🔐VERSION CONTROL & CHANGE AUTHORITY

Document: Master Index — Post-Freeze Addendums (v5)
Status: UNLOCKED (Living, Append-Only)
Current Version: v5
Effective Date: Immediately

Versioning Rules (Authoritative)

1. This document is maintained as a **single living addendum**.
2. The **version number MUST be incremented** whenever:
3. New governance is added
4. Existing governance is clarified or expanded
5. Any rule affecting validation, monetization, page contracts, or vertical behavior is introduced
6. Version increments apply to the **entire document**, not individual sections.
7. Older versions are superseded upon export and replacement in project files.

Lock / Unlock Semantics

- **LOCKED:**
  The document is authoritative, enforceable, and safe to execute against.

- **UNLOCKED:**
  The canvas may be amended, but changes do not take effect until:

- The version number is incremented

- The document is re-exported
- The project file is replaced

Enforcement Rule

Execution, validation, and repo changes MUST target **the highest exported version** of this document.

If the version referenced by an execution effort does not match the latest exported file, execution MUST pause.

Governance Principle

> There is **one addendum**, **one version**, and **one source of truth at a time**.

Master Index — Post-Freeze Addendums (Authoritative, Append-Only)

🔒 PREAMBLE INHERITANCE & GOVERNING AUTHORITY

This addendum **inherits and is governed by the full preamble of Master Index v2.1 (Frozen Canonical)**.

All principles defined in the Master Index v2.1 preamble — including but not limited to:

- content-before-code
- no summarization or compression
- no regeneration of approved text
- deterministic, auditable policy language
- strict separation between governance and execution

apply to this addendum **as if fully restated herein**.

In the event of any perceived ambiguity, the Master Index v2.1 preamble **controls and governs interpretation** of this addendum.

Governance Declaration

Canonical Artifact Completeness Rule (v1.0)

> **Canonical Artifact Completeness Rule (Hard Fail):** Any document marked *Canonical* must be a fully self-contained, literal artifact and contain **no placeholders, stubs, editor notes, shorthand, or references to previously approved content** (e.g., “unchanged,” “same as above,” “TBD”). Violation is a hard fail and blocks locking and conversion.

Machine-Only Marker & Accordion Convertibility Rule (v1.0)

> **Machine-Only Marker & Accordion Convertibility Rule (Hard Fail):** All canonical guide documents must contain valid machine-only markers for every guide and section, and must be directly convertible to JSON and HTML where each numbered section renders as a discrete

accordion panel; missing markers, malformed markers, or non-convertible section structures constitute a hard fail.

This canvas is a **governance document**, not a working chat.

It is governed by the same principles as the **Master Index v2.1 preamble**, including:

- Content-before-code
- No summarization or compression
- No regeneration of approved text
- Deterministic, auditable policy language
- Clear separation between governance and execution

This canvas is **policy-only**. No execution steps. No code. No examples.

Authoritative Canvas Creation Prompt (Embedded)

The following prompt is embedded verbatim as authoritative governance language. It defines the required initialization, custody, and enforcement rules for this canvas and any future re-creation of it. This text is preserved **exactly as written** and must not be summarized, rewritten, or paraphrased.

📌CANVAS CREATION PROMPT — MASTER INDEX POST-FREEZE ADDENDUMS (AUTHORITATIVE)

> Create a **canvas document** titled: **“Master Index — Post-Freeze Addendums (Authoritative, Append-Only)”**
>
> This canvas is a **governance document**, not a working chat.
>
> It is governed by the same principles as the **Master Index v2.1 preamble**, including:
>
> - Content-before-code
> - No summarization or compression
> - No regeneration of approved text
> - Deterministic, auditable policy language
> - Clear separation between governance and execution
>
> **Scope & Authority**
>
> - This canvas is the **sole durable source of truth** for all post-freeze addendums to Master Index v2.1
> - Nothing in this canvas overrides the frozen Master Index v2.1
> - This canvas supplements v2.1 only
> - This canvas is append-only unless explicitly instructed otherwise

Phase Control

- This canvas is **policy-only**
- No execution steps
- No file generation unless explicitly instructed
- No code
- No examples

Canon & Custody Rules

- All text added to this canvas is **authoritative policy language**
- Do not summarize, rewrite, paraphrase, or “clean up” any addendum text
- If text is pasted or referenced, it must be preserved verbatim

Initialization Instruction

- Initialize the canvas by loading the FULL TEXT of: **“Master Index v2.1 — Post-Freeze Addendum (Policy Memo: Guides, Verticals & Canon Control)”**
- Load the text **in full**, exactly as written

File Generation Constraint (MANDATORY)

- If asked to generate a PDF, DOCX, or other file from this canvas:

“Generate a file that contains the FULL text exactly as printed above — no summarizing, no shortening, no paraphrasing. Before you give me a link, confirm: (1) section headings match exactly (2) the file includes every paragraph (3) provide a total word count.”

**Addendum Management Rules

🦯MASTER INDEX ADDENDUM — PERSONAL INJURY (DISTRIBUTION VERTICAL & PHASE GOVERNANCE)

Status: UNLOCKED (Append-Only) **Effective:** Immediately **Applies to:** Personal Injury (PI) vertical only

---

I. STRATEGIC CLASSIFICATION (AUTHORITATIVE)

Personal Injury (PI) is governed as a Distribution Vertical.

PI is not evaluated or monetized under authority-based sponsorship models by default.

Reason:

- PI advertisers do not purchase authority or visibility
- PI advertisers purchase **control over intake and case flow**
- Therefore, PI monetization constitutes a **distribution contract**, not a content sponsorship

PI exists in the system as:

- Educational publisher content (Phase-1 compatible)
- With controlled intake and monetization enabled only under explicit phase governance

---

II. MONETIZATION MODE ENUM (CANONICAL LAW)

Every PI city MUST be classified under exactly one monetization mode.

The following enum is **authoritative**:

- **EDUCATION_ONLY**
- **DISTRIBUTION_SPONSORED_DIRECTORY**
- **FULL_SITE_SPONSORSHIP**

Rendering behavior, disclosures, validation, and CTA availability are governed by this enum.

If a city lacks an explicit monetization mode assignment, validation MUST fail.

III. PHASE GOVERNANCE (LOCKED)

Phase 1 — Foundation (Non-Monetized)

- All PI cities default to **EDUCATION_ONLY**
- Directory renders neutrally
- No sponsored placement
- No intake CTA
- No indication of availability

Purpose:

- Establish educational authority
- Validate guide performance
- Prepare infrastructure without monetization

Phase 2 — Distribution (Primary PI Monetization Phase)

Phase 2 enables **DISTRIBUTION_SPONSORED_DIRECTORY**.

Characteristics:

- Directory remains present and intact
- One firm may purchase **exclusive sponsored placement** within a city directory
- Sponsorship affects **presentation**, not directory existence

Critical Rule:

> Sponsored directory behavior is permitted **only** during Phase 2 or later.

Phase 2 is the **default monetization phase for PI**.

Phase 3 — Full-Site Sponsorship (Optional / Non-Primary)

Phase 3 enables **FULL_SITE_SPONSORSHIP**.

Characteristics:

- Entire PI site or vertical is sponsored by a single firm
- Global disclosures required
- Directory may remain but sponsorship scope is site-wide

This phase:

- May be offered (e.g., \$75K)
- Is not required for PI success
- Does not replace Phase-2 distribution as the core model

IV. SPONSORED DIRECTORY BEHAVIOR (PHASE-2 LOCKED)

When a city is classified as **DISTRIBUTION_SPONSORED_DIRECTORY**:

- The directory **MUST remain rendered**
- The directory **MUST NOT be removed, hidden, or replaced**

Sponsored Placement Rules

- One sponsored firm block appears **at the top of the directory**
- Sponsored block may include:
- Firm name
- Location relevance

- CTA (e.g., "Request a confidential consultation")

Directory Integrity Rules

- Non-sponsored directory entries remain visible below
- No rankings or ordering claims may be implied
- No comparative language is permitted

Sponsored placement modifies **emphasis**, not structure.

V. DISCLOSURE REQUIREMENTS (NON-NEGOTIABLE)

When any sponsored placement exists, an inline disclosure MUST appear immediately above the sponsored block.

Canonical Disclosure (Concise, Required):

> **Disclosure:** Paid placement. We partner with one licensed personal injury firm per market. No outcomes are guaranteed.

Rules:

- This disclosure is **monetization-specific** and does not replace or duplicate the global disclaimer or footer
- The disclosure MUST be proximate to the sponsored block
- The disclosure MUST appear only when a sponsored placement exists
- Omission, alteration, or relocation of the disclosure is a validation failure.

VI. VALIDATION RULES (POLICY-LEVEL)

Validation MUST fail if any of the following are true:

- A PI city lacks a monetization mode assignment

- A city marked **EDUCATION_ONLY** renders:

- Sponsored blocks

- CTAs

- Intake signaling

- A city marked **DISTRIBUTION_SPONSORED_DIRECTORY**:

- Omits the directory
- Omits required disclosure
- Renders more than one sponsored firm
- A city marked **FULL_SITE_SPONSORSHIP**:
- Omits global sponsorship disclosure

Monetization mode violations are **hard failures**, not warnings.

