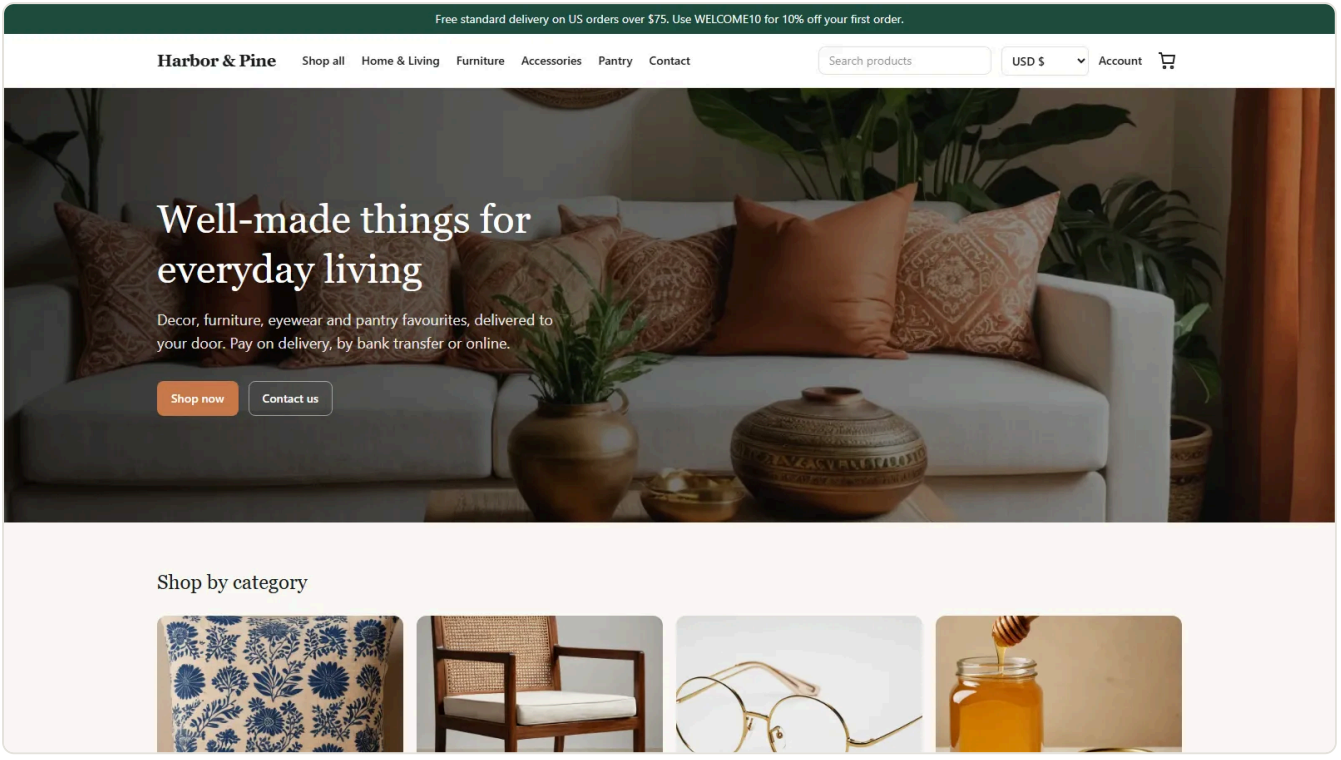
Code Craft Store: Next.js Ecommerce

Installation & owner's guide · version 1.0.0 · October 2026

Next.js 16 · React 19 · Prisma 6 · SQLite / PostgreSQL / MySQL

This guide takes you from the download to a live shop: installing, the first-run setup wizard, adding products with variants, payments, delivery, taxes, currencies, orders and going live with your licence key. The demo shop is called **Harbor & Pine**; every name, address and product can be changed in the admin.



The storefront home page of the demo shop (announcement bar, currency switcher, categories, WhatsApp button).

