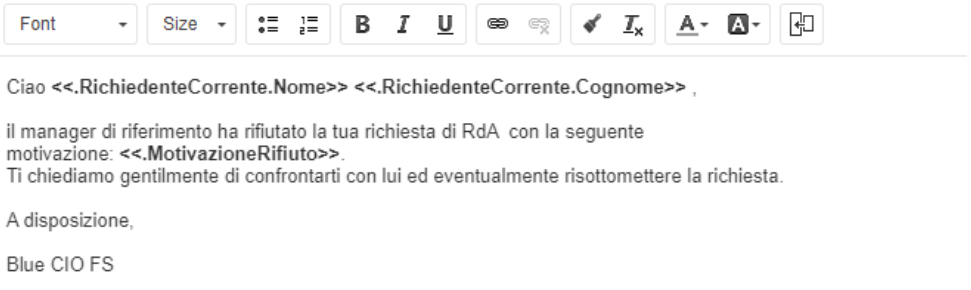


**Figura 58:** Corpo della mail di rifiuto con motivazione

**Figura 59:** Corpo della mail di rifiuto senza motivazione

### 4.4.6 *Cruscotto CIO*: Menu e Homepage

Dopo aver completato l'implementazione del processo di invio e approvazione di una richiesta di acquisto, si è iniziato a creare una piattaforma per i membri di Blue CIO che consente loro di gestire le richieste.

Innanzitutto, è stato creato un nuovo portale web tramite l'interfaccia *Portal* della sezione *Channels* di App Studio. Il portale utilizzato è quello denominato *User Portal*.

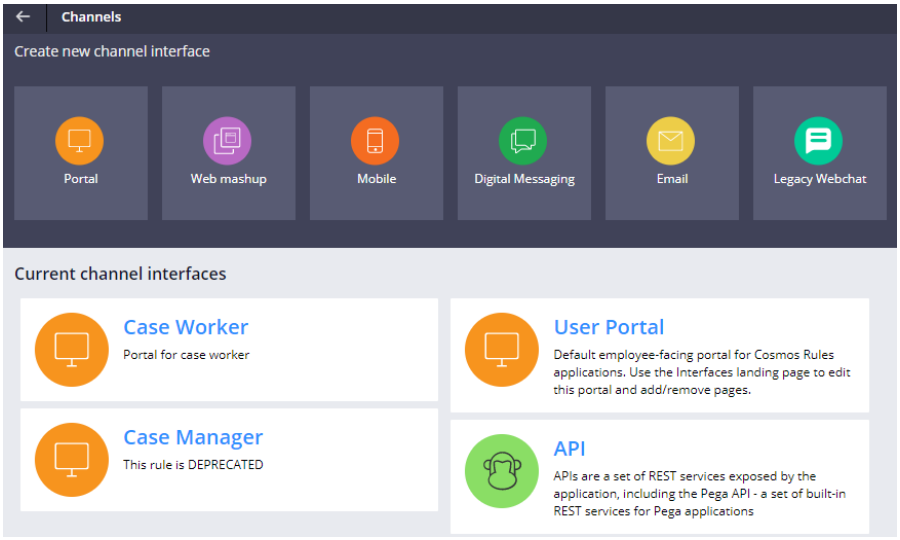


**Figura 60:** Interfaccia della sezione *Channels* di App Studio

In App Studio è anche possibile specificare alcune voci del menu, selezionabili da una lista fornita da Pega. In questo caso è stata inserita la sezione *Home*.

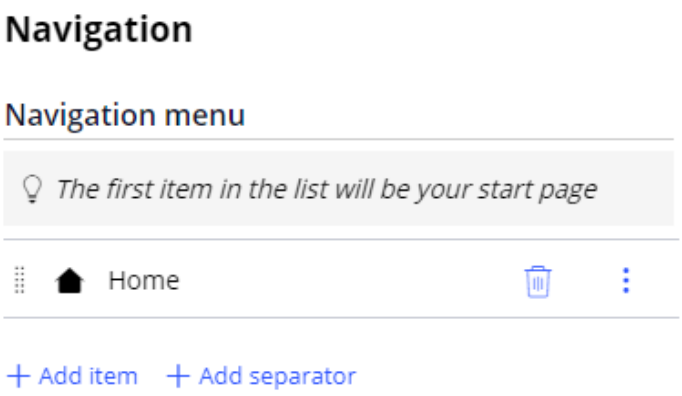


**Figura 61:** Voce *Home* del menu inserita da App Studio

È stato richiesto, però, di aggiungere altre due voci nel menu che non sono presenti nella lista. Per poterli inserire, è stata personalizzata la section *MainNavigation* in Dev Studio. Sono state inserite le due voci *Configurazione BU* e *Report*, a cui sono state aggiunte le icone del meccanismo (*pi-gear-solid*) e del chart (*pi-chart-bar-solid*).

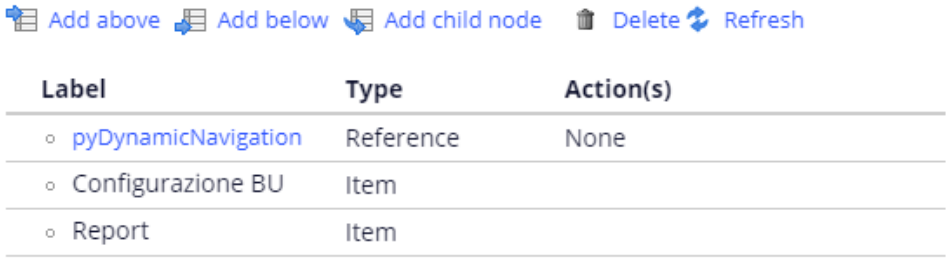


**Figura 62:** Voci del menu inserite nella section *MainNavigation* in Dev Studio

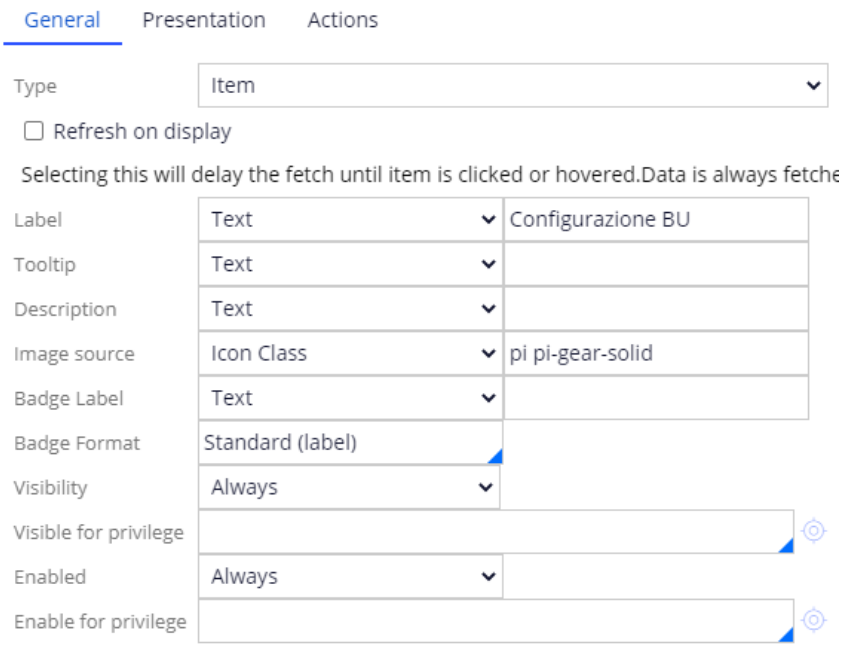


**Figura 63:** Item *Configurazione BU*

Successivamente, è stata implementata l'homepage del portale. La richiesta è stata quella di avere una homepage che presentasse la lista di tutte le richieste di acquisto sottomesse. Per fare ciò è stata creata una tabella che presenti varie informazioni sulle RdA sottomesse.

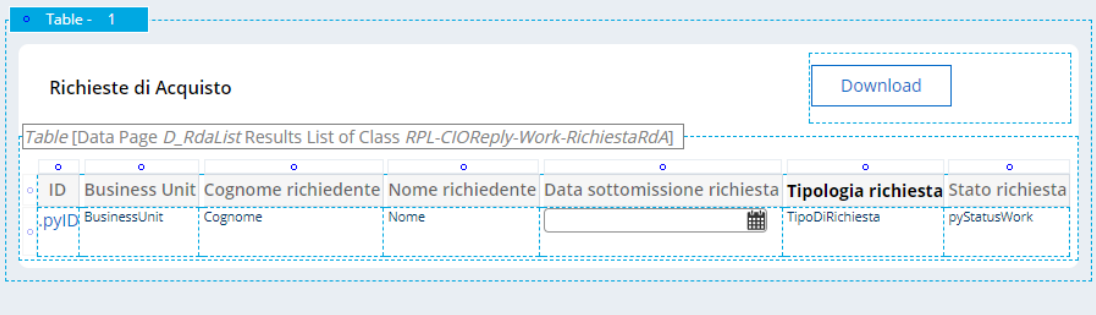


**Figura 64:** Tabella con la lista delle richieste di acquisto

Tramite ogni intestazione delle colonne della tabella è possibile anche filtrare le richieste, anche se sarebbe possibile, in sviluppi successivi, integrare dei filtri più funzionali per la gestione delle RdA.

Il pulsante *Download* della **Figura 64**, infine, permette di scaricare un file Excel in cui sono presenti tutte le richieste sottomesse. Il bottone scatena una *Activity*, tramite cui viene generato il file excel desiderato.

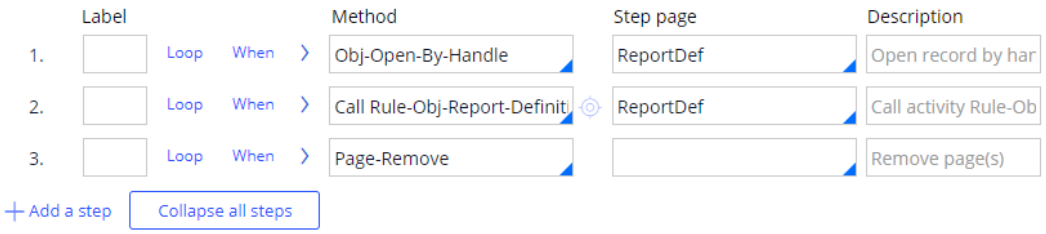


**Figura 65:** Activity *DownloadExcel*

# 5. Risultati del progetto

La piattaforma sviluppata nel progetto è stata realizzata per andare incontro a due necessità in particolare: quella di standardizzare il processo di richiesta di acquisto con una migliore gestione delle code di lavoro – *Fase 1* - e quello di fornire un portale ai membri del team Blue CIO per gestire le richieste (controllo richieste, gestione stati, ricezione o annullamento ordini, controllo costi…) – *Fase 2*. L'obiettivo dichiarato a inizio progetto era quello di provare a concludere la *fase 1* del progetto, e iniziare l'impostazione della *fase 2*. Entrambi gli obiettivi sono stati raggiunti. Ora è possibile utilizzare la piattaforma per la fase di sottomissione RdA, con un processo standardizzato, un flow preciso, controlli definiti e una coda di lavoro definita. È possibile per i membri Blue CIO, inoltre, accedere al portale di gestione delle RdA, con una homepage definita e la possibilità di scaricare un file excel con tutte le richieste.

In seguito, verranno mostrate alcune schermate in *front-end* sul risultato finale del lavoro svolto che è stato descritto nel capitolo precedente, presentando alcune funzionalità di cui si è parlato.

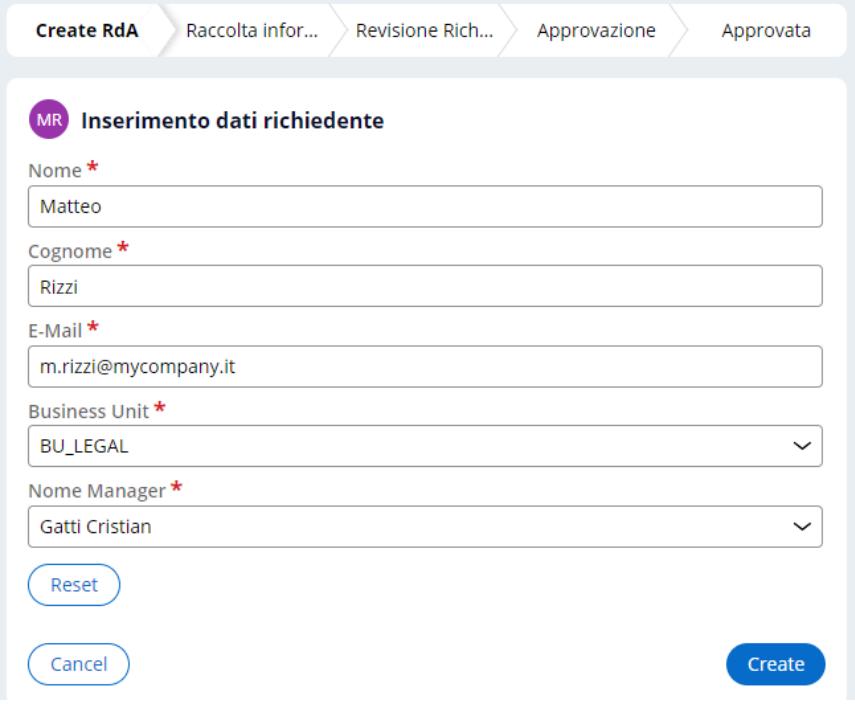


**Figura 66:** Esempio di inserimento dati richiedente

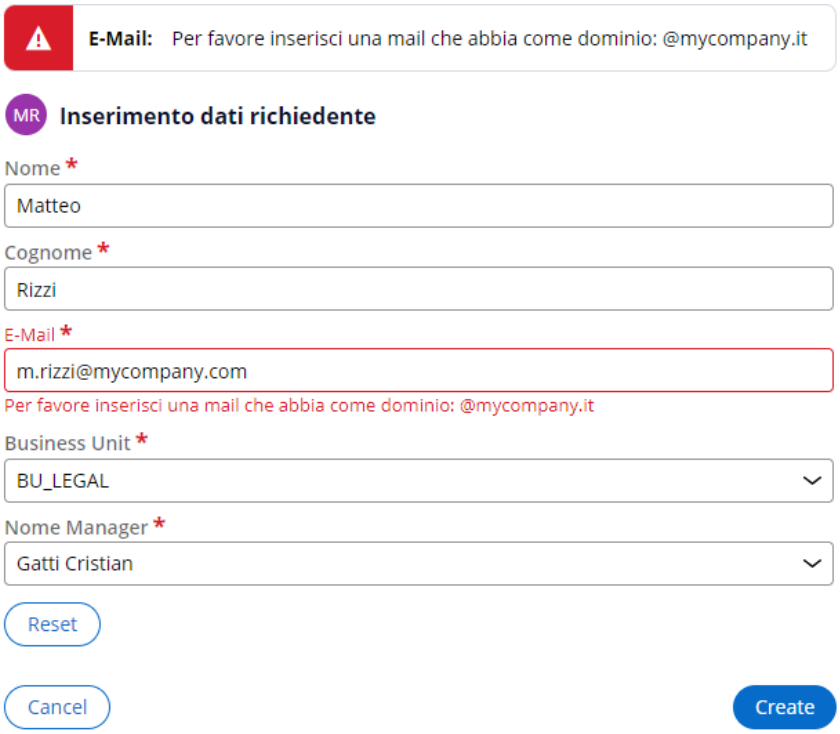


**Figura 67:** Esempio di inserimento dati richiedente con e-mail sbagliata

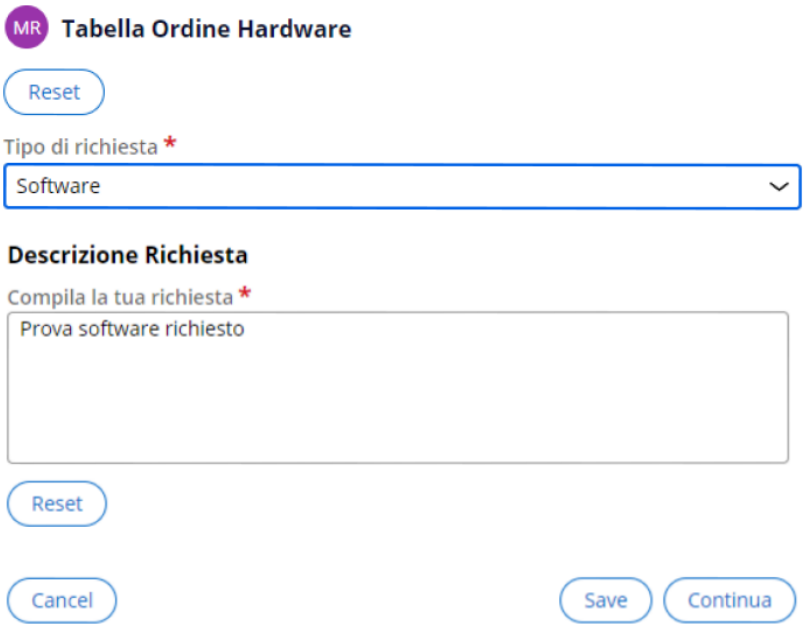


**Figura 68:** Esempio inserimento dati del bene richiesto - Software

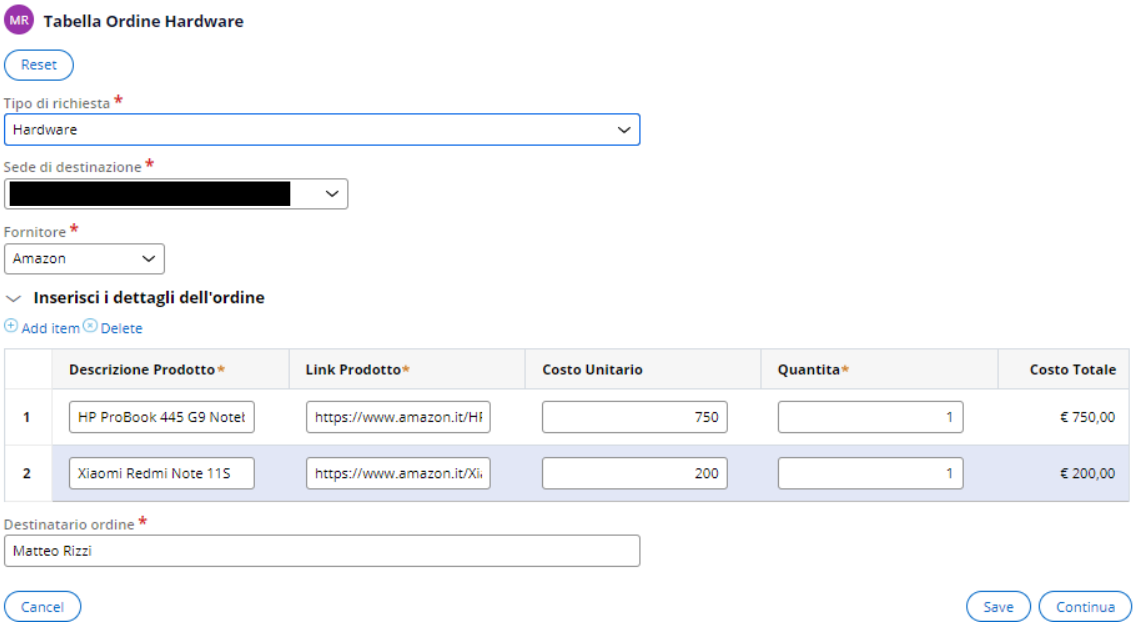


**Figura 69:** Esempio inserimento dati del bene richiesto – Hardware

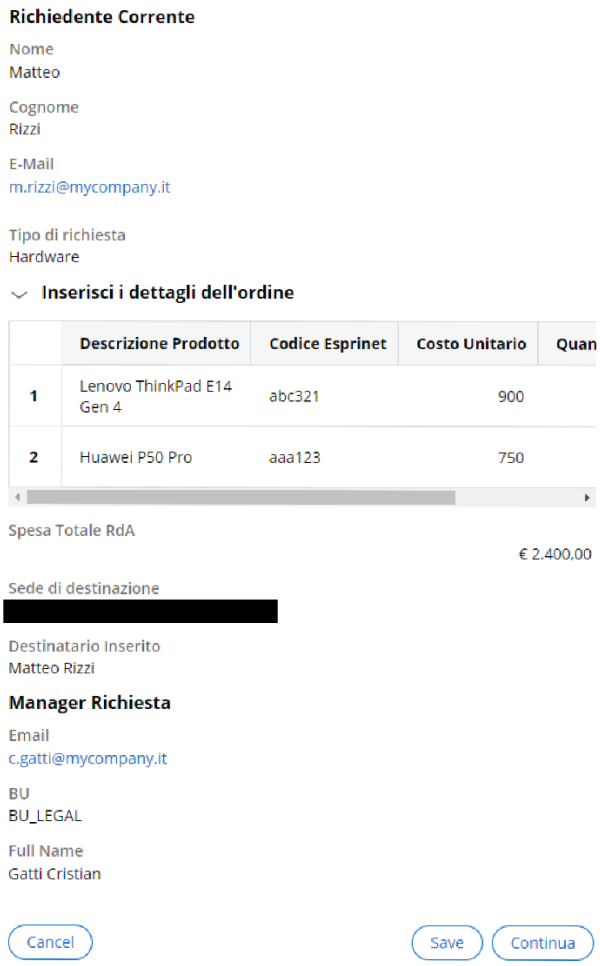


**Figura 70:** Esempio revisione richiesta - Hardware

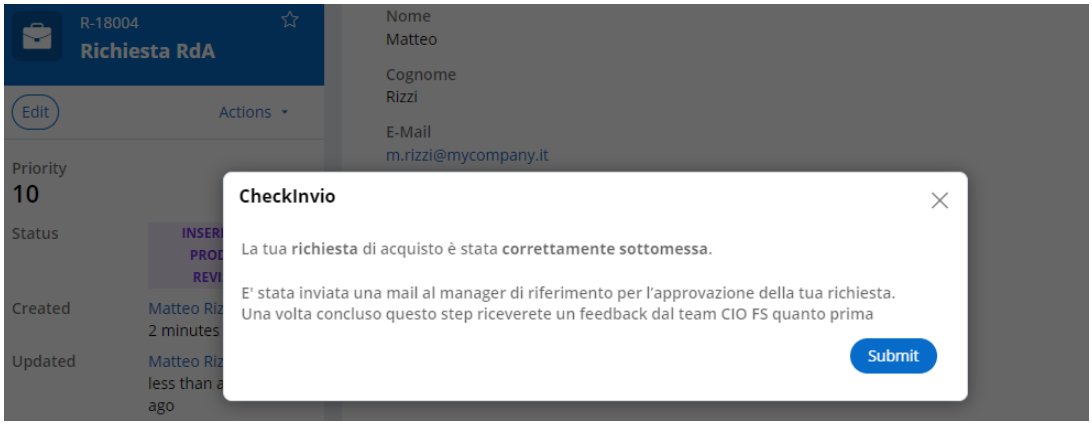


**Figura 71:** Modal dialog della Thank you page

Sent: 08/mar/23 09:52:52
From: default@PegaSample.com
To: bluecio@mycompany.it,m.rizzi@mycompany.it,c.gatti@mycompany.it
Subject: Richiesta di approvazione del manager

Ciao **Gatti Cristian**,

**Matteo Rizzi** ha appena emesso la richiesta RdA per i seguenti articoli:

**Inserisci i dettagli dell'ordine**

| | Descrizione Prodotto | Codice Esprinet | Costo Unitario | Quantita | Costo Totale |
|---|---|---|---|---|---|
| **1** | Lenovo ThinkPad E14 Gen 4 | abc321 | 900 | 1 | € 900,00 |
| **2** | Huawei P50 Pro | aaa123 | 750 | 2 | € 1.500,00 |

Ti preghiamo di approvare o rifiutare la richiesta tramite questo link: R-22008

Cordiali Saluti,

Team Blue CIO FS

**Figura 72:** Esempio di e-mail inviata al manager per l'approvazione - Hardware

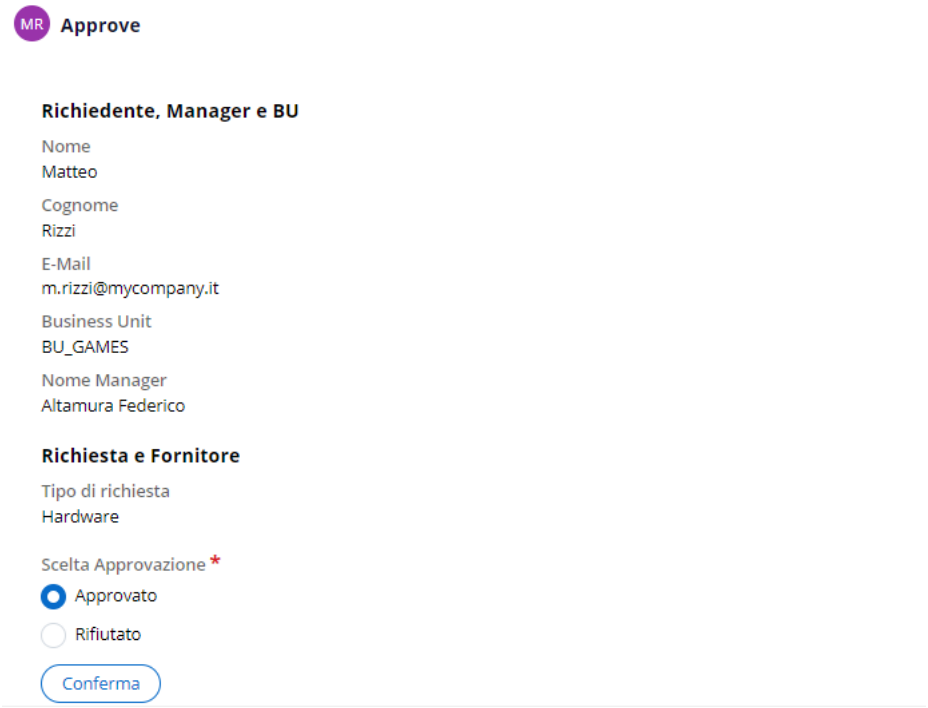


**Figura 73:** Esempio fase approvazione del manager

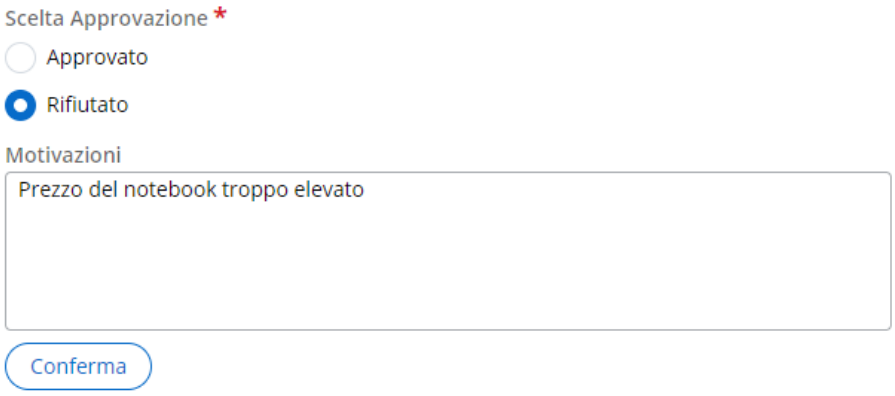


**Figura 74:** Esempio fase approvazione del manager - Richiesta rifiutata

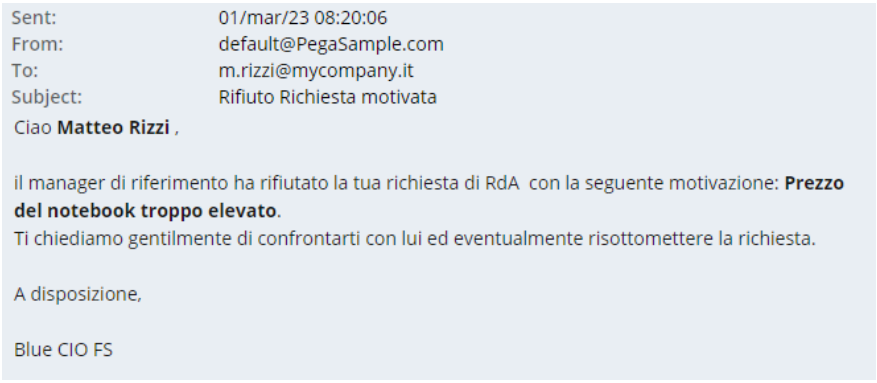


**Figura 75:** Esempio mail di rifiuto inviata al richiedente

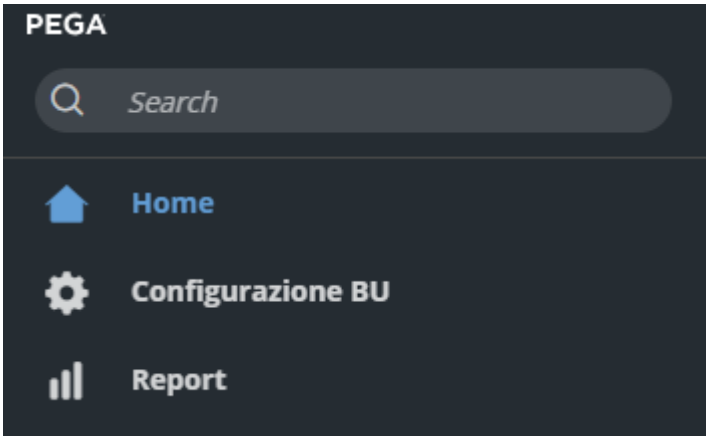


**Figura 76:** Menu del portale *Cruscotto CIO*

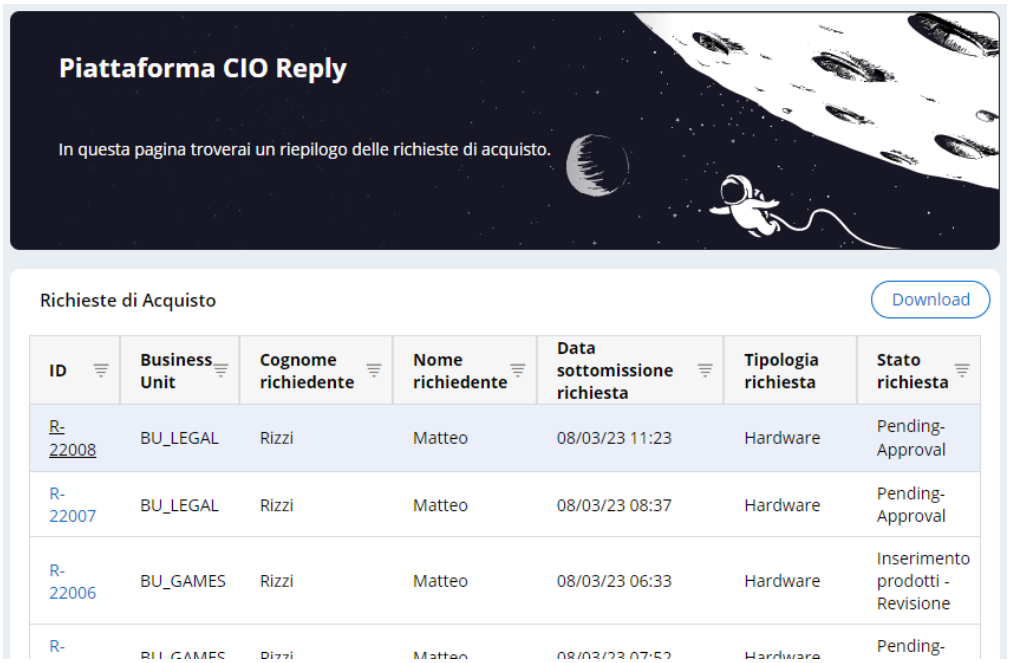


**Figura 77:** Interfaccia dell'homepage del *Cruscotto CIO*

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Case ID | Business Unit | Work status | Business Unit | Tipo di richiesta | Nome | Cognome | Update Date/Time |
| 2 | R-22008 | BU_LEGAL | Pending-Approval | BU_LEGAL | Hardware | Matteo | Rizzi | 08/03/23 11:23 |
| 3 | R-22007 | BU_LEGAL | Pending-Approval | BU_LEGAL | Hardware | Matteo | Rizzi | 08/03/23 08:37 |
| 4 | R-22006 | BU_GAMES | Inserimento prodotti - Revisione | BU_GAMES | Hardware | Matteo | Rizzi | 08/03/23 06:33 |
| 5 | R-22005 | BU_GAMES | Pending-Approval | BU_GAMES | Hardware | Matteo | Rizzi | 08/03/23 07:52 |
| 6 | R-22004 | BU_GAMES | Pending-Approval | BU_GAMES | Hardware | Matteo | Rizzi | 08/03/23 07:30 |
| 7 | R-22003 | BU_GAMES | Pending-Approval | BU_GAMES | Hardware | Matteo | Rizzi | 08/03/23 07:21 |
| 8 | R-22001 | BU_GAMES | Inserimento prodotti - Revisione | BU_GAMES | Hardware | Matteo | Rizzi | 08/03/23 05:29 |
| 9 | R-20001 | BU_AUDIT | Pending-Approval | BU_AUDIT | Hardware | Matteo | Rizzi | 06/03/23 07:07 |
| 10 | R-19003 | BU_GAMES | Resolved-Completed | BU_GAMES | Hardware | Roberto | Cutaia | 01/03/23 20:37 |
| 11 | R-19002 | BU_GAMES | Resolved-Rejected | BU_GAMES | Software | Roberto | Cutaia | 01/03/23 20:34 |
| 12 | R-19001 | BU_GAMES | Resolved-Rejected | BU_GAMES | Hardware | Roberto | Cutaia | 01/03/23 20:14 |
| 13 | R-18005 | BU_GAMES | Resolved-Rejected | BU_GAMES | Hardware | Matteo | Rizzi | 01/03/23 09:50 |
