



UNIVERSITÀ DI TRENTO

Dipartimento di Ingegneria e Scienza dell'Informazione

Corso di Laurea in
Informatica

Elaborato finale

Titolo

Sottotitolo (alcune volte lungo - opzionale)

Supervisore
Prof. Alberto Montresor

Laureando
Filippo Momesso

Anno accademico 2020/2021

Ringraziamenti

...thanks to...

Indice

1	Contesto Aziendale	3
1.1	Cluster Reply	3
1.2	Progetti, Clienti e Prodotti	3
1.3	Struttura team di lavoro	4
1.4	Metodologia di lavoro	4
1.5	Strumenti a supporto dei processi	6
2	Analisi delle tecnologie	6
2.1	Microsoft Dynamics CRM	6
2.1.1	Moduli Funzionali	6
2.1.2	Entità e Record	8
2.1.3	Queue	8
2.1.4	Moduli (Forms)	9
2.1.5	Ricerca e Ricerca Avanzata	9
2.1.6	Web Resources	9
2.1.7	Workflow	10
2.1.8	Plugin	11
2.1.9	Soluzioni	14
2.2	Microsoft Power Automate	14
2.2.1	Flussi Cloud	14
2.3	Microsoft AI Builder	17
2.3.1	Creazione di un modello personalizzato	17
3	Primo Progetto	21
3.1	Cliente e obiettivo	21
3.1.1	Processo Planning Request	21
3.1.2	Creazione Planning Request in dettaglio	22
3.2	Sviluppo del progetto	23
3.2.1	Modello AI Builder	23
3.2.2	Flusso Power Automate	24
3.3	Considerazioni sul progetto	26
4	Secondo Progetto	27
4.1	Processo di business	27
4.2	Sviluppo del progetto	28
4.2.1	AI Builder	28
4.2.2	Flussi Power Automate	29
4.2.3	Plugin	30
4.2.4	Gestione del Case	32
4.3	Considerazioni sul progetto	33
5	Conclusioni	33
	Bibliografia	33

A Codice sorgente	35
A.1 Output del Modello di Elaborazione di Moduli	35
A.2 Codice sorgente Plugin di conversione Lead	47
A.3 Codice sorgente Plugin per l'invio di email con preventivo	50

1 Contesto Aziendale

Reply è una società specializzata in consulenze, *system integration* e servizi digitali con un focus sulla concezione, design e implementazione di soluzioni basate sulle nuove tecnologie e i nuovi canali di comunicazione. Operativa dal 1996, collabora con importanti realtà aziendali di diversi settori al fine di definire e sviluppare modelli di business resi possibili dai nuovi paradigmi tecnologici quali *Artificial Intelligence*, *Big Data*, *Cloud Computing*, *Digital Communication*, *Internet of Things* e *Social Networking*. Il campo di azione di Reply è quello delle aziende dei settori bancario e finanziario, industriale e dei servizi, delle telecomunicazioni, dell'energia e della pubblica amministrazione. I principali servizi offerti da Reply sono:

- **Consulenze** su strategie, comunicazione, processi aziendali e tecnologie;
- **System Integration** di soluzioni software esistenti;
- **Gestione**, monitoraggio e sviluppo continuo di sistemi e applicazioni software.

Il gruppo è formato da decine di società secondo un modello a rete e nel corso degli anni si è conquistato una posizione di prestigio nel panorama europeo e mondiale. Il fatturato 2020 si attesta a 1,25 miliardi di Euro [8].

1.1 Cluster Reply



Figura 1.1: Logo Cluster Reply Srl

Cluster Reply è la società del gruppo Reply specializzata in servizi di consulenza e di integrazione di sistemi su tecnologie Microsoft. Opera in Italia in collaborazione con le altre aziende del gruppo specializzate in tali tecnologie. È presente sul territorio con sedi a Milano, Padova, Roma, Torino, Trieste, Bologna e Silea.

Per quanto riguarda la struttura, Cluster Reply è divisa in diverse sezioni, ognuna dedicata a uno specifico settore di business. Si ha una sezione dedicata a Microsoft Azure, una al sistema Microsoft ERP (*Enterprise Resource Planning*), una al settore *Manufacturing*, una al settore *Financial* (banche e assicurazioni) e infine una sezione dedicata alla *Customer Experience*.

È in quest'ultima sezione che ho svolto la mia attività di tirocinio formativo. Essa è altamente specializzata nella consulenza e realizzazione di sistemi custom basati su Microsoft Dynamics 365, la linea di applicazioni aziendali intelligenti per la pianificazione di risorse aziendali e la gestione delle relazioni con i clienti. Di questa ampia gamma di prodotti, durante il tirocinio ho avuto la possibilità di approfondire la conoscenza delle applicazioni CRM (*Customer Relationship Management*), Sales Hub, Customer Service Hub e un'applicazione custom specificatamente sviluppata per un cliente.

1.2 Progetti, Clienti e Prodotti

Tra i progetti più importanti dell'azienda (non coperti da segreto professionale) si può menzionare il "sistema di automazione delle attività nella gestione degli affitti arretrati" sviluppato per il cliente Notting Hill Genesis (NHG). La soluzione proposta da Cluster Reply consiste di una

soluzione CRM basata sulla piattaforma Microsoft Dynamics 365 per l'automazione delle attività degli utenti e la comunicazione verso i clienti. Questa soluzione permette la gestione di un flusso automatizzato per la gestione degli affitti arretrati, in grado di guidare gli utenti NHG e automatizzare le comunicazioni (SMS, email e lettere) verso i clienti, gestendone le tempistiche, i template dinamici da utilizzare e salvando l'intera documentazione su Microsoft SharePoint in caso di rinvio legale [5].

Un altro prodotto confezionato e pubblicato sul Microsoft Store da Cluster Reply è il motore di configurazione di *Workflow*¹ "Configurable Workflow & SLAs Engine". Grazie a questo prodotto software da integrare in Microsoft Dynamics 365 è possibile semplificare la configurazione e la gestione dei *workflow* e degli SLA (*Service Level Agreement*) [6].

1.3 Struttura team di lavoro

In Cluster Reply il personale è suddiviso in Consultant, Senior Consultant, Manager e Senior Manager. Per quanto riguarda le prime due figure, le mansioni dipendono dal ruolo svolto all'interno del team. In generale un Consultant o Senior Consultant può essere uno sviluppatore software oppure un analista funzionale. Quest'ultimo si occupa della prima interazione con il cliente, analizzando il problema e proponendo le soluzioni, raccoglie i requisiti e in collaborazione con chi si occupa dello sviluppo progetta la soluzione. L'analista, inoltre, si occupa della scrittura del documento di analisi funzionale. Gli sviluppatori invece, come si può intuire, si occupano della messa in pratica e della realizzazione del progetto e della scrittura della documentazione tecnica.

I Manager e Senior Manager sono le figure che si occupano degli elementi meno tecnici ma ugualmente importanti nell'azienda ovvero l'ambito *business* ed economico, le relazioni con i clienti e la gestione dei progetti (avvio, pianificazione, divisione dei compiti, controllo e chiusura progetto).

1.4 Metodologia di lavoro

Per la gestione dei progetti vengono utilizzate tre metodologie a seconda di quale sia la più adatta alla specifica istanza in cui ci si trova:

- Metodologia Waterfall
- Metodologia Agile
- Metodologia Reply

Metodologia Waterfall In questo modello di gestione del progetto viene utilizzata la classica metodologia "a cascata". Il processo di realizzazione segue un andamento strutturato in una sequenza lineare di passi, in generale così definiti:

1. analisi dei requisiti
2. progettazione
3. sviluppo
4. collaudo
5. rilascio
6. manutenzione

In Cluster Reply questa metodologia viene scelta quando è ben chiaro fin dall'inizio del progetto quali siano le necessità e i desideri del cliente e quindi i requisiti risultino ben definiti e non si prospettano possibilità di mutazione degli stessi.

¹In Microsoft Dynamics 365 un *Workflow* o flusso di lavoro è un processo che permette di automatizzare operazioni che non necessitano di interazione da parte dell'utente.

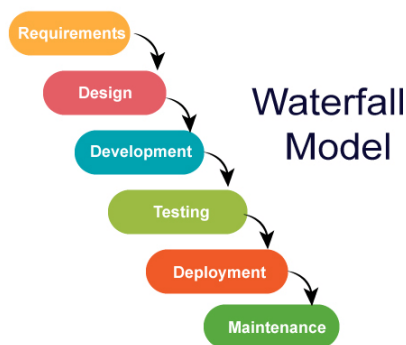


Figura 1.2: Modello a cascata

Metodologia Agile Questo modello, al contrario della metodologia Waterfall, rientra nella categoria dei modelli iterativi: ogni iterazione viene chiamata *sprint* ed è di breve durata, all'incirca un paio di settimane. Ogni *sprint* deve comprendere tutte le fasi necessarie per rilasciare un piccolo incremento nelle funzionalità software, come mostrato in Figura 1.3. L'obiettivo di ogni iterazione è quello di consegnare al cliente software "consegnabile", ovvero funzionante e di buona qualità anche se per le funzionalità fornite non è considerabile completo. Questo al fine di coinvolgere maggiormente il cliente nel processo di sviluppo e avere la possibilità di rivalutare i requisiti e le priorità a progetto in corso, riducendo il rischio di fallimento.

Cluster Reply utilizza questa metodologia nei casi in cui il cliente non abbia le idee perfettamente chiare riguardo i requisiti e sia necessario un metodo di lavoro più agile e aperto ai cambiamenti.



Figura 1.3: *Sprint* nella metodologia Agile

Metodologia Reply Questa metodologia può essere considerata come una versione intermedia tra Agile e Waterfall in quanto fa suoi diversi aspetti sia dell'una che dell'altra. Viene mantenuta quindi una programmazione lineare e sequenziale del progetto secondo le varie fasi successive. La differenza rispetto a Waterfall, tuttavia, è che il cliente viene periodicamente coinvolto nel progetto e aggiornato sull'andamento, mostrandogli ciò che è stato fatto fino a quel momento. Il fine è quello di ridurre il rischio di fallimento del progetto evitando che il cliente si veda consegnare un prodotto diverso dalle aspettative. Durante la fase di *solution review*, la revisione periodica con il cliente, è comunque possibile rivedere i piani e fare modifiche a progetto in corso, a discapito dell'efficienza e dei tempi di sviluppo e consegna del prodotto.

Si opta per questa metodologia qualora il cliente manifesti la necessità di monitorare frequentemente lo stato di avanzamento del prodotto e il tipo di progetto mal si prestasse all'utilizzo della metodologia Agile.

Indistintamente dalla metodologia utilizzata, resta necessario sottolineare l'importanza delle fasi di analisi di fattibilità e dei requisiti che si concretizzano nell'elaborazione del documento di analisi funzionale. Quest'ultimo, oltre a definire cosa deve essere fatto, è un'importante tutela dell'azienda dal punto di vista commerciale ed economico in quanto stabilisce cosa il cliente ha richiesto e quindi per cosa sta pagando. Aggiunte o modifiche rispetto a quanto approvato

dal cliente nel documento di analisi funzionale, sono da considerarsi non incluse nell'accordo commerciale e quindi da saldare separatamente.

1.5 Strumenti a supporto dei processi

Durante le fasi di sviluppo vengono utilizzati diversi strumenti a supporto delle varie operazioni.

Strumenti di gestione del progetto

Per la pianificazione e organizzazione del progetto Cluster Reply fa largo uso di Microsoft Project. Questo software permette la creazione dei diagrammi GANTT del progetto, la definizione dei task e delle *milestone* e il monitoraggio del loro stato, oltre che alla realizzazione di reportistica come i diagrammi dell'*effort*. Per la gestione dei budget e delle giornate e orari dedicati alla consuntivazione commesse viene utilizzato invece Geco, un software interno a Reply.

Strumenti di sviluppo

I linguaggi principalmente utilizzati sono C# e .NET per lo sviluppo di plugin e azioni, a questo si aggiunge l'utilizzo delle librerie Microsoft XRM SDK e Microsoft CRM SDK. Viene inoltre utilizzato Javascript per la parte frontend mentre per la manipolazione di database principalmente T-SQL. L'ambiente di sviluppo usato principalmente è Microsoft Visual Studio. Per quanto riguarda il *version control* del codice e i *backlog* tecnici viene utilizzato Azure DevOps.

2 Analisi delle tecnologie

In questo capitolo viene effettuata un'analisi dettagliata delle tecnologie Microsoft apprese e utilizzate durante il tirocinio.

2.1 Microsoft Dynamics CRM

Microsoft Dynamics CRM¹ è un pacchetto software per la gestione delle relazioni con il cliente sviluppato da Microsoft. Seppure interamente customizzabile grazie al framework proprietario Microsoft XRM SDK basato su .NET, di base si concentra sui settori Vendite, Marketing e Servizio Clienti.

Il CRM può essere infatti utilizzato per aumentare la produttività delle vendite e l'efficacia del marketing per un'azienda, gestire l'intera catena di assistenza clienti e fornire informazioni sull'andamento dei social media, business intelligence e molte altre caratteristiche e funzionalità pronte all'uso. L'interazione utente con Microsoft Dynamics CRM avviene attraverso l'interfaccia web, la quale è ottimizzata anche per l'utilizzo mobile e tablet oltre che desktop.

2.1.1 Moduli Funzionali

Il CRM è interamente progettato sui moduli funzionali Sales, Marketing e Service Management. Questa divisione in moduli è dovuta al fatto che un'azienda, nell'utilizzare il CRM per gestire i propri processi, necessita che gli utenti del settore vendite, ad esempio, abbiano a disposizione delle funzionalità specifiche disponibili nel modulo Sales, parimenti gli utenti del settore marketing con il modulo Marketing e gli utenti del servizio clienti con il modulo Service Management.

¹Devo davvero ringraziare Tutorialspoint perché la sua guida [7], su cui mi sono basato per l'elaborazione di questa sezione, mi ha chiarito moltissimi dubbi che la documentazione ufficiale (estremamente confusionaria) e i colleghi in azienda non sono riusciti a risolvermi.

Modulo Sales

Il modulo Sales del CRM è progettato per gestire l'intero ciclo di vita di un nuovo cliente. Si compone dei seguenti sotto-moduli:

- **Leads** - Rappresenta una persona o un'organizzazione che può diventare un potenziale cliente. Questo è il primo passaggio per l'inserimento di un potenziale cliente nel sistema.
- **Opportunities** - Rappresenta una potenziale vendita al cliente. Quando un Lead mostra interesse nell'offerta, viene convertita in Opportunity. Un'Opportunity può essere vinta o persa.
- **Accounts** - Rappresenta un'azienda con cui si ha una relazione. Quando un'Opportunity è vinta, viene convertita in un Account o un Contact.
- **Contacts** - Rappresenta una persona o un individuo con cui l'azienda ha una relazione. In genere, un Contact è un cliente (ad esempio tutti gli intestatari di un conto presso per una banca).
- **Competitors** - Gestisce i concorrenti di mercato dell'azienda.
- **Products** - Gestisce i prodotti offerti dall'azienda ai clienti.
- **Quotes** - Preventivo formale di prodotti o servizi proposti a prezzi specifici a un potenziale cliente.
- **Orders** - Quando un Quote viene accettato da un cliente viene convertito in un Order.
- **Invoices** - Fattura generata da un ordine.

Modulo Marketing

Il modulo Marketing del CRM è progettato per gestire l'intero processo di marketing di un'azienda per i suoi clienti esistenti e potenziali. Si compone dei seguenti sotto-moduli, i quali funzionano in coordinazione con il modulo Sales:

- **Marketing Lists** - Fornisce un metodo per raggruppare Contact, Account e Lead e interagire con essi attraverso l'invio di email promozionali, dettagli di eventi, newsletter e altre comunicazioni rilevanti per il cliente. È possibile definire dei criteri per creare le Marketing List.
- **Campaigns** - Servono a misurare l'efficacia e il completamento di uno specifico risultato, come l'introduzione di un nuovo prodotto o l'incremento della quota di mercato e può includere diversi canali di comunicazione come email e altre forme pubblicitarie.
- **Quick Campaigns** - Simile a una Campaign ma può essere messa in relazione con un solo tipo di Activity².

Modulo Service

Il modulo Service del CRM è progettato per gestire e tracciare le operazioni di servizio clienti di un'azienda come il supporto ai servizi basati su incidenti/casi, pianificazione di interventi di supporto ai clienti, eccetera. Si compone dei seguenti sotto-moduli:

- **Cases** - Permette di tracciare una qualsiasi richiesta, problema o lamentela di un cliente. Un Case ha un processo di risoluzione composto di vari stati che terminano con la risoluzione o la chiusura del Case.
- **Knowledge Base** - Mantiene una collezione di tutte le domande e risposte più comuni.

²Vedi sottosezione Gestione attività.

- **Contracts** - In relazione con Cases indica i contratti attivi del cliente.
- **Resources/Resource Groups** - Rappresenta le persone, strumenti, luoghi o l'attrezzatura necessaria per fornire un servizio. Possono essere utilizzate per risolvere uno specifico problema di un cliente.
- **Services** - Rappresentano i servizi di assistenza che l'azienda offre ai clienti.
- **Service Calendar** - Permette di organizzare le operazioni di assistenza.

Gestione attività

Tutti i moduli precedentemente trattati fanno uso del modulo Activity Management del CRM. Un'*activity* rappresenta qualsiasi tipo di interazione con il cliente come telefonate, email, lettere, appuntamenti e altro ancora. Queste *activity* possono essere messe in relazione con le altre entità trattate precedentemente appartenenti ai vari moduli del CRM.

2.1.2 Entità e Record

Un'*entità* è utilizzata per modellare dei dati nel CRM. Contact, Case, Account e Lead sono tutte entità del CRM che possono essere istanziate in record. A livello concettuale un'entità è equivalente a una tabella di un database relazionale.

