

Automazione dei processi fiscali e contabili

Un metodo operativo per scegliere i processi, usare gli strumenti di intelligenza artificiale, trasformare le regole professionali in software e verificare il risultato. Il metodo viene applicato alla riclassificazione di un bilancio infrannuale dal reporting IAS/IFRS agli schemi civilistici italiani.

Riccardo Zedda, dottore commercialista

Contenuti

I	Capire cosa automatizzare	03
II	Scegliere gli strumenti	04
III	Usare l'intelligenza artificiale	05
IV	Verificare il risultato	06
V	Chicche pratiche	07
VI	Il problema da risolvere	09
VII	Input e condizioni	10
VIII	Brainstorm e memoria	11

IX	Progetto di sviluppo	12
X	Prima versione	13
XI	Follow-up e test	14
XII	Logica contabile	15
XIII	Rettifiche IAS/IFRS e OIC	16
XIV	Audit indipendente	17
XV	Rilascio	18
XVI	La professione	19

Partire dal processo, non dallo strumento

Scelgo un'attività che conosco bene e che ripeto. Gli input devono essere abbastanza stabili, le regole spiegabili e il risultato controllabile. Non parto dal processo più complesso. Parto da quello in cui posso confrontare il tool con un working paper già affidabile.

- 01 **Ricorre.** La eseguo ogni mese, trimestre o anno e assorbe tempo ogni volta.
- 02 **I dati sono leggibili.** I file hanno una struttura riconoscibile oppure posso normalizzarli.
- 03 **Le regole si possono scrivere.** So spiegare cosa fare nei casi ordinari e nelle principali eccezioni.
- 04 **Il risultato si può verificare.** Dispongo di quadrature, controlli o output storici con cui confrontarlo.

Se almeno tre risposte sono positive, vale la pena costruire un primo prototipo. Se una regola vive solo nella testa di chi esegue il processo, prima va scritta.

Assegnare un ruolo a ogni ambiente

Non serve conoscere tutti i modelli. Serve sapere quale lavoro affidare a ciascuno. Questa è la combinazione che uso io perché mi trovo bene, ma può essere sostituita. Terrei comunque separate progettazione, sviluppo e controllo.

- 01 **Capire e progettare — ChatGPT.** Metto ordine nel processo, preparo il brief, chiarisco le regole e costruisco i test.
- 02 **Scrivere il codice — Claude.** Sviluppo l'applicativo a partire da specifiche già approvate e lavoro per versioni piccole.
- 03 **Leggere molti documenti — Gemini o Perplexity.** Esamino corpus ampi e individuo i passaggi da verificare sulle fonti ufficiali.
- 04 **Conservare la memoria — Obsidian.** Tengo insieme fonti, mapping, decisioni, test, assunzioni e versioni del progetto.
- 05 **Verificare la sostanza — banche dati professionali.** Controllo norme, prassi e principi contabili. Il modello non sostituisce la fonte.

Aprire la chat e insegnare il processo

Non inizierei chiedendo di scrivere il codice. Aprirei una chat nuova e trasferirei il processo un pezzo alla volta. Il modello deve capire cosa voglio ottenere, come sono fatti i dati e dove deve fermarsi.

- 01 **Scrivi l'obiettivo.** Una frase concreta: voglio ottenere questo file partendo da questi dati.
- 02 **Carica input reali.** Usa almeno un file ordinario e uno con un'anomalia già conosciuta.
- 03 **Insegna come leggerli.** Spiega colonne, segni, codici, totali, righe da ignorare e dati mancanti.
- 04 **Spiega le regole.** Scrivi se accade X fai Y. Se manca Z fermati e chiedi. Non lasciare decisioni implicite.
- 05 **Definisci output e controlli.** Indica formato, campi, quadrature, messaggi di errore e livello di dettaglio.
- 06 **Fatti restituire il brief.** Prima del codice chiedi una sintesi completa e correggi ciò che il modello ha capito male.

