PEACOCK — THE STRIKE ORCHESTRATOR (V21.3 OMEGA) Requirements Specification

1. EXECUTIVE SUMMARY

- Executive Overview
  - Localhost-first, Matrix-themed React SPA that converts a static dashboard into a state-aware "Neural Journey."
  - Orchestrates single-shot or up to 500-file batch AI "strikes" via a six-state branching machine with sub-200 ms UI feedback.
- Key Guarantees
  - Local-first operation on MX Linux localhost; zero server-side dependencies required.
  - Crash-proof browser storage (IndexedDB + fallbacks) and deterministic state recovery on reload.
  - UI responsiveness target: click→visual change ≤200 ms; first paint goal ≤500 ms.
- Release & Scope
  - Version V21.3 OMEGA — localhost distribution; English-only baseline; Matrix visual motif.
  - Primary audience: researchers/operators running strikes locally; offline workflows prioritized.

2. SYSTEM CLASSIFICATION

- Type and Complexity
  - Type: Single Page Application (SPA) — High complexity (offline first, local storage, orchestration machine).
- Primary Constraints
  - Must run offline-first on MX Linux localhost with zero server-side dependencies.
  - Web-only stack: React SPA, Web Workers for parsing, IndexedDB for persistence.
  - Security constraint: API keys stored locally; network calls only over HTTPS when used.
- Stakeholders & Users
  - Primary users: operators executing MANUAL or BATCH strikes; secondary: auditors reviewing dossiers.
  - System administrators and developers responsible for local installs and crash recovery.
  - Non-technical users need simple HUD controls: Start Over, Settings, About and clear toast/modal feedback.

3. FUNCTIONAL REQUIREMENTS (MUST-HAVES)

- FR-01 → FR-05: Identity, HUD, Stage, Mode, Manual
  - FR-01 Identity Node — Auto-load one of 7 peacock*.png assets; render 128×128 avatar + "Support the Mission" donation link + GitHub repo link.
- FR-06 → FR-08: Wrap, Strike, Homing
  - FR-02 Command HUD — Start Over, Settings, About; Start Over clears IndexedDB vaults and returns to IDLE; Settings opens API-key modal; About opens static markdown overlay.
  - FR-03 Stage Button Bar — Buttons N, S, F, E, H trigger state transitions to MODE_SELECTION; SVG neural path grows from button to split-node; active button glows and line animates (≤300 ms).
- FR-03 → FR-05: Stage, Mode, Manual Console
  - FR-04 Mode Split Node — User selects MANUAL or BATCH; transition to CONSOLE_SETUP or FACTORY_FLOOR; show dual-console dropzone or batch ingest perimeter.
  - FR-05 Manual Console — File drop/browse or optional paste; validate MIME (txt, md, json, pdf, docx); compute SHA-256 hash; stream to nexusIn store; show immutable "Master Protocol" + editable "Phase Payload."

4. ADDITIONAL FUNCTIONAL REQUIREMENTS & NFRS

- FR-06 → FR-09: Wrap, Strike, Homing, Auto-Handshake
  - FR-06 Wrap Prompt — Merge Master + Payload; inject header "### NEXUS RESPONSE ###"; compress to Blob; open Strategic Review Editor in full-width; state = STRATEGIC_REVIEW.
  - FR-07 Strike Authorization — User clicks STRIKE; POST to API Gateway with retry/back-off; stream response to Live CLI flare; update card status and emit audio ping; output SUCCESS | FAILURE badge.
  - FR-08 Homing Retraction — On strike completion animate card scale 1→0 and translate back into parent button; button turns Matrix Green; state = AUTO-HANDSHAKE; neural line auto-extends.
- FR-08 → FR-11: Homing, Auto-Handshake, Batch Ingest, Dossier Export
  - FR-09 Auto-Handshake — Copy output payload to next stage's input store and auto-focus the next button; show toast "Handed off to SPARK."
  - FR-10 Batch Ingest — Accept 1