1. What you need

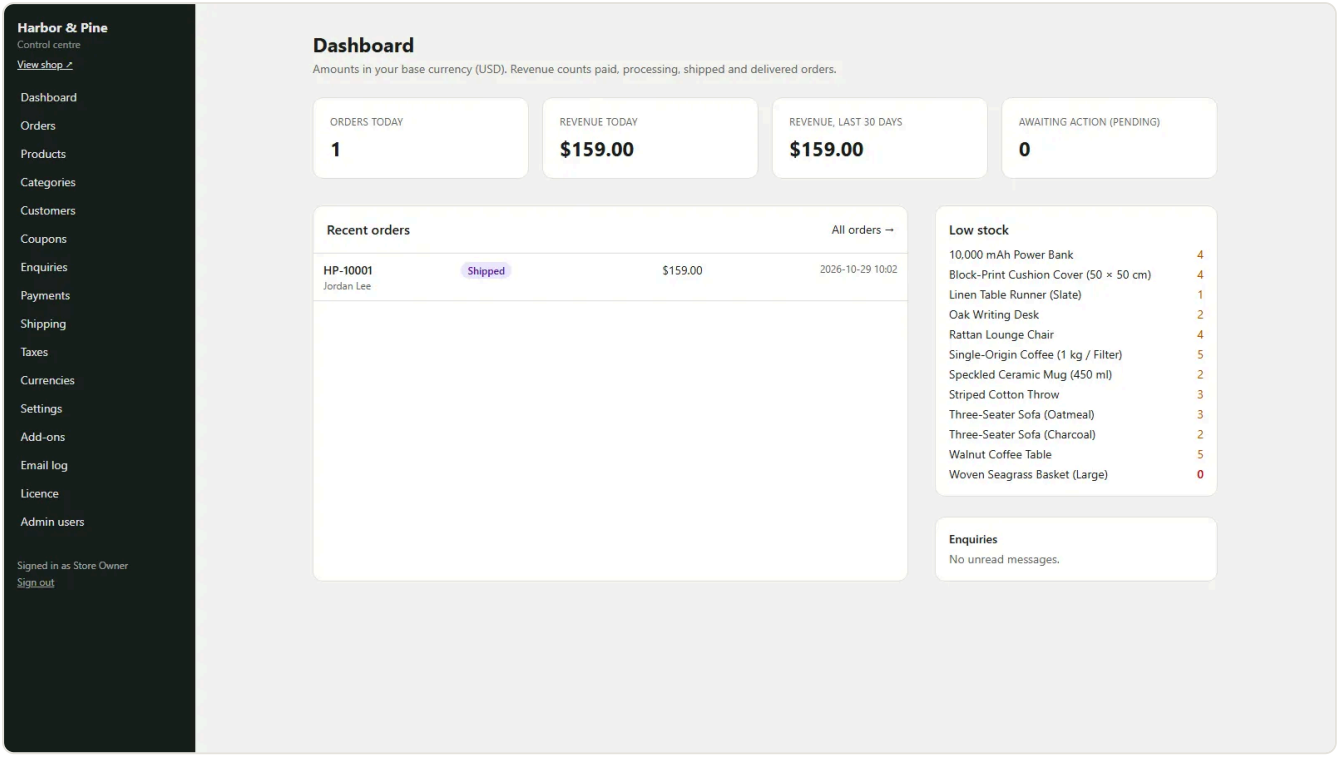
- Node.js 20.9 or later (22 LTS recommended) on your computer and on your host.
- A host that runs Node.js apps: a VPS, Hostinger/cPanel "Node.js app", Render, Railway or similar. Shared PHP-only hosting is not enough.
- For the database: nothing (SQLite file) or a PostgreSQL/MySQL database from your host.
- Your licence key from the Code Craft purchase email (only needed on the live domain).

2. Install on your computer

```
npm install
copy .env.example .env      (macOS/Linux: cp .env.example .env)
npm run setup               creates the database + demo shop
npm run dev                 opens on http://localhost:3000
```

Open `http://localhost:3000/admin`. The first time, the setup wizard asks for your shop name and creates your admin login (password of at least 10 characters, stored as a bcrypt hash). After that the wizard is closed for good; more admins can be added in **Admin** → **Admin users**.