UN'ISTRUZIONE CHE SI PUÒ COPIARE

«Voglio costruire un applicativo che produca questo risultato. Questi sono gli input e vanno letti in questo modo. Queste sono le regole e le eccezioni. Se manca un dato, fermati. L'output deve superare questi controlli. Non scrivere ancora codice. Restituisci prima il brief e indicami cosa non è chiaro.»

Provare il risultato prima di usarlo

Un applicativo non è corretto perché si apre o perché produce un numero. Prima dello sviluppo preparo i casi di prova e il risultato atteso. Dopo lo sviluppo provo a farlo sbagliare.

- 01 **Caso ordinario.** Un file pulito con risultato già verificato a mano.
- 02 **Caso limite.** Un segno invertito, un conto nuovo, un saldo zero o una fattispecie rara.
- 03 **Input errato.** Colonne mancanti, formato sbagliato o dati incoerenti. Il tool deve fermarsi.
- 04 **Risultato atteso.** Numeri, classificazioni e quadrature preparati prima di vedere il prototipo.

Eseguo tutti i casi e registro PASS o FAIL. Controllo le fonti normative, apro le voci e arrivo ai conti sottostanti. Una quadratura aritmetica non dimostra che la classificazione sia corretta.

Infine apro una chat nuova. Le fornisco codice, brief e matrice di test, ma non la storia dello sviluppo. Le chiedo di cercare errori, assunzioni non supportate e casi non gestiti. La stessa chat che ha costruito il tool non dovrebbe certificarlo.

ISTRUZIONE DI AUDIT

«Per ogni controllo restituisci PASS o FAIL, motivazione, evidenza e correzione. Non cercare di confermare il risultato. Cerca il modo in cui potrebbe essere sbagliato.»

Piccoli accorgimenti che evitano molti errori

01

Il canarino del contesto

Chiedi al modello di iniziare ogni risposta con il tuo nome. Se smette, il contesto potrebbe essersi deteriorato. È un allarme, non una prova.

02

Il test dello scoiattolo

Chiedi cosa dicono i file sugli scoiattoli. La risposta corretta è che non ne parlano. Se inventa, non sta usando bene le fonti.

03

Il passaggio di consegne

Prima di cambiare chat, fatti preparare una pagina con obiettivo, decisioni prese, errori già trovati, file utili e prossimo passo.

04

Sintesi, non copia

Nel passaggio di consegne non copiare interi documenti. Scrivi cosa è stato deciso, perché e dove si trova la fonte.

05

Una produce, una controlla

Una chat costruisce. Una seconda chat, senza la storia del lavoro, cerca errori e contraddizioni.

06

I controlli vengono prima

Definisci i criteri PASS o FAIL prima del codice. Dopo rischieresti di adattare il controllo al risultato.

07

Registro delle assunzioni

Tieni visibili le ipotesi usate dal tool. Ogni assunzione non verificata deve restare aperta.

08

Prova a smentirlo

Non chiedere solo se funziona. Chiedi quale input lo farebbe fallire e quale caso contraddice la regola.

P A R T E S E C O N D A

Un caso completo: riclassifica del bilancio secondo la IV Direttiva

Dalla lettura dei principi al rilascio validato. Un applicativo che riceve in ingresso i report gestionali di un gruppo redatti in logica IAS/IFRS e restituisce i prospetti civilistici di cui agli articoli 2424, 2425 e 2435-bis del codice civile, gestendo le rettifiche book-to-book in partita doppia.

Il percorso seguito è sempre lo stesso: prima chiarisco il processo in chat, poi affido a Obsidian ciò che deve restare, poi apro un progetto Claude con quelle stesse fonti, e da lì procedo per versioni e follow-up fino al rilascio.

Un bilanciino OIC serve anche durante l'anno

Nel gruppo molte società tengono la contabilità quotidiana e il reporting secondo logiche IAS/IFRS. Il bilancio annuale viene poi predisposto secondo il codice civile e gli OIC.

Durante l'anno può però servire un bilanciino civilistico. Un caso concreto è l'istruttoria per un rimborso IVA, quando l'autorità chiede una situazione infrannuale leggibile secondo criteri italiani.

