

CHECKLIST

PRIMA DI PAGARE UN'AGENZIA: LA CHECKLIST DEL FOUNDER PER NON FARSI FREGARE (ANCHE SE NON SEI TECNICO)

- > Scopri se sei pronto a delegare (o se stai per buttare tempo e budget)
- > Proteggi il tuo progetto dalle fregature più comuni
- > Metti paletti prima di firmare e rimani sempre in controllo



EDIZIONE PDF

PIONEER 

INTRO

Se stai per delegare lo sviluppo, questa pagina può salvarti mesi (e budget)

Quando sei founder di una startup high-tech, il rischio non è “scegliere lo stack sbagliato”.

Il rischio vero è: pagare per confusione.

- Preventivi che sembrano chiari... finché non partono gli “extra”.
- Settimane senza demo.
- Scope che cambia ogni due giorni.
- Decisioni tecniche prese “perché sì”.
- Dipendenza totale dal fornitore (“senza di noi non puoi andare avanti”).

Questa checklist ti serve per una cosa: tornare in controllo.

Non devi diventare tecnico. Devi avere chiarezza, regole e segnali oggettivi.

Per ogni punto segna:

✅ Sì, è chiaro / già definito (2 punti)

⚠️ Più o meno / manca un pezzo (1 punto)

❌ No / non esiste (0 punti)

Il tuo semaforo

🟢 40–56 (Verde): puoi delegare, ma con paletti non negoziabili

🟡 23–39 (Giallo): stai per comprare rework → chiarisci prima di firmare

🔴 0–22 (Rosso): rischio “progetto zombie” → prima metti ordine e governance

Se hai più di 5 ❌ tra Chiarezza e Controllo & Proprietà, fermati.

Non è “prudenza”: è autoprotezione.

Non devi diventare tecnico. Devi avere chiarezza e controllo.

E quando hai chiarezza e controllo, prendi decisioni più sicure e smetti di pagare “per il caos”.

Alex Pagnoni

<https://alexpagnoni.com>

<https://pioneerx.tech>



CHIAREZZA

Cosa stai comprando davvero?

Qui si evita lo scenario: "pensavo fosse incluso".

- 1 So dire l'obiettivo in **1 frase** (risultato business, non "fare l'app")
- 2 Ho scritto cosa **NON** è incluso (anti-sorprese)
- 3 Ho **3 metriche** di successo (anche grezze, ma misurabili)
- 4 Ho una lista **Must / Should / Not now** (MVP)
- 5 C'è un **owner interno** che approva priorità (tu o una persona tua)
- 6 Il **brief** include vincoli (budget, tempi, integrazioni, compliance)
- 7 È chiaro se stiamo facendo **MVP** o **produzione** (qualità, sicurezza, scalabilità)

Red flags immediati

"Ci pensiamo strada facendo" (traduzione: paghi rework)

Preventivo "a giornate" senza output e criteri di consegna



PROCESSO

Come eviti i "ci aggiorniamo"

Qui si evita lo scenario: "sono passate 3 settimane e non ho visto nulla".

- 8 C'è una cadenza fissa: **update settimanale + demo** (non solo call)
- 9 Esiste una **Definition of Done** (test minimi, deploy, doc essenziale)
- 10 Backlog e task tracking sono **visibili e condivisi** (Linear/Jira/Notion)
- 11 Ogni settimana esce un output verificabile: **demo/release/report**
- 12 C'è una regola per gestire i cambi: **richiesta → impatto → approvazione**
- 13 I rischi vengono detti **subito** (non "scoperti" alla consegna)
- 14 Comunicazione chiara: **un canale + un responsabile** (chi decide cosa)

Red flags immediati

"Ti facciamo vedere quando è pronto"

Nessun backlog visibile = zero controllo



TRASPARENZA TECH

Capire senza diventare CTO

Qui si evita lo scenario: "pensavo fosse incluso".

- 15 Le scelte tech sono spiegate in linguaggio founder: **perché / impatto / rischi**
- 16 Lo stack è motivato (non moda): se è complesso, c'è una ragione concreta
- 17 Sicurezza minima: ruoli/accessi, backup, gestione credenziali
- 18 Staging + produzione: è chiaro **come si testa** prima di andare live
- 19 Monitoring minimo: log/errori (se qualcosa si rompe, lo sai subito)
- 20 Qualità: test minimi + gestione bug (non "fixiamo dopo")

Red flags immediati

"È standard" ma non sanno dirti trade-off e rischi

Overkill per MVP senza motivazione



CONTROLLO & PROPRIETÀ

La parte che ti salva dal lock-in

Qui si evita lo scenario: “non ho le chiavi”.

- 21 Il codice è sotto **tuo controllo** (repository GitHub/GitLab del cliente)
- 22 Accessi/credenziali sono gestiti da te (password manager/vault)
- 23 Documentazione minima: come avviare / deployare / orientarsi (anche 1 pagina)
- 24 Proprietà IP chiara (codice, design, asset) + consegna garantita
- 25 Piano di handover: cosa succede se cambi fornitore
- 26 Dipendenze/licenze/tool critici dichiarati (costi ricorrenti inclusi)
- 27 Budget trasparente: incluso vs extra + condizioni (anti fatture sorpresa)
- 28 Progressi verificabili con prove (demo + accessi + tracking), non a sensazione

Red flags immediati

“Repository nostro, poi te lo passiamo alla fine”

Hosting/analytics/domini in mano al fornitore

Cosa fare col risultato (senza complicarti la vita)

Se sei in Verde

Prima di firmare, rendi non negoziabili questi 3 punti:

1. Repo e accessi tuoi (D21–D22)
2. Demo settimanale + backlog condiviso (B8–B11)
3. Change management scritto (B12)

Se sei in Giallo

Non firmare finché non sistemi:

- Obiettivo + scope + metriche + priorità + owner (A1–A5)
- Repo + accessi (D21–D22)

Azione rapida (45 minuti): fai una call e chiedi all'agenzia di produrre entro 48h:

- 1 pagina: obiettivo + esclusioni + metriche
- piano 2–4 settimane con demo e output

Se sei in Rosso

Non è "pessimismo": è protezione.

Prima metti ordine:

- Obiettivo + must/should + vincoli (anche grezzi, ma scritti)
- Governance minima (demo, tracking, repo tuo)

Poi chiedi preventivi su una base chiara.

BONUS - Mini-Canvas Anti-Caos

Compilalo e allegalo al tuo brief. Ti farà risparmiare settimane.

1) Obiettivo (1 frase): _____

2) Utente target: _____

3) Problema che risolvi: _____

4) Must-have MVP (max 5):

-
-
-
-
-

5) Non incluso (max 5):

-
-
-
-
-

6) Metriche (max 3): _____

7) Vincoli (tempo/budget/compliance): _____

8) Owner decisioni: _____