VII. GOVERNANCE GUARANTEE

This addendum ensures:

- Clean transition between unpaid and paid states
- Compliance-first monetization
- Clear value proposition for sponsoring firms
- Zero ambiguity for operators, VAs, or validators

PI monetization is governed as a **controlled distribution system**, not an advertising free-for-all.

END ADDENDUM

APPENDIX A — VA OPERATING SOP (AUTHORITATIVE)

Status: UNLOCKED (Append-Only) **Applies to:** All VAs, contractors, operators acting on this system **Governs:** Human execution layer only

I. PURPOSE

This appendix defines **what human operators (VAs) are permitted to do**, **what they must do**, and **when they must stop**.

It exists to ensure:

- deterministic execution
- zero silent drift
- clean escalation boundaries

- auditability of all human work

This SOP is **binding**.

II. AUTHORITY & BOUNDARIES

VA Authority Includes

A VA **may**:

- Create draft content **only when explicitly instructed**
- Move approved text between formats **without modification**
- Run validation or checklists
- Flag failures, gaps, or ambiguities
- Prepare artifacts for review

VA Authority Explicitly Excludes

A VA **may not**:

- Invent or modify policy
- Interpret intent beyond written instruction
- Rewrite “approved” or “canonical” text
- Skip required phases
- “Fix” validation failures silently
- Proceed when a hard failure is detected

III. REQUIRED ORDER OF OPERATIONS (NON-NEGOTIABLE)

All VA work must follow this order:

1. **Confirm Canonical Source**

2. Identify the governing Master Index version + addendum

3. If unclear → STOP

4. **Confirm Phase**

5. Authoring / QA / Conversion / Validation / Release

6. If phase is ambiguous → STOP

7. **Execute Assigned Task Only**

8. No adjacent cleanup

9. No anticipatory changes

10. **Run Required Checklist**

11. QA checklist

12. Validation checklist

13. Phase-specific checklist

14. **Escalate or Deliver**

15. Deliver only after PASS

16. Escalate immediately on FAIL

Deviation is prohibited.

IV. HARD STOP CONDITIONS (IMMEDIATE ESCALATION)

A VA must **stop work and escalate** if any of the following occur:

- Missing canonical source
- Conflicting instructions
- Validation failure
- Missing required artifact
- Ambiguous approval state
- Any request that violates Master Index governance

“No response” is preferable to an incorrect response.

V. DELIVERABLE STANDARDS

Any VA-delivered artifact must be:

- Complete
- Lossless relative to source
- Clearly labeled (draft / review / final)
- Traceable to a governing document

Partial or speculative delivery is invalid.

VI. GOVERNANCE GUARANTEE

This SOP ensures:

- VAs cannot accidentally change policy

- Errors surface early
- Accountability is explicit
- Scale does not introduce chaos

APPENDIX B — QA SIGN-OFF CHECKLIST (CANONICAL)

Status: UNLOCKED (Append-Only) **Applies to:** All guides, all verticals

I. PURPOSE

This checklist converts the **Canonical Guide QA Rubric** into a **binary sign-off artifact**.

A guide **cannot be locked** unless all items pass.

II. QA SIGN-OFF (PASS / FAIL)

Layer 0 — Structural Integrity

-

Layer 1 — Primary Question Alignment

-

Layer 2 — Decision Completeness

-

Layer 3 — Situational Specificity

-

Layer 4 — LLM Extractability

-

Layer 5 — Human Helpfulness

-

Layer 6 — Authority & Depth

-

III. SIGN-OFF RULE

- Any FAIL → Guide is **not canonical**
- Partial passes are invalid
- Word count does not override logic failure

APPENDIX C — VERTICAL LAUNCH SOP (AUTHORITATIVE)

Status: UNLOCKED (Append-Only) **Applies to:** All current and future verticals

I. PURPOSE

This SOP defines the **only permitted sequence** for launching or expanding a vertical.

Its goal is to make vertical launches **predictable, repeatable, and boring**.

II. REQUIRED PHASES (ORDERED)

Phase 1 — Canonical Authoring

Required:

- Single canonical authoring surface (Canvas MD)
- Immutable guide slugs
- Primary questions defined per guide

Prohibited:

- Conversion
- JSON generation
- Repo changes

Phase 2 — QA & Lock

Required:

- Full QA checklist pass (Appendix B)
- Explicit “LOCKED” declaration

- No unresolved failures

Prohibited:

- Partial locking
- Conditional approval

Phase 3 — Conversion Preparation

Required:

- Canonical source exported
- Section counts confirmed
- Word counts confirmed

Prohibited:

- Regeneration
- Interpretation
- Rewriting

Phase 4 — Mechanical Conversion

Required:

- Lossless format transformation
- No placeholders
- No stubs

Conversion ≠ Delivery.