Ricostruire ogni volta il prospetto a mano significa ripetere mappature, controlli sui segni, verifiche delle rettifiche e quadrature. Il problema non è tecnico. È un processo professionale già noto che viene rieseguito molte volte.

SITUAZIONE ATTUALE

Trial balance e report IAS/IFRS disponibili. Raccordo OIC già usato in sede annuale. Il lavoro infrannuale è ancora manuale.

OBIETTIVO

Caricare i file standard e ottenere un bilanciino OIC navigabile, riconciliato, documentato e pronto per la review.

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	-------	----------

Gli input esistono già e sono abbastanza stabili

Prima di aprire una chat verifico cosa possiedo davvero. In questo caso non parto da zero. Il lavoro annuale ha già prodotto gran parte della conoscenza necessaria.

TRIAL BALANCE File Excel standard con codice conto, descrizione, saldo e periodo.	IL PROCESSO PASSA IL TEST <i>Ricorre, i dati sono leggibili, le regole possono essere scritte e il risultato si confronta con un bilancio già predisposto.</i>
BALANCE SHEET E P&L Report gestionali omogenei fra le società del gruppo.	
MAPPING ANNUALE Associazione fra conto IFRS e voce o classe civilistica OIC già utilizzata in sede di bilancio d'esercizio.	OUTPUT ATTESO Stato Patrimoniale e Conto Economico OIC, drill-down fino ai conti, elenco dei conti non mappati, rettifiche book-to-book, quadrature ed esportazione del working paper.

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	-------	----------

Prima chiarisco il processo, poi lo affido alla memoria del progetto

01 · Apro una chat nuova (ChatGPT o Gemini, per corpus più ampi) e descrivo il problema in termini professionali. Non chiedo ancora codice: voglio capire quali regole mancano e trasformare il lavoro annuale in un brief utilizzabile.

- a Spiego perché serve il bilanciino e quale risultato voglio ottenere.
- b Carico un file ordinario, un file con anomalie e il mapping già usato.
- c Insegno come leggere colonne, segni, totali, codici e righe da scartare.
- d Chiedo domande, casi limite e una prima struttura di input, regole, output e controlli.

ISTRUZIONE UTILE

Non scrivere codice. Prima ricostruisci il processo, indicami le informazioni mancanti e restituisci un brief che io possa correggere.

02 · Trasferisco su Obsidian ciò che deve restare del brief: non l'intera chat, solo le decisioni. Ogni blocco diventa un file Markdown separato, così nel passo successivo posso allegare a Claude solo le fonti pertinenti.

● ChatGPT

+ Nuova chat

PROGETTI

IV Direttiva — brief

Bridge IAS/IFRS → OIC

CHAT

Perimetro schema abbreviato

Riconciliazione utile netto

Riclassifica IV Direttiva — brief di progetto

R PROMPT

Devo predisporre uno strumento che, ricevendo in ingresso i report gestionali del gruppo redatti secondo principi IAS/IFRS, restituisca lo Stato Patrimoniale e il Conto Economico riclassificati secondo lo schema civilistico italiano (artt. 2424, 2425, 2435-bis c.c.). Non redigere codice: struttura input, regole di transito e output.

AI CHATGPT

Il brief può essere articolato come segue.

Input

Trial balance IAS/IFRS, mapping annuale già in uso, note sulle differenze di rappresentazione.

SALVO IN OBSIDIAN — FILE MARKDOWN SEPARATI

brief.md

mapping-conti.md

differenze-IAS-OIC.md

casi-test.md

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	-------	----------

In Claude trasferisco lo stesso materiale, già ordinato

03 · Apro un progetto dedicato e carico, come fonti, gli stessi file Markdown organizzati nel passo precedente in Obsidian: brief, mapping dei conti, differenze IAS/IFRS e OIC, casi di test. Non ricopio la chat di ChatGPT: passo le decisioni già filtrate.

Nella pratica, avere fonti già ordinate aiuta il modello a fare più domande prima del codice, invece di scrivere subito sulla base di ipotesi implicite.

ISTRUZIONI PERMANENTI

Leggi sempre le fonti prima di proporre modifiche. Non inventare regole mancanti. Mantieni un changelog. Non alterare funzioni già testate senza segnalarlo.

