Stripe payments — add-on for the Code Craft HTML + PHP shop templates

Take card, Apple Pay and Google Pay payments on **Stripe Checkout**, Stripe's hosted payment page. The order is marked *Paid* after the Checkout Session is read back from Stripe, or a signed webhook arrives, and the amount matches.

Product page: https://coodecraft.com/market/product/php-shop-stripe-addon

Works with the 12 Code Craft shop templates (site core 1.1 / shop engine 1.0 or later). It does not work with WordPress or with the non-shop templates.

How it works

1. When the order is placed, the add-on creates a **Checkout Session** (`POST /v1/checkout/sessions`) for the exact total, with an idempotency key, and sends the customer to Stripe.
2. On the way back it **re-reads the session** from Stripe (`GET /v1/checkout/sessions/{id}`) and only accepts `payment_status: paid` for that order's own session.
3. A **webhook** (`checkout.session.completed`, `checkout.session.async_payment_succeeded`) is checked with the `Stripe-Signature` header: HMAC-SHA256 of `timestamp.body` with your `whsec_` secret, at most 5 minutes old.
4. The shop engine compares currency and amount with the order before marking it paid, and never records the same payment twice.

Install

1. Upload the `stripe` folder into your website's `addons/` folder (so you have `addons/stripe/addon.php`).
2. Open **Admin** → **Add-ons**, switch on **Stripe payments** and open its **Settings**.
3. Paste your add-on licence key (from your purchase email; not needed on localhost) and your Stripe details (below).
4. In Stripe → Developers → Webhooks, add the endpoint URL shown on the add-on settings screen (`…/cc/endpoint.php?addon=stripe&action=webhook`) with the events **checkout.session.completed** and **checkout.session.async_payment_succeeded**, then paste its signing secret.
5. Open **Admin** → **Shop** → **Payments**: the method is listed there. Keep it ticked and set its position.
6. Place a test order with your `sk_test_` key and Stripe's test card 4242 4242 4242 4242 before you switch to live keys.

Settings

Setting	Meaning
Add-on licence key	Your key for this add-on, checked with coodecraft.com like the website's own licence (always active on localhost). Until it is active the method is not offered.
Secret key	`sk_test_…` / `sk_live_…` (or a restricted `rk_` key with Checkout Sessions write access). Never shown again.
Webhook signing secret	The `whsec_…` secret of your webhook endpoint.
Name at checkout	What customers see in the list of payment methods.

How it was tested

Real PHP 8.3 (`php -S`) on copies of the Kora and Pustak shop templates, with every Stripe API call answered by a local mock server (`config.php` → `api_base`), 47 checks per shop together with the other gateway add-ons: Checkout Session created with the server total, the right currency, an idempotency key and a success URL carrying `{CHECKOUT_SESSION_ID}`; return before payment → pending; return with another session id → refused; return after payment → session re-read → paid; webhook with a tampered signature → 400; webhook older than 5 minutes → 400; signed webhook with a lower amount → not paid; valid webhook → paid.

Limits and honest notes

- Tested only against **mocked** Stripe responses and locally forged callbacks/webhooks. It was **not** tested with real money or the live Stripe service. Try a test-mode payment yourself before going live.
- You need your own Stripe account. Stripe fees, payouts, refunds and disputes are handled in your Stripe dashboard; refunds are not sent from the website (set the order to *Refunded* yourself after refunding).
- The amount charged is the order total in your base currency, or in the customer's currency if you chose that in **Shop settings** (at your rate).
- Stripe does not support every currency/country combination; check Stripe's list for your account.

Files

- `addon.php`: the manifest (settings shown in Admin → Add-ons).
- `stripe.php`: the code (payment method, return/callback and webhook endpoints).
- Details of the add-on contract: `cc/ADDONS.md` on your website.

The settings screen

Admin → Add-ons → stripe, on a local copy of the Kora shop template:

