

# IPTCompile - One-Page App Summary (Repo-Based)

---

## What it is

IPTCompile is a Windows PowerShell WinForms app that builds an IPT Excel report from Seal Test files and optional CSV input. It validates and aggregates data into a standard output workbook and can enrich report metadata from SharePoint.

## Who it is for

Primary persona: internal QC/PQC analysts working with LSP batches, Seal Test files, and IPT production reporting workflows.

## What it does

- Finds LSP folders and auto-lists matching CSV, Seal NEG/POS, and Worksheet files; manual browse is also supported.
- Builds reports from a fixed template (`output\_template-v4.xlsx`) using EPPlus.
- Optionally writes and validates signatures in Seal files (cell B47) across active sheets.
- Imports CSV data (with streaming for large files), extracts assay/stats, and sorts rows for downstream processing.
- Runs a compiled RuleBank-driven rule engine and can output a CSV summary debug sheet.
- Compares NEG/POS values, flags mismatches, and populates key tabs such as Information, Seal Test Info, STF Sum, and equipment/control material sections.
- Optionally connects to SharePoint via PnP.PowerShell to append production-order fields, then saves/open the report and writes audit/log records.

## How it works (architecture)

- Installer layer: `IPTCompile\_Install.bat` -> `Launcher/Install.bat` copies launcher files locally and creates a desktop shortcut.
- Launcher layer: `Run.ps1` self-updates launcher files, sets env vars/log root, then calls `Launcher.ps1`.
- Cache/run layer: `Launcher.ps1` mirrors app files from source to `C:\IPTCompile` (version + mutex guarded) and runs `Main.ps1`.
- App layer: `Main.ps1` (WinForms UI) orchestrates modules (`Config`, `DataHelpers`, `RuleEngine`, `SignatureHelpers`, `Logging`, `Splash`, `UiStyling`).
- Data flow: user input/files -> parsing + checks + rule engine + optional SharePoint lookup -> populated workbook -> temp `.xlsx` output + audit logs.

## How to run (minimal)

- On Windows, run `IPTCompile\_Install/IPTCompile\_Install.bat` once.
- Start the created desktop shortcut `IPTCompile (Local)` (it launches `Run.ps1` -> `Launcher.ps1` -> app).
- In the app, enter an LSP and click `Sok filer` (or select files manually).
- Select Seal NEG and POS (CSV optional), then click `Skapa rapport`; output is saved to `%TEMP%` and opened in Excel.
- Exact minimum supported Windows/PowerShell/Excel versions: Not found in repo.