

Overview & Integration Flow

XPOS ESDC exposes a local HTTP service that a Point of Sale (the built-in POS, or your own POS/ERP) uses to fiscalise sales. This guide is the integration reference for developers building against that service.

Base URL

The service listens on port **8888** on all interfaces of the machine running XPOS ESDC:

```
http://localhost:8888      (POS on the same PC)
http://<esdc-host>:8888    (POS elsewhere on the LAN)
```

Two API surfaces are available:

- **v3** — the current API, under `/api/v3`. Use this for new integrations.
- **Legacy V2** — `/api/attention`, `/api/status`, `/api/sign`, for compatibility with older POS integrations. See the "Legacy V2 API" page.

Interactive reference (Swagger)

The running service publishes a live, always-current OpenAPI document and an interactive UI:

```
http://localhost:8888/swagger/index.html
```

The Swagger schema is authoritative for exact field names and shapes — this guide explains the flow and the important fields; Swagger is the definitive contract for the build you are integrating with.

The integration flow

A POS follows this sequence:

1. **Check availability** — `GET /api/v3/attention`. Confirms the SDC service is up and a reader is present.
2. **Read status** — `GET /api/v3/status`. Tells you whether a PIN is required (`isPinRequired`), whether an audit is required, the SDC date/time, the device identity, and the active tax

rates.

3. **Verify PIN (once per session)** — if `isPinRequired` is true, `POST /api/v3/pin` with the operator's PIN. The SDC caches it in memory and signs subsequent invoices without it. The PIN is cleared on restart, so this step repeats after the app restarts.
4. **Sign invoices** — `POST /api/v3/invoices` for each sale. The response contains the verification URL, QR code, journal (receipt text) and signature.
5. **Print / store** — render the receipt from the response, or fetch the rendered PDF later from history.

Idempotency (RequestId)

On `POST /api/v3/invoices`, send a unique `RequestId` request header. If the same `RequestId` is retried (after a timeout or dropped connection), the SDC returns the **same** already-signed invoice instead of signing again — the header is echoed back on the response. Always send a fresh `RequestId` per sale and reuse it only when retrying that exact sale.

Content type & CORS

- Requests and responses are JSON (`Content-Type: application/json`), except `POST /api/v3/pin`, whose body is the PIN as plain text.
- CORS is open for `GET`, `POST` and `OPTIONS` from any origin, so a browser-based POS can call the service directly.
- If the POS runs on another machine, allow inbound TCP **8888** through the Windows firewall on the SDC host.

Revision #1

Created 2026-08-13 08:49:59 UTC by Shanil Verma

Updated 2026-08-13 08:49:59 UTC by Shanil Verma