Oltre alle entità presenti di default nel CRM è possibile definire nuove entità custom in base alle necessità specifiche, oppure modificare quelle esistenti aggiungendo o eliminando campi.

Il CRM fornisce undici tipi di campi:

- Single Line of Text
- Option Set (Dropdown)
- Two Options (Radio Button)
- Image
- Whole Number
- Floating Point Number
- Decimal Number
- Currency
- Multiple Lines of Text
- Date and Time
- Lookup

A partire dalla versione 2011 inoltre è disponibile un particolare tipo di campo chiamato Party List. Questo tipo consente di mappare una relazione tra entità come nel caso del tipo Lookup ma con la differenza che quest'ultimo può mappare una relazione con una singola entità mentre un campo di tipo Party List permette di avere una relazione con entità multiple. Per esempio, una email può essere associata a un Contact, un User o una Queue.

2.1.3 Queue

Le Queue sono contenitori usati per raccogliere record del CRM che siano da completare o che richiedano un'azione, come per esempio il completamento di un *task* o la chiusura di un Case. Sono molto utili quando è necessario avere una lista centralizzata di lavoro in attesa di essere completato, oppure quando sia necessario ordinare le attività in base al tipo o agli utenti assegnati per completarle. Inoltre, possono essere utilizzate per suddividere in categorie e assegnare priorità alle Activity e ai Case. [4]

2.1.4 Moduli (Forms)

Per creare, aggiornare o modificare un record nel CRM si utilizzano i Form. Ad ogni entità possono dunque essere associati uno o più Form. A seconda dell'opzione di visualizzazione scelta, del modulo funzionale selezionato o del tipo di utente che interagisce con il CRM possono essere visualizzati Form diversi. I form possono essere creati attraverso un'interfaccia apposita presente nelle impostazioni di customizzazione del CRM.

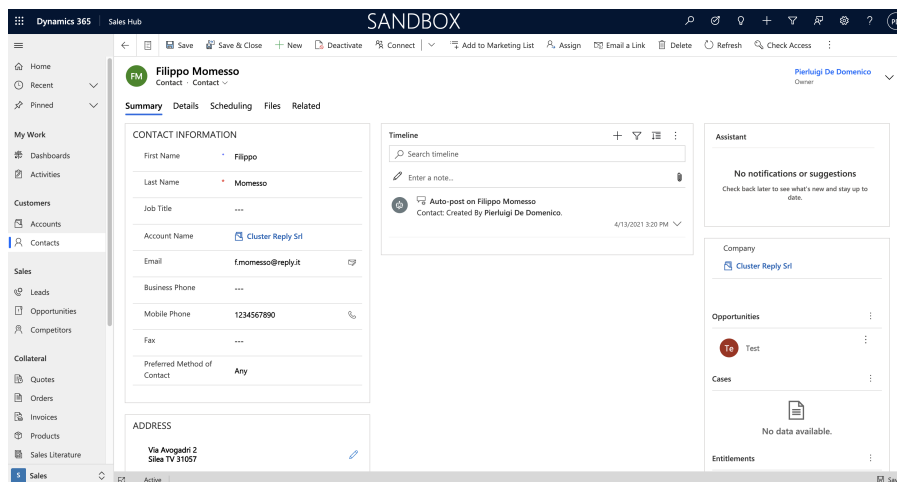


Figura 2.1: Esempio di form per la modifica di un record Contact

2.1.5 Ricerca e Ricerca Avanzata

Una delle più importanti funzionalità predefinite di Microsoft Dynamics CRM si trova nelle sue capacità di ricerca, le quali forniscono la possibilità di costruire *query* e filtri molto avanzati senza la necessità da parte dell'utente di conoscere linguaggi di programmazione o di querying.

Di default la vista a griglia di ogni entità supporta una funzionalità di Ricerca Veloce mediante una barra di ricerca posizionata nell'interfaccia utente in alto a destra come si può osservare in Figura 2.2.

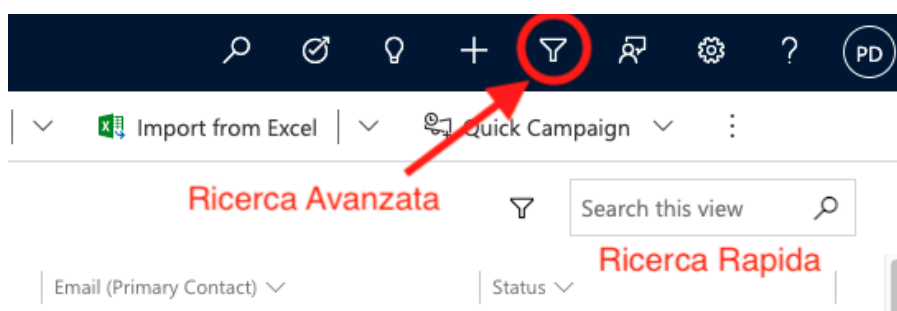


Figura 2.2: Opzioni di ricerca rapida e avanzata.

Cliccando invece sull'icona cerchiata in rosso in Figura 2.2 si accede alla ricerca avanzata, disponibile in una nuova finestra. La ricerca avanzata del CRM è una delle funzionalità più utili e potenti disponibili di default. In questa finestra, come si vede in Figura 2.3, è possibile selezionare l'entità di cui di vogliono cercare i record, applicare filtri e criteri di raggruppamento e salvare i risultati come viste (*views*) personali.

2.1.6 Web Resources

Le *Web Resources* nel CRM sono i file virtuali salvati nel database del CRM e usati per implementare le funzionalità della pagine web del CRM. Questi file possono essere HTML, Javascript, Silverlight o di qualsiasi altro tipo supportato. Nonostante il CRM venga fornito da Microsoft con una serie di funzionalità di base, spesso si rende necessario estendere e personalizzare queste

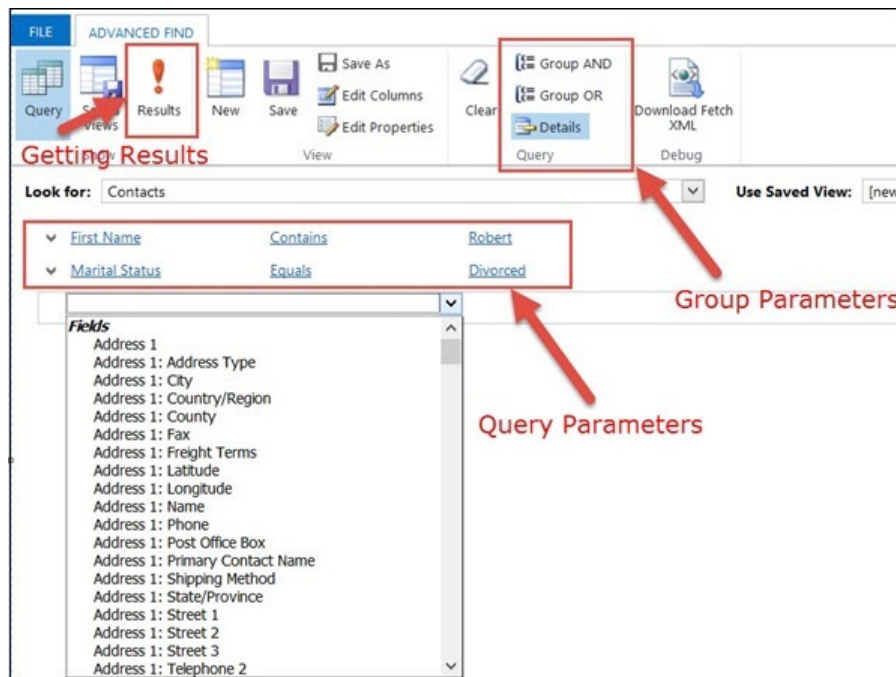


Figura 2.3: Finestra di ricerca avanzata.

funzionalità per rispettare e implementare i requisiti del progetto in questione. Tali estensioni possono avvenire in genere in due modi:

- **Estensione lato client** - Usando le Web Resources e il *Form Scripting*.
- **Estensione lato server** - Mediante *Plugin*, *Workflow* e *Web Services*.

Per capire quando può essere necessario l'utilizzo delle Web Resources del CRM prendiamo in considerazione i seguenti esempi:

- Si rende necessario fare ulteriori validazioni lato client sui campi di un form del CRM.
- È necessario costruire una o più pagine completamente custom che utilizzino dati provenienti da sistemi esterni al CRM.
- Si vogliono applicare modifiche visive o funzionali all'interfaccia grafica standard del CRM.
- Si vuole richiamare l'esecuzione di servizi web esterni in seguito ad azioni lato client, senza dover scomodare l'utilizzo di plugin o workflow lato server.

L'accesso a una Web Resource può avvenire mediante il suo URL univoco. Dato che le Web Resource vengono salvate nel database del CRM, possono essere caricate come singolo file o come collezione eterogenea di file (HTML, Javascript, ecc.) oppure possono essere create e/o modificate direttamente dal CRM, mediante un pannello apposito. Questo consente inoltre di semplificare i passaggi in caso di migrazione da un ambiente a un altro, come per qualsiasi altra personalizzazione del CRM. In Tabella 2.1 vi è un elenco dei principali tipi di web resource supportati.

2.1.7 Workflow

I *Workflow* permettono di automatizzare processi di business all'interno del CRM. Possono essere creati utilizzando le funzionalità del CRM oppure mediante lo sviluppo di codice .NET, consigliato nel caso di workflow più complessi. I workflow possono essere eseguiti in background oppure in tempo reale e possono anche richiedere l'input dell'utente.

L'esecuzione di un workflow può essere avviata automaticamente, in base a specifiche condizioni, oppure manualmente dall'utente (ad esempio tramite la pressione di un pulsante o la

Tipo di Web Resource	Esempio
Pagina Web (HTML)	È possibile creare una qualsiasi pagina HTML e inserirla in un form del CRM.
Fogli di stile (CSS)	Qualsiasi file css che può essere usato insieme ai file HTML.
Script (Javascript)	Qualsiasi tipo di codice lato client per manipolare campi, valori, effettuare validazioni, ecc.
Dati (XML)	Usati per salvare impostazioni o dati di configurazione in modo statico.
Immagini (PNG, JPG, GIF, ICO)	Qualsiasi immagine da utilizzare nel CRM.
Silverlight (XAP)	Qualsiasi applicazione Microsoft Silverlight da utilizzare nel CRM.
Fogli di stile (XSL)	Da usare per trasformare dati XML.

Tabella 2.1: Tipi di Web Resource supportati

selezione di un'opzione nel CRM). Internamente i workflow sono implementati utilizzando Windows Workflow Foundation, una tecnologia Microsoft che fornisce un'API, un motore di workflow e un designer per implementare workflow all'interno di applicazioni .NET.

I workflow del CRM possono essere eseguiti in maniera sincrona o asincrona. In genere, si utilizza l'approccio asincrono, facendo eseguire il workflow in background in quanto in questo modo si può limitare l'utilizzo di risorse del sistema.

L'esecuzione di un workflow può avvenire in seguito a specifici eventi definiti dal *Message* del workflow (ovvero il tipo di evento sul quale un Workflow può essere registrato) i quali possono essere creazione, modifica di uno o più valori o eliminazione di un record. Inoltre, un workflow può avere uno *scope* (in italiano "ambito di lavoro") che può essere User, Business Unit, Parent Child Business Unit o Organization. È possibile quindi specificare su quali record potrà essere eseguito il workflow in base all'utente proprietario dei record e del workflow.

Un workflow dunque non è altro che una sequenza di passaggi che vengono eseguiti sul CRM. Possono essere condizionali, di attesa o azioni. I primi due tipi sono auto esplicativi mentre nel terzo si può avere ad esempio la creazione o l'aggiornamento di un record, l'assegnamento di un record a un utente, l'invio di un'email, l'esecuzione di uno step custom programmato in .NET da uno sviluppatore oppure l'interruzione dell'esecuzione del workflow.

2.1.8 Plugin

Un *Plugin* è pezzo di software che si integra con Microsoft Dynamics CRM per modificarne o estenderne il comportamento standard. I plugin si comportano come gestori di eventi (*event handler*) e vengono eseguiti in seguito a un particolare evento nel CRM, specificato in fase di registrazione del plugin. Possono essere scritti in linguaggio C# oppure Visual Basic e possono essere eseguiti in modalità sincrona o asincrona.

Alcuni esempi di scenari in cui un Plugin può essere utilizzato sono:

- È necessario eseguire delle operazioni in modo automatico in seguito all'aggiornamento di alcuni determinati campi di un record, oppure aggiornare altri record collegati in seguito alla modifica o alla creazione di un record nel CRM.
- Si vuole chiamare un servizio web esterno in seguito a un evento come la creazione o l'aggiornamento di un record.
- È necessario compilare in modo dinamico i valori di alcuni campi di un record.

- Si vuole automatizzare dei processi come l'invio di email in seguito a specifici eventi nel CRM.

La pipeline di un plugin è divisa in molteplici fasi su cui può essere registrata la logica del plugin. La fase (in inglese *stage*) indica in quale punto del ciclo di esecuzione del plugin deve essere eseguito il codice. In Tabella 2.2 si possono consultare le fasi per cui è possibile registrare un plugin.

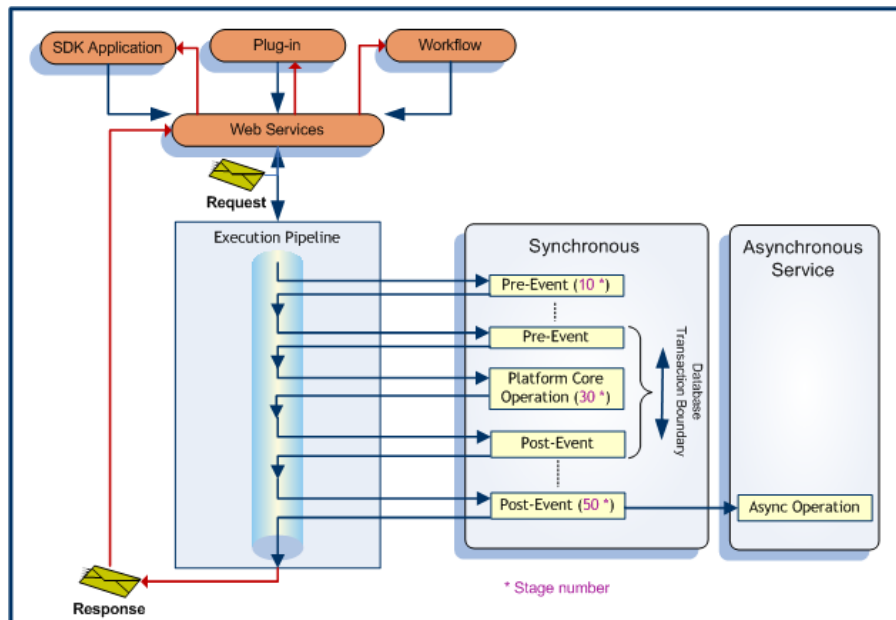


Figura 2.4: Pipeline di un Plugin

Evento	Nome della fase	Descrizione
Pre-Event	Pre-validation	Fase della pipeline per i plugin che devono essere eseguiti prima della MainOperation. I plugin registrati in questa fase possono essere eseguiti all' esterno della transazione del database.
Pre-Event	Pre-operation	Fase della pipeline per i plugin che devono essere eseguiti prima della MainOperation. I plugin registrati in questa fase sono eseguiti all' interno della transazione del database.
Platform Core Operation	MainOperation	L'operazione principale eseguita dal sistema, come creazione, aggiornamento, eliminazione di un record. Nessun plugin può essere registrato per questa fase.
PostEvent	Post-operation	Fase della pipeline per i plugin che devono essere eseguiti dopo della MainOperation. I plugin registrati in questa fase sono eseguiti all' interno della transazione del database.

Tabella 2.2: Fasi del ciclo di esecuzione di un Plugin

Ogni volta che il CRM invoca un evento, come ad esempio il salvataggio di un record, viene eseguita una sequenza di azioni. Per prima cosa l'evento innesca la chiamata al CRM Organization Web Service e l'esecuzione viene fatta passare attraverso le fasi della pipeline. Internamente le informazioni vengono trasmesse mediante un messaggio di tipo `OrganizationRequest` il quale viene intercettato dai plugin di tipo `Pre-Event` che possono modificarne le informazioni prima

che venga passato alla Platform Core Operation. Dopodiché il messaggio viene trasformato in un OrganizationResponse che viene intercettato dai plugin Post-Operation i quali possono modificare le informazioni prima. Infine, l'esecuzione viene ritornata all'applicazione che ha invocato l'evento.

Il campo *Message* specifica, come nel caso dei Workflow, il tipo di evento su cui il plugin è registrato. Per esempio, un plugin può essere registrato su un Create Message di un'entità Contact. In questo caso il codice del plugin verrebbe eseguito ogni qual volta un record Contact viene creato. Per le entità di default del CRM sono supportati più di 100 message diversi, mentre per le entità custom la scelta è più limitata.

Differenze tra Workflow e Plugin

Sia i Workflow che i Plugin possono essere utilizzati per estendere e le funzionalità del CRM. In molti casi i due approcci sono intercambiabili e possono essere utilizzati uno al posto dell'altro senza nessun problema. Tuttavia, la documentazione ufficiale Microsoft [1] suggerisce alcune linee guida riportate in Tabella 2.3. Inoltre, i colleghi più esperti mi hanno spiegato che in generale preferiscono usare i plugin in caso di logica sincrona oppure molto complessa, mentre i workflow per la logica asincrona o per processi semplici e da eseguire in seguito a una richiesta dell'utente, come ad esempio l'invio automatico di email.

Criteria	Plugin	Workflow
Eseguito prima o dopo la Core Platform Operation	Viene eseguito immediatamente prima o dopo la core operation (sincrono). Può essere messo in coda ed eseguito dopo la core operation (asincrono)	Viene messo in coda ed eseguito dopo la core operation
Impatto sulle performance del CRM	Plugin sincroni possono aumentare i tempi di risposta del CRM in quanto fanno parte dei processi della piattaforma. Un plugin mal implementato può bloccare il CRM	L'impatto negativo sui tempi di risposta del CRM è minimo.
Restrizioni di sicurezza	Per registrare un plugin è necessario un utente con i privilegi di System Admin o System Customizer che sia membro del Deployment Administrator group	Gli utenti possono creare workflow in modo interattivo all'interno dell'interfaccia web. Tuttavia, per poter registrare un workflow è necessario avere gli stessi privilegi di sicurezza richiesti per i plugin.
Migliore per operazioni che richiedono molto o poco tempo	I plugin ad esecuzione sincrona andrebbero usati per processi brevi mentre i plugin asincroni per operazioni più intensive.	Indifferentemente per processi brevi o lunghi.
Persistenza del processo e dei dati	I plugin vengono eseguiti fino al completamento	I workflow possono essere messi in pausa, posposti, cancellati e ripresi mediante chiamate all'SDK o dall'utente mediante l'interfaccia web del CRM. Lo stato di un Workflow viene salvato automaticamente prima di essere messo in pausa o posposto.

Tabella 2.3: Come scegliere se utilizzare un Plugin o un Workflow

2.1.9 Soluzioni

Le soluzioni sono il modo con cui è possibile firmare, impacchettare e mantenere le unità software che estendono il CRM [2]. Qualsiasi customizzazione, estensione, o configurazione può essere impacchettata, organizzata e distribuita usando le soluzioni. Una soluzione può essere esportata come file .zip e importata in seguito in un'altra istanza di Dynamics 365.

Tipi di soluzioni

Esistono tre tipi di soluzioni: la Default System Solution, le soluzioni Managed e le soluzioni Unmanaged.

La prima contiene tutti i componenti definiti di default in Microsoft Dynamics CRM senza alcuna customizzazione. Questi componenti possono essere modificati e le versioni modificate possono essere inserite in soluzioni di tipo Managed o Unmanaged.

Una soluzione Managed è una soluzione che si intende distribuire e installare nel CRM del cliente. In particolare può essere installata sulla soluzione di default o su altre soluzioni Managed. Su una soluzione di questo tipo non è possibile quindi aggiungere o rimuovere componenti, è tuttavia permessa la modifica dei componenti presenti.

Una soluzione Unmanaged è una soluzione da considerarsi ancora in fase di sviluppo e che non si intende distribuire. In una soluzione Unmanaged è possibile aggiungere, rimuovere, modificare ed eliminare componenti. Ogni nuova soluzione inoltre è impostata di default come di tipo Unmanaged.

2.2 Microsoft Power Automate

Microsoft Power Automate, in precedenza conosciuto come Microsoft Flow, è un servizio online per la creazione di flussi di lavoro *low-code/no-code* in grado di automatizzare operazioni su diverse app e servizi, sia sviluppati da Microsoft che non. È possibile connettersi infatti a più di 220 servizi e gestire i dati sia in cloud che in locale come ad esempio con SharePoint o Microsoft SQL Server. La lista di connettori ad app e servizi disponibili inoltre, è in continua crescita in quanto Microsoft lavora a stretto contatto con i vari partner per estendere le possibilità di Power Automate.

I due principali tipi di flussi di lavoro supportati da Power Automate sono i flussi cloud e i flussi desktop.

Flussi Cloud Si dividono in tre tipologie:

- Flussi automatizzati - Permettono di creare un'automazione che viene attivata da un evento come l'arrivo di un email da una persona specifica.
- Flussi istantanei - Permettono di avviare un'automazione facendo click su un pulsante. È possibile automatizzare le azioni ripetitive come ad esempio l'invio di un promemoria su Microsoft Teams
- Flussi pianificati - Permettono di pianificare un'automazione ad una specifica ora come ad esempio il caricamento giornaliero di dati in SharePoint o in un database

Flussi Desktop Vengono utilizzati per automatizzare attività ripetitive sul Web o sul desktop. È possibile ad esempio creare flussi per organizzare documenti mediante azioni su file e cartelle, estrarre dati da siti web e archivarli su file Excel, inserire fatture in software gestionali e molto altro.

2.2.1 Flussi Cloud

In Microsoft Power Automate un flusso è rappresentato mediante un diagramma a blocchi come si può vedere in Figura 2.5. Ogni flusso inizia sempre con un blocco di tipo *trigger*, il quale permette di avviare l'esecuzione del flusso in seguito al verificarsi di un evento specificato. La maggior parte dei connettori disponibili offre diversi trigger predefiniti da utilizzare per avviare

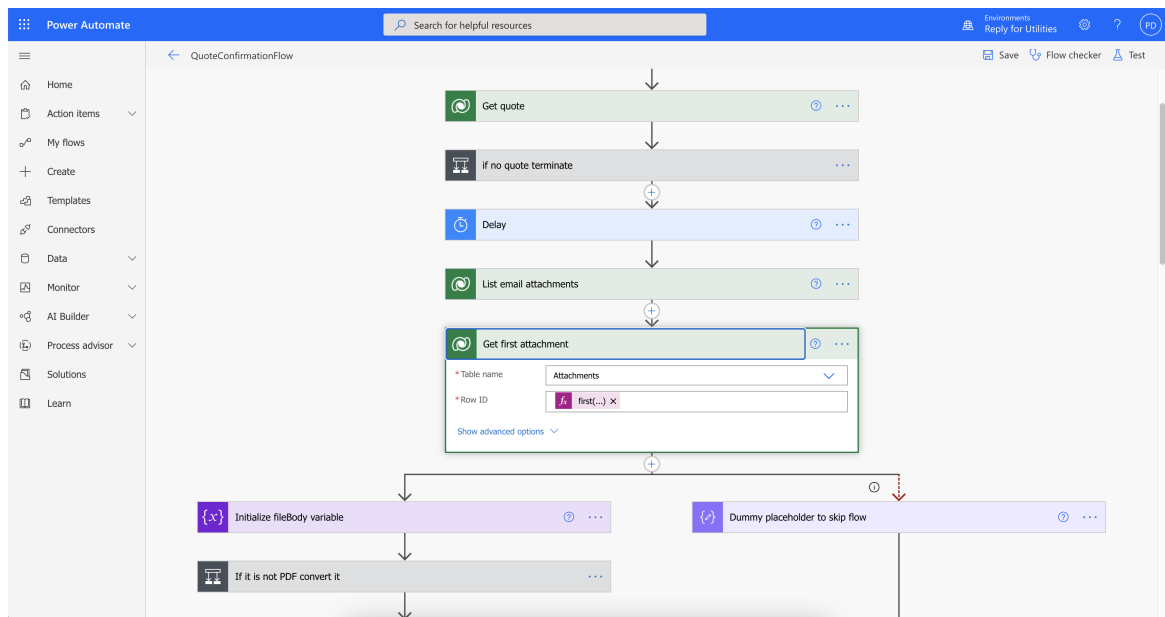


Figura 2.5: Esempio di un flusso Power Automate

i flussi. Tutti gli altri blocchi utilizzabili sono blocchi di tipo *action* e permettono di eseguire operazioni come la creazione di record nel CRM, la chiamata ad API esterne, l'invio di email e altro, a seconda di ciò che viene reso disponibile dai vari connettori.

Un connettore non è altro che un proxy o un wrapper intorno a un'API che consente al servizio sottostante di comunicare con Microsoft Power Automate. Consente agli utenti di connettere i loro account e sfruttare un set di trigger e azioni predefiniti per sviluppare i flussi di lavoro. Microsoft mette a disposizione numerosi connettori (gratuiti e a pagamento) già pronti per essere utilizzati ed è possibile inoltre per gli sviluppatori creare dei connettori custom per poter integrare il proprio servizio web o API nell'ecosistema Power Automate.

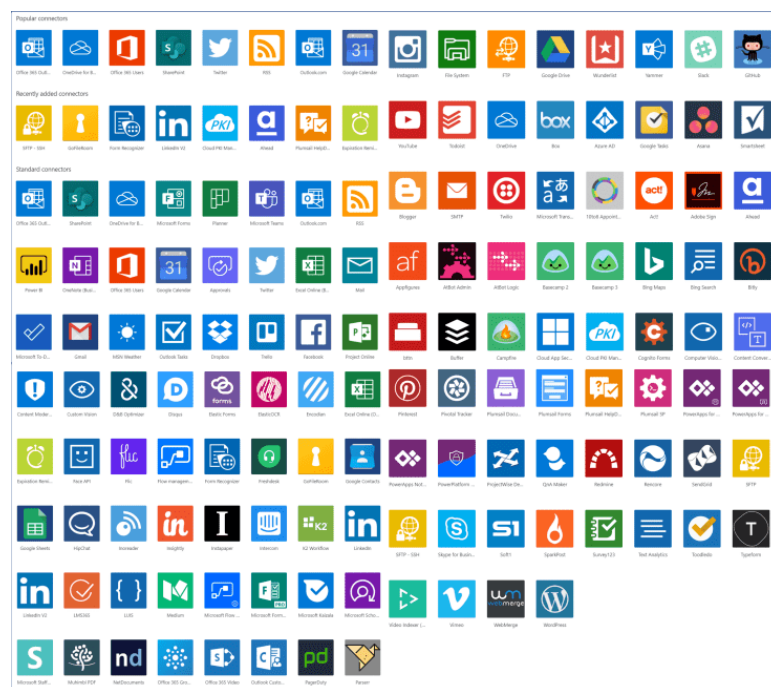


Figura 2.6: Elenco dei connettori gratuiti di Power Automate.

Trigger e Action

Attraverso l'interfaccia grafica è possibile creare blocchi trigger e action, ai quali si possono passare dati o espressioni in maniera molto semplice e intuitiva come in Figura 2.7 nella quale è visibile una action per la creazione di un file su dropbox che utilizza i dati di un tweet restituiti da un trigger. Attraverso il pannello laterale *Dynamic Content* si possono inserire i valori dinamici ritornati da action o trigger precedenti. È possibile inserire espressioni in linguaggio OData all'interno dei campi di testo oppure espressioni più complesse in linguaggio *Workflow Definition Language* mediante il pannello laterale *Expression*[3].

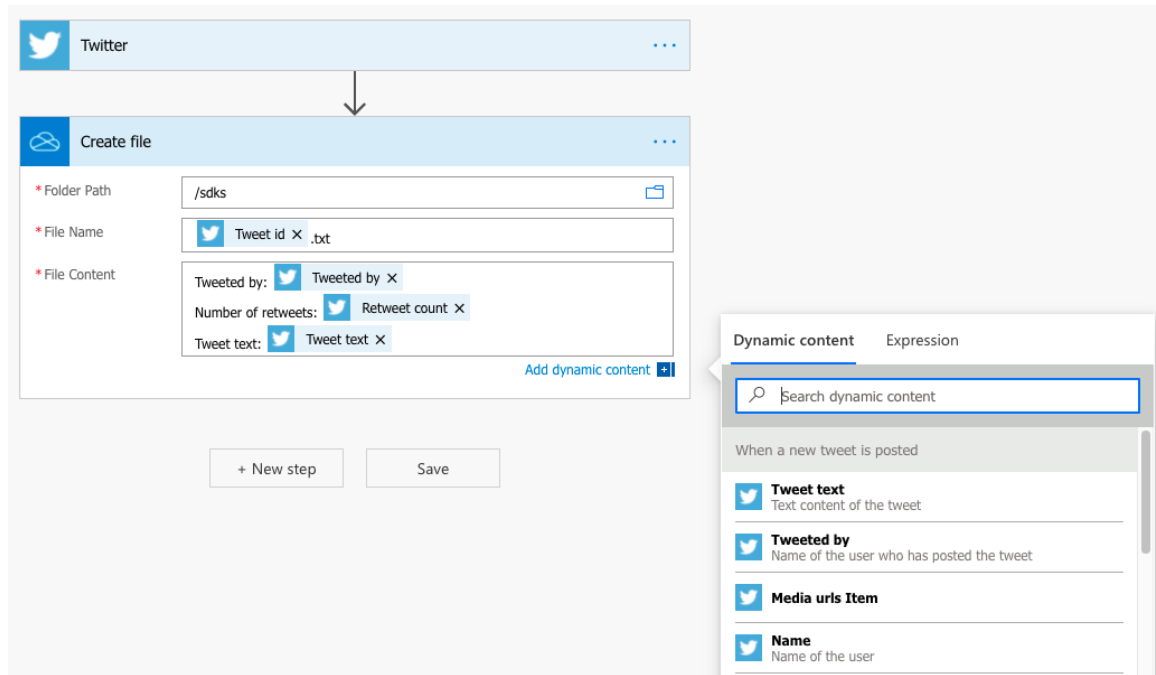


Figura 2.7: Esempio di blocco di tipo action.

Dietro a questa semplice e intuitiva interfaccia grafica non si nasconde altro che un oggetto JSON definito rispettando le specifiche del linguaggio sopracitato. Di seguito il listato corrispondente all'action usata come esempio.

```
1 {
2   "inputs": {
3     "host": {
4       "connectionName": "shared_onedriveforbusiness",
5       "operationId": "CreateFile",
6       "apiId": "/providers/Microsoft.PowerApps/apis/shared_onedriveforbusiness"
7     },
8     "parameters": {
9       "folderPath": "/sdks",
10      "name": "@{triggerOutputs()?['body/TweetId']}.txt",
11      "body": "Tweeted by: @{triggerOutputs()?['body/TweetedBy']}\nNumber of retweets:
              @{triggerOutputs()?['body/RetweetCount']}\nTweet text:
              @{triggerOutputs()?['body/TweetText']}"
12    },
13     "authentication": "@parameters('$authentication')"
14   },
15   "runtimeConfiguration": {
16     "contentTransfer": {
17       "transferMode": "Chunked"
18     }
19   }
20 }
```

2.3 Microsoft AI Builder

Microsoft AI Builder è una funzionalità di Power Platform che permette l'utilizzo di modelli di intelligenza artificiale per ottimizzare i processi aziendali. Essendo parte di Power Automate, anche AI Builder sposa la filosofia low-code/no-code permettendo all'utente di utilizzare modelli di machine learning preconfezionati senza la necessità di competenze di programmazione o *data science*. Essendo una tecnologia completamente closed-source Microsoft non ha reso disponibili molte informazioni sull'architettura e il funzionamento dietro le quinte di AI Builder. L'unica informazione disponibile è che viene sfruttata la stessa tecnologia del pacchetto di servizi Microsoft Azure AI, anch'essa closed-source.

In AI Builder è possibile scegliere tra diversi tipi di modelli a seconda dello scenario di utilizzo. Questi si dividono in modelli predefiniti e modelli personalizzati. La differenza tra le due tipologie si trova nel fatto che nel primo caso il modello è fornito pronto all'uso, mentre nel secondo è necessario creare, sottoporre a training e pubblicare per l'uso il modello in questione.

Modelli predefiniti:

- Estrazione di parole chiave
- Rilevamento lingua
- Analisi del sentimento
- Traduzione testo
- Lettore di biglietti da visita
- Riconoscimento del testo
- Elaborazione ricevute

Modelli personalizzati:

- Classificazione categoria
- Estrazione di entità
- Stima
- Elaborazione moduli
- Rilevamento oggetti

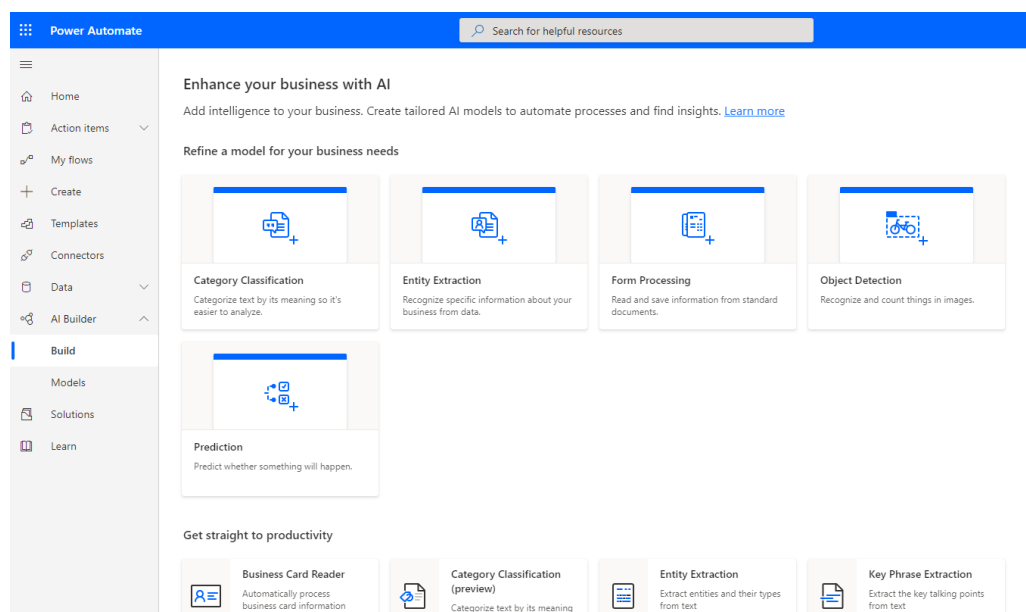


Figura 2.8: Schermata home di AI Builder.

2.3.1 Creazione di un modello personalizzato

Nel caso di modelli personalizzati in fase di creazione di un modello è necessario fornire un dataset di training. Essendo una feature low-code/no-code i passaggi necessari sono abbastanza semplici. Per fornire il dataset di training si hanno due possibilità a seconda del tipo di dati che il modello in questione prende come input: testo o immagini/documenti. Se il modello effettua analisi di testo il dataset di training dovrà essere fornito utilizzando una tabella Dataverse, un altro servizio Microsoft che permette di archiviare e gestire dati utilizzati nelle applicazioni Microsoft.

Questa tabella dovrà contenere i dati di training secondo le specifiche definite per il singolo modello. In altre parole, modelli diversi richiedono dati diversi, etichettati in modo diverso. Se il modello lavora su dati come documenti o immagini, invece, si dovrà seguire la procedura guidata all'interno di AI Builder, che permetterà di caricare ed etichettare i documenti su cui fare il training.

Una volta terminato il training del modello è possibile testarlo, con altri dati non etichettati e valutarne le performance. Dopodiché il modello è pronto per essere pubblicato ed utilizzato in Microsoft Power Automate. Esiste infatti un connettore apposito per AI Builder.

Esempio: modello di elaborazione moduli

Il modello di elaborazione moduli fornito da AI Builder consente di leggere e salvare informazioni da documenti standard come fatture o documenti fiscali. Automatizzando questo processo è possibile risparmiare molto tempo evitando, ad esempio, che l'utente debba passare ore a trascrivere a mano fatture all'interno di un gestionale.

Il modello, tuttavia, presenta ancora alcune limitazioni, come ad esempio il formato del file (PDF, JPG, PNG), il numero di pagine e la dimensione del documento (non più di 500 pagine per i PDF e 50 MB) e il supporto al solo alfabeto latino. Tra le limitazioni più importanti, inoltre, vi è il mancato supporto al riconoscimento di caselle di controllo, pulsanti di opzione e firme. Queste ultime tre limitazioni in particolare sono state un problema in fase di sviluppo dei progetti.

Per creare il modello personalizzato di elaborazione dei moduli è necessario per prima cosa specificare i nomi dei campi e delle tabelle che dovranno essere estratti dal documento. In seguito, si dovranno caricare i file su cui avverrà il training del modello. AI Builder a questo punto analizzerà i documenti caricati in modo da rilevare la presenza di campi testuali e tabelle. L'utente dovrà selezionare manualmente i campi che si vogliono riconoscere e contrassegnarli con l'etichetta corrispondente come in Figura 2.9; allo stesso modo è possibile contrassegnare tabelle come in Figura 2.10. Non sono supportate le tabelle estese su più pagine, pertanto è necessario definirle come tabelle separate durante il primo passaggio della procedura.

Terminato il passaggio di etichettatura di campi e tabelle sui documenti di training, AI Builder procede con il training del modello.

Al termine del training è possibile effettuare un test rapido sulle performance del modello caricando manualmente un documento. In seguito all'analisi, verranno mostrati i campi ed eventuali tabelle riconosciute e una percentuale di *confidence* per ciascun campo. Questo dato indica quanto il modello ritiene di essere stato preciso nel riconoscimento di quel valore. In Figura 2.11 un esempio. Dopodiché per utilizzare un modello all'interno di app e servizi Microsoft è necessario pubblicarlo mediante l'apposito pulsante.

Utilizzo in Power Automate

Per utilizzare un modello AI Builder in un flusso Power Automate è sufficiente utilizzare l'action del connettore AI Builder corrispondente al tipo di modello necessario. Per utilizzare un modello di elaborazione dei moduli, ad esempio, bisogna utilizzare la action *Process and save information from forms* e selezionare il modello pubblicato, il tipo di documento e passare il file stesso come contenuto dinamico. Nell'esempio in Figura 2.12 il documento viene recuperato da un trigger manuale che richiede l'inserimento di un file da parte dell'utente. Nulla vieta tuttavia che il documento da analizzare possa essere l'allegato di un'email oppure un file in Onedrive o Sharepoint. Sarà necessario semplicemente utilizzare una action del relativo connettore per recuperare il file desiderato.

Come output viene ritornato un oggetto JSON contenente dati diversi a seconda del tipo di modello utilizzato. Nel caso di esempio per ogni campo vengono ritornati, il valore corrispondente, il punteggio di confidence, le coordinate del riquadro che racchiude il valore, il numero di pagina in cui si trova il valore e altri dati. In Appendice A.1 è disponibile un esempio di oggetto JSON restituito. I dati contenuti in questo oggetto inoltre vengono interpretati da Power Automate e resi disponibili come contenuto dinamico per essere utilizzati nel flusso da altre action.

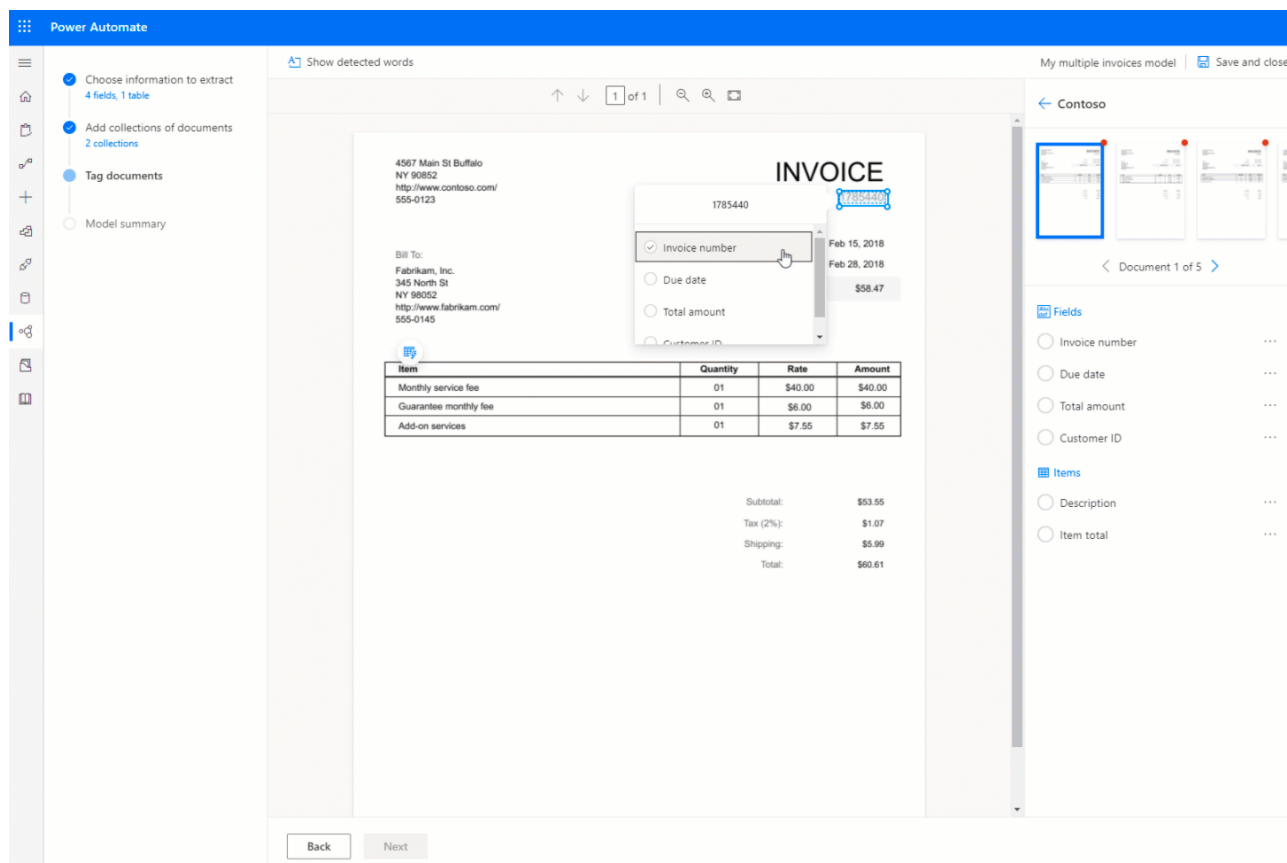


Figura 2.9: Etichettatura di campi testuali in AI Builder.

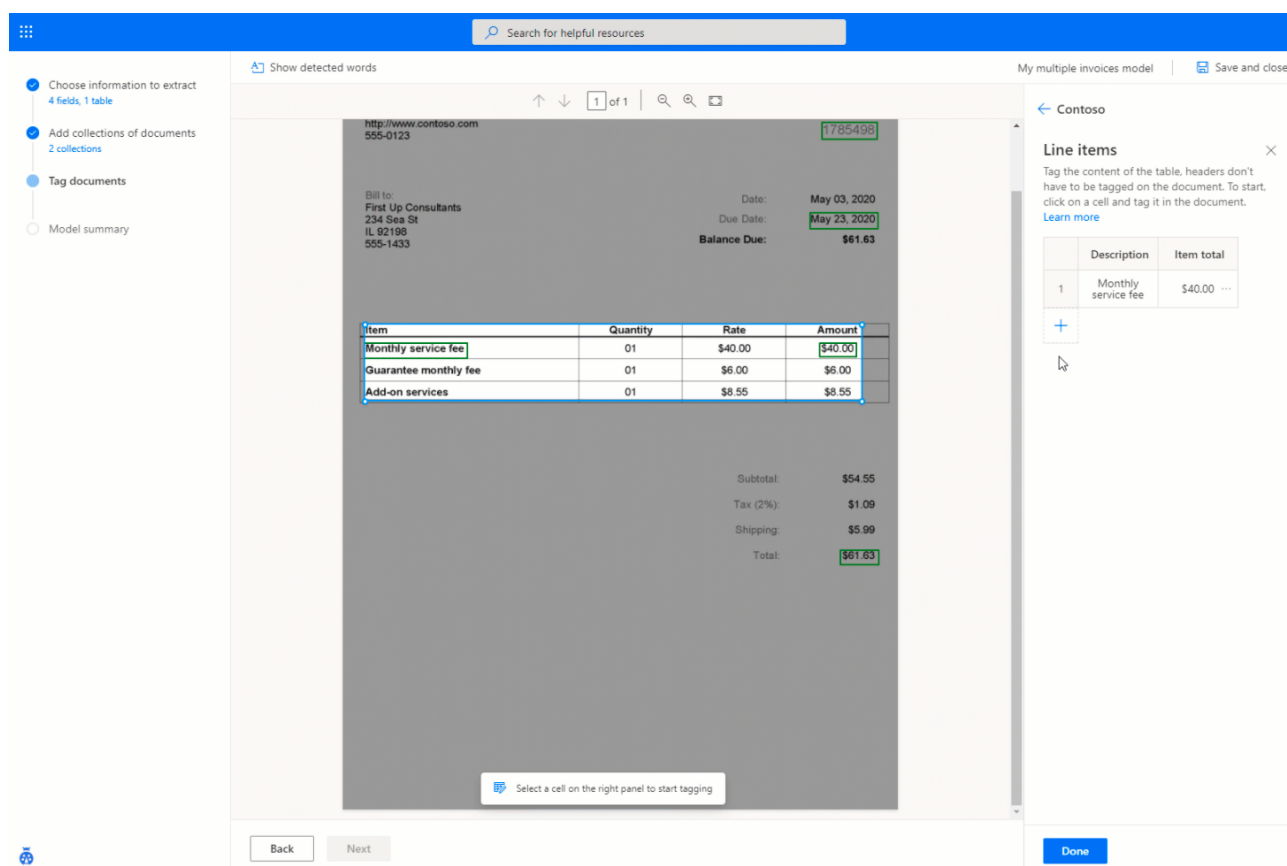


Figura 2.10: Etichettatura di tabelle in AI Builder.

INVOICE

DATE
Nov 12, 2020

INVOICE NO
1975

ADATUM CORPORATION
234 France Ave
WA 76328
www.adatum.com
sales@adatum.com

INVOICE TO
Fabrikam, Inc.
345 North St
NY 98052

Customer Id: 5791

SALESPERSON
Klarissa Wolf

PAYMENT DUE DATE
Due on 11/15/2020

Items (Quantity, Description, Total)
Confidence score
100%

QUANTITY	DESCRIPTION	UNIT PRICE	LINE TOTAL
01	Safe pool alarm	\$32.99	\$32.99
09	Filter cartridge	\$28.99	\$260.91
01	Pool pump	\$154.99	\$154.99

Figura 2.11: Esempio dell'esito del test rapido su un documento.

Manually trigger a flow

File Content | My Document

+ Add an input

↓

Process and save information from forms

- * AI Model: Contoso Invoices AI Builder model
- * Document type: JPEG Image
- * Form: File Content

Figura 2.12: Esempio di utilizzo di una action AI Builder in un flusso Power Automate.

Criticità di AI Builder

Nonostante AI Builder sia uno strumento molto potente per gli utilizzi per cui è proposto, risulta anche molto limitante. Prendendo ad esempio il modello di elaborazione di documenti, che è quello con cui ho avuto più a che fare durante i due progetti, si ottengono risultati positivi solo quando i documenti analizzati sono tra loro identici nella struttura a quelli utilizzati per il training. Le uniche differenze infatti devono trovarsi nei dati contenuti; qualora posizionamento, font e dimensione del testo differiscano, il modello fallisce. Microsoft tuttavia non menziona in

nessun modo questo dettaglio nella documentazione.

Inoltre, avendo potuto lavorare con le varie tecniche di machine learning durante il corso *Introduction to Machine Learning*, mi sono trovato impossibilitato a utilizzare le metodologie apprese, in particolare riguardo la validazione e il testing, in quanto AI Builder non fornisce alcun modo per poter effettuare testing su larga scala e quindi raccogliere dati e analizzarli. In sostanza bisogna fidarsi dello strumento e credo che in ambito business non sia una buona pratica. Chiaramente essendo un prodotto mirato a una classe di utenti non esperta credo che il compromesso sia comunque buono.

3 Primo Progetto

Il primo progetto sviluppato è stato scelto da parte del mio supervisore per permettermi di apprendere i meccanismi generali di funzionamento delle tecnologie trattate nel Capitolo 2, ponendo particolare attenzione su AI Builder.

Fino a quel momento, inoltre, il team di sviluppo non aveva mai avuto l'occasione di utilizzare AI Builder e necessitava di capirne potenzialità e limiti. Quindi, al termine del progetto, ho dovuto fare un resoconto riguardo alla mia esperienza in modo da decidere se AI Builder fosse uno strumento valido da utilizzare in progetti futuri.

Il progetto da me svolto durante la prima fase dello stage, infatti, non era destinato al rilascio al cliente, con l'eccezione che, nel caso in cui avesse avuto un riscontro positivo, avrebbe potuto essere proposto come estensione a ciò che l'azienda stava già sviluppando.

3.1 Cliente e obiettivo

Il cliente per cui è stato svolto il progetto è un'importante società di trasporti ferroviari inglese, il cui nome per questioni di riservatezza non è divulgabile. Questa società si occupa nello specifico della gestione e organizzazione degli interventi di controllo e manutenzione delle linee ferroviarie. Operando solamente nel settore gestionale, per la realizzazione materiale delle operazioni l'azienda fa affidamento ad altre società partner.

Come si può facilmente ipotizzare la pianificazione di questi interventi è estremamente complessa, in quanto è necessario organizzare diverse componenti: deviazione delle tratte ferroviarie, scelta dell'azienda che deve effettuare l'intervento, scelta di giornate e orari in cui effettuare i lavori, schieramento di mezzi e attrezzature, ecc... Cluster Reply si è occupata quindi dello sviluppo di una soluzione software custom, basata su Microsoft Dynamics 365, che permettesse di semplificare e rendere più efficiente queste operazioni.

All'interno di questa soluzione, il mio compito è stato quello di automatizzare uno dei processi aziendali del cliente, senza però modificare le metodologie esistenti.

3.1.1 Processo Planning Request

Dietro all'organizzazione di un intervento di controllo o manutenzione si configura un complesso ma ben delineato iter burocratico che fa pieno affidamento sul CRM, dal momento in cui viene avviato, fino alla sua conclusione positiva o negativa che sia.

Il processo è così composto:

1. **Creazione della *Planning Request*** - In seguito all'arrivo di un'email con uno specifico allegato, un addetto crea un record di tipo Planning Request mediante uno specifico form. E' anche possibile associare l'email ad una Planning Request esistente, ad un Contact o ad un Work Party.
2. **Approvazione della *Planning Request*** - Composta da tre fasi:
 - Fase di pianificazione
 - Fase di bozza

- Fase di pubblicazione


Una Planning Request è l'entità su cui questo processo è costruito. Essa è basata sull'entità standard Incident del CRM, su cui sono state effettuate diverse modifiche in modo da adattarla alle necessità del caso. Essenzialmente, sono stati aggiunti diversi campi e associazioni con altre entità come, ad esempio, l'entità Job, la quale invece modella una attività che deve essere eseguita da un Work Party, entità che rappresenta un'azienda partner. La customizzazione del CRM per questo cliente comprende, inoltre, altre funzionalità di importanza marginale per questa trattazione.

Più nello specifico, l'obiettivo assegnatomi è quello di automatizzare la prima fase e parte della seconda, ovvero la creazione della Planning Request e la sua pianificazione. Essendo infatti una semplice operazione di trascrizione del contenuto dell'allegato nel sistema, la sua automatizzazione permetterebbe il risparmio di molto tempo per i dipendenti del cliente.

Per la realizzazione del progetto è quindi stato scelto di creare un flusso cloud Power Automate per la creazione e la pianificazione della Planning Request a partire dai dati estratti dall'allegato mediante un modello di elaborazione di moduli di AI Builder opportunamente creato e allenato.

3.1.2 Creazione Planning Request in dettaglio

Il processo di creazione di una Planning Request inizia in seguito all'arrivo di un'email indirizzata ad un apposita casella di posta. Le email inviate a questo indirizzo, alle quali deve essere allegato un documento appositamente compilato (come in Figura 3.1), vengono assegnate automaticamente ad una Coda (Queue) del CRM, chiamata Power Planning Queue.



AP POWER PLANNING

ARRANGEMENTS (Office Use Only)	YES
PLANNED	
TESTERS	

PLANNING REQUEST FORM
TO AVOID DELAYS PLEASE READ NOTES BELOW

1. Complete sections 1 to 7, then return to Operations Planning for processing. Should you not receive confirmation of receipt the next business day, please call the Planning Section for assistance.
2. Planning Requests must be submitted to Operations Planning, giving a **minimum of 8 weeks notice**. Work of an urgent nature will only be accepted if agreed in advance by a Technical Planning Engineer.
3. When work requires a Resource, you will need to specify the requirements and present a valid Job Number, please contact us on 42667 for details.
4. Ensure that the Method Statement reference number related to the particular work is placed in the respective box.
5. Please ensure that the relevant **XXXXXXXXXX** Project Engineer / Manager is aware and agreed your proposed dates of work.
6. For further information when making a planning application, **please refer to the XXXXXXXXXX**

Planning Request No	Week

1. Request To:

Section: Operations Planning	Telephone: Auto: 42421 Ext: 0207-918- 2421	Facsimile: Auto: 42370 Ext: 0207-918-2370
----------------------------------------	--------------------------------------------------	-------------------------------------------------

2. Request From:

Section: P&E	Telephone: 07764 429068	Facsimile:
----------------------------	-----------------------------------	------------

3. Date(s) of requirement (Please attach programme if more than 1 weeks duration)

Week No.	Wk Comm.	Start Date	Finish Date	Time (on/off)	Mon	Tue	Wed	Thur	Fri	Sat	Sun
	05/04/21	08/04/21	10/04/21	TH	D	D	D	D	D	D	D
					N	N	N	N	N	N	N

Shade or Circle Shift Required

4. Details of Work

Location: Northfields Substation	Project/ Maintenance Title DC Maintenance	Description of work & Plant item Maintenance of Red, Green, Mauve & Yellow Disc Panel
--------------------------------------------	-----------------------------------------------------	-----------------------------------------------------------------------------------------------------

5. Additional information


Thursday 8th April – Rack out Red & Green Sections to enable Disconnection Panel Maintenance
Saturday 10th April – Rack out Mauve & Yellow Section to enable Disconnection Panel Maintenance

6. Arrangements Required (Specify When Made By Requester)

Resources (AP Power/3 rd Party) & Date/Shift Nil	Plant Outage & Date/Shift Thursday 8th April Red & Green Saturday 10th April Mauve & Yellow
Access Booking & Date/Shift Thursday 8th & Saturday 10th April TH	Method Statement/Waiver No. WD-02470.5
Project Number 1020884	

OA Reference Number
The OA must be completed by the LU Project Engineer / Manager or representative who is responsible for undertaking the work on site.

7. Prepared by

Name J Postlethwaite	Signature 	Date 3rd March 2021
--------------------------------	--------------------------------------------------------------------------------------------------	------------------------------------------

8. Restrictions/Remarks (to be completed by Operations Planning)

Details

9. Provisional Approval Granted by Operations Planning

Name	Signature	Date	Approved (Yes/No)

Form 086/1

Figura 3.1: Esempio di documento Planning Request

Se per qualche motivo il documento viene inviato in maniera fisica (ad esempio mediante fax o posta), l'addetto si dovrà occupare di scannerizzarlo e inviarlo alla Power Planning Queue.

Per la creazione di una Planning Request viene sfruttata la funzionalità standard di promozione di un elemento di una Queue in un altro record (come ad esempio un Lead). In questo modo la Planning Request creata erediterà le informazioni, ad esempio il contatto, dall'email. Quest'ulti-

ma, inoltre, verrà associata alla Planning Request in modo che venga aggiunta allo storico delle comunicazioni ad essa relative. In aggiunta, il sistema consente di creare una Planning Request anche senza partire da un elemento di una coda. In questo caso sarà l'addetto a dover inserire manualmente le informazioni richieste.

Una Planning Request appena creata entra in fase di approvazione. Le entità del CRM con cui si ha a che fare durante questa fase del processo sono:

- Planning Requests
- Jobs
 - Job Schedules
 - Job Resources
 - Work Documents
 - Job Arrangements
 - Job Outages

Tra queste entità vi sono le seguenti relazioni: Planning Request è in relazione 1 a N con Jobs mentre quest'ultima è in relazione 1 a N con Job Schedules, Job Resources, Work Documents, Job Arrangements e Job Outages.

L'addetto ha il compito di compilare manualmente i campi dei relativi form del CRM, creando così dei record, sulla base del contenuto del documento in allegato all'email che avvia il processo di creazione di una Planning Request. Ciò che mi è stato chiesto di fare è di automatizzare quanto più possibile del processo appena visto in modo da ridurre al minimo l'intervento umano.

3.2 Sviluppo del progetto

Lo sviluppo è stato diviso in due fasi:

- Creazione e training del modello AI Builder per la raccolta automatica dei dati dal documento.
- Sviluppo del flusso cloud Power Automate.

3.2.1 Modello AI Builder

Per l'estrazione automatica dei dati dal documento è stato scelto di utilizzare il modello personalizzato di elaborazione di moduli di AI Builder.

Il mio supervisore ha specificato che era sufficiente estrarre le seguenti informazioni:

- Work Type
- Work Party
- Resources
- Location
- Originator Reference
- Description
- Comment
- Schedule (tabella che specifica giorni e fasce orarie di lavoro)

Dopo aver scaricato dal database del CRM un dataset di 500 documenti compilati appartenenti a Planning Request già esistenti, ho effettuato un'estrazione casuale di 20 documenti mediante un semplice script Python e li ho utilizzati come training set per il modello AI Builder, completando la

irration)

Mon	Tue	Wed	Thur	Fri	Sat	Sun
D	D	D	D	D	D	D
N	N	N	N	N	N	N
'Shade or Circle' Shift Required						

(a) Tabella senza modifiche.

Mon	Tue	Wed	Thur	Fri	Sat	Sun
D	D	D	D	D	D	D
N	N	N	N	N	N	N
'Shade or Circle' Shift Required						

(b) Altra tabella senza modifiche.

irration)

Mon	Tue	Wed	Thur	Fri	Sat	Sun
D	D	D	D	D	D	D
	N	N	N	N	N	N
'Shade or Circle' Shift Required						

(c) Tabella modificata con annerimento totale.

irration)

Mon	Tue	Wed	Thur	Fri	Sat	Sun
	X					
'Shade or Circle' Shift Required						

(d) Tabella modificata con carattere 'X'.

Figura 3.2: Esempi di tabelle con e senza modifiche.

fase di etichettatura manuale descritta in Sezione 2.3.1. Il numero di documenti scelto per il training è stato di 20 documenti per trovare un compromesso tra il tempo necessario all'etichettatura e il rischio di *overfitting*¹.

Problematiche incontrate

Durante l'utilizzo del modello sono sorte diverse problematiche. Sebbene il training fosse stato effettuato su un campione di documenti eterogeneo nei contenuti ma con la stessa struttura, il modello ha manifestato cattiva performance in caso di documenti con campi dimensionati o posizionati diversamente, anche se in maniera lieve, rispetto ai documenti utilizzati per il training.

Particolare attenzione è stata dedicata, inoltre, alla tabella *Schedule*, che è risultata difficile da trattare a causa delle limitazioni di AI Builder. Come spiegato in Sezione 2.3.1, non è supportato il riconoscimento di caselle di controllo, funzionalità che sarebbe stata quanto necessario per la corretta estrazione dei dati dalla tabella. Essa infatti viene compilata evidenziando, cerchiando o annerendo la cella desiderata. Estrarre dalla tabella giorni e fasce orarie desiderate è risultato pertanto impossibile.

Non potendo intervenire su AI Builder, ho tentato strade alternative apportando modifiche ai documenti stessi e creando nuovi modelli allenati su dataset appositamente preparati. In Figura 3.2 alcuni esempi.

Nonostante questi tentativi nei documenti forniti per il testing la tabella *Schedule* non veniva mai correttamente rilevata. Nel caso del tentativo basato sull'annerimento totale della cella desiderata, essendo il riconoscimento di caselle di controllo non supportato, era facilmente prevedibile che ciò non avvenisse. Il tentativo con i caratteri 'X', invece, si basava sull'assunto che il modello è stato progettato per il riconoscimento di campi testuali e quindi avrebbe potuto avere maggiori probabilità di successo nel rilevare i caratteri 'X' nelle celle corrispondenti. Anche questo tentativo non ha avuto esito positivo in quanto le tabelle non venivano rilevate affatto (l'oggetto JSON di output risultava vuoto).

È stato deciso dunque di ignorare la tabella *Schedule* e, di conseguenza, evitare poi di compilare i campi dell'entità Job Schedule ad essa corrispondente. La parte di processo che ho dovuto automatizzare termina dunque con la creazione del Job.

3.2.2 Flusso Power Automate

Lo sviluppo del flusso cloud su Power Automate è stato abbastanza semplice e diretto e non ho riscontrato grosse difficoltà. In Figura 3.3 è presente il flusso completo mentre in Figura 3.4 è

¹Condizione in cui un modello di machine learning si adatta in maniera esatta al set di dati di training. Come conseguenza, il modello non generalizza e quindi avrà una cattiva performance su dati mai visti.

presente il blocco di creazione del record Planning Request.

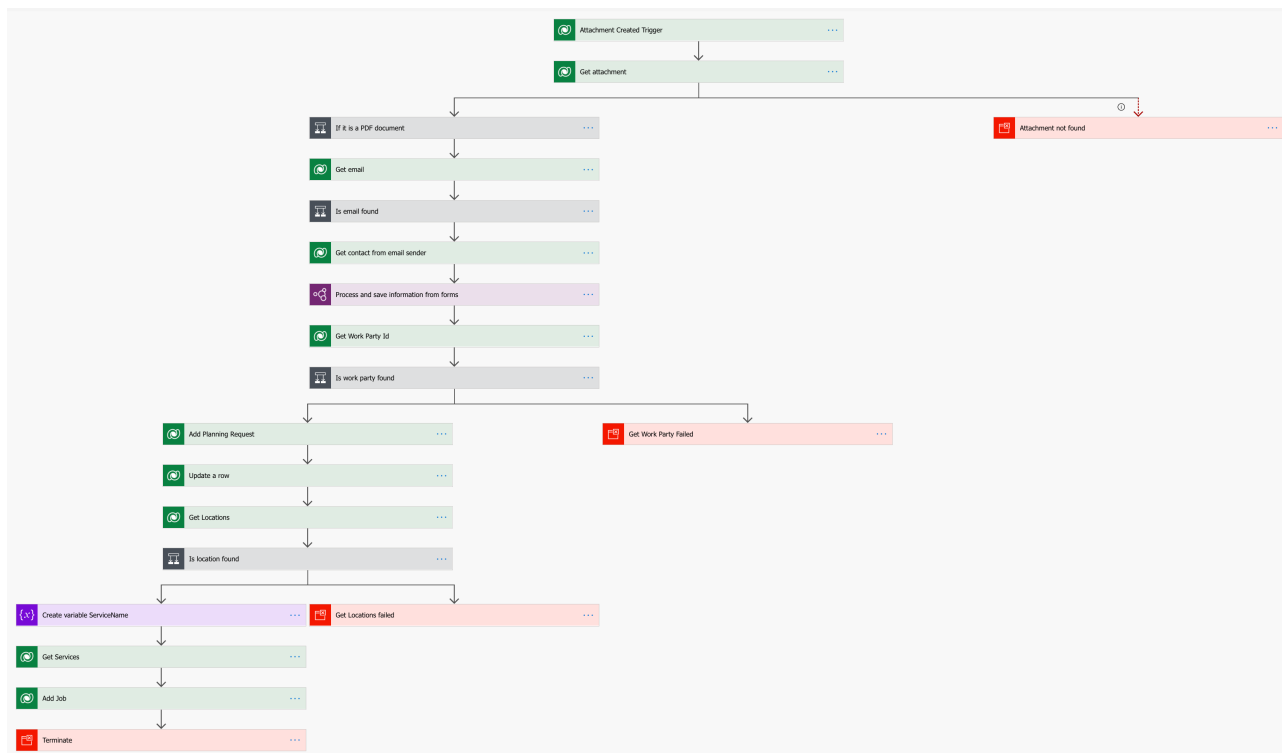


Figura 3.3: Flusso cloud Power Automate per l'automatizzazione della creazione di Planning Request.

Il flusso si compone di diverse fasi:

1. Trigger sulla creazione di un record Attachment nel CRM, verifica che l'allegato che ha attivato il trigger sia di tipo PDF e verifica che l'allegato in questione sia in relazione con un record Email.
2. Elaborazione del documento mediante AI Builder
3. Ottenimento dell'ID univoco del Work Party da associare alla Planning Request, a partire dal valore estratto dal documento.
4. Creazione di un record Planning Request e aggiornamento del campo Regarding dell'email per cui il flusso è in esecuzione, in modo da associare l'email alla Planning Request.
5. Ottenimento dell'ID univoco della Location da associare al Job relativo alla Planning Request, a partire dal valore estratto dal documento.
6. Ottenimento dell'ID univoco del Service da associare Job relativo alla Planning Request, a partire dal valore estratto dal documento.
7. Creazione del Job, utilizzando i dati ottenuti in precedenza e quelli estratti dal documento.

È importante notare come sia stato necessario utilizzare un trigger sulla creazione di un record Attachment invece che Email. Questo perché quando il CRM riceve un'email, questa viene salvata per prima e in seguito eventuali allegati. Utilizzando un trigger sulla creazione di un record Email, il flusso sarebbe stato attivato anche per quelle email senza allegato, a discapito delle performance.

Add Planning Request

- * Table name: Planning Requests
- * Case Title:
- * Cost Area (Cost Areas):
- * Delivery Area (Delivery Areas):
- * Primary Contact (Contacts):
- * Receipt Date:
- Assigned To (Owners): Unique identifier of the user or team who owns the case.
- Case Type: Information that specifies the type of case.
- Description:
- DO NOT USE_Priority: Priority of the case.
- Parent Case (Planning Requests): Choose the parent case for a case.
- Status Reason: Reason for the status of the case.
- Subject (Subjects): Choose the subject for the case, such as catalog request or product complaint.
- Work Party (Contacts): Select the customer account or contact to provide a quick link to additional cus
- Work Party (Work Parties):

[Show advanced options](#)

Figura 3.4: Blocco di creazione di una Planning Request.

3.3 Considerazioni sul progetto

Al termine dello sviluppo, posso dirmi non pienamente soddisfatto del risultato finale. Sebbene il flusso cloud implementato sia funzionante, in uno scenario di utilizzo reale non consentirebbe un alleggerimento significativo del carico di lavoro manuale che gli operatori del cliente avrebbero da svolgere.

Il processo di estrazione dei valori e compilazione automatica della Planning Request e delle entità ad essa associate ha un tasso di successo di circa il 10%. I test svolti utilizzando moduli estratti dal dataset menzionato in Sezione 3.2.1 hanno avuto come esito, nella maggior parte dei casi, un'interruzione dell'esecuzione del flusso oppure una Planning Request mal compilata.

Questo comportamento può essere dovuto principalmente al fatto che i moduli, essendo compilati a mano da personale umano, a volte non contengono gli stessi nominativi di Work Party, Location o altri campi presenti nel CRM. I valori di testo estratti dal modello di elaborazione di moduli, quindi, non danno nessun risultato in seguito alla ricerca del relativo record ed essendo le entità utilizzate strettamente interconnesse da relazioni, non risulta possibile completare il processo in modo corretto.

Mantenendo l'automatizzazione del processo come obiettivo, una soluzione a questa problematica potrebbe essere modificare la metodologia con cui l'azienda partner comunica la richiesta di effettuare un intervento. Infatti, informatizzando la fase di compilazione della richiesta mediante un'interfaccia web standardizzata si risolverebbe il problema in modo semplice ed efficace. I campi, ad esempio, potrebbero essere compilati attraverso dei menù *dropdown* dotati di funzionalità di ricerca.

Tuttavia, dopo aver proposto questa soluzione al mio supervisore, mi è stato riferito che Cluster Reply ci aveva già pensato ma non era una strada percorribile in quanto avrebbe creato troppa discontinuità con le metodologie attualmente in uso e sarebbe stato un investimento che il cliente non avrebbe scelto di fare.

4 Secondo Progetto

Il secondo progetto assegnatomi durante il periodo di tirocinio mi ha permesso di approfondire ancora di più le tecnologie trattate nel Capitolo 2 e di utilizzare le competenze di programmazione apprese durante il percorso di Laurea, sviluppando dei Plugin in linguaggio C#.

La motivazione che ha spinto il supervisore aziendale ad affidarmi questo progetto risiede nel fatto che l'azienda necessitava di una *demo proof-of-concept* da utilizzare come esempio delle possibilità di customizzazione del CRM per i potenziali clienti. Il progetto sviluppato infatti modella un processo di business¹ per le società che operano nel settore della fornitura energetica per aziende.

Lo scenario fittizio che il tutor aziendale, insieme all'analista funzionale, ha progettato per me è quello di un'azienda che offre forniture di energia elettrica e gas per aziende. Vista l'alta mole di lavoro, tale società necessita di un sistema che permetta di velocizzare e rendere più efficiente il processo di creazione di preventivi e offerte personalizzate per i propri clienti e quindi ha commissionato la soluzione di questo problema a Cluster Reply.

4.1 Processo di business

Il processo di business che mi è stato chiesto di modellare consiste nella gestione semi-automatica di una richiesta di fornitura di energia personalizzata per una azienda cliente, basata sul processo standard del CRM Lead-Opportunity-Quote-Order del modulo funzionale Sales e sui Case del modulo funzionale Service. In Figura 4.1 un diagramma del processo modellato.