COME MI PRESENTO

Non sono uno sviluppatore. Guidami nelle scelte, fammi domande quando una regola non è chiara e proponi una modifica alla volta. Prima di scrivere spiegami cosa cambierai e come potrò verificarlo.

Claude

PROGETTO

Riclassifica IV Direttiva

FONTI

brief.md

mapping-conti.md

differenze-IAS-OIC.md

casi-test.md

changelog.md

CHAT RECENTI

Correzione segno SP2

Rettifica leasing IFRS 16

Riclassifica IV Direttiva — sviluppo

PROMPT

Ho sempre voluto sviluppare uno strumento di questo tipo, ma non sono uno sviluppatore. Conosco il processo fiscale e ti fornirò tutta la documentazione. Aiutami a progettare e sviluppare il tool spiegandomi anche le scelte tecniche.

AI

CLAUDE

Ho letto **brief.md** e **mapping-conti.md**. Prima di scrivere codice, tre domande sui casi limite: come tratto un conto reversibile a saldo zero? La mappa annuale copre già le società in sviluppo? Dove si ferma il tool se un conto non è mappato?

Prossimo passo proposto

V1 — solo caricamento Excel e lettura colonne. Nessuna logica di mapping ancora.

LE FONTI CARICATE SONO LE STESSA DI OBSIDIAN

brief.md

mapping-conti.md

differenze-IAS-OIC.md

casi-test.md

changelog.md

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	-------	----------

Riccardo Zedda · Automazione dei processi fiscali

12

La prima versione deve fare poco ma funzionare

Chiedo un tool che giri in locale e lavoro per blocchi. Ogni versione aggiunge una funzione verificabile senza riscrivere ciò che è già stabile.

- V1 Caricamento del file Excel e lettura corretta delle colonne.
- V2 Applicazione del mapping ed elenco dei conti non mappati.
- V3 Stato Patrimoniale, Conto Economico e drill-down conto per conto.
- V4 Rettifiche, verifiche, esportazione Excel e PDF.

LA REGOLA

Non chiedo subito il prodotto finale. Una prima versione semplice rende più facile capire se il modello ha letto bene i dati e se la struttura del progetto è solida.

Riclassifica IV Direttiva · V3		Esporta Excel	Esporta PDF
Stato Patrimoniale	Conto Economico	Conti non mappati · 0	Verifiche
ATTIVO			
B.I	Immobilizzazioni immateriali		961.173
B.II	Immobilizzazioni materiali		20.158
B.III	Immobilizzazioni finanziarie		138.895.237
C.II	Crediti		49.205.585
C.IV	Disponibilità liquide		1.288.598
Totale attivo			191.391.833

Prima interfaccia navigabile: mapping, conti e quadrature visibili fin da V3.

Ogni follow-up parte da un errore osservabile

Controllo pulsanti, schede, navigazione e messaggi prima di entrare nei numeri. Il tool deve reggere anche prima di essere corretto. Ogni correzione approvata entra nel **changelog.md** del progetto, così la versione successiva parte da uno stato noto.

- 01

Caricamento. Un file standard e uno con colonne o righe anomale.
- 02

Navigazione. Pulsanti, schede, drill-down, ritorno alla vista precedente.
- 03

Lettura input. Codici, descrizioni, segni, periodo, valuta e righe da ignorare.
- 04

Elaborazione. Mapping, conti non mappati e rettifiche.
- 05

Output. Quadrature, Excel, PDF e riconciliazione del risultato.
- 06

Errori. Il tool si ferma quando manca un dato e non prova a indovinare.

IL FOLLOW-UP CHE FUNZIONA

Con questo file ottengo questo risultato. Mi aspetto invece questo comportamento. Individua la causa, correggi senza toccare le funzioni già testate, restituisci il file completo e indicami cosa provare.

Il mapping non basta da solo

Il mapping annuale indica la destinazione OIC del conto. Il tool deve però conoscere anche la natura del saldo e le fattispecie che possono cambiare rappresentazione da un periodo all'altro.