3. The control centre

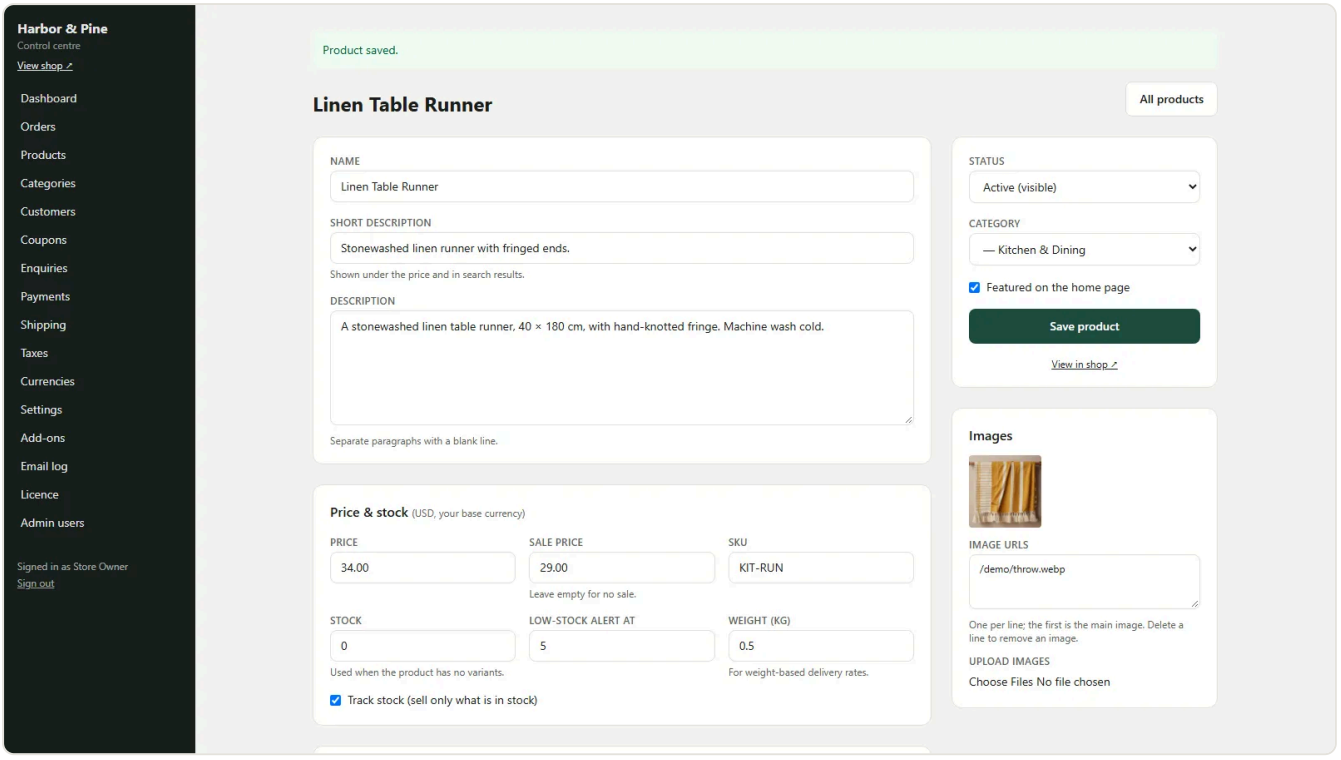


Dashboard: orders today, revenue, items awaiting action, recent orders and low-stock items.

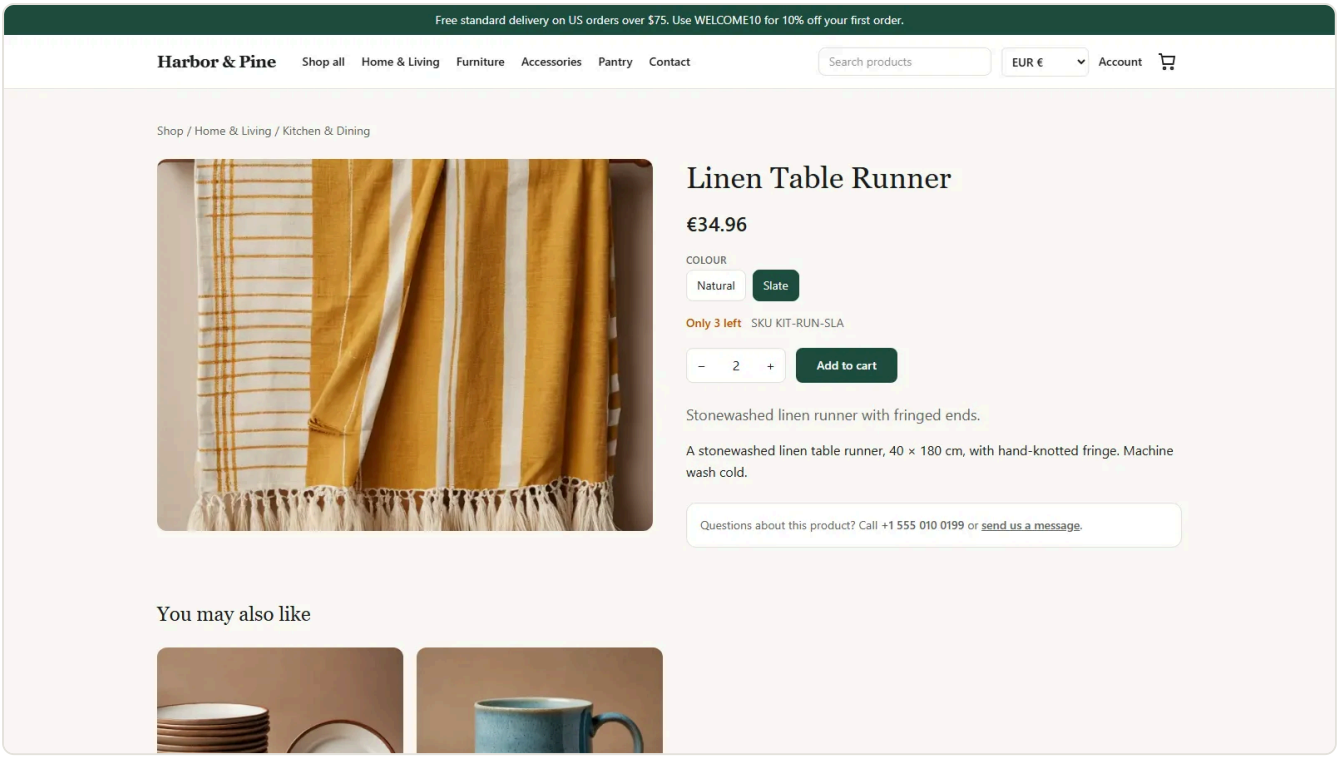
The left menu holds everything: Orders, Products, Categories, Customers, Coupons, Enquiries, Payments, Shipping, Taxes, Currencies, Settings, Add-ons, Email log, Licence and Admin users.

