

2026

Timesheet per aziende di consulenza IT e software: come tracciare le ore e fatturare i progetti senza fogli Excel

Cosa trovi in questa guida

1	Il problema dei timesheet caotici	3
2	Struttura ore, progetti e clienti	5
3	Fatturazione e marginalità reale	7
4	Dal timesheet alla busta paga	9

Il problema dei timesheet caotici

La maggior parte delle software house sa quante ore ha venduto al cliente, ma non quante ne ha davvero consumate. Il risultato è che i progetti in perdita sembrano in pareggio, e quelli sani sembrano normali. Finché non hai un sistema preciso di tracciamento ore, stai gestendo la tua azienda alla cieca.

IN SINTESI

- Senza timesheet strutturati, le ore fatturabili si perdono tra chat, email e fogli Excel diversi per ogni progetto.
- Collegare le ore ai progetti ti permette di calcolare la marginalità reale, non quella stimata a inizio lavori.
- I dati del timesheet alimentano anche le buste paga: straordinari e reperibilità devono essere tracciati con precisione per essere pagati correttamente.
- Passare da fogli manuali a un flusso automatizzato non richiede mesi: con il metodo giusto, si fa in quattro fasi concrete.

Dove spariscono le ore fatturabili

Un developer lavora su tre progetti nella stessa settimana. Alla fine della giornata non ricorda esattamente quante ore ha dedicato a ciascuno, oppure segna tutto in un unico blocco senza distinguere il cliente. Questo accade ogni giorno in centinaia di società di consulenza IT, e il costo non è visibile subito: emerge solo quando vai a fatturare e ti accorgi che mancano ore, oppure quando chiudi i conti di un progetto e scopri che hai lavorato il 30% in più di quanto preventivato.

Il problema non è la pigrizia dei tuoi developer. È che non esiste un sistema che renda la registrazione delle ore semplice e immediata quanto aprire WhatsApp. Se registrare le ore è lento o macchinoso, le persone lo rimandano, lo fanno in blocco a fine settimana e inevitabilmente dimenticano i dettagli.

La conseguenza diretta è che fatturi meno di quello che hai prodotto, oppure fatturi in ritardo perché devi ricostruire le ore a posteriori prima di emettere la nota.

I sintomi di un sistema rotto

Ecco i segnali che indicano che il tuo sistema di timesheet attuale ha un problema strutturale:

- I developer segnano le ore in blocco a fine settimana o a fine mese, non ogni giorno.
- Hai più di un file Excel o Google Sheet attivo contemporaneamente per tracciare le ore, uno diverso per ogni progetto o cliente.
- Non riesci a rispondere in tempo reale alla domanda 'quante ore abbiamo bruciato su questo progetto fino a oggi?'.

- Le fatture al cliente escono con ritardo perché prima devi raccogliere i dati dalle fonti sparse e riconciliarli a mano.
- Non sai quale progetto sta consumando più ore rispetto al budget, finché non è troppo tardi per intervenire.
- Non hai un processo chiaro per portare le ore di straordinario dal timesheet alla busta paga.

Quanto costa davvero il disordine

Un esempio concreto su una software house con 10 developer, tariffa media 500 €/giorno, 220 giorni lavorativi l'anno.

Scenario	Ore perse per developer/anno	Fatturato non recuperato
Timesheet manuali, compilazione settimanale	~20 ore	~12.500 € per developer
Nessun timesheet, ore stimate a fine progetto	~40 ore	~25.000 € per developer
Timesheet digitale compilato quotidianamente	~2 ore	~1.250 € per developer

Perché l'Excel non basta

Un foglio Excel funziona quando hai due o tre progetti attivi e un team di quattro persone. Appena la complessità sale, il sistema si inceppa: i file si moltiplicano, le versioni divergono, nessuno sa qual è quello aggiornato. Soprattutto, Excel non ti dà alert quando un progetto sta sforando il budget di ore, non calcola automaticamente le ore straordinarie e non è collegato alla fatturazione.

Passare a un sistema strutturato non significa solo 'usare un software diverso'. Significa ridefinire il flusso: come si apre un progetto, come si assegnano le persone, come si registrano le ore, come i dati arrivano al commercialista o all'ufficio paghe. Questo flusso, una volta impostato, gira quasi da solo.

Struttura ore, progetti e clienti

Prima di scegliere uno strumento, devi avere chiaro come vuoi strutturare i dati. Progetti, sottoattività, clienti e tipologie di ore devono essere configurati in modo coerente, altrimenti il sistema ti darà numeri che non sai come leggere. Questo capitolo ti aiuta a costruire una tassonomia funzionale per una società di consulenza IT.

