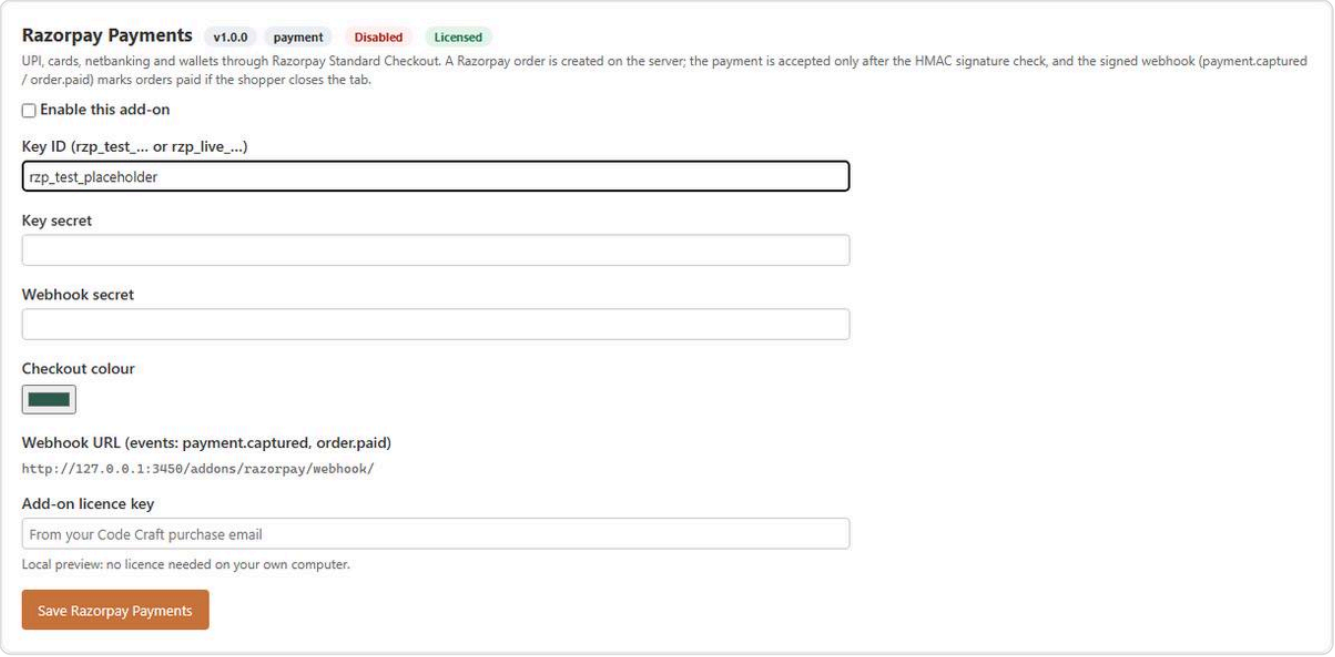
CODE CRAFT · ADD-ON GUIDE · VERSION 1.0

Razorpay Payments Add-on

For Code Craft Store: Django Ecommerce (sold separately). You need the store installed first.



The add-on's card in Admin → Add-ons.

What it does and how to install it

Takes UPI, cards, netbanking and wallets through Razorpay Standard Checkout. A Razorpay order is created on the server for the exact amount; after payment the HMAC signature `order_id|payment_id` is verified with your key secret before the order is marked paid. A signed webhook (`payment.captured` / `order.paid`) covers shoppers who close the tab.

1. Upload the `razorpay` folder as `addons/razorpay/` and restart the app.
2. Razorpay Dashboard → Account & Settings → API keys: generate a **test** key (Key ID and Key secret).
3. Webhooks → Add new: the URL shown on the add-on card (`https://yourdomain.com/addons/razorpay/webhook/`), a secret of your choice, events `payment.captured` and `order.paid`.
4. Admin → Add-ons → Razorpay: Key ID, Key secret, Webhook secret, licence key, Enable, Save.
5. Admin → Payment methods: tick Razorpay. Test in test mode, then switch to live keys.

Razorpay mainly charges INR; enable INR in Admin → Currencies if your base currency is different.

> Honest testing note: this add-on follows the provider's official API documentation. Its automated tests answer every provider call with a simulated response (including forged and expired signatures and wrong amounts); it has not processed real payments. Always run a test or sandbox payment before going live.

Where things are

- `addon.json` : id, version, type and the settings shown in the admin.
- `hooks.py` : the calls to the provider.
- `urls.py` : return and webhook endpoints under `/addons/razorpay/` .

Each add-on has its own licence key (in your purchase email and at coodecraft.com/account). Help: coodecraft.com/contact.