fabrick

Fabrick Checkout

Manuale utente

Version 1.0.1

Modulo PrestaShop 8 / 9

Destinatari: merchant / amministratori e-commerce

Documento: agosto 2026

Modulo PrestaShop 8 / 9 | Versione modulo **1.0.1**
Destinatari: merchant / amministratori e-commerce
Documento: settembre 2026

---

1. Introduzione

Fabrick Checkout integra i pagamenti **Fabrick Payment Orchestra** in PrestaShop tramite **REST API** e **Web Component** <fbk-checkout>.

Funzionalità principali:

- Pagamento standard (carta e metodi APM abilitati sullo shopLogin)
- Checkout su **pagina dedicata (Redirect)**
- **Pay by Link** (opzionale)
- **Salvataggio carta** con opt-in del cliente e gestione da Account (opzionale)
- Notifiche server-to-server e callback buyer
- Cron di riconciliazione (es. Pay by Bank)
- Strumento rimborsi dal backoffice
- **Multinegozio (multishop):** configurazione e credenziali **per negozio**

---

2. Requisiti

- PrestaShop **8.0+** o **9.x**
- Account Fabrick con **Payment Orchestra**, Web Components e API REST abilitati
- Credenziali: **Shop Login** e **API Key** (sandbox o produzione)
- HTTPS sul dominio del negozio (consigliato / obbligatorio in produzione)
- Per Apple Pay / Google Pay: prerequisiti dominio e attivazione lato Fabrick (vedi documentazione ufficiale)

---

3. Installazione

1. Caricare l'archivio o copiare la cartella fabrickcheckout in modules/ del negozio.
2. Backoffice -> **Moduli** -> cercare **Fabrick Checkout** -> **Installa**
3. Aprire **Configura** e compilare le impostazioni (capitolo 4).
4. Verificare che il modulo sia **attivo** e disponibile per le valute del carrello.

Dopo un aggiornamento senza reinstallazione, aprire almeno una volta la pagina di configurazione del modulo (self-heal di hook/config se necessario).

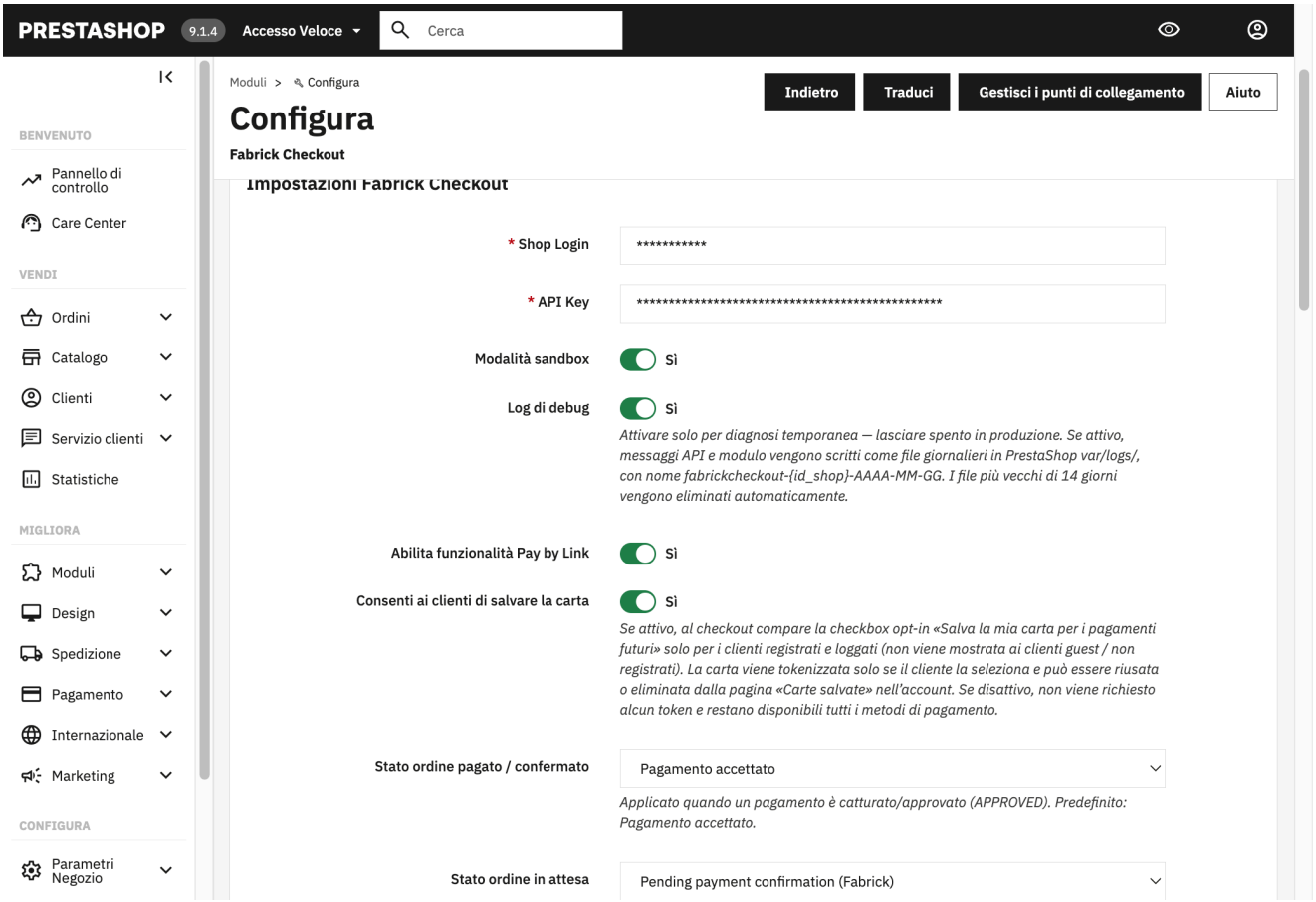
---

4. Configurazione backoffice

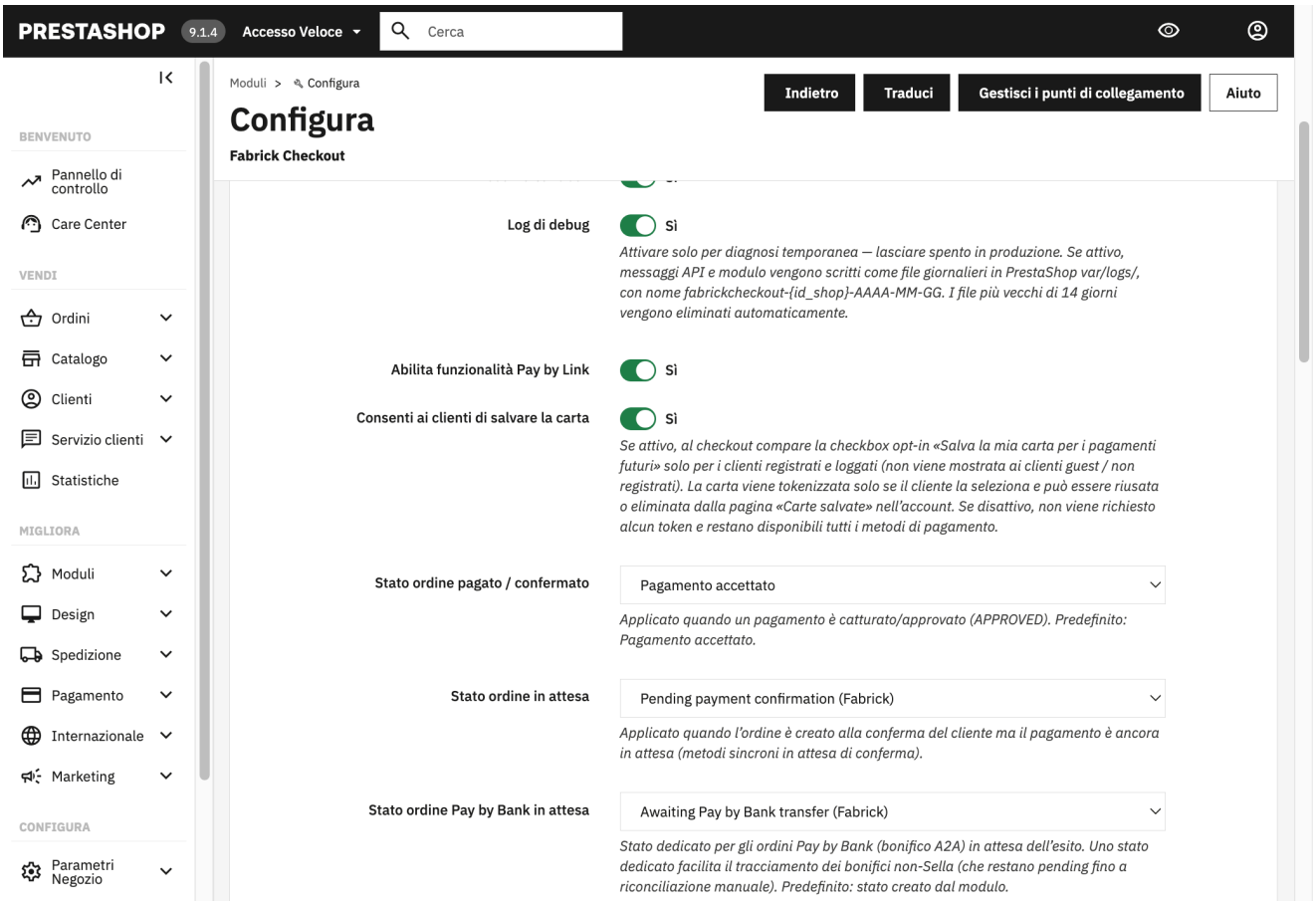
Percorso: **Moduli -> Fabrick Checkout -> Configura**

Multinegozio: se PrestaShop ha più shop, selezionare il negozio desiderato nell'intestazione backoffice **prima** di aprire o salvare la configurazione. Le impostazioni non si applicano in blocco a "Tutti i negozi".

Esempio della pagina di configurazione (valori Shop Login / API Key offuscati):



Configurazione Fabrick Checkout - Shop Login e API Key (backoffice)



Configurazione Fabrick Checkout - opzioni funzionali (backoffice)

4.1 Credenziali e ambiente

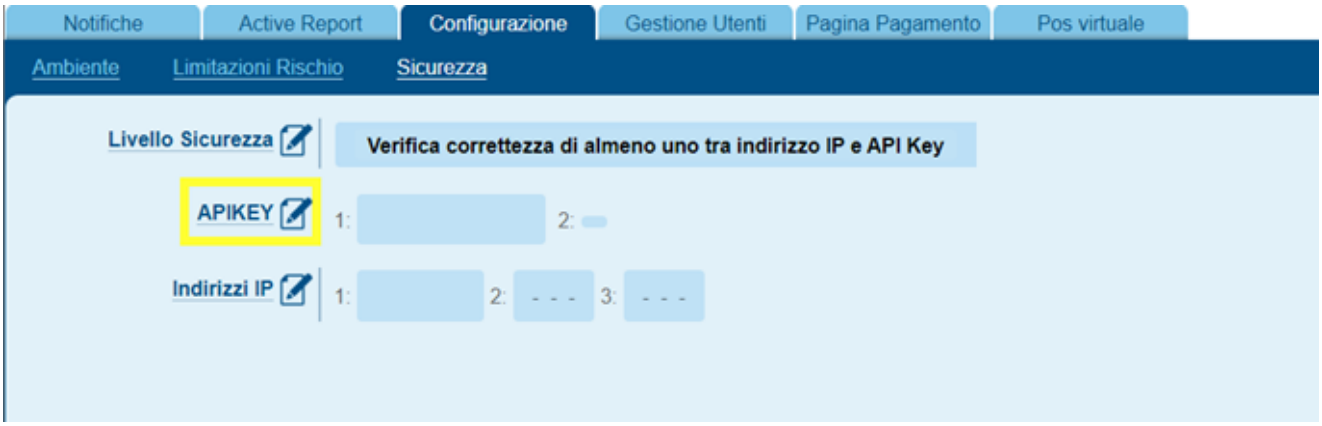
Impostazione	Descrizione
Shop Login	Codice merchant Fabrick
API Key	Chiave API (non condividere)
Modalità sandbox	Sì = ambiente di test; No = produzione
Log di debug	Scrive log giornalieri in var/logs/fabrickcheckout-{id_shop}-AAAA-MM-GG (solo per diagnosi; disattivare in produzione se non serve)

4.1.1 Back-office merchant Fabrick

Nel **back-office merchant** messo a disposizione da Fabrick è possibile recuperare le credenziali (**Shop Login** e **API Key**) da usare poi nella configurazione del modulo in PrestaShop.

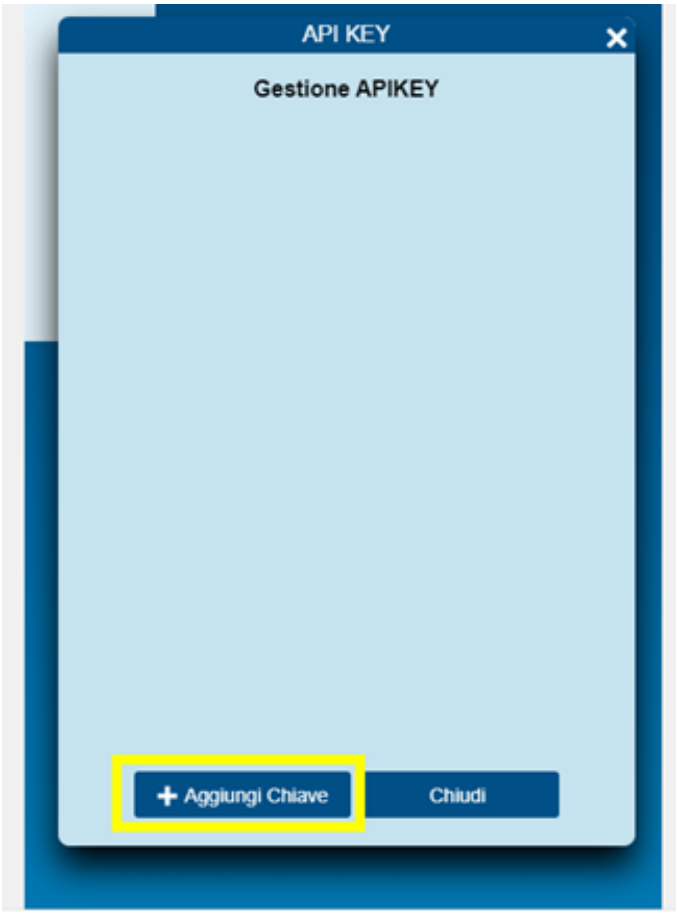
Creazione e copia dell'API Key

1. Andare in **Configurazione -> Sicurezza -> APIKEY** e creare il valore richiesto.



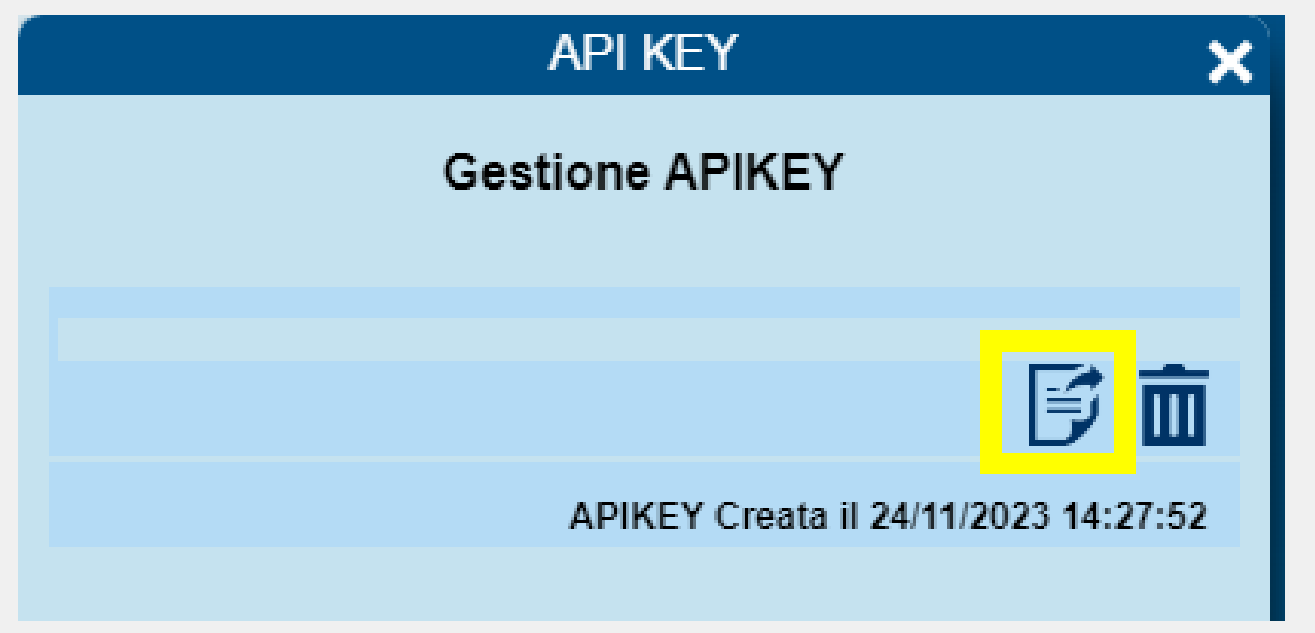
Back-office Fabrick - Configurazione > Sicurezza, sezione APIKEY

2. Premere **Aggiungi Chiave**.



Back-office Fabrick - Gestione APIKEY, Aggiungi Chiave

3. Usare l'icona di **copia** (evidenziata nello screenshot) per copiare la chiave negli appunti e incollarla nel campo **API Key** del modulo PrestaShop.



Back-office Fabrick - copia API Key

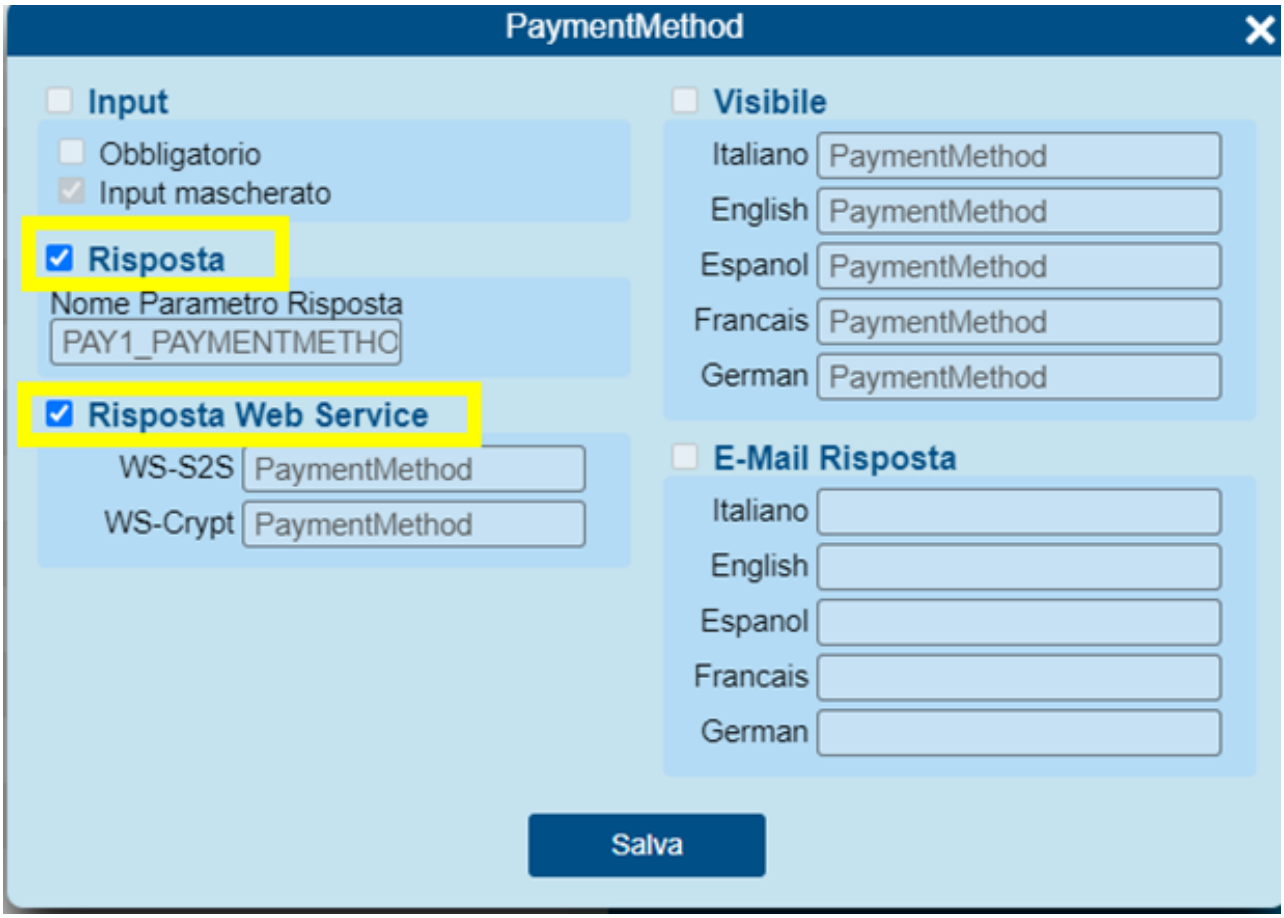
Parametri in risposta (Pagina di Pagamento)

In **Pagina di Pagamento -> Campi & Parametri** abilitare in risposta i parametri **PaymentMethod**, **Token**, **Token ExpiryMonth** e **Token ExpiryYear** su entrambe le pagine. Di seguito l'esempio con **PaymentMethod**; gli altri seguono la stessa logica.



Back-office Fabrick - elenco parametri, PaymentMethod

Nella scheda del parametro, flaggare le voci **Risposta** e **Risposta Web Service**, poi salvare.



Back-office Fabrick - PaymentMethod: Risposta e Risposta Web Service

4.2 Opzioni funzionali

Impostazione	Descrizione
Abilita funzionalità Pay by Link	Mostra l'opzione Pay by Link al checkout
Consenti ai clienti di salvare la carta	Mostra la checkbox opt-in "Salva la mia carta" e la pagina Carte salvate in Account

4.3 Stati ordine

Configurare gli stati PrestaShop per:

- Pagamento confermato / accettato
- Pagamento in attesa (generico)
- Pay by Bank in attesa
- Pay by Link in attesa / pagato
- Pagamento fallito

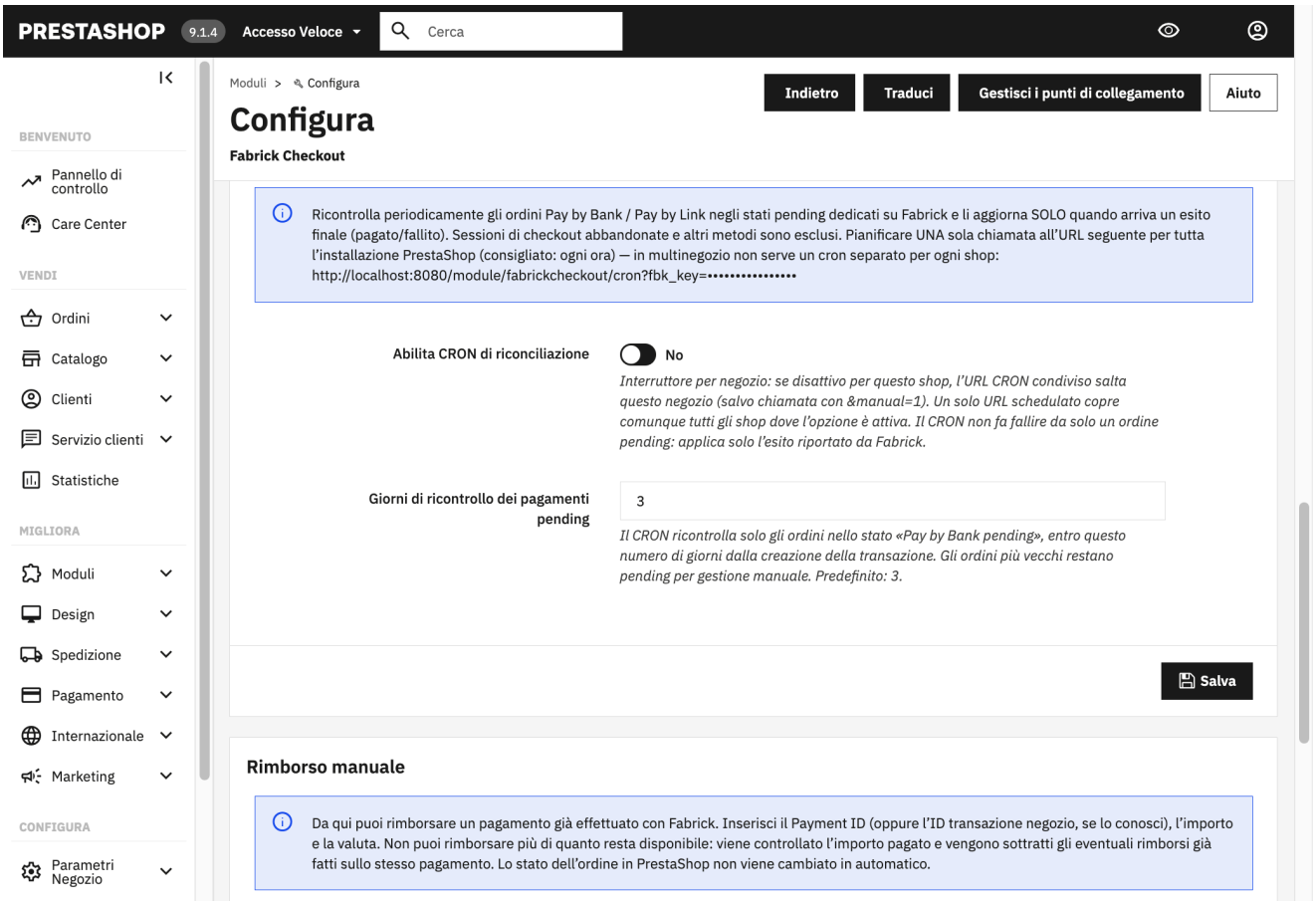
Il modulo crea stati dedicati all'installazione; è possibile rimapparli su stati già esistenti.

4.4 Pay by Link - validità

Validità Pay by Link (ore): durata di validità del link (tipicamente fino a 48-49 ore secondo regole Fabrick).

4.5 Riconciliazione pagamenti in attesa (CRON)

Sezione backoffice: **Riconciliazione pagamenti in attesa (CRON)**.



Riconciliazione pagamenti in attesa (CRON) - backoffice

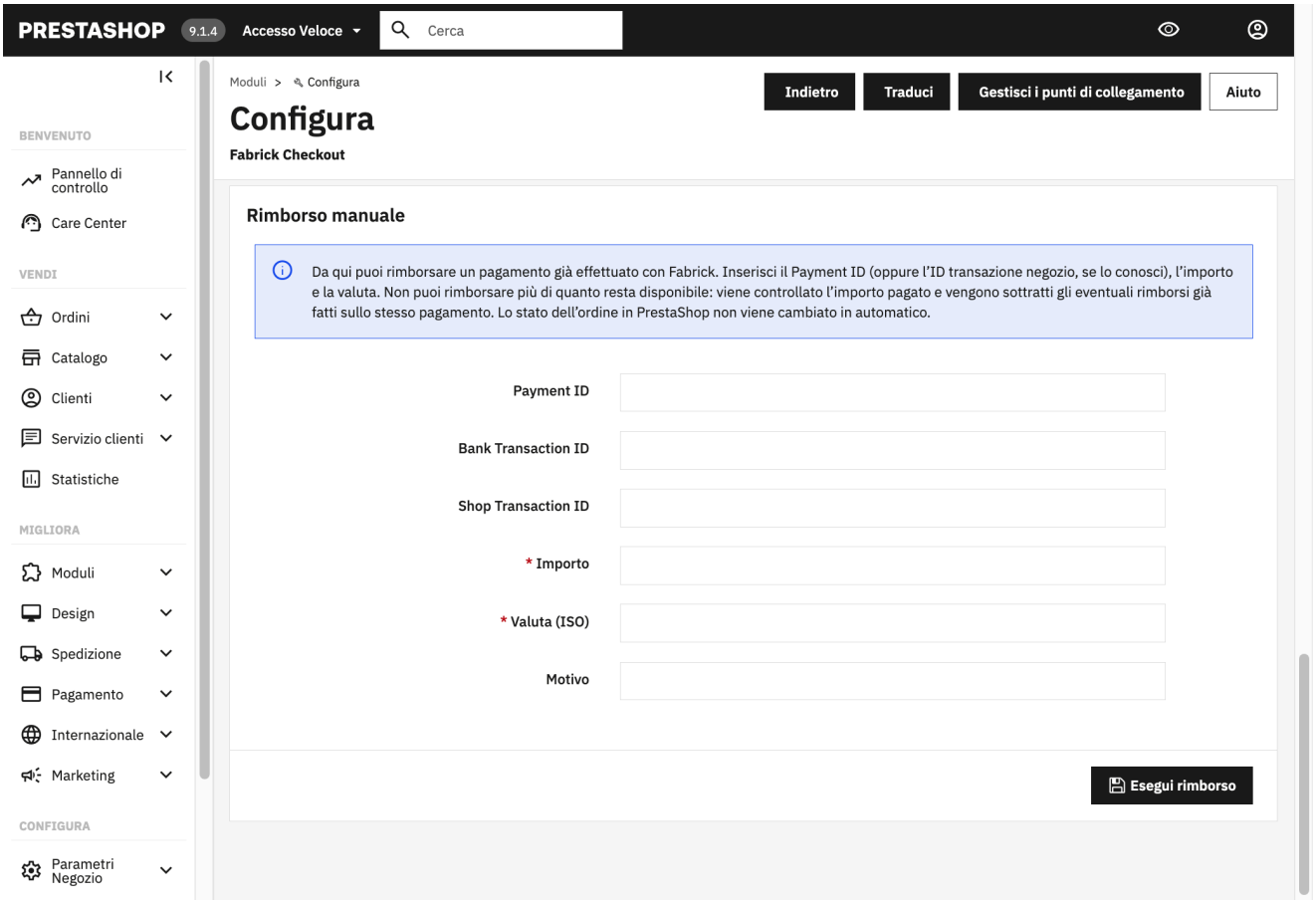
Impostazione	Descrizione
Abilita CRON di riconciliazione	Attiva la logica di riconciliazione
Giorni di ricontrollo dei pagamenti pending	Finestra in giorni per le verifiche

Pianificare **una sola** chiamata periodica (es. oraria) all'URL cron mostrato in configurazione. In **multinegozio** non serve un cron per ogni shop: lo stesso URL (dominio principale, senza /shop2/) riconcilia tutti i negozi dove "Abilita CRON" è attivo.

Esempio: https://tuodominio/module/fabrickcheckout/cron?fbk_key=...
(La fbk_key nell'URL è quella del negozio predefinito; non va condivisa pubblicamente.)

4.6 Rimborso manuale

Sezione backoffice: **Rimborso manuale**.



Rimborso manuale - backoffice

Da qui puoi rimborsare un pagamento già effettuato con Fabrick. Inserisci il **Payment ID** (oppure l'ID transazione negozio se lo conosci), l'importo, la valuta e un eventuale motivo; poi premi **Esegui rimborso**.

- **Non puoi rimborsare più di quanto resta da rimborsare**: se hai già fatto uno o più rimborsi parziali sullo stesso pagamento, conta solo la differenza ancora disponibile.
- Lo **stato dell'ordine in PrestaShop non viene aggiornato** automaticamente. Se serve, aggiornalo a mano dalla scheda ordine.

---

5. Modalità di visualizzazione checkout

Il modulo usa la modalità **Redirect** (pagina dedicata con Web Component).

- Il cliente sceglie **Pay with Fabrick**.
- Viene reindirizzato a una **pagina dedicata** del modulo con il Web Component.
- Con tokenizzazione attiva, la checkbox "Salva la mia carta" è nel form di pagamento PrestaShop e viene inviata al create della sessione.

---

6. Flussi frontend

6.1 Pagamento standard

1. Cliente completa carrello e checkout fino al pagamento.
2. Seleziona **Pay with Fabrick** (e, se previsto, opt-in salvataggio carta).
3. Completa il pagamento nel Web Component Fabrick (carta o APM disponibili).
4. Al termine: conferma ordine PrestaShop (o stato pending secondo il metodo).

6.2 Pay by Link

Se abilitato, al checkout compare un'opzione dedicata. Il cliente riceve/usa il link secondo la configurazione Fabrick; l'ordine può restare in stato pending fino all'esito.

6.3 Carte salvate

Se la tokenizzazione è attiva e il cliente ha salvato almeno una carta:

- Al checkout può apparire **Pay with saved card** (brand / ultime cifre / scadenza).
- Dall'**Account cliente** -> link **Saved cards** / carte salvate: elenco e cancellazione (revoca anche lato Fabrick quando possibile).

6.4 Callback e notifiche

- **Callback browser:** ritorno del cliente dopo il pagamento.
- **Notifica S2S:** aggiornamento affidabile dello stato lato server (non disabilitare URL pubblici del modulo).

---

7. Sandbox e test

1. Attivare **Modalità sandbox**.
2. Usare le **carte di test** ufficiali Fabrick (es. frictionless / challenge / autenticazione 3DS fallita).
3. Verificare creazione ordine, stati e, se usato, salvataggio carta.
4. Con **Log di debug** attivo, controllare var/logs/fabrickcheckout-{id_shop}-AAAA-MM-GG sul server.
5. Su un pagamento **rifiutato** (DECLINED, es. carta di test con 3DS fallito) il cliente torna allo step di pagamento e vede in **testata** un avviso che il pagamento non è stato completato, con il motivo restituito da Fabrick quando disponibile (es. "Autorizzazione negata").

Non usare carte reali in sandbox.

---

8. Produzione - checklist

[ ] Modalità sandbox = **No**
[ ] Credenziali di produzione
[ ] Log di debug = **No** (o limitato)
[ ] HTTPS e URL di callback/notify raggiungibili da Fabrick
[ ] Cron pianificato se si usano metodi asincroni (es. Pay by Bank)

[ ] Test E2E su carta e APM abilitati
[ ] Apple Pay / Google Pay: dominio verificato se richiesti

9. Risoluzione problemi

Se qualcosa non funziona come previsto, conviene partire dalle verifiche più semplici (configurazione e cache) e solo dopo approfondire log e rete.

Il metodo di pagamento non compare al checkout.

Controllare che il modulo sia installato e attivo, che la valuta del carrello sia tra quelle abilitate e che Shop Login e API Key siano correttamente impostati. In alcuni casi anche restrizioni di paese o valuta sul profilo Fabrick possono nascondere il metodo.

Pagina bianca dopo "Pay with Fabrick".

Spesso dipende dalla cache del tema o di Smarty: svuotarla e riprovare. Controllare inoltre la console del browser per errori di caricamento dello script Fabrick (fbk-elements.js) e che non ci siano blocchi CSP o di rete.

La carta non viene salvata.

La tokenizzazione deve essere attiva in backoffice e il cliente deve aver spuntato "Salva la mia carta" **prima** di completare il pagamento. Nei log, con debug attivo, la sessione pagata deve mostrare requestToken=MASKEDPAN e save_card_requested=1.

L’ordine resta in stato pending.

Può essere normale per metodi asincroni (ad esempio Pay by Bank). Verificare che le notifiche server-to-server arrivino, che il cron sia pianificato se previsto e che gli stati ordine in configurazione siano quelli attesi.

Dopo un pagamento fallito non compare nessun avviso.

Comportamento atteso del modulo aggiornato: al ritorno da buyerKO / DECLINED deve comparire in testata del checkout un messaggio di pagamento non completato (eventualmente con il motivo Fabrick). Se manca, svuotare la cache PrestaShop, verificare che il modulo sia aggiornato e che l’hook actionFrontControllerInitAfter sia registrato (basta aprire la configurazione del modulo in backoffice per lo self-heal).

Il rimborso non cambia lo stato ordine.

Lo strumento **Rimborso manuale** rimborsa su Fabrick e non permette di superare quanto resta da rimborsare (anche dopo più rimborsi parziali). Lo stato in PrestaShop va aggiornato a mano in **Ordini**, se necessario.

Dove leggere i log.

Con **Log di debug** attivo, le chiamate API sono in var/logs/ del negozio PrestaShop, file giornaliero fabrickcheckout-{id_shop}-AAAA-MM-GG (es. fabrickcheckout-1-2026-09-04). Restano al massimo 14 giorni. In produzione lasciare il debug attivo solo per il tempo necessario alla diagnosi.

© Fabrick | Modulo PrestaShop Fabrick Checkout 1.0.1