La gerarchia base: cliente, progetto, attività

Il modello più efficace per una software house è a tre livelli: cliente (chi paga), progetto (il contratto o l'incarico specifico), attività (il tipo di lavoro svolto in quel progetto). Questo schema ti permette di aggregare i dati in modi diversi a seconda dell'esigenza: tutte le ore su un cliente, tutte le ore su un progetto specifico, oppure quanto tempo si spende in analysis rispetto a development su tutti i progetti.

Ogni voce del timesheet deve sempre avere questi tre campi compilati: senza il cliente, non puoi fatturare; senza il progetto, non puoi calcolare la marginalità; senza l'attività, non puoi capire dove va davvero il tempo del tuo team.

Alcune aziende aggiungono un quarto livello (sprint o milestone), utile se lavori in metodologie agili. È una scelta opzionale, ma aiuta molto a collegare il timesheet alla pianificazione del progetto.

Tipologie di ore da distinguere sempre

Non tutte le ore sono uguali. Distinguere le tipologie nel timesheet ti serve sia per fatturare correttamente sia per gestire le buste paga:

- Ore ordinarie fatturabili: il lavoro svolto per il cliente all'interno del contratto, quelle che genereranno ricavo.
- Ore ordinarie non fatturabili: riunioni interne, formazione, pre-sales, attività amministrative. Vanno tracciate comunque per capire il costo reale della struttura.
- Ore straordinarie: quelle oltre l'orario contrattuale, che devono essere autorizzate dall'azienda e gestite in busta paga. Non è il developer che decide da solo di fare straordinario.
- Reperibilità on-call: il developer è disponibile fuori orario senza lavorare attivamente. Si traccia come voce separata perché ha una retribuzione specifica nel CCNL.
- Ore su progetto interno: sviluppo di prodotti propri, ricerca e sviluppo, attività non collegate a un cliente.

Esempio di struttura per una software house

Una configurazione tipica per un'azienda di consulenza IT con 15 persone e tre clienti attivi.

Livello	Esempio concreto	A cosa serve
Cliente	Banca Rossi S.p.A.	Aggregare ricavi e ore per emettere la fattura

Progetto	Migrazione core banking Q12026	Calcolare la marginalità per singolo incarico
Attività	Analysis / Development / Testing / PM	Capire dove si concentra lo sforzo del team
Tipo ora	Fatturabile / Non fatturabile / Straordinario / On-call	Fatturazione corretta e gestione buste paga

Le regole di compilazione per il team

Un sistema funziona solo se tutti lo usano allo stesso modo. Queste sono le regole minime da condividere con il team:

- Le ore si registrano ogni giorno, non a fine settimana: dopo 48 ore i dettagli si perdono e le stime diventano imprecise.
- Ogni voce deve avere cliente, progetto e attività: una riga 'varie' o 'altro' non è accettata nel sistema.
- Le ore straordinarie vanno segnalate prima di essere fatte, non a posteriori: il sistema deve ricevere l'autorizzazione del PM prima che l'ora venga registrata come straordinario.
- Le ore non fatturabili vanno registrate lo stesso: servono per calcolare il costo reale delle persone e non scompaiono solo perché non generano ricavo.
- Il PM ha visibilità in tempo reale su tutte le ore del progetto: non serve mandare report via email, il dato è già lì.

Budget ore e alert di sforamento

Ogni progetto dovrebbe avere un budget di ore definito all'apertura, non solo un budget economico. Se hai stimato 200 ore per una fase di sviluppo e a metà progetto ne hai già consumate 150, vuoi saperlo adesso, non quando emetti la fattura finale.

Un buon sistema ti avvisa automaticamente quando un progetto ha consumato una certa percentuale del budget (di solito 70-80%) così hai ancora margine per intervenire: rinegoziare con il cliente, ridurre lo scope, oppure decidere consapevolmente di assorbire il costo extra. Senza questo alert, lo scopri solo a chiusura progetto, quando non puoi fare nulla.

Fatturazione e marginalità reale

Tracciare le ore è il primo passo. Il secondo è trasformare quei dati in fatture accurate e in numeri di marginalità su cui puoi prendere decisioni. Molte aziende di consulenza IT sanno il fatturato ma non la marginalità per progetto: sanno quanto hanno incassato, non quanto ci è rimasto dopo aver pagato le persone.

Dal timesheet alla fattura senza ricostruzioni manuali

Il flusso ideale è questo: le ore approvate dal PM nel timesheet vengono estratte automaticamente con un click, raggruppate per cliente e progetto, e trasformate in una distinta che il tuo amministrativo usa per emettere la fattura. Non c'è nessun passaggio manuale di copia-incolla da Excel a un altro documento.

Questo è particolarmente importante se fatturi a giornata o a consuntivo, perché il cliente si aspetta un dettaglio preciso delle ore lavorate a supporto della fattura. Se queste ore arrivano da un sistema strutturato, il report è già pronto e professionale. Se le ricostruisci a mano ogni volta, il rischio di errore è alto e il tempo speso è significativo.

Nei contratti a corpo (fixed price), il timesheet non cambia il valore della fattura, ma ti serve comunque per capire se il progetto ha consumato più o meno ore di quelle stimate, e quindi se hai guadagnato o perso rispetto alle aspettative.

Calcolare la marginalità vera per progetto

La marginalità reale di un progetto si calcola sottraendo il costo effettivo delle ore al ricavo. Per farlo hai bisogno di tre dati collegati:

- Il ricavo del progetto: il valore del contratto o le ore fatturate moltiplicate per la tariffa giornaliera concordata.
- Il costo orario interno di ogni developer: non la RAL divisa per le ore, ma il costo aziendale totale (RAL più contributi INPS di circa il 30%, più TFR di circa il 7,7%) diviso per le ore produttive annue effettive.
- Le ore consumate per tipo: separare le ore fatturabili da quelle non fatturabili ti dice anche qual è il tuo 'tasso di utilizzo' (billable ratio), cioè quanta parte del tempo pagato ai developer genera davvero ricavo.
- I costi fissi allocati al progetto: se vuoi una marginalità completa, devi decidere come spalmare i costi di struttura (PM, vendita, amministrazione) sui singoli progetti.

Esempio di marginalità su un progetto reale

Un progetto a tariffa giornaliera con due developer senior per tre mesi.

Voce	Calcolo	Valore
Ricavo fatturato	60 giorni x 2 developer x 550 €/giorno	66.000 €

Costo developer (costo aziendale)	60 giorni x 2 developer x 320 €/giorno*	38.400 €
Ore non fatturabili (riunioni, PM)	20 ore x 2 developer x 40 €/ora	1.600 €
Marginalità lorda	66.000 - 38.400 - 1.600	26.000 € (39%)

Nota sul costo giornaliero interno

Il costo giornaliero interno di un developer si calcola a partire dal costo aziendale annuo totale. Per un developer con una RAL di 42.000 €, il costo annuo per l'azienda è circa 42.000 + 12.600 (contributi INPS ~30%) + 3.234 (TFR ~7,7%) + INAIL, arrivando intorno a 58.000-60.000 € annui. Diviso per circa 220 giorni lavorativi effettivi, ottieni un costo interno di circa 265-270 €/giorno. Nota che questa è la base: se consideri anche i giorni di ferie, malattia e formazione in cui il developer non è produttivo per il cliente, il costo per giornata fatturabile sale ulteriormente.

Conoscere questo numero ti permette di valutare se la tua tariffa giornaliera garantisce un margine sostenibile, oppure se stai vendendo troppo a basso costo per coprire davvero i costi.

Billable ratio: l'indicatore che molti ignorano

Il billable ratio è la percentuale delle ore totali lavorate che sono effettivamente fatturabili a un cliente. In una software house sana, questo valore si aggira tipicamente tra il 70% e l'85% per i developer. Sotto il 60% significa che troppo tempo va in attività interne non remunerate.

Tracciare il billable ratio per persona e per team nel tempo ti dice subito se c'è un problema di sottoutilizzo (il developer è poco impegnato su progetti cliente), o se al contrario stai bruciando le persone su lavoro interno senza che questo generi ricavo. È un KPI che puoi costruire solo se il timesheet distingue chiaramente ore fatturabili e non fatturabili.

Dal timesheet alla busta paga

Il timesheet non serve solo alla fatturazione: è anche la fonte da cui devono partire i dati per le buste paga. Gli straordinari non autorizzati non esistono, la reperibilità va pagata secondo il CCNL applicato, e ogni ora extra deve avere un'origine tracciabile. Se questo flusso non è automatizzato, il rischio è pagare troppo, pagare troppo poco o pagare in modo non conforme.

Straordinari: le regole di base

Prima di parlare di flussi tecnici, è utile chiarire cosa dice la normativa sugli straordinari, perché molte aziende IT gestiscono questa voce in modo approssimativo:

- Lo straordinario è il lavoro svolto oltre l'orario contrattuale normale, che per la maggior parte dei contratti è 40 ore settimanali.
- Il limite massimo è di 48 ore medie settimanali (calcolate su un periodo di riferimento) e, salvo che il CCNL preveda altro, non si possono superare 250 ore di straordinario in un anno.
- Tra la fine di una giornata lavorativa e l'inizio della successiva devono passare almeno 11 ore di riposo: un developer che finisce a mezzanotte non può ricominciare alle 7 del mattino.
- Il developer non può fare straordinario di propria iniziativa: deve essere autorizzato dall'azienda prima. Un'ora registrata come straordinario senza autorizzazione non è automaticamente dovuta.
- Gli straordinari vanno retribuiti con la maggiorazione prevista dal CCNL applicato. Per le aziende di consulenza IT, il CCNL più diffuso è il CCNL Terziario, Distribuzione e Servizi (Confcommercio), che prevede maggiorazioni diverse per straordinari diurni, notturni e festivi.

Reperibilità on-call: una voce a sé

La reperibilità è una condizione in cui il developer deve essere raggiungibile fuori dall'orario di lavoro e pronto a intervenire se necessario, ma non sta lavorando attivamente. Non va confusa con lo smart working o con il lavoro da remoto: sono situazioni diverse.

La reperibilità ha una sua retribuzione specifica, definita dal CCNL o dall'accordo individuale, che si applica indipendentemente dal fatto che il developer venga effettivamente chiamato o meno. Le ore in cui viene chiamato e lavora concretamente vanno invece registrate come ore ordinarie o straordinarie, a seconda di quando cadono.

Nel timesheet, la reperibilità deve essere una voce separata, non sommata alle ore lavorate. Questo ti permette di calcolare correttamente sia la retribuzione del mese sia il costo effettivo del servizio di reperibilità per l'azienda.

Il flusso che collega timesheet e paghe

Questo è il percorso che le ore devono fare dal timesheet alla busta paga, con i passaggi chiave:

- Il PM approva le ore del team a fine mese (o a fine settimana, se il ciclo paghe lo richiede): l'approvazione è il momento in cui le ore diventano dati ufficiali.
- Il sistema estrae automaticamente le ore straordinarie e di reperibilità già categorizzate, senza che qualcuno debba rileggere tutti i fogli ore a mano.
- I dati vengono trasmessi all'ufficio paghe (interno o studio esterno) in un formato strutturato: numero di ore straordinarie per dipendente, tipo (diurno, notturno, festivo), e ore di reperibilità.
- L'addetto paghe inserisce questi dati nel cedolino applicando le maggiorazioni previste dal CCNL applicato: se usi il CCNL Terziario (Confcommercio), le percentuali di maggiorazione sono definite contrattualmente.
- Il cedolino finale riporta separatamente la quota di retribuzione ordinaria e le voci aggiuntive legate agli straordinari e alla reperibilità, con piena tracciabilità dell'origine.

Metodo in quattro fasi per automatizzare il flusso

Un percorso realistico per passare dai fogli Excel a un sistema integrato, con tempi indicativi.

Fase	Cosa fai	Tempo indicativo
1. Mappa	Definisci la struttura cliente/progetto/attività e le tipologie di ore che vuoi tracciare	1-2 settimane
2. Configura	Imposti il software di timesheet con i tuoi progetti attivi, i tuoi clienti e le tariffe giornaliere	1 settimana
3. Forma il team	Fai una sessione breve con i developer, mostri come si registra un'ora, stabilisci la regola 'ogni giorno'	2-3 giorni
4. Collega paghe e fatturazione	Definisci il flusso di esportazione dati verso l'ufficio paghe e verso la fatturazione, testa per un mese pilota	2-4 settimane

Siamo Jet HR

Jet HR è una tech company nata nel 2022 – da un'idea di Marco Ogliengo e Francesco Scalabrino, supportata da alcuni tra i migliori imprenditori e fondi d'investimento italiani – per risolvere un problema reale: il peso della burocrazia sulle aziende in Italia.

Jet HR è una soluzione innovativa per pagare stipendi e automatizzare processi HR, affiancati da un team di supporto con tempi di risposta record e un Consulente del Lavoro dedicato.

Jet HR è una piattaforma che integra tutte le funzioni HR essenziali, specificatamente sviluppate per la normativa italiana, tra cui: assunzione di nuovi dipendenti; gestione di smart-working, ferie, permessi e malattia; rilevazione presenze; note spese; compliance salute e sicurezza; creazione e distribuzione dei cedolini; pagamento degli stipendi; e altro ancora.

Per saperne di più

jethr.com