| 14 | R-18004 | BU_LEGAL | Pending-Approval | BU_LEGAL | Software | Matteo | Rizzi | 01/03/23 09:45 |
| 15 | R-18001 | BU_GAMES | Resolved-Completed | BU_GAMES | Hardware | Giovanni | Bianchi | 01/03/23 09:12 |

**Figura 78:** File excel generato dal pulsante *Download* dell'homepage

I benefici apportati all'azienda alla fine del progetto sono di efficienza e di risparmio di tempo. Questi vantaggi sono rivolti sia a chi crea una richiesta, in quanto non deve pensare a cosa inserire nel corpo della e-mail di richiesta, bensì deve solamente seguire il processo guidato e inserire le informazioni richieste, ma anche e soprattutto per i membri del team Blue CIO, in quanto non devono perdere tempo nel controllare se tutte le informazioni necessarie sono state inserite, e non devono, eventualmente, scrivere al richiedente per le eventuali informazioni omesse. Ciò porta ad un efficientamento delle tempistiche per il processamento delle RdA.

È stato possibile, poi, dividere l'assegnazione delle richieste per ogni membro Blue CIO, in modo da gestire meglio le code di lavorazione e da avere la divisione delle richieste in base alla Business Unit di appartenenza del manager che ha approvato la richiesta in maniera chiara ed univoca.

Con l'impostazione della *fase 2*, invece, si è iniziato ad intravedere il beneficio di una visione d'insieme ed una centralizzazione dei controlli nella gestione delle richieste, grazie all'homepage che permette di visualizzare tutte le richieste, ai filtri che permettono di visualizzare solo un determinato tipo di richieste – che però sono migliorabili tramite filtri personalizzati, e alla possibilità di scaricare un file Excel con le richieste senza doverle inserire a mano, come invece era d'uso fare precedentemente.

I costi sostenuti per la realizzazione del progetto non sono stati giudicati molto rilevanti. In generale, le tre fonti di costo maggiore sono: costi di gestione del progetto, costi di supporto tecnico e costi legati al mantenimento della macchina virtuale utilizzata per lo sviluppo della piattaforma. In ogni caso, questi costi sono stati valutati come inferiori rispetto ai benefici dell'applicazione.