- 01
- Voce OIC.** La mappa collega il codice conto alla famiglia o alla voce civilistica.
- 02
- Segno.** Per conti reversibili il saldo consuntivo determina attivo, passivo, costo o ricavo.
- 03
- Eccezione.** Un conto nuovo o ambiguo resta non mappato. Il sistema non decide da solo.

CONTO CORRENTE INFRAGRUPO

La mappa identifica la famiglia civilistica. Il segno effettivo decide se il saldo è un credito o un debito verso il gruppo.

IVA E POSTE TRIBUTARIE

IVA a credito o a debito, imposte correnti, ritenute e rapporti di consolidato possono invertirsi fra un periodo e l'altro. La classificazione segue il saldo e la regola documentata.

PRINCIPIO DI ARRESTO

Quando il dato non consente una decisione univoca, il tool espone il problema e chiede una scelta. Non trasforma un dubbio professionale in una regola nascosta.

Le differenze IAS/IFRS entrano come scritture book-to-book

Alcune differenze non si risolvono spostando un saldo. Richiedono una scrittura Dare e Avere applicata al solo prospetto civilistico, senza modificare la contabilità gestionale originaria.

Leasing — IFRS 16 e OIC 12. Sorno del diritto d'uso e della passività, ricostruzione dei canoni.

Costi di sviluppo — IAS 38 e OIC 24. Verifica dei requisiti di capitalizzazione e ammortamento.

Crediti e derivati — IFRS 9 e OIC 15/20/32. Raccordo fra ECL, costo ammortizzato e classificazione.

TFR — IAS 19 e OIC 31. Eliminazione della valutazione attuariale, ricostruzione del debito civilistico.

Avviamento — IFRS 3, IAS 36 e OIC 24. Raccordo fra impairment e ammortamento sistematico OIC.

Lavori interni — IAS/IFRS e OIC 12. Costi per natura in B.6/B.7 con contropartita A.4.

Rettifica IAS/IFRS → OIC

Capitalizzazione dei lavori interni — transito per natura e compensazione via A.4

Voce IV Direttiva	Dare	Avere
B.6 — materie prime	30.000,00	0,00
B.7 — servizi	70.000,00	0,00
A.4 — incrementi immob.	0,00	100.000,00

✓ Dare = Avere, la rettifica quadra

La rettifica può essere salvata solo quando Dare e Avere coincidono.

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	-------	----------

Quando il tool mi soddisfa, provo a smentirlo

Congelo la versione e apro un ambiente nuovo. Uso un modello potente e, quando possibile, diverso da quello impiegato nello sviluppo. Non gli racconto come siamo arrivati al risultato. Gli consegno codice, brief, mapping, fonti, file di prova e matrice dei test.

L'audit del modello non chiude il lavoro. Riproduco i difetti segnalati e testo di nuovo i casi critici. Un PASS non sostituisce la mia review.

ISTRUZIONE DI AUDIT

Non cercare conferme. Cerca il modo in cui il prodotto può sbagliare. Per ogni controllo restituisci PASS o FAIL, evidenza, impatto e correzione proposta.