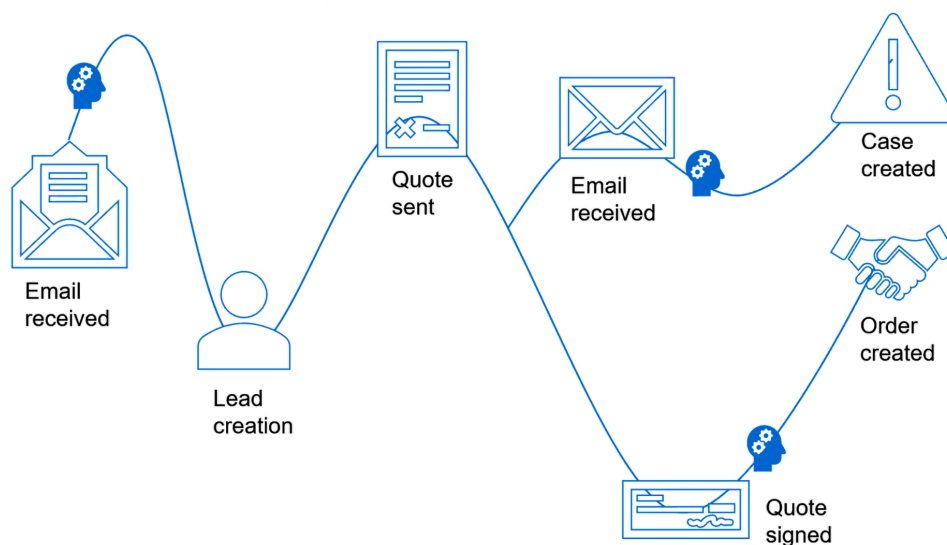


Figura 4.1: Processo di business modellato dal progetto.

Nel dettaglio si compone dei seguenti passaggi:

1. Arrivo di un email da parte di un potenziale cliente con in allegato un modulo compilato. Il modulo contiene informazioni riguardanti il cliente stesso (anagrafica) e le necessità di fornitura di energia per le quali ha effettuato la richiesta di preventivo.
2. Il documento viene processato automaticamente mediante AI Builder e viene creato un nuovo record Lead nel CRM. Nel caso in cui il documento non venga processato correttamente

¹Anche se semplificato.

o si verificano errori viene inviata una email di cortesia al potenziale cliente che notifihi l'errore.

3. In seguito alla creazione del Lead quest'ultimo viene automaticamente *qualificato* in un record Opportunity mediante un Plugin.
4. Un operatore compila manualmente il preventivo (Quote) di fornitura basandosi sui dati raccolti in precedenza.
5. Quando il preventivo viene contrassegnato come "attivo", viene avviato un secondo plugin che invia automaticamente un'email al cliente contenente il preventivo come allegato.
6.
 - Se il cliente risponde all'email con il preventivo firmato (con il controllo effettuato automaticamente grazie a un modello AI Builder), il Quote si considera vinto e viene convertito in Order per poi essere ultimato da un operatore.
 - Se il cliente risponde all'email in qualsiasi altro modo, un modello di classificazione di categoria di AI Builder analizza il testo dell'email. Sulla base dell'output di questo modello, viene creato un record Case e assegnato a un'apposita coda del modulo funzionale Service per essere processato da un operatore.

4.2 Sviluppo del progetto

In seguito alla definizione dettagliata dei requisiti ho proceduto con lo sviluppo del progetto che ha richiesto l'utilizzo di Power Automate, AI Builder, Plugin e customizzazioni di form e regole di *routing* per l'instradamento dei Case nelle varie Queue.

4.2.1 AI Builder

Sono stati utilizzati due diversi tipi di modelli AI Builder:

- Modello personalizzato per l'elaborazione di moduli - utilizzato per l'estrazione di dati dal modulo di richiesta preventivo e per verificare l'accettazione del preventivo.
- Modello predefinito di classificazione delle categorie per classificare il contenuto dell'email di risposta all'invio del preventivo.

Modulo di richiesta preventivo

Avendo praticamente carta bianca su come sviluppare il modulo ho potuto evitare le problematiche riscontrate durante il primo progetto. In Figura 4.2 un esempio di modulo. Per questo progetto, come indicato dal mio supervisore, non mi sono soffermato in modo eccessivo sul training del modello e ho utilizzato semplicemente un campione di 5 documenti compilati. Per brevità, inoltre, non riporto l'elenco dei campi riconosciuti, in quanto corrispondono alle celle compilabili del documento in Figura 4.2 a fronte.

Documento con preventivo accettato

Per l'accettazione del preventivo si è optato per l'apposizione di data e firma. È stato dunque creato un modello di elaborazione di moduli che rilevasse nome del cliente, ID del Quote, data e firma. Tuttavia, come riportato in Sezione 2.3.1, il rilevamento di firme non è ancora supportato da AI Builder ma stando a informazioni interne giunte a Cluster Reply, lo sarà in futuro. Pertanto, in fase di validazione del documento, la firma verrà semplicemente ignorata e i controlli verranno effettuati solamente sui restanti campi analizzati. In Figura 4.3 un esempio di documento utilizzato per questo modello. Anche in questo caso per il training è stato utilizzato un campione di 5 documenti compilati.

Richiesta di preventivo

Nome	Filippo
Cognome	Momesso
Email	f.momesso@reply.it
Cellulare	1234567890
Nome azienda	Cluster Reply Srl



Sede legale:

Via	Via Avogadri 2	Città	Silea
CAP	31057	Provincia	TV

Sede per cui è richiesta la fornitura (lasciare vuoto se coincide con sede legale):

Via	Via Avogadri 2	Città	Silea
CAP	31057	Nazione	TV

Breve descrizione delle necessità di fornitura dell'azienda:

Cluster Reply è la società del gruppo Reply specializzata in servizi di consulenza e di integrazione di sistemi su tecnologie Microsoft. Opera in Italia e collabora con altre aziende del gruppo che sono partner Microsoft in Brasile, Germania, Regno Unito e Stati Uniti.

Settore di business	Financial	Tipo di fornitura	Gas & Electricity
---------------------	-----------	-------------------	-------------------

Tipo di contratto attuale	Industriale
---------------------------	-------------

Classe energetica	A+	Superficie immobile (m ²)	2500
-------------------	----	---------------------------------------	------

Consumi elettricità (kWh)					
Gen-Feb	Mar-Apr	Mag-Giu	Lug-Ago	Set-Ott	Nov-Dic
420	480	450	420	480	450
Destinazione d'uso principale			Ufficio		
Impianto di generazione proprio			Impianto fotovoltaico		
Tensione di allacciamento (V)			380		

Consumi gas (smc)					
Gen-Feb	Mar-Apr	Mag-Giu	Lug-Ago	Set-Ott	Nov-Dic
200	300	398	200	300	398
Destinazione d'uso principale			Office heating		

Note:

5 Questa è un'annotazione di prova.

(b) Pagina 2.

(a) Pagina 1.

Figura 4.2: Esempio di modulo utilizzato.

Offerta

ID OFFERTA: QUO-01007-2757F5
 AGGIORNAMENTO: 5
 TOTALE FATTURA: €95.02
 VALIDA DAL:
 VALIDA FINO AL:
 VENDITORE: PIERLUIGI DE DOMENICO

Quote Request Gas & Electricity

CLIENTE FATTURAZIONE:
 Cluster Reply Srl
 Via Avogadri 2,
 Silea, TV, 31057

DESCRIZIONE:
 Descrizione offerta test

PRODOTTI

DESCRIZIONE	QUANTITÀ	PREZZO UNITARIO	TOTALE
test	1.00000	€100.00	€95.00
Fornitura elettrica - 230V - 1,5 kW	1.00000	€0.02	€0.02
SUBTOTALE			€90.02
IVA			€5.00
SPEDIZIONE E GESTIONE			
TOTALE DOVUTO			€95.02

Data:	12/03/2021	Firma:	Cluster Reply Srl
-------	------------	--------	-------------------

Figura 4.3: Esempio di preventivo compilato e firmato per accettazione.

4.2.2 Flussi Power Automate

Sono stati sviluppati due flussi cloud: il primo copre le fasi del processo che vanno dalla ricezione dell'email con allegato fino alla creazione del Lead; il secondo, invece, si occupa di gestire quanto avviene dopo l'invio del preventivo, ovvero la risposta positiva o negativa del cliente.

Flusso di creazione del Lead

In Figura 4.4 è riportato il primo flusso cloud sviluppato, il quale si occupa di creare il Lead a partire dai dati estratti dal documento ricevuto come allegato dell'email.

Si compone di diverse fasi:

1. Trigger sulla creazione di un record Attachment.
2. Controlli su destinatario dell'email e tipo di documento.
3. Conversione del documento in PDF se l'allegato non è in formato PDF.
4. Estrazione dei dati dal documento mediante modello AI Builder.
5. Operazioni di validazione sui dati estratti
6. Creazione del Lead e aggiornamento del campo Regarding dell'email in modo da associarla al Lead.

Flusso di elaborazione della risposta del cliente

Il flusso che viene eseguito in seguito alla ricezione di un email di risposta a un preventivo è più articolato in quanto necessita di gestire due possibilità: offerta accettata oppure offerta rifiutata.

Si compone delle seguenti fasi:

1. Trigger sulla creazione di un record Email e controllo che l'email sia associata ad un record Quote. Il CRM si occupa in automatico di creare le relazioni se un'email viene ricevuta come risposta ad un'email associata ad un record mediante il campo Regarding.
2. Recupero di un eventuale allegato. Se presente il processo procede con:
 - (a) Conversione del documento in formato PDF se non è in formato PDF.
 - (b) Estrazione dei dati dal documento mediante modello AI Builder.
 - (c) Validazione sui dati estratti con controllo sulle confidence. Un valore medio di confidence sui valori riconosciuti inferiore a 0,7 rende non valido il documento.
 - (d) Se il documento è valido e firmato, il Quote è considerato vinto e pertanto convertito in Ordine.
3. Se il documento non è presente o risulta non valido, segue l'analisi del testo dell'email mediante il modello di classificazione di categorie.
4. Creazione del Case il cui campo Description contiene l'output del modello utilizzato nella fase precedente e aggiornamento del campo Regarding dell'email in maniera tale da associarla al Case appena creato.

4.2.3 Plugin

Sono stati sviluppati due Plugin che svolgono le seguenti operazioni:

- Conversione del Lead in Opportunity.
- Invio automatico di email con preventivo in allegato.

Come strumenti di sviluppo ho utilizzato Microsoft Visual Studio unitamente a Microsoft XRM Toolbox, utile per il debugging e la registrazione dei plugin nella soluzione del CRM.

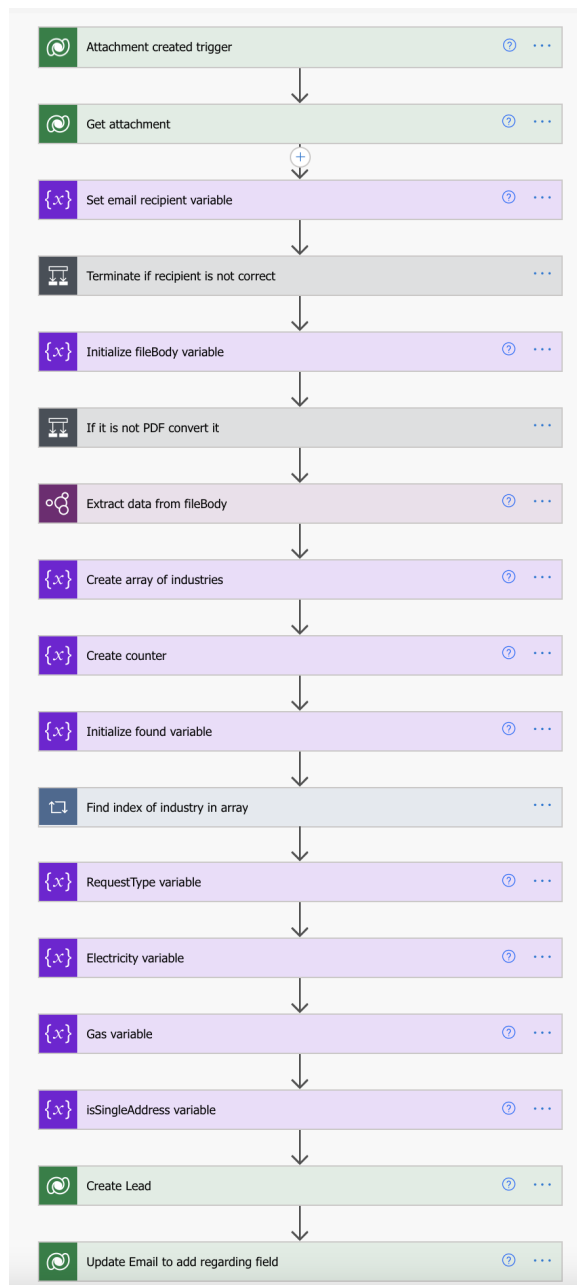


Figura 4.4: Flusso di creazione del Lead

Conversione del Lead in Opportunity

Per la conversione del Lead in Opportunity ho scelto di utilizzare un Plugin registrato sul Create Message in fase Post-operation dell'entità Lead. In questo modo la sua esecuzione viene avviata al termine della creazione del Lead.

Mediante un'istanza di `IOrganizationService` si ottiene il contesto in cui il plugin viene eseguito, e in questo modo è possibile, ad esempio, avere informazioni sul record che ha avviato l'esecuzione. La classe `ITracingService` permette di stampare dei log sul CRM.

Per prima cosa il codice effettua dei controlli sull'esistenza di un Account o Contact a cui associare l'Opportunity una volta convertita in Lead. Con la classe `QueryExpression` è possibile, infatti, interrogare il CRM per ottenere dei record, come fosse una `SELECT` su un database. Successivamente viene utilizzato un `Message QualifyLeadRequest`, che fornisce la funzionalità standard di conversione di un Lead in Opportunity. A quest'ultimo vengono passati come parametri il Lead da qualificare ed eventuali riferimenti a Contact o Account se già esistenti a sistema. `QualifyLeadRequest`, infatti, si occupa di default di creare Contact e Account per il potenziale

cliente a partire dal Lead e, per evitare problemi con possibili duplicati, ho deciso di disabilitare il controllo automatico e gestire i duplicati manualmente.

Altre operazioni svolte sono l'aggiornamento di alcuni campi dell'Opportunity creatasi dopo l'esecuzione del suddetto Message. In Appendice A.2 è presente il codice sorgente completo del Plugin.

Invio di email con preventivo

Per l'invio automatico dell'email con il preventivo, ho deciso di utilizzare un Plugin registrato sull'Update Message in fase Post-operation dell'entità Quote.

Per prima cosa viene controllato che il campo *statecode* sia stato aggiornato ad *attivo*, altrimenti l'esecuzione viene interrotta. Dopodiché viene generato il documento PDF con il preventivo, codificato in formato *base64*, mediante una chiamata all'Organization Web Service, in questo modo:

```
1 OrganizationRequest createPdfRequest = new OrganizationRequest("ExportPdfDocument");
2 createPdfRequest["EntityTypeCode"] = 1084; // Quote typecode
3 createPdfRequest["SelectedTemplate"] = new EntityReference("documenttemplate",
4   documentTemplateId);
5 createPdfRequest["SelectedRecords"] = "[\"{" + quote.Id.ToString() + "}\"]";
6
7 OrganizationResponse createPdfResponse =
8   (OrganizationResponse)service.Execute(createPdfRequest);
9
10 return Convert.ToBase64String((byte[])createPdfResponse["PdfFile"]);
```

Viene quindi creata l'email a partire da un template presente sul CRM. Per fare ciò prima si recupera il template attraverso una QueryExpression e, successivamente, il template viene istanziato in un'email grazie a una chiamata al Message IstantiateTemplateRequest. L'email viene poi completata con i valori *destinatario* e *mittente* e vien quindi creato un record Attachment sul CRM utilizzando il documento in formato *base64* generato in precedenza. Infine, l'email viene inviata mediante un Message SendEmailRequest.

In Appendice A.3 è presente il codice sorgente completo del plugin.

4.2.4 Gestione del Case

Nell'eventualità che il flusso di elaborazione della risposta del cliente rilevi un documento non valido oppure non presente, procede con la creazione di un record Case la cui descrizione contiene l'output del modello di classificazione di categorie. Le categorie possibili sono le seguenti:

- Issues
- Customer Service
- Documentation
- Price & Billing
- Compliment
- Staff

Per facilitare la gestione dei Case da parte degli addetti del settore Service ho creato quattro Queue, in cui i Case vengono distribuiti grazie a delle *routing rules* (in italiano *regole di instradamento*), una funzionalità del CRM che permette di redistribuire automaticamente i record in Queue sulla base di regole appositamente definite. In Tabella 4.1 è possibile osservare come ho scelto di distribuire le sei tipologie di Case nelle rispettive Queue.

Tipologia di Case	Queue
Issues Customer Service	Problemi
Documentation	Documentazione
Price & Billing	Prezzo e Fatturazione
Compliment Staff	Generale

Tabella 4.1: Tipi di Web Resource supportati

4.3 Considerazioni sul progetto

Diversamente dal precedente, mi ritengo soddisfatto dell'esito di questo progetto. Salvo qualche intoppo durante lo sviluppo, risolto grazie all'aiuto dei colleghi più esperti, sono riuscito a portare a termine quanto richiesto e ho ottenuto un feedback positivo da parte del mio supervisore aziendale sul lavoro svolto. Quest'ultimo, inoltre, mi ha fatto sapere che, vista la qualità del risultato, sarebbe stato realmente utilizzato come *demo* da mostrare ai potenziali clienti di Cluster Reply.