# 6. Conclusioni

L'elaborato di tesi ha posto l'attenzione su vari aspetti che negli ultimi anni si stanno rivelando sempre di maggiore importanza. L'argomento descritto nel primo capitolo, non oggetto del progetto effettuato in azienda, è stato utile per la società Blue Reply Srl per poter comprendere le potenzialità delle piattaforme di Real Time Interaction Management. L'azienda, infatti, ha iniziato, tramite vari progetti di tesi, uno studio sul Customer Decision Hub di Pega, dei suoi potenziali utilizzi e dei vantaggi che può portare. Lo scopo portato a termine nella tesi è stato quello di individuare un ambito di applicazione della piattaforma - quello del marketing relazionale, di capire come si può avere un vantaggio nell'ambito indicato - attraverso le Next-Best-Action, e come può essere utilizzato il Customer Decision Hub nel contesto. Inoltre, è stato individuato il vantaggio economico riscontrato dall'utilizzo del CDH ed è stato effettuato un confronto tra le principali piattaforme che offrono questo servizio, riscontrando come la piattaforma di Pega, di cui Blue Financial Services possiede la licenza, sia attualmente la vera leader in questo mercato.

Per quanto riguarda il progetto svolto in azienda, invece, sono stati già evidenziati nel capitolo precedente gli obiettivi raggiunti come risultato finale della piattaforma, completa e soddisfacente per quanto riguarda la *fase 1*, mentre è stata avviata l'impostazione per la *fase 2*.

Nell'ultimo Sprint Retrospective è stato analizzato non solo l'andamento dello sprint, ma in generale anche dell'intero progetto. In generale, l'applicazione della metodologia Agile è stata efficace. Le User Stories definite dal Product Owner sono state chiare e ben definite e la collaborazione tra i due Citizen Developers è stata positiva, grazie anche al supporto dell'azienda e alla buona organizzazione generale del progetto. L'adozione del metodo Agile ha anche permesso di poter avere sempre una piattaforma funzionante, anche se non interamente completata, oltre a ridurre il time-to-market. Inoltre, grazie al supporto degli esperti della piattaforma Pega Platform è stato possibile imparare a padroneggiare le

funzionalità offerte dalla piattaforma di sviluppo low-code e a sfruttarne al meglio le potenzialità. Ciò ha portato ad ottenere risultati alla fine di ogni Sprint, sia dal punto di vista delle funzionalità, che da quello grafico, rendendo l'esperienza gratificante per gli sviluppatori.

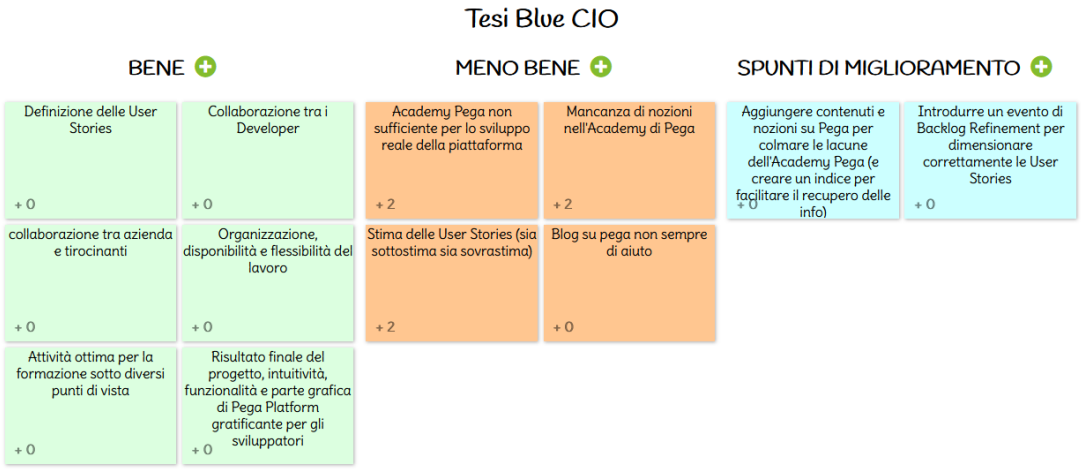


**Figura 79:** Sprint Retrospective finale sul progetto

Tra i limiti del progetto, invece, è stata individuata una non totale adeguatezza dell'Academy di Pega per lo sviluppo in low-code su Pega Platform. Il corso di quasi quaratacinque ore offerto dalla società di software, infatti, aiuta ad avere delle nozioni di base, e tramite le *Challenge* è stato possibile avere un primo approccio pratico alla piattaforma. Questo, però, non è bastato ad affrontare un progetto reale, e i primi passi mossi nel progetto sono stati disorientanti e difficili per gli sviluppatori, ed è servito un supporto ulteriore degli esperti della piattaforma per avere una padronanza delle funzionalità. Come miglioramento di questo problema, è stato proposto di aggiungere contenuti e nozioni complementari all'Academy. Un altro limite riscontrato è stata una stima iniziale delle User Stories non perfetta, per questo è stato proposto di introdurre l'evento di Backlog Refinement per poter dimensionare correttamente le User Stories.

L'avviamento della *fase 2* della piattaforma apre sicuramente la strada ad ulteriori sviluppi del progetto, con nuove implementazioni che possono essere attuate per migliorare la gestione delle richieste di acquisto da parte dei membri

del team Blue CIO. Un tool molto utile, ad esempio, può essere quello di una pagina di report delle spese: in questo modo, sarà possibile capire quale Business Unit spende maggiormente in hardware e software e quali sforano il budget disponibile, oltre a poter osservare quale fornitore è quello verso cui si effettuano più richieste o si tende a spendere maggiormente. Inoltre, può essere implementata la funzionalità di controllo delle sottomissioni degli ordini, per capire lo stato degli stessi, oppure per avere una miglior visione di ciò che è stato fatto. Sarebbe possibile anche poter gestire le urgenze per quelle richieste che magari hanno un bisogno prioritario di essere processate, perché particolarmente necessarie.

Per poter implementare un effettivo invio delle e-mail automatiche presentate nei capitoli precedenti, infine, sarebbe necessario configurare una casella di posta (ad esempio quella di Gmail) collegata a Pega.

