# **Pre-Coding Essentials (Component: crates/vm\_report/src/render\_json.rs, Version/FormulaID: VM-ENGINE v0) — 64/89**

## **1) Goal & Success**

**Goal:** Serialize the ReportModel to **JSON** that mirrors Doc 7’s fixed **sections, fields, precision, and data sources**—no extra data.

**Success:** Output includes sections in **exact order** with **one-decimal** percentages; approval-denominator sentence appears when ballot type is **approval**.

## **2) Scope**

**In scope:** Convert ReportModel into a deterministic JSON structure (stable key order), preserving Doc 7 wording/fields and preformatted numerics.

**Out of scope:** HTML/CSS templates, assets, maps; any recomputation of gates/percentages (must already be in the model).

## **3) Inputs → Outputs (with schemas/IDs)**

**Input:** ReportModel built solely from **Result**, optional **FrontierMap**, and **RunRecord**.

**Output:** JSON object with sections **§1–§10** and **Fixed footer**, matching Doc 7 bindings (no extra fields).

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

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

None computed here. All VM-VAR values are **displayed** per bindings; approval ballots require the **approval-rate denominator** sentence.

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

rust

CopyEdit

pub fn render\_json(model: &ReportModel) -> String;

// helpers

fn to\_ordered\_json(model: &ReportModel) -> serde\_json::Value; // stable section order, stable key order

fn write\_footer\_ids(run: &RunRecordDb, result: &ResultDb, frontier: Option<&FrontierMapDb>) -> FooterJson;

*(Footer content strictly from RunRecord/Result IDs; optional FrontierMap ID.)*

## **7) Algorithm Outline (bullet steps)**

**Section order:** emit §1→§10 exactly as Doc 7 lists them.

**Ballot paragraph:** if ballot type is **approval**, include the mandatory **approval-rate denominator** sentence.

**Frontier section:** include only if a **FrontierMap** was produced; map statuses/diagnostics from FrontierMap and mirror per-unit flags.

**Sensitivity:** include 2×3 **±1pp/±5pp** table only if CompareScenarios exists; otherwise "N/A (not executed)".

**Integrity & footer:** list identifiers (FID, Engine, REG, PS, TLY label, RNG seed if used, Run UTC, Result ID, optional FrontierMap ID) and duplicate fixed footer line.

**Precision:** ensure all percentages/margins are **one decimal** (model should already be formatted; renderer must not round again).

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

Called by vm\_report::lib after ReportModel creation; **reads artifacts only** via the model; strictly offline.

## **9) Determinism & Numeric Rules**

**No double rounding**; keep one-decimal strings as-is.

**Stable key order** for map-like structures in JSON to avoid diff noise.

**No external assets** (JSON is pure data).

## **10) Edge Cases & Failure Policy**

**Validation failed**: emit sections and texts per Doc 7 fallbacks; **omit Frontier**; mark outcome **Invalid**.

**Gates failed**: render up to panel with ❌ flags; outcome **Invalid (gate failed: …)**; **omit Frontier**.

**Mediation/protected flags**: include diagnostics counts under Outcome; label is **Marginal** (already set upstream).

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

Sections appear in Doc 7 order; **approval sentence** present for approval ballots.

Frontier only when FR exists; diagnostics mirror FR + per-unit flags.

Sensitivity logic respected (table vs “N/A”).

Footer identifiers sourced **verbatim** from RunRecord/Result; values match fixed footer line.