5 Conclusioni

Bibliografia

- [1] Peter Hecke. When to use plug-ins. [https://docs.microsoft.com/en-us/previous-versions/dynamics-crm4/developer-articles/dd393303\(v=crm.6\)?redirectedfrom=MSDN#when-to-use-plug-ins](https://docs.microsoft.com/en-us/previous-versions/dynamics-crm4/developer-articles/dd393303(v=crm.6)?redirectedfrom=MSDN#when-to-use-plug-ins), 2009.
- [2] Microsoft. Introduction to solutions. <https://docs.microsoft.com/it-it/dynamics365/customerengagement/on-premises/developer/introduction-solutions?view=op-9-1>, 2017.
- [3] Microsoft. Schema reference guide for the workflow definition language in azure logic apps. <https://docs.microsoft.com/it-it/azure/logic-apps/logic-apps-workflow-definition-language>, 2017.
- [4] Microsoft. Creare e gestire le code di base. <https://docs.microsoft.com/it-it/dynamics365/customer-service/set-up-queues-manage-activities-cases>, 2021.
- [5] Cluster Reply. Automazione delle attività nella gestione degli affitti arretrati. <https://www.reply.com/cluster-reply/it/automazione-delle-attivita-nella-gestione-degli-affitti-arretrati-in-NHG>.
- [6] Cluster Reply. Configurable workflow & slas engine. <https://www.reply.com/cluster-reply/it/configurable-workflow-slas-engine>.
- [7] Tutorialspoint. Microsoft dynamics crm - quick guide. https://www.tutorialspoint.com/microsoft_crm/microsoft_crm_quick_guide.htm.
- [8] Business Wire. Reply: Il c.d.a. approva il progetto di bilancio per l'esercizio 2020. <https://www.businesswire.com/news/home/20210315005472/it/>, March 2015.

Allegato A Codice sorgente

In questo allegato si trovano i listati citati durante la trattazione.

A.1 Output del Modello di Elaborazione di Moduli

```
1 {
2   "@odata.context":
3     "https://replyforutilities.crm4.dynamics.com/api/data/v9.1/$metadata#Microsoft.Dynamics.CRM.",
4   "responsev2": {
5     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
6     "operationStatus": "Success",
7     "predictionOutput": {
8       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
9       "pageCount": 1,
10      "layoutName": "Adatum",
11      "layoutConfidenceScore": 0.964,
12      "readResults@odata.type": "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
13      "readResults": [],
14      "tables": {
15        "@odata.type": "#Microsoft.Dynamics.CRM.expando",
16        "Items": {
17          "@odata.type": "#Microsoft.Dynamics.CRM.expando",
18          "displayName": "Items",
19          "entries@odata.type": "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
20          "entries": [
21            {
22              "@odata.type": "#Microsoft.Dynamics.CRM.expando",
23              "Line_002ab8055a39c0312b2ebd69806b8c27b87": {
24                "@odata.type": "#Microsoft.Dynamics.CRM.expando",
25                "value": "$32.99",
26                "location": {
27                  "@odata.type": "#Microsoft.Dynamics.CRM.expando",
28                  "pageNumber": 1,
29                  "boundingBox": {
30                    "@odata.type": "#Microsoft.Dynamics.CRM.expando",
31                    "left": 0.7617647058823529,
32                    "top": 0.5372727272727272,
33                    "width": 0.055882352941176494,
34                    "height": 0.012272727272727324,
35                    "polygon": {
36                      "@odata.type": "#Microsoft.Dynamics.CRM.expando",
37                      "coordinates@odata.type":
38                        "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
39                      "coordinates": [
40                        {
41                          "@odata.type": "#Microsoft.Dynamics.CRM.expando",
42                          "x": 0.7617647058823529,
43                          "y": 0.5372727272727272
44                        }
45                      ]
46                    }
47                  }
48                }
49              }
50            ]
51          }
52        }
53      }
54    }
55  }
```

```

43         {
44             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
45             "x": 0.8176470588235294,
46             "y": 0.5372727272727272
47         },
48         {
49             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
50             "x": 0.8176470588235294,
51             "y": 0.5495454545454546
52         },
53         {
54             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
55             "x": 0.7617647058823529,
56             "y": 0.5495454545454546
57         }
58     ]
59 }
60 }
61 }
62 },
63 "Quantity": {
64     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
65     "value": "01",
66     "location": {
67         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
68         "pageNumber": 1,
69         "boundingBox": {
70             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
71             "left": 0.08,
72             "top": 0.5381818181818182,
73             "width": 0.01705882352941175,
74             "height": 0.010454545454545494,
75             "polygon": {
76                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
77                 "coordinates@odata.type":
78                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
79                 "coordinates": [
80                     {
81                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
82                         "x": 0.08,
83                         "y": 0.5381818181818182
84                     },
85                     {
86                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
87                         "x": 0.09705882352941175,
88                         "y": 0.5381818181818182
89                     },
90                     {
91                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
92                         "x": 0.09705882352941175,
93                         "y": 0.5486363636363637
94                     },
95                     {
96                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
97                         "x": 0.08,
98                         "y": 0.5486363636363637
99                     }
100                 ]
101             }
102         }

```

```

103 },
104 "Description": {
105     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
106     "value": "Safe pool alarm",
107     "location": {
108         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
109         "pageNumber": 1,
110         "boundingBox": {
111             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
112             "left": 0.2776470588235294,
113             "top": 0.5381818181818182,
114             "width": 0.12823529411764706,
115             "height": 0.013181818181818183,
116             "polygon": {
117                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
118                 "coordinates@odata.type":
119                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
120                 "coordinates": [
121                     {
122                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
123                         "x": 0.2776470588235294,
124                         "y": 0.5381818181818182
125                     },
126                     {
127                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
128                         "x": 0.40588235294117647,
129                         "y": 0.5381818181818182
130                     },
131                     {
132                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
133                         "x": 0.40588235294117647,
134                         "y": 0.5513636363636364
135                     },
136                     {
137                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
138                         "x": 0.2776470588235294,
139                         "y": 0.5513636363636364
140                     }
141                 ]
142             }
143         }
144     },
145     "Unit_002e5018fc9450544ffc2f82e17655cf519": {
146         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
147         "value": "$32.99",
148         "location": {
149             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
150             "pageNumber": 1,
151             "boundingBox": {
152                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
153                 "left": 0.5735294117647058,
154                 "top": 0.5372727272727272,
155                 "width": 0.05529411764705883,
156                 "height": 0.012272727272727324,
157                 "polygon": {
158                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
159                     "coordinates@odata.type":
160                         "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
161                     "coordinates": [

```

```

162         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
163         "x": 0.5735294117647058,
164         "y": 0.5372727272727272
165     },
166     {
167         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
168         "x": 0.6288235294117647,
169         "y": 0.5372727272727272
170     },
171     {
172         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
173         "x": 0.6288235294117647,
174         "y": 0.5495454545454546
175     },
176     {
177         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
178         "x": 0.5735294117647058,
179         "y": 0.5495454545454546
180     }
181 ]
182 }
183 }
184 }
185 }
186 },
187 {
188     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
189     "Line_002ab8055a39c0312b2ebd69806b8c27b87": {
190         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
191         "value": "$260.91",
192         "location": {
193             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
194             "pageNumber": 1,
195             "boundingBox": {
196                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
197                 "left": 0.7617647058823529,
198                 "top": 0.5681818181818182,
199                 "width": 0.06294117647058828,
200                 "height": 0.012272727272727213,
201                 "polygon": {
202                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
203                     "coordinates@odata.type":
204                         "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
205                     "coordinates": [
206                         {
207                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
208                             "x": 0.7617647058823529,
209                             "y": 0.5681818181818182
210                         },
211                         {
212                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
213                             "x": 0.8247058823529412,
214                             "y": 0.5681818181818182
215                         },
216                         {
217                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
218                             "x": 0.8247058823529412,
219                             "y": 0.5804545454545454
220                         },
221                         {
222                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",

```



```

222         "x": 0.7617647058823529,
223         "y": 0.5804545454545454
224     }
225 ]
226 }
227 }
228 }
229 },
230 "Quantity": {
231     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
232     "value": "09",
233     "location": {
234         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
235         "pageNumber": 1,
236         "boundingBox": {
237             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
238             "left": 0.08,
239             "top": 0.5690909090909091,
240             "width": 0.01941176470588235,
241             "height": 0.010454545454545494,
242             "polygon": {
243                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
244                 "coordinates@odata.type":
245                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
246                 "coordinates": [
247                     {
248                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
249                         "x": 0.08,
250                         "y": 0.5690909090909091
251                     },
252                     {
253                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
254                         "x": 0.09941176470588235,
255                         "y": 0.5690909090909091
256                     },
257                     {
258                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
259                         "x": 0.09941176470588235,
260                         "y": 0.5795454545454546
261                     },
262                     {
263                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
264                         "x": 0.08,
265                         "y": 0.5795454545454546
266                     }
267                 ]
268             }
269         }
270     },
271     "Description": {
272         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
273         "value": "Filter cartridge",
274         "location": {
275             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
276             "pageNumber": 1,
277             "boundingBox": {
278                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
279                 "left": 0.2782352941176471,
280                 "top": 0.5690909090909091,
281                 "width": 0.11823529411764705,

```

```

282 "height": 0.013181818181818183,
283 "polygon": {
284   "@odata.type": "#Microsoft.Dynamics.CRM.expando",
285   "coordinates@odata.type":
286     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
287   "coordinates": [
288     {
289       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
290       "x": 0.2782352941176471,
291       "y": 0.5690909090909091
292     },
293     {
294       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
295       "x": 0.39647058823529413,
296       "y": 0.5690909090909091
297     },
298     {
299       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
300       "x": 0.39647058823529413,
301       "y": 0.5822727272727273
302     },
303     {
304       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
305       "x": 0.2782352941176471,
306       "y": 0.5822727272727273
307     }
308   ]
309 }
310 },
311 "Unit_002e5018fc9450544ffc2f82e17655cf519": {
312   "@odata.type": "#Microsoft.Dynamics.CRM.expando",
313   "value": "$28.99",
314   "location": {
315     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
316     "pageNumber": 1,
317     "boundingBox": {
318       "@odata.type": "#Microsoft.Dynamics.CRM.expando",
319       "left": 0.5735294117647058,
320       "top": 0.5681818181818182,
321       "width": 0.05529411764705883,
322       "height": 0.012272727272727213,
323       "polygon": {
324         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
325         "coordinates@odata.type":
326           "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
327         "coordinates": [
328           {
329             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
330             "x": 0.5735294117647058,
331             "y": 0.5681818181818182
332           },
333           {
334             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
335             "x": 0.6288235294117647,
336             "y": 0.5681818181818182
337           },
338           {
339             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
340             "x": 0.6288235294117647,

```

```

341         "y": 0.5804545454545454
342     },
343     {
344         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
345         "x": 0.5735294117647058,
346         "y": 0.5804545454545454
347     }
348 ]
349 }
350 }
351 }
352 }
353 },
354 {
355     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
356     "Line_002ab8055a39c0312b2ebd69806b8c27b87": {
357         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
358         "value": "$154.99",
359         "location": {
360             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
361             "pageNumber": 1,
362             "boundingBox": {
363                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
364                 "left": 0.7617647058823529,
365                 "top": 0.5990909090909091,
366                 "width": 0.0658823529411765,
367                 "height": 0.012272727272727213,
368                 "polygon": {
369                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
370                     "coordinates@odata.type":
371                         "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
372                     "coordinates": [
373                         {
374                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
375                             "x": 0.7617647058823529,
376                             "y": 0.5990909090909091
377                         },
378                         {
379                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
380                             "x": 0.8276470588235294,
381                             "y": 0.5990909090909091
382                         },
383                         {
384                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
385                             "x": 0.8276470588235294,
386                             "y": 0.6113636363636363
387                         },
388                         {
389                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
390                             "x": 0.7617647058823529,
391                             "y": 0.6113636363636363
392                         }
393                     ]
394                 }
395             }
396         },
397         "Quantity": {
398             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
399             "value": "01",
400             "location": {

```

```

401     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
402     "pageNumber": 1,
403     "boundingBox": {
404         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
405         "left": 0.08,
406         "top": 0.6,
407         "width": 0.01705882352941175,
408         "height": 0.010454545454545494,
409         "polygon": {
410             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
411             "coordinates@odata.type":
412                 "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
413             "coordinates": [
414                 {
415                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
416                     "x": 0.08,
417                     "y": 0.6
418                 },
419                 {
420                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
421                     "x": 0.09705882352941175,
422                     "y": 0.6
423                 },
424                 {
425                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
426                     "x": 0.09705882352941175,
427                     "y": 0.6104545454545455
428                 },
429                 {
430                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
431                     "x": 0.08,
432                     "y": 0.6104545454545455
433                 }
434             ]
435         }
436     },
437 },
438 "Description": {
439     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
440     "value": "Pool pump",
441     "location": {
442         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
443         "pageNumber": 1,
444         "boundingBox": {
445             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
446             "left": 0.2782352941176471,
447             "top": 0.6,
448             "width": 0.08647058823529413,
449             "height": 0.013181818181818183,
450             "polygon": {
451                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
452                 "coordinates@odata.type":
453                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
454                 "coordinates": [
455                     {
456                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
457                         "x": 0.2782352941176471,
458                         "y": 0.6
459                     },

```

```

460         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
461         "x": 0.3647058823529412,
462         "y": 0.6
463     },
464     {
465         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
466         "x": 0.3647058823529412,
467         "y": 0.6131818181818182
468     },
469     {
470         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
471         "x": 0.2782352941176471,
472         "y": 0.6131818181818182
473     }
474 ]
475 }
476 }
477 }
478 },
479 "Unit_002e5018fc9450544ffc2f82e17655cf519": {
480     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
481     "value": "$154.99",
482     "location": {
483         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
484         "pageNumber": 1,
485         "boundingBox": {
486             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
487             "left": 0.5735294117647058,
488             "top": 0.5990909090909091,
489             "width": 0.06529411764705884,
490             "height": 0.012272727272727213,
491             "polygon": {
492                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
493                 "coordinates@odata.type":
494                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
495                 "coordinates": [
496                     {
497                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
498                         "x": 0.5735294117647058,
499                         "y": 0.5990909090909091
500                     },
501                     {
502                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
503                         "x": 0.6388235294117647,
504                         "y": 0.5990909090909091
505                     },
506                     {
507                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
508                         "x": 0.6388235294117647,
509                         "y": 0.6113636363636363
510                     },
511                     {
512                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
513                         "x": 0.5735294117647058,
514                         "y": 0.6113636363636363
515                     }
516                 ]
517             }
518         }
519     }

```

```

520     }
521 ],
522 "columns@odata.type": "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
523 "columns": [
524     {
525         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
526         "name": "Quantity"
527     },
528     {
529         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
530         "name": "Description"
531     },
532     {
533         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
534         "name": "Unit price"
535     },
536     {
537         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
538         "name": "Line total"
539     }
540 ]
541 }
542 },
543 "labels": {
544     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
545     "Total_0098ada642839a03b55e11ffdc9a8ddca": {
546         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
547         "displayName": "Total amount",
548         "value": "$457.87",
549         "confidence": 0.99,
550         "valueLocation": {
551             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
552             "pageNumber": 1,
553             "boundingBox": {
554                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
555                 "left": 0.13588235294117648,
556                 "top": 0.8663636363636363,
557                 "width": 0.07999999999999999,
558                 "height": 0.015454545454545388,
559                 "polygon": {
560                     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
561                     "coordinates@odata.type":
562                         "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
563                     "coordinates": [
564                         {
565                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
566                             "x": 0.13588235294117648,
567                             "y": 0.8663636363636363
568                         },
569                         {
570                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
571                             "x": 0.21588235294117647,
572                             "y": 0.8663636363636363
573                         },
574                         {
575                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
576                             "x": 0.21588235294117647,
577                             "y": 0.8818181818181817
578                         },
579                         {
580                             "@odata.type": "#Microsoft.Dynamics.CRM.expando",

```

```

580         "x": 0.13588235294117648,
581         "y": 0.8818181818181817
582     }
583 ]
584 }
585 }
586 }
587 },
588 "Invoice_85149f36226f35d3c565e5f0daa1f797": {
589     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
590     "displayName": "Invoice number",
591     "value": "1975",
592     "confidence": 0.995,
593     "valueLocation": {
594         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
595         "pageNumber": 1,
596         "boundingBox": {
597             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
598             "left": 0.36705882352941177,
599             "top": 0.15863636363636366,
600             "width": 0.05470588235294116,
601             "height": 0.014999999999999958,
602             "polygon": {
603                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
604                 "coordinates@odata.type":
605                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
606                 "coordinates": [
607                     {
608                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
609                         "x": 0.36705882352941177,
610                         "y": 0.15863636363636366
611                     },
612                     {
613                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
614                         "x": 0.42176470588235293,
615                         "y": 0.15863636363636366
616                     },
617                     {
618                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
619                         "x": 0.42176470588235293,
620                         "y": 0.17363636363636362
621                     },
622                     {
623                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
624                         "x": 0.36705882352941177,
625                         "y": 0.17363636363636362
626                     }
627                 ]
628             }
629         }
630     },
631 "Due_0020e1cb6d30fa3f17ee1e50b2bcf1967374": {
632     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
633     "displayName": "Due date",
634     "value": "Nov 12, 2020",
635     "confidence": 0.993,
636     "valueLocation": {
637         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
638         "pageNumber": 1,
639         "boundingBox": {

```

```

640     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
641     "left": 0.07764705882352942,
642     "top": 0.15863636363636366,
643     "width": 0.1511764705882353,
644     "height": 0.017272727272727245,
645     "polygon": {
646         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
647         "coordinates@odata.type":
648             "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
649         "coordinates": [
650             {
651                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
652                 "x": 0.07764705882352942,
653                 "y": 0.15863636363636366
654             },
655             {
656                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
657                 "x": 0.2288235294117647,
658                 "y": 0.15863636363636366
659             },
660             {
661                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
662                 "x": 0.2288235294117647,
663                 "y": 0.1759090909090909
664             },
665             {
666                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
667                 "x": 0.07764705882352942,
668                 "y": 0.1759090909090909
669             }
670         ]
671     }
672 }
673 },
674 "Customer62566c648cebabc3d75a987bb13e3bd7": {
675     "@odata.type": "#Microsoft.Dynamics.CRM.expando",
676     "displayName": "Customer ID",
677     "value": "5791",
678     "confidence": 0.994,
679     "valueLocation": {
680         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
681         "pageNumber": 1,
682         "boundingBox": {
683             "@odata.type": "#Microsoft.Dynamics.CRM.expando",
684             "left": 0.19294117647058823,
685             "top": 0.35318181818181815,
686             "width": 0.0388235294117647,
687             "height": 0.011363636363636354,
688             "polygon": {
689                 "@odata.type": "#Microsoft.Dynamics.CRM.expando",
690                 "coordinates@odata.type":
691                     "#Collection(Microsoft.Dynamics.CRM.crmbaseentity)",
692                 "coordinates": [
693                     {
694                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
695                         "x": 0.19294117647058823,
696                         "y": 0.35318181818181815
697                     },
698                     {
699                         "@odata.type": "#Microsoft.Dynamics.CRM.expando",

```



```

699         "x": 0.23176470588235293,
700         "y": 0.353181818181815
701     },
702     {
703         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
704         "x": 0.23176470588235293,
705         "y": 0.3645454545454545
706     },
707     {
708         "@odata.type": "#Microsoft.Dynamics.CRM.expando",
709         "x": 0.19294117647058823,
710         "y": 0.3645454545454545
711     }
712 ]
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }

```

Listing A.1: Oggetto JSON di output ritornato dal modello di elaborazione di moduli utilizzato come esempio in Sezione 2.3.1.