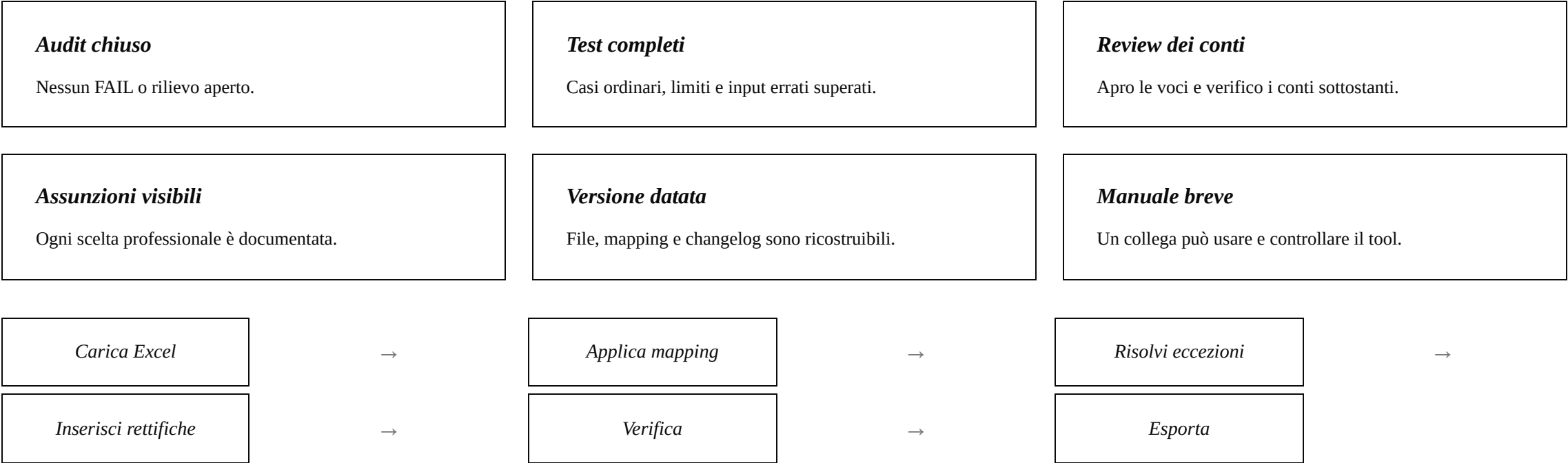
AREE DELLA RUBRIC

<i>Normativa</i>	<i>Quadratura</i>	<i>Mapping</i>
<i>Casi limite</i>	<i>Tracciabilità</i>	<i>Regressioni</i>

PROBLEMA	INPUT	BRAINSTORM	PROGETTO	SVILUPPO	TEST	LOGICA	RETTIFICHE	AUDIT	RILASCIO
----------	-------	------------	----------	----------	------	--------	------------	--------------	----------

Il tool entra in uso solo dopo l'ultima review

A questo punto il flusso è semplice e ripetibile. Carico i file, applico il mapping, risolvo i conti non mappati, inserisco le rettifiche necessarie, verifico le quadrature e genero il working paper.



Dove si sposta il valore

Il metodo descritto qui non è rivoluzionario. È utile perché può essere replicato. Una volta imparato a preparare contesto, regole, test e rilascio, lo stesso approccio può essere applicato a una LIPE, a un modello 770, a una riconciliazione IVA, a un controllo di coerenza o a un altro adempimento ricorrente.

Questo cambia il valore economico di alcune prestazioni. Quando un'attività può essere documentata, testata e automatizzata in tempi brevi, la parte puramente compilativa perde valore. Il cliente percepisce il risultato e non le ore necessarie a produrlo.

Il valore si sposta sulla progettazione del processo, sulla lettura delle eccezioni e sull'assunzione di responsabilità. Sono competenze che la professione possiede già, ma che spesso rimangono nascoste dietro molte ore di lavoro manuale.

Per me l'opportunità è usare meno tempo per ricostruire dati e più tempo per giudicarli. La responsabilità professionale non viene trasferita al modello. Rimane in capo a chi usa lo strumento, lo valida e decide di affidarsi al risultato.

Il primo progetto deve essere utile

Il primo applicativo che ho sviluppato con questo metodo non era destinato a un cliente. Serviva a gestire un'attività che mi occupava circa due giornate ogni trimestre.

La prima versione richiese pochi giorni. La validazione richiese più tempo. Dovetti costruire casi di prova, correggere il modo in cui avevo descritto alcune regole e ripartire più di una volta. È stata la parte più utile, perché mi ha obbligato a rendere esplicito un processo che fino a quel momento eseguivo quasi in automatico.

Da allora quelle due giornate vengono usate diversamente. Non sono diminuite le responsabilità. È diminuita la parte ripetitiva che le nascondeva.

Il consiglio con cui chiudo è molto semplice. Partire da un processo conosciuto bene, con dati reali e un risultato facile da verificare. Il primo obiettivo non deve essere costruire qualcosa di impressionante. Deve essere costruire qualcosa che si possa usare, controllare e spiegare a un collega.

Riccardo Zedda

Dottore commercialista