# **Lean pre-coding sheet — 83/89**

**Component:** crates/vm\_app/src-tauri/icons/icon.png (app icon asset)  
 **Version/FormulaID:** VM-ENGINE v0

## **1) Goal & success**

**Goal:** Provide a **local, bundled** application icon for the desktop app; no external fetches.

**Success:** App packages and displays this icon on Win/macOS/Linux; build performs **no network** calls for assets.

## **2) Scope**

**In:** Binary image file at src-tauri/icons/icon.png referenced by tauri.conf.json bundle settings. (Asset is shipped with the app.)

**Out:** Report glyphs/visual icons used inside reports (Doc 7B §2.2) — those are drawn by the renderer, not this file.

## **3) Inputs → outputs**

**Inputs:** The PNG file itself.

**Outputs:** Included in the packaged desktop app (no runtime downloads).

## **4) Entities/Tables (minimal)**

## **5) Variables (only ones used here)**

N/A.

## **6) Functions (signatures only)**

N/A (static asset).

## **7) Algorithm outline (practical steps)**

Place icon.png at crates/vm\_app/src-tauri/icons/.

Ensure tauri.conf.json points to packaged assets only; **no external assets** are allowed.

Commit the binary (do **not** fetch at build or runtime).

## **8) State flow (very short)**

Build → Tauri bundles local assets (UI + icons) → App shows icon; runtime remains offline.

## **9) Determinism & numeric rules**

Not applicable; asset presence must not alter **Result/RunRecord** bytes. Canonical JSON rules remain unaffected.

## **10) Edge cases & failure policy**

Missing/incorrect path in tauri.conf.json ⇒ packaging error; fix by pointing to the local file. (No network fallback permitted.)

Do not embed remote fonts/styles via the icon path or metadata; **all** visuals are local.

## **11) Test checklist (must pass)**

App packages and displays the icon on all targets **without** any HTTP/DNS.

Removing icon.png makes packaging fail (expected); restoring it fixes the build.

Reports remain self-contained (no external assets).