In conclusione, il progetto di tesi ha mostrato come è stato possibile, anche per sviluppatori non professionisti, realizzare una applicazione tramite una piattaforma low-code, offrendo anche funzionalità e controlli più complessi, in tempi brevi, e lasciando, inoltre, le porte aperte per ulteriori sviluppi, potendo garantire una risoluzione ancor più pertinente e di supporto al problema affrontato.

# Lista degli acronimi

- **AI:** Artificial Intelligence
- **API:** Application Programming Interface
- **BU:** Business Unit
- **CBA:** Commonwealth Bank of Australia
- **CDH:** Customer Decision Hub
- **CEE:** Customer Engagement Engine
- **CIO:** Chief Information Officer
- **CLV:** Customer Life Value
- **CRM:** Customer Relationship Management
- **CX:** Costumer Experience
- **ESB:** Enterprise Service Bus
- **IoT:** Internet of Things
- **IT:** Information Technology
- **KPI:** Key Performance Indicators
- **LCDP:** Low-Code Development Platform
- **NBA:** Next Best Action
- **NBO:** Next Best Offer
- **NBX:** Next Best Experience
- **NPV:** Net Present Value
- **PV:** Present Value
- **RdA:** Richiesta di Acquisto
- **ROI:** Return On Investment
- **RPA:** Robotic Process Automation
- **RTIM:** Real-Time Interaction Management
- **SOA:** Service Oriented Architecture
- **TEI:** Total Economic Impact

# Bibliografia


[1] **RYMER John**. *Why You Need To Know About Low-Code, Even If You're Not Responsible For Software Delivery* [Online]

[2] **SYDLE**. *Low Code: Why Is It Essential for Any Business?* [Online]

[3] **MEDALLIA**. *What is Real-Time Interaction Management (RTIM) – and Why is Everyone Talking About It.* [Online]

[4] **WARNER Rusty**. *Invest In Real-Time Interaction Management To Elevate Your Long-Term Customer Experience Strategy.* [Online]

[5] **MILHEIRO João Luís Trindade**. *Next Best Action – A Data-Driven Marketing Approach*. 2019

[6] **WALKER Rob, NOLAN Matt**. *Crossing The Chasm: From Traditional Marketing To One-To-One Customer Engagement*. 2017

[7] **PEGA Academy**. *Cross-Sell on the Web* [Online]

[8] **PEGA Documentation**. *Customer Decision Hub 8.8* [Online]

[9] **DAVENPORT Thomas H., DALLEMULE Leandro, LUCKER John.** *Know What Your Customers Want Before They Do.* 2011

[10] **WALKER Rob, NOLAN Matt**. *Crossing The Chasm: From Traditional Marketing To One-To-One Customer Engagement*. 2021

[11] **FADZEYEVA Julia, DESAI Sanitra**. *The Total Economic Impact™ of Pega Customer Decision Hub.* 2020

[12] **ROGERS Jess.** *How Artificial Intelligence Is Changing The Face Of Banking.* 2022

[13] **CROZIER Ry.** *CBA System Suggests 20M Customer Conversations A Day.* [Online]

[14] **PEGA.** *CBA: Enabling Meaningful Customer Conversations.* [Online]

[15] **WARNER Rusty.** *The Forrester Wave: Real-Time Interaction Management, Q2 2022.* 2022

[16] **MICROSOFT POWERAPPS.** *Sviluppo Di App Low-Code E No-Code A Confronto.* [Online]

[17] **PEGA Academy**. *System Architect* [Online]

[18] **PEGA Documentation**. *Pega Platform 8.8.* [Online]

[19] **TREMOLADA Luca.** *Low Code o No Code? La Programmazione Per Tutti Conquista Aziende e Metaversi.* 2022

[20] **DAHLBERG Daniel.** *Developer Experience Of A Low-Code Platform: An Exploratory Study.* 2020

[21] **REDHAT.** *Intelligent Process Automation And The Emergence Of Digital Automation Platforms.* 2018

[22] **ZHAOHANG Yan.** *The Impacts Of Low/No-Code Development On Digital Transformation And Software Development.* 2021

[23] **MONDAYBLOG.** *Build Custom Apps Fast -2023: Guide To Low-Code Platforms.* [Online]

[24] **SIROTNAK Casey, DESAI Sanitra.** *The Total Economic Impact™ Of Pega Platform For Low Code.* 2020

[25] **VINCENT Paul, IIJIMA Kimihiko, LEOW Adrian, WEST Mike, MATVITSKYY Oleksandr.** *Magic Quadrant for Enterprise Low-Code Application Platforms.* 2022

[26] **FAGARASAN Cristian et al.** *Agile, waterfall and iterative approach in information technology projects.* 2021. s.l.: 1 IOP Conf. Ser.: Mater. Sci. Eng. 1169 012025

[27] **SENERATH Udesh S.** *Waterfall Methodology, Prototyping and Agile Development.* 2021

[28] **HÜBNER Felix**. *Slides Del Corso di Project Management: Agile Project Management. Think Different*. 2021. s.l.: Karlsruhe Institute of Technology – VSE Group

[29] **BECK Kent, BEEDLE Mike, CUNNINGHAM Ward et al.** *Manifesto Per Lo Sviluppo Agile Di Software.* [Online] Disponibile all'indirizzo: http://www.agilemanifesto.org/iso/it/manifesto.html

[30] **SCHWABER Ken, SUTHERLAND Jeff.** *The Scrum Guide. The Definitive Guide To Scrum: The Rules Of The Game.* 2020

[31] **DEEMER Pete, BENEFIELD Gabrielle, LARMAN Craig, VODDE Bas.** *A Lightweight Guide To The Theory And Practice Of Scrum. Version 2.0*. 2012

[32] **ZUCCARELLA Veronica.** *Agile Project Management per la gestione di progetti nell'industria delle costruzioni.* 2017

[33] **INGROSSO Alessandro.** *Quali sono i pilastri e le cerimonie di Scrum?*. 2018

[34] **DEMARTINI Claudio Giovanni, GATTESCHI Valentina.** *Slides Del Corso di Gestione Dell'Innovazione E Sviluppo Prodotto ICT*. 2021. s.l.: Politecnico di Torino

[35] **PEGA Academy.** *Pega Express Delivery.* [Online]

[36] **BLUE Reply.** *Tesi: Sviluppo Agile Di Una Applicazione Low-Code.* [Online]

# Ringraziamenti