

CDC Dashboard

User Guide

Chain Driven Cycles · Sligo

Updated 14 April 2026

CDC Dashboard — User Guide

Day-to-day reference for staff at Chain Driven Cycles using the dashboard at <https://cdc-dashboard.fly.dev>. The test environment at <https://cdc-dashboard-test.fly.dev> mirrors prod but runs against a separate database — safe for trying things out.

Last updated: 2026-04-14

Getting in

Open the dashboard URL in any modern browser and sign in with your username and password. On iOS or Android, use the browser's "Add to Home Screen" option to install it as a PWA — the repairs intake page (</repairs/new>) and used-bikes quick-add (</used-bikes/add>) are optimised for that. Staff accounts are managed from the Users page; module-level access is gated, so what you see in the sidebar is what you have permission to use.

The sidebar groups modules into **Workshop**, **Inventory**, **Finance**, plus **Settings** and **Users**. On mobile the sidebar collapses into a hamburger menu.

Dashboard (home)

Lands you on a summary view with KPIs, recent repair activity, cache stats and shortcuts into the main workflows. Nothing here is editable — it's read-only, pulled live.

Workshop

Repairs

The core workshop tool. Repair tickets move through a status kanban from booked-in to ready for collection.

Booking a bike in. Use </repairs/new> on a phone or tablet at the counter. Fill in the customer, phone number, bike, reported issue and priority. The form talks to Shopify in the background so customer and bike records get linked. Walk-in customers without a Shopify record get created automatically.

Working a ticket. Open a ticket from the repairs list to see tabs for **Messages, Photos, Parts, Timeline, Warranty** and **History**. Parts added here flow into the ordering workflow and the timeline logs every status change, note and SMS so there's an audit trail.

Notifying the customer. SMS is sent via Twilio when a ticket moves to `ready_for_collection`, and manual messages can be fired from the Messages tab. Irish numbers are normalised automatically.

Desktop vs mobile. The repairs list renders as a table on desktop and as cards on phones — both views are live.

Parts Ordering

Aggregates parts requested across open repair tickets and pushes them into per-supplier orders. Work from newest requirement downward; mark items ordered, then invoiced, then received. The module records the full lifecycle in `repair_events` so you can see who ordered what and when. When a PO is confirmed (see PO Processing), matching parts will eventually auto-advance the repair ticket to `ready_for_work` under Phase N — for now that step is manual.

Warranty

Track warranty claims against suppliers — create a claim, link the affected parts from a repair ticket, attach photos and follow it to reimbursement. Stale claims (no activity for a while) get surfaced with a warning.

Bike Specs

A catalogue of bike model specifications so mechanics can look up the right replacement part without chasing manuals. Search by brand and model, or add a new spec from the form. Structured spec fields and a fuller brand/model seed list are planned but not yet in.

Inventory

Products

A live view of the Shopify catalogue — search, filter, paginate through 500+ products. Use it to check stock, prices and status, or for bulk edits (price %, tag changes, status toggles). Data is cached locally; the Cache page shows freshness.

PO Processing

Upload a supplier invoice as PDF, Excel or CSV. The system auto-detects the supplier (10 supported: Cara Trade, JHI, Madison, Chicken CycleKit, Favero, Bottecchia, Bike Outlet and others), parses the lines, runs multi-pack splitting and price-change detection, and gives you a review screen. Confirm the lines and inventory syncs to Shopify. Live scraping (via ProductFetcher) fills gaps when a line isn't in the local cache.

Supplier login changes. Several scrapers keep cached sessions — flag it in Settings when you change a supplier password so the scraper can be reset before it silently fails.

Cara Stock

Cara Trade-specific stock view built on their feed. Useful for quick availability checks when a customer is on the counter.

Cache

Shows what's in the local product cache, when each entry was last refreshed, and stats on hit/miss rates. The buttons here let you force a refresh of the Shopify catalogue or clear stale entries. Rarely needed day-to-day, but the first place to look when a product page shows wrong prices.

Used Bikes

Staff listings for second-hand stock plus a public browse page and trade-in enquiries. `/used-bikes/add` is a quick-add PWA for adding a bike from the shop floor — photos, spec, price, and publish flags for Shopify, DoneDeal and Facebook Marketplace. Trade-in enquiries land here too.

Finance

B2W Invoicing

Pulls Bike to Work orders from Shopify so you can generate a Xero invoice to the employer in one go. Xero OAuth is still being wired in — connect it from Settings once available.

Settings

App configuration, Shopify and Xero connection status, PO defaults, and password change. Changes are stored in `config.json`; secrets live in `.env` on the server and aren't editable from the UI.

Users

Staff accounts and per-module permissions. Only signed-in users with the Users permission see this page.

Tips that matter

- **Cache-busting:** if a page looks stale right after a deploy, a hard refresh (Ctrl+Shift+R / Cmd+Shift+R) picks up the new JS/CSS version.
 - **Test vs prod:** the test URL is for experimenting. Anything you do there doesn't touch the live Shopify store.
 - **Mobile first:** the intake forms (repairs, used bikes) are designed to work one-handed at the workshop bench — install them as PWAs.
 - **Logs:** server-side logs are in `logs/app.log` if something looks wrong and you need to open a bug.
-

Known gaps / coming soon

- Real products page with bulk actions (B.1 — in progress, due next)
- B2W → Xero invoice automation (B.2)
- Xero OAuth connect flow in Settings (B.5)
- Phase J (Trade-in), Phase L (Strava service reminders), Phase N (PO→Repairs auto-advance), Phase P (Claims assessment)

For a full roadmap, see `ROADMAP.md` and `docs/PHASE_PROMPTS.md`.