4. Products and variants

1. **Products → Add product.** Enter the name, descriptions, price (in your base currency), optional sale price, SKU, stock, low-stock alert level and weight, choose a category and upload photos.
2. For variants, type the option names (for example `Size, Colour`) and save. A table appears: fill one row per combination with its own SKU, price, sale price and stock. Leave the variant price empty to use the product price.
3. Untick *Track stock* for made-to-order items. Draft products are hidden from the shop.



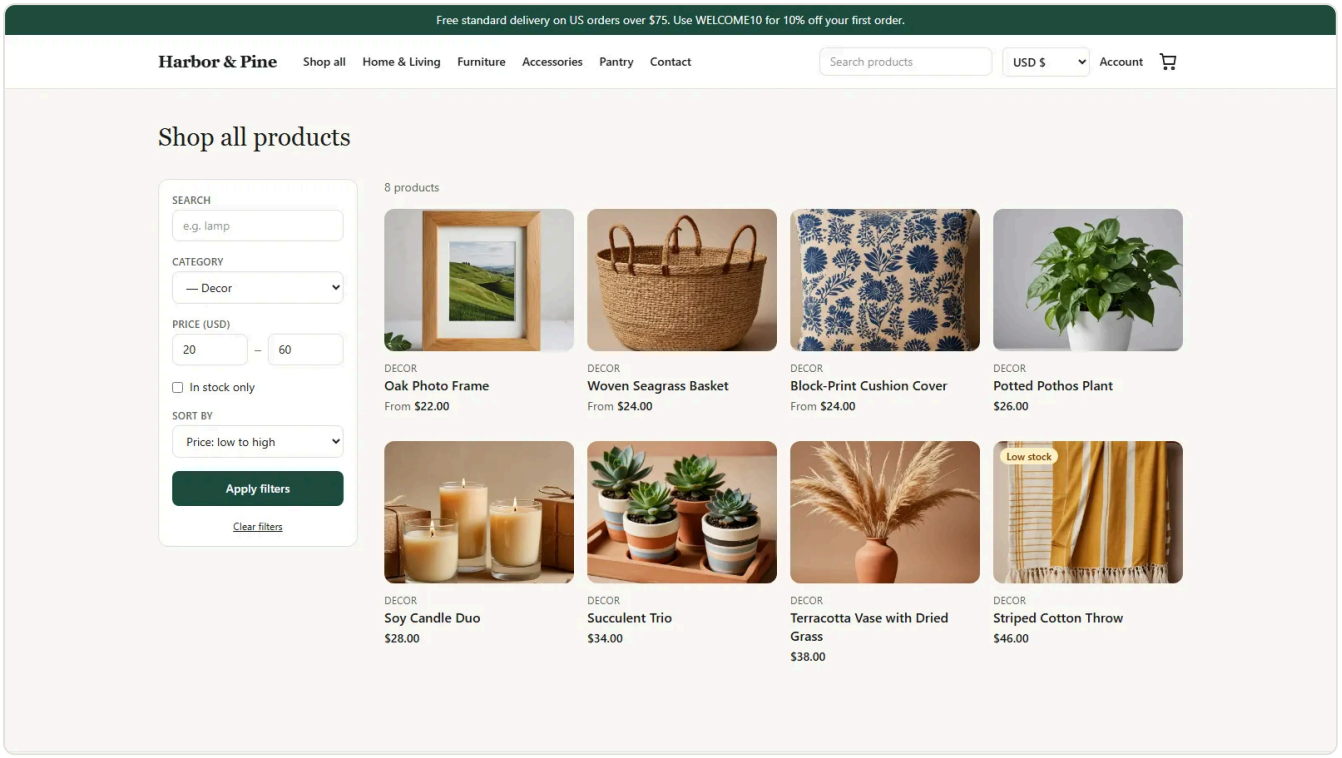
Editing a product: price and stock in the base currency, category, images and SEO fields.



The same product in the shop: the shopper picks a colour and sees that variant's price and stock.

5. Categories, search and filters

Categories can have one level of sub-categories (for example Home & Living → Decor). Each has its own image, description, SEO title and meta description. The shop page has search, a category filter, a price filter in the shopper's own currency, an in-stock filter and sorting.



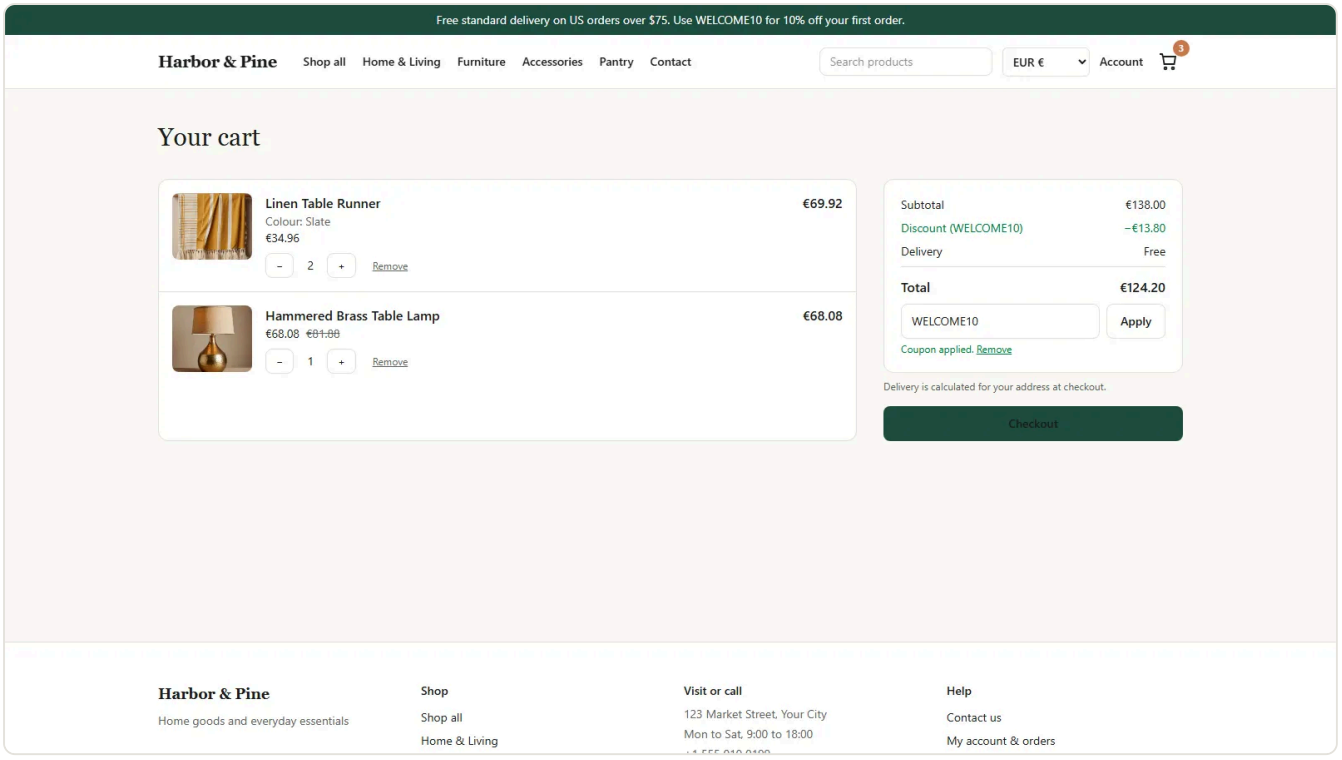
Shop page filtered by category and price, sorted by price.

6. Currencies

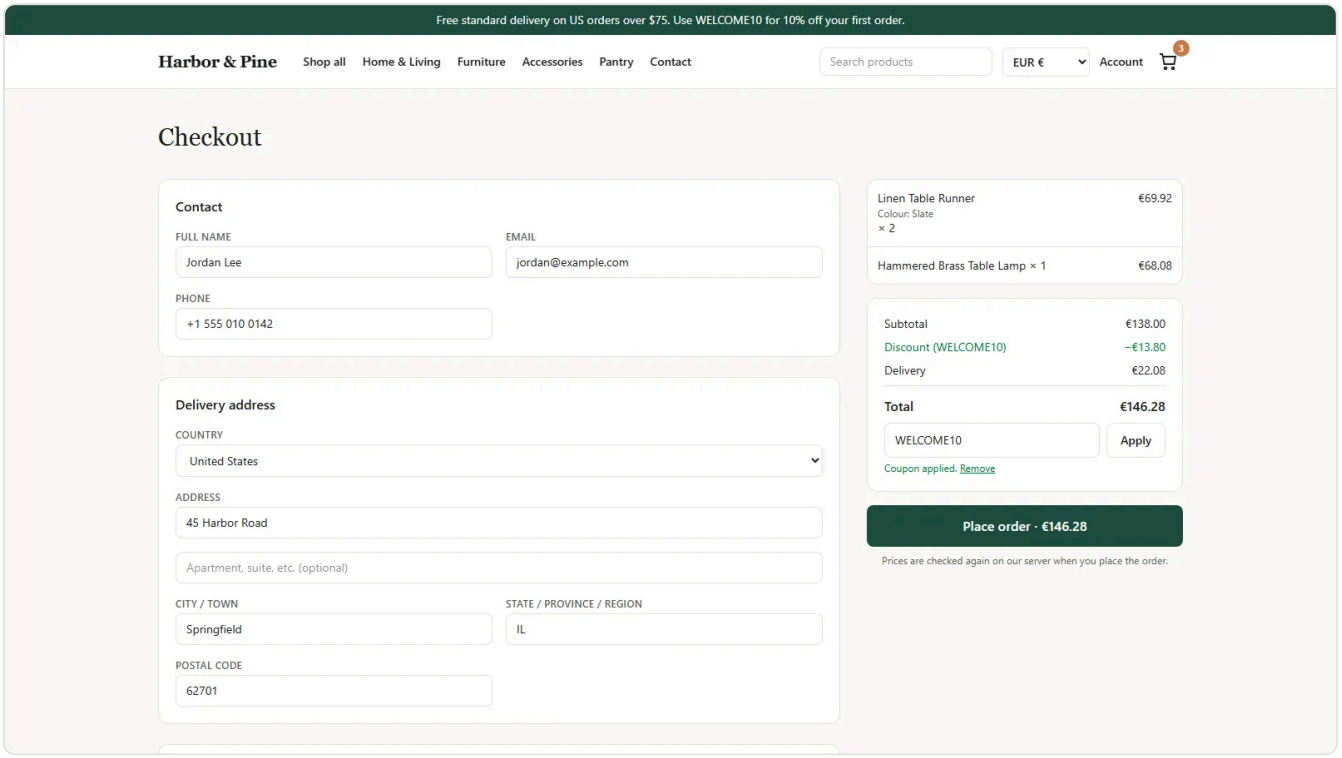
Admin → **Currencies**: choose the base currency (the one your prices are entered in), tick the currencies shoppers may choose, and set each one's rate (1 base = rate), symbol, decimals and symbol position. The shopper's choice is remembered, and each order stores the currency and the rate used. The optional Auto currency rates add-on updates rates daily; manual rates remain the fallback.

7. Coupons, cart and checkout

Admin → **Coupons**: percent or fixed amount, minimum spend, expiry date, usage limit, on/off. Every total (prices, discount, delivery, tax) is recalculated on the server when the order is placed, so a changed price in the browser has no effect, and stock is reserved in the same database transaction.



Cart in euros with the WELCOME10 coupon applied.



Checkout: contact, an address form for any country, delivery methods and payment methods.

8. Payments

Method	What the shopper sees
Cash on delivery	Pays when the order arrives.
Bank transfer	Your bank details (Admin → Payments) on the order page and in the email.
UPI	A QR code and your UPI ID (only shown once you enter a UPI ID).
Pay on pickup	Only offered when the shopper chooses local pickup.
Stripe / PayPal / Razorpay	Separate add-ons: install, enter your keys, and they appear here.

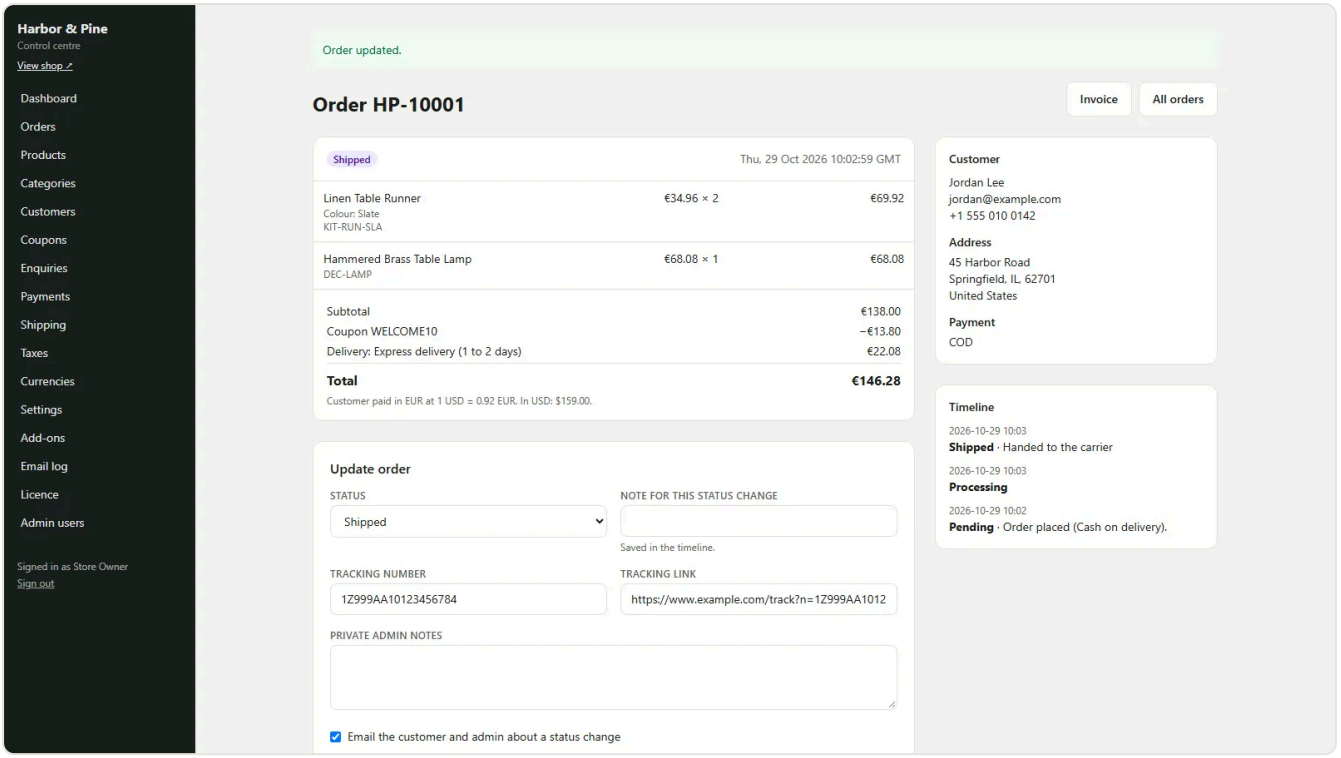
Tick or untick methods, rename them and set their order in **Admin** → **Payments**.

9. Shipping and taxes

Admin → **Shipping**: create zones by country code (`US` , `GB, IE, DE` …; `*` = every other country). Each zone has its own methods: flat rate, free shipping over an amount, a table rate by weight or order value (lines like `2:14.00` = up to 2 kg costs 14.00), and local pickup.

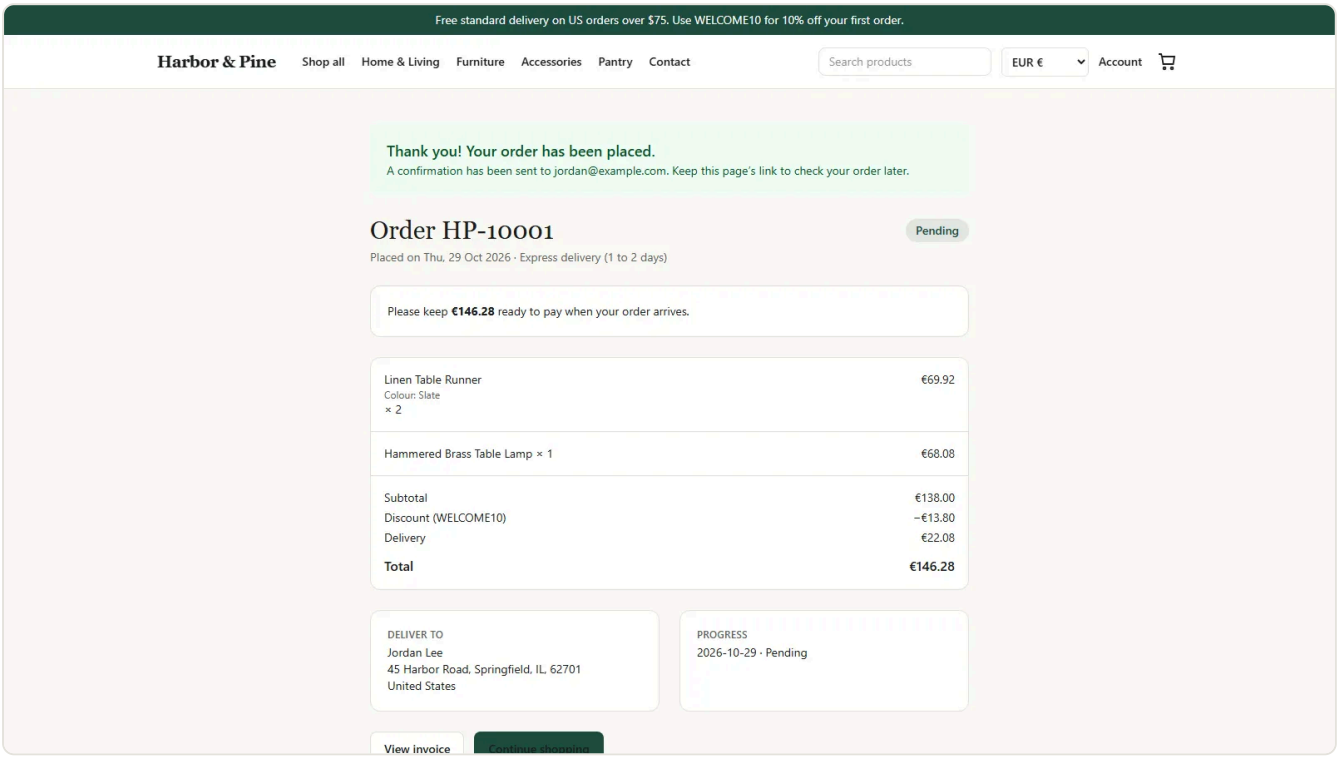
Admin → **Taxes**: one rate per country or per state/region (for example US / NY 8.875%), and whether your prices already include tax. Check the rules for your business with an accountant.