A.2 Codice sorgente Plugin di conversione Lead

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.ServiceModel;
5  using System.Text;
6  using System.Threading.Tasks;
7  using Microsoft.Crm.Sdk.Messages;
8  using Microsoft.Xrm.Sdk;
9  using Microsoft.Xrm.Sdk.Client;
10 using Microsoft.Xrm.Sdk.Query;
11 using UtilityFlowDevSolution.Classes;
12
13 namespace UtilityFlowDevSolution {
14     /* Nel costruttore mettere tutti gli step del plugin
15     * costruttore public nomeclasse () : base(typeof(nomeclasse)) {
16     *     RegisteredEvents.Add(new Tuple<int, string, string,
17         Action<LocalPluginContext>>(Plugin.PipelineStage.postOperation, "Create", "lead",
18         ))
19     * }
20     */
21     public class Lead_QualifyLead : Plugin {
22         public Lead_QualifyLead() : base(typeof(Lead_QualifyLead)) {
23             RegisteredEvents.Add(new Tuple<int, string, string,
24                 Action<LocalPluginContext>>(Plugin.PipelineStage.PostOperation, "Create",
25                 "lead", QualifyLead));
26         }
27
28         private void QualifyLead(LocalPluginContext context) {
29             //Extract the tracing service for use in debugging sandboxed plug-ins.
30             ITracingService tracingService = context.TracingService;

```

```

28
29 // Obtain the target entity from the input parameters.
30 Entity entity = context.GetDefaultPostImage();
31
32 // Verify that the target entity represents a lead.
33 // If not, this plug-in was not registered correctly.
34 if (entity.LogicalName != "lead")
35     return;
36
37 try {
38     IOrganizationService service = context.OrganizationService;
39     Guid accountId = new Guid();
40     Guid contactId = new Guid();
41     Guid opportunityId = new Guid();
42
43
44     // Get companyname to check if account exists
45     String companyName = entity.GetAttributeValue<string>("companyname");
46     String partitaIva = entity.GetAttributeValue<string>("rfu_partitaiva");
47     Boolean accountExists = false;
48
49
50     // Check if the account already exist
51     QueryExpression accountQuery = new QueryExpression("account");
52     accountQuery.NoLock = true;
53     accountQuery.ColumnSet = new ColumnSet("accountid");
54     accountQuery.Criteria.AddCondition(new ConditionExpression("name",
55         ConditionOperator.Like, companyName));
56     if (partitaIva != null) accountQuery.Criteria.AddCondition(new
57         ConditionExpression("accountnumber", ConditionOperator.Like, partitaIva));
58     accountQuery.PageInfo.ReturnTotalRecordCount = true;
59     EntityCollection resultsAccount = service.RetrieveMultiple(accountQuery);
60
61     // Get email to check if contact exists
62     String email = entity.GetAttributeValue<string>("emailaddress1");
63     String mobile = entity.GetAttributeValue<string>("mobilephone");
64     Boolean contactExists = false;
65
66     // Check if the contact already exist
67     QueryExpression contactQuery = new QueryExpression("contact");
68     contactQuery.NoLock = true;
69     contactQuery.ColumnSet = new ColumnSet("contactid");
70     contactQuery.Criteria.AddCondition(new ConditionExpression("emailaddress1",
71         ConditionOperator.Like, email));
72     if (mobile != null) contactQuery.Criteria.AddCondition(new
73         ConditionExpression("mobilephone", ConditionOperator.Like, mobile));
74     contactQuery.PageInfo.ReturnTotalRecordCount = true;
75     EntityCollection resultsContact = service.RetrieveMultiple(contactQuery);
76
77     if (resultsAccount.TotalRecordCount > 0) {
78         accountExists = true;
79         foreach (Entity item in resultsAccount.Entities) {
80             accountId = item.Id; // get account id
81             break;
82         }
83     }
84     if (resultsContact.TotalRecordCount > 0) {
85         contactExists = true;
86         foreach (Entity item in resultsContact.Entities) {
87             contactId = item.Id; // get contact id
88             break;
89         }
90     }
91 }

```

```

85     }
86 }
87
88 // Retrieve the organization's base currency ID for setting the
89 // transaction currency of the opportunity.
90 var query = new QueryExpression("organization");
91 query.ColumnSet = new ColumnSet("basecurrencyid");
92 var result = service.RetrieveMultiple(query);
93 var currencyId = (EntityReference)result.Entities[0]["basecurrencyid"];
94
95 QualifyLeadRequest qualify = new QualifyLeadRequest {
96     CreateOpportunity = true,
97     OpportunityCurrencyId = currencyId,
98     LeadId = new EntityReference(entity.LogicalName, entity.Id),
99     Status = new OptionSetValue(OptionSet.Lead.StatusCode.Qualified)
100 };
101
102 if (contactExists && !accountExists) {
103     // Create account and use existing contact as OpportunityCustomerId
104     qualify.CreateAccount = true;
105     qualify.OpportunityCustomerId = new EntityReference("contact", contactId);
106 } else if (!contactExists && accountExists) {
107     // Create contact and use existing account as OpportunityCustomerId
108     qualify.CreateContact = true;
109     qualify.OpportunityCustomerId = new EntityReference("account", accountId);
110 } else if (contactExists && accountExists) {
111     // Use existing account as OpportunityCustomerId
112     qualify.OpportunityCustomerId = new EntityReference("account", accountId);
113     //if contact is not associated to account
114 } else {
115     qualify.CreateAccount = true;
116     qualify.CreateContact = true;
117 }
118
119 //Disable DuplicateDetection
120 qualify.Parameters.Add("SuppressDuplicateDetection", true);
121
122 QualifyLeadResponse qualifyRes = (QualifyLeadResponse)service.Execute(qualify);
123
124 foreach (var resEntity in qualifyRes.CreatedEntities) {
125     //NotifyEntityCreated(entity.LogicalName, entity.Id);
126     if (resEntity.LogicalName == "account") {
127         accountId = resEntity.Id;
128     }
129     if (resEntity.LogicalName == "contact") {
130         contactId = resEntity.Id;
131     }
132     if (resEntity.LogicalName == "opportunity") {
133         opportunityId = resEntity.Id;
134         tracingService.Trace("OpportunityID: " + opportunityId);
135     }
136 }
137
138 // Update account if newly created to add account number
139 if (!accountExists) {
140     Entity retrievedAccount = new Entity("account", accountId);
141     Entity account = new Entity("account");
142     account.Id = retrievedAccount.Id;
143     account["accountnumber"] = partitaIva;
144     service.Update(account);
145     tracingService.Trace("Updated account " + accountId);

```

```

146     }
147
148     String[] fieldsToUpdate = new string[] {
149         "rfu_consumielettgenfeb",
150         "rfu_consumielettmaggiu",
151         "rfu_consumielettmarapr",
152         "rfu_consumielettnovdic",
153         "rfu_consumielettlugago",
154         "rfu_consumielettsetott",
155         "rfu_consumigasgenfeb",
156         "rfu_consumigaslugago",
157         "rfu_consumigasmaggiu",
158         "rfu_consumigasmarapr",
159         "rfu_consumigasnovdic",
160         "rfu_consumigassetott",
161         "rfu_electricityintendeduse",
162         "rfu_energyrating",
163         "rfu_floorarea",
164         "rfu_gasintendeduse",
165         "rfu_impiantoproprio",
166         "rfu_requestedutility",
167         "rfu_tensioneallacciamento",
168         "rfu_tipocontratto"
169     };
170
171     // Add custom field's data to created opportunity
172     Entity retrievedOpportunity = new Entity("opportunity", opportunityId);
173     tracingService.Trace(retrievedOpportunity.ToString());
174     Entity opportunity = new Entity("opportunity");
175     opportunity.Id = retrievedOpportunity.Id;
176
177     if (!qualify.CreateContact) {
178         opportunity["parentcontactid"] = new EntityReference("contact", contactId);
179     }
180
181     foreach (String field in fieldsToUpdate) {
182         opportunity[field] = entity.Contains(field) ? entity[field] : null;
183     }
184
185     tracingService.Trace("before update opportunity");
186     service.Update(opportunity);
187     tracingService.Trace("After update opportunity");
188
189 } catch (Exception ex) {
190     tracingService.Trace("Lead_QualifyLead Plugin: {0}", ex.ToString());
191     throw new InvalidPluginExecutionException("An error occurred in the
        Lead_QualifyLead plug-in.", ex);
192 }
193 }
194 }
195 }

```

Listing A.2: Codice sorgente del Plugin di conversione del Lead in Opportunity.

A.3 Codice sorgente Plugin per l'invio di email con preventivo

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;

```

```

4 using System.Text;
5 using System.ServiceModel;
6 using System.Threading.Tasks;
7 using Microsoft.Crm.Sdk.Messages;
8 using Microsoft.Xrm.Sdk;
9 using Microsoft.Xrm.Sdk.Client;
10 using Microsoft.Xrm.Sdk.Query;
11 using UtilityFlowDevSolution.Classes;
12
13 namespace UtilityFlowDevSolution {
14     public class Quote_Update_GenerateAndSendPdf : Plugin {
15         public Quote_Update_GenerateAndSendPdf() :
16             base(typeof(Quote_Update_GenerateAndSendPdf)) {
17             RegisteredEvents.Add(new Tuple<int, string, string,
18                 Action<LocalPluginContext>>(Plugin.PipelineStage.PostOperation, "Update",
19                 "quote", GenerateAndSendPdf));
20         }
21
22         private void GenerateAndSendPdf(LocalPluginContext context) {
23             //Extract the tracing service for use in debugging sandboxed plug-ins.
24             ITracingService tracingService = context.TracingService;
25
26             // Obtain the target entity from the input parameters.
27             Entity entity = context.GetDefaultPostImage();
28
29             // Verify that the target entity represents a quote.
30             // If not, this plug-in was not registered correctly.
31             if (entity.LogicalName != "quote")
32                 return;
33             // Verify that target statecode attribute exists (it means that it has been
34             // updated)
35             if (!entity.Contains("statecode"))
36                 return;
37             // Verify that the target statecode has been updated to "Active"
38             if (((OptionSetValue)entity["statecode"]).Value != new OptionSetValue(1).Value)
39                 return;
40
41             try {
42                 tracingService.Trace("Plugin triggered");
43                 IOrganizationService service = context.OrganizationService;
44
45                 Guid documentTemplateId = getTemplateId(service);
46
47                 // Get quote that triggered the plugin
48                 Entity quote = service.Retrieve("quote", entity.Id, new ColumnSet(true));
49
50                 String b64File = generateEncodedPdf(service, quote, documentTemplateId);
51                 String attachmentName = "OffertaPersonalizzata.pdf";
52
53                 //tracingService.Trace(b64File);
54
55                 // create email
56                 Entity email = generateEmail(service, quote);
57                 Guid emailId = service.Create(email);
58
59                 Guid attachmentId = createAttachment(service, b64File, attachmentName, emailId);
60
61                 SendEmailRequest sendEmailReq = new SendEmailRequest {
62                     EmailId = emailId,
63                     IssueSend = true,
64                 };

```

```

61
62     SendEmailResponse sendEmailResp =
        (SendEmailResponse)service.Execute(sendEmailReq);
63 } catch (Exception ex) {
64     tracingService.Trace("Lead_QualifyLead Plugin: {0}", ex.ToString());
65     throw new InvalidPluginExecutionException("An error occurred in the
        Quote_Update_GenerateAndSendPdf plug-in.", ex);
66 }
67 }
68
69 private Guid getTemplateId(IOrganizationService service) {
70     Guid documentTemplateId = new Guid();
71     QueryExpression templateQuery = new QueryExpression("documenttemplate");
72     FilterExpression filter = new FilterExpression(LogicalOperator.Or);
73     filter.AddCondition("name", ConditionOperator.Like, "Riepilogo Offerta");
74     filter.AddCondition("name", ConditionOperator.Like, "Quote Summary");
75     templateQuery.Criteria.AddFilter(filter);
76     templateQuery.ColumnSet.AddColumns("name", "documenttemplateid", "content");
77
78     EntityCollection resultsTemplate = service.RetrieveMultiple(templateQuery);
79
80     foreach (Entity template in resultsTemplate.Entities) {
81         if (template["name"].ToString().Contains("Riepilogo")) {
82             return template.Id;
83         }
84         if (template["name"].ToString() == "Quote Summary") {
85             documentTemplateId = template.Id;
86         }
87     }
88
89     if (documentTemplateId == Guid.Empty) {
90         throw new Exception("Template not found.");
91     }
92     return documentTemplateId;
93 }
94
95 private String generateEncodedPdf(IOrganizationService service, Entity quote, Guid
    documentTemplateId) {
96     // Request to create pdf
97     OrganizationRequest createPdfRequest = new
        OrganizationRequest("ExportPdfDocument");
98     createPdfRequest["EntityTypeCode"] = 1084; // Quote typecode
99     createPdfRequest["SelectedTemplate"] = new EntityReference("documenttemplate",
        documentTemplateId);
100     createPdfRequest["SelectedRecords"] = "[" + quote.Id.ToString() + "]";
101
102     OrganizationResponse createPdfResponse =
        (OrganizationResponse)service.Execute(createPdfRequest);
103
104     return Convert.ToBase64String((byte[])createPdfResponse["PdfFile"]);
105 }
106
107 private Entity generateEmail(IOrganizationService service, Entity quote) {
108     Entity email = generateEmailFromTemplate(service, quote);
109     return completeEmail(quote, email);
110 }
111
112 private Entity generateEmailFromTemplate(IOrganizationService service, Entity quote)
    {
113     //Create a query expression to get necessary email quote template
114     QueryExpression queryBuildInTemplates = new QueryExpression {

```

```

115         EntityName = "template",
116         ColumnSet = new ColumnSet("templateid", "title"),
117         Criteria = new FilterExpression()
118     };
119     queryBuildInTemplates.Criteria.AddCondition("title", ConditionOperator.Equal,
120         "Template Offerta Email");
121     EntityCollection templateEntityCollection =
122         service.RetrieveMultiple(queryBuildInTemplates);
123
124     Guid emailTemplateId = new Guid();
125
126     if (templateEntityCollection.Entities.Count > 0) {
127         emailTemplateId =
128             (Guid)templateEntityCollection.Entities[0].Attributes["templateid"];
129     } else {
130         throw new ArgumentException("Standard Email Templates are missing");
131     }
132
133     // Create the request
134     InstantiateTemplateRequest emailUsingTemplateReq = new InstantiateTemplateRequest {
135         // Use a built-in Email Template of type "quote".
136         TemplateId = emailTemplateId,
137
138         // The regarding Id is required, and must be of the same type as the Email
139         // Template.
140         ObjectId = quote.Id,
141         ObjectType = "quote"
142     };
143
144     InstantiateTemplateResponse emailUsingTemplateResp =
145         (InstantiateTemplateResponse)service.Execute(emailUsingTemplateReq);
146     return emailUsingTemplateResp.EntityCollection[0];
147 }
148
149 private Entity completeEmail(Entity quote, Entity email) {
150     Guid clientId = ((EntityReference)quote.Attributes["customerid"]).Id;
151     Guid senderId = ((EntityReference)quote.Attributes["ownerid"]).Id; // Quote owner
152         is the sender
153
154     // Create the 'From:' activity party for the email
155     Entity fromParty = new Entity("activityparty");
156     fromParty["partyid"] = new EntityReference("systemuser", senderId);
157
158     // Create the 'To:' activity party for the email
159     Entity toParty = new Entity("activityparty");
160     toParty["partyid"] = new EntityReference("account", clientId);
161
162     email["to"] = new Entity[] { toParty };
163     email["from"] = new Entity[] { fromParty };
164     email["directioncode"] = true;
165     email["regardingobjectid"] = new EntityReference("quote", quote.Id);
166
167     return email;
168 }
169
170 private Guid createAttachment(IOrganizationService service, String b64File, String
171     attachmentName, Guid emailId) {
172     // Create attachment
173     Entity attachment = new Entity("activitimimeattachment");
174     attachment["body"] = b64File;
175     attachment["filename"] = attachmentName;

```

```
169     attachment["objectid"] = new EntityReference("email", emailId);
170     attachment["objecttypecode"] = "email";
171     return service.Create(attachment);
172 }
173 }
174 }
```

Listing A.3: Codice sorgente del Plugin di invio email con preventivo.