Stripe payments for Code Craft Store (Next.js)

Add-on version 1.0.0 · sold separately · licence item `nextjs-store-stripe-addon`

Card, Apple Pay, Google Pay and local payment methods through **Stripe Checkout** (hosted payment page).

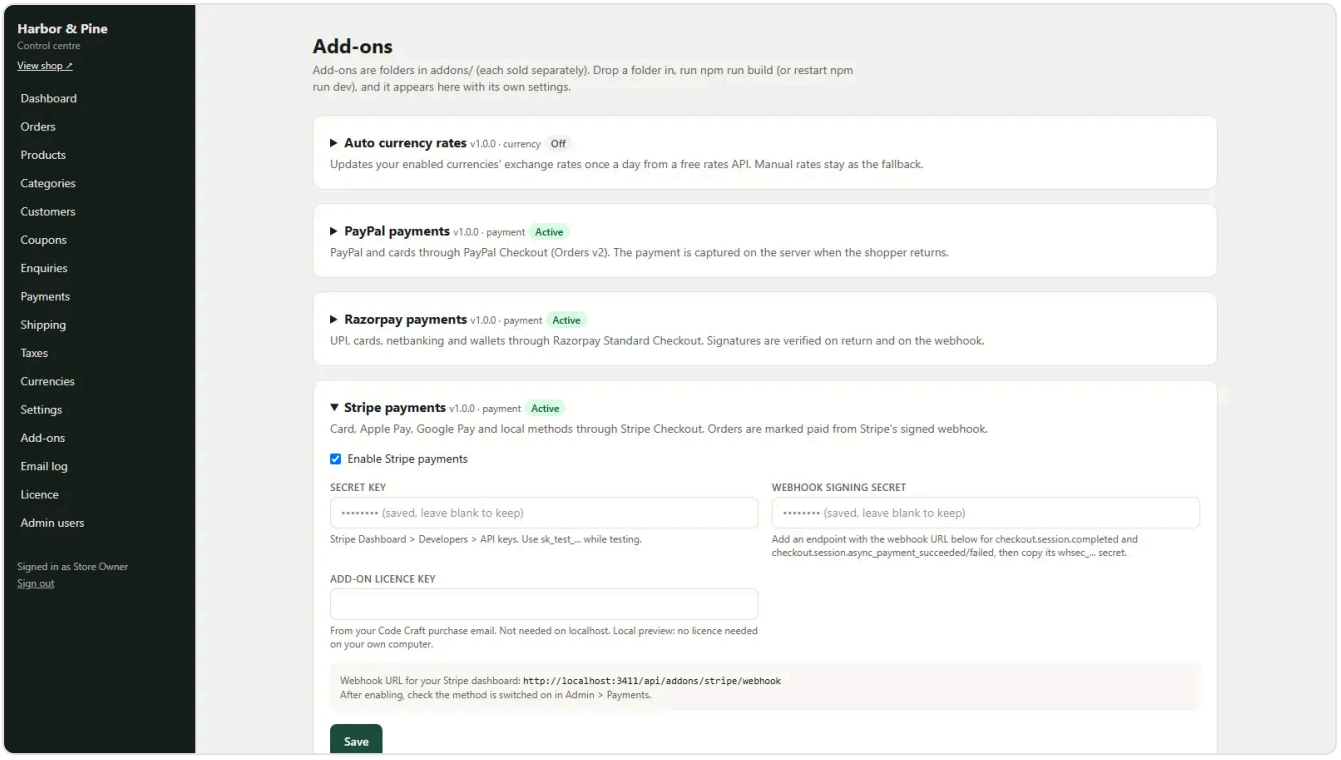
How it works

- At checkout the shopper picks *Card / wallet (Stripe)*. The store creates a Checkout Session (POST /v1/checkout/sessions) for the exact order total in the order currency, with an idempotency key, and redirects to Stripe.
- When the shopper returns, the add-on reads the session back from Stripe (it never trusts the redirect alone) and the store checks amount and currency before marking the order **Paid**.
- Stripe also calls your webhook. The add-on verifies the `Stripe-Signature` header (HMAC-SHA256 of timestamp + raw body, 5-minute tolerance) before accepting it.

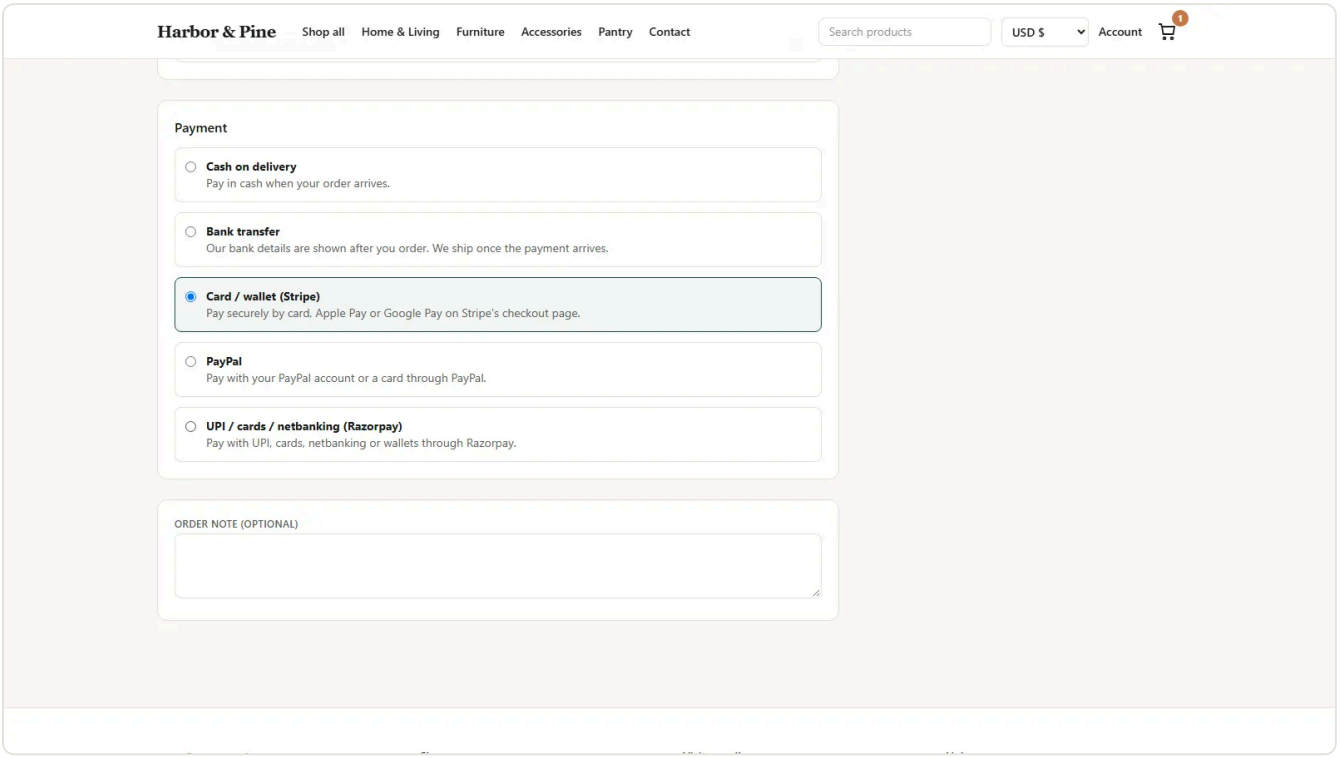
Settings

- **Secret key**: Stripe Dashboard > Developers > API keys (sk_test_... while testing, sk_live_... when live).
- **Webhook signing secret**: Dashboard > Developers > Webhooks > Add endpoint `https://your-shop.com/api/addons/stripe/webhook` with events checkout.session.completed, checkout.session.async_payment_succeeded, checkout.session.async_payment_failed and checkout.session.expired. Copy its whsec_... secret.

Refunds are made in the Stripe Dashboard; then set the order to Refunded in the store.



Stripe settings in Admin → Add-ons, with the webhook URL to copy.



The gateways as payment choices at checkout.

Install

1. You need Code Craft Store (Next.js Ecommerce) 1.0 or later.
2. Copy this folder into the store as `addons/stripe/` (keep the folder name).
3. Run `npm run build` and restart the store (or restart `npm run dev`).

4. Open **Admin > Add-ons > Stripe payments**, tick Enable, fill in the settings and paste the add-on licence key (not needed on localhost).
5. Payment add-ons also appear in **Admin > Payments**, where you can rename them and set their order.

Tests

From the store folder: `npm test`. The tests answer every gateway API call with a mock, so they run without keys and never contact the provider.

Honest notes

This add-on follows the provider's official API documentation and is tested against mocked API responses. Before taking real orders, place a test order with your provider's test/sandbox keys.

Support: https://coodecraft.com/contact