Phase 5 — Validation

Required:

- Structural validation
- Page contract validation
- Regression checks

Any failure blocks release.

Phase 6 — Release

Only permitted after:

- All validation passes
- All artifacts verified
- Governance intact

III. HARD FAILURE CONDITIONS

A vertical launch must halt if:

- Any phase is skipped
- Canonical source is ambiguous
- Validation fails
- Governance conflicts exist

IV. GOVERNANCE GUARANTEE

This SOP ensures:

- No vertical launches prematurely
- No content drift
- No validator surprises
- No “how did this get live?” moments

END OF APPENDICES A–C

APPENDIX D — DEFINITIONS & CANONICAL TERMS (AUTHORITATIVE)

Status: UNLOCKED (Append-Only) **Applies to:** All governance, validation, and execution contexts

I. PURPOSE

This appendix defines canonical terms used throughout the Master Index v2.1 and all post-freeze addendums.

Its purpose is to eliminate interpretive drift, especially as work is delegated across humans, tools, and time.

II. CANONICAL DEFINITIONS

Canonical
Text, structure, or artifacts that are approved, locked, and serve as the sole source of truth. Canonical material may not be modified except through an explicit governance change.

Lock / Locked
A declared state indicating that content has passed all required QA and validation gates. Locked content may be referenced, converted, or validated, but not rewritten.

Append-Only
A governance rule permitting additions but prohibiting silent edits, rewrites, or deletions of previously approved material.

Phase
A governance-controlled stage defining what actions are permitted (e.g., authoring, monetization, conversion). Phases are sequential and non-overlapping.

Distribution Vertical
A vertical where monetization is based on controlled routing or intake behavior rather than authority, visibility, or impressions (e.g., Personal Injury).

Sponsored Directory
A directory that remains structurally intact while allowing one paid placement to modify emphasis or prominence, subject to disclosure and validation rules.

Validator
An automated or manual system that checks structural, semantic, and governance compliance. Validators enforce failure conditions but do not determine usefulness or quality.

QA (Quality Assurance)
A decision-layer evaluation determining whether content is complete, helpful, specific, and safe to lock. QA is required in addition to validation.

Conversion
A mechanical transformation of canonical content into another format. Conversion alone does not imply correctness or deliverability.

Delivery
Conversion plus a full sanity, syntax, and validation pass. Delivery may only occur after all checks pass.

III. INTERPRETATION RULE

If a term is used ambiguously or inconsistently in execution, the definition in this appendix prevails.

APPENDIX E — VALIDATOR & QA MAPPING (REFERENCE-ONLY, AUTHORITATIVE)

Status: UNLOCKED (Append-Only) **Applies to:** Validator design, QA execution, and regression prevention

I. PURPOSE

This appendix maps **QA intent** to **validator enforcement** so that automated checks and human review operate on the same mental model.

It is reference-only and does not redefine existing validator logic.

II. QA LAYER → VALIDATOR RESPONSIBILITY MAP

QA Layer	What It Ensures	Validator Role	Failure Type
Layer 0 — Structural Integrity	Required sections and markers exist	Enforce presence, order, markers	Hard Fail
Layer 1 — Primary Question	Guide answers its core question	Detect missing or unanswered title question	Hard Fail
Layer 2 — Decision Completeness	Decision logic is complete	Flag missing required decision blocks	Hard Fail
Layer 3 — Situational Specificity	Scenario-resolved guidance	Detect shell or generic patterns	Hard Fail
Layer 4 — LLM Extractability	Self-contained, quotable logic	Flag missing answer blocks	Hard Fail
Layer 5 — Human Helpfulness	Clarity and confidence	Human QA only	Lock Blocker
Layer 6 — Authority & Depth	Risk-appropriate depth	Human QA only	Lock Blocker

III. REGRESSION PROTECTION

Validators must additionally support:

- Page type contract enforcement

- Required zone detection (FAQ, directory, guides block)
- Ad slot resolution checks
- Golden-city regression checks

Loss of any previously required structure is a **catastrophic failure**.

IV. GOVERNANCE CLARIFICATION

Validators:

- Enforce structure and safety
- Prevent silent regressions

Validators do not:

- Judge usefulness
- Approve locking
- Replace human QA

END OF APPENDICES A–E