10. Orders



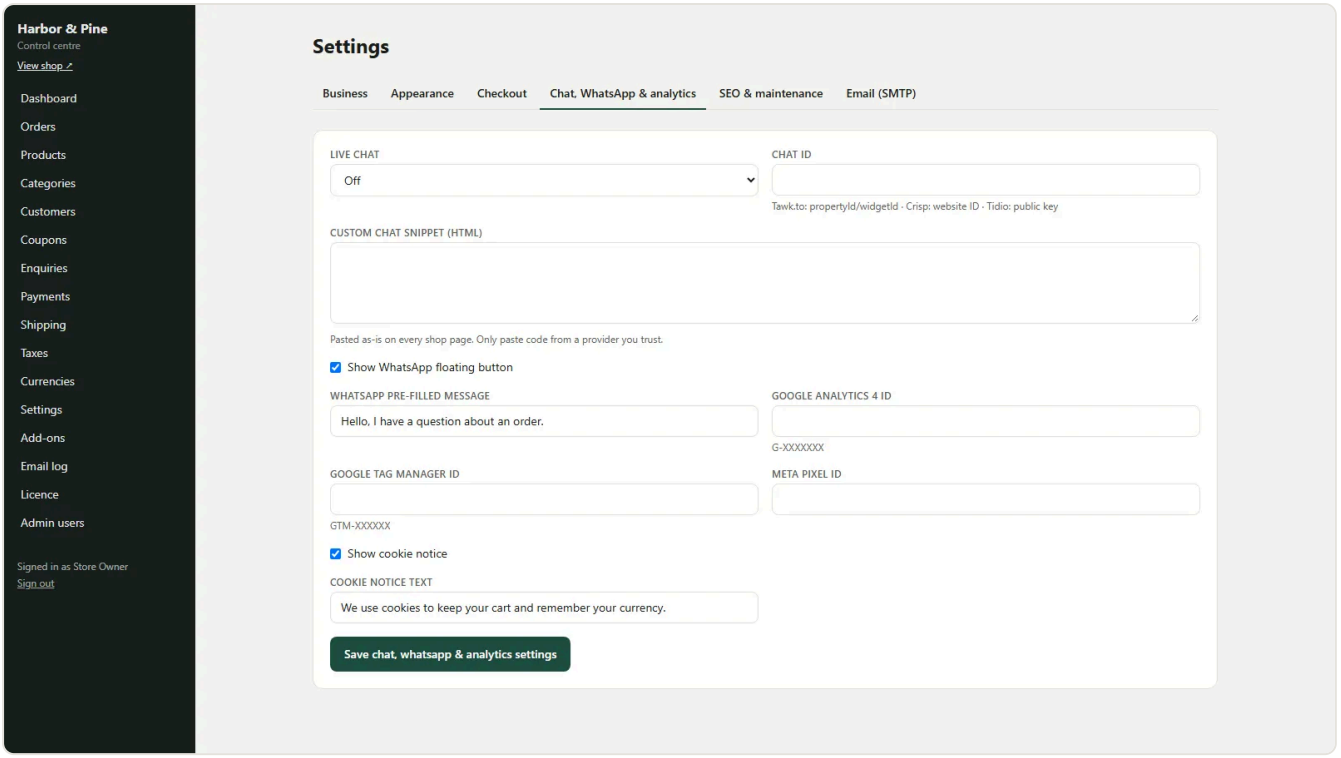
An order in the admin: items, totals in the shopper's currency, status, tracking, notes and timeline.

Statuses: pending → paid → processing → shipped → delivered, or cancelled / refunded (both put the items back into stock). Each change can email the customer and you. Add a tracking number and link, private notes, print the invoice, or export orders to CSV from the Orders list.



What the customer sees after ordering: status, payment instructions, items and a link to the invoice.

11. Settings: your business, chat, WhatsApp and analytics



Settings → Chat, WhatsApp & analytics.

- **Business:** shop name, logo, email, phone, WhatsApp number, address, Google Maps embed, hours, social links.
- **Appearance:** main and accent colours, home page headline/text/image, announcement bar.
- **Checkout:** order number prefix, default country, guest checkout on/off, pickup address.
- **Chat, WhatsApp & analytics:** Tawk.to, Crisp or Tidio live chat (or your own snippet), the WhatsApp button and message, GA4, Google Tag Manager, Meta Pixel and the cookie notice.
- **SEO & maintenance:** home page title and description; maintenance mode shows a holding page to everyone except signed-in admins.
- **Email (SMTP):** until this is filled in, emails are saved in **Admin** → **Email log** instead of being sent, so you can check them first.

12. Going live

1. Upload the project (without `node_modules`) or clone it on your server.
2. In `.env` set `SITE_URL=https://your-domain.com`, a long random `CRON_SECRET`, and `DATABASE_URL`. For PostgreSQL/MySQL also change `provider` in `prisma/schema.prisma`.
3. Run `npm install`, `npm run setup` (first time only), `npm run build` and `npm start` (your host's Node app manager can run `npm start` for you).
4. Open `/admin`, create your admin login, then **Admin** → **Licence**: paste your key and press *Save and activate*.

5. Optional: ask your host's cron to open `https://your-domain.com/api/cron?key=YOUR_CRON_SECRET` hourly.
6. Back up the database and the `storage/uploads` folder regularly.

> Until the licence is active on the live domain, checkout and the contact form are switched off and the admin shows a yellow banner. On localhost the shop always works. The key is re-checked about once a week; if coodecraft.com cannot be reached, an active shop keeps working for 30 days.

13. Add-ons

Add-ons are folders you copy into `addons/`, then run `npm run build`. They appear in **Admin → Add-ons** with their own settings and licence key. Available: Stripe, PayPal, Razorpay and Auto currency rates. Developers can write their own with the contract in `src/addons/sdk.ts` (payment, shipping, currency and marketing hooks).

14. Help

Installation questions: coodecraft.com/contact · Licence keys and downloads: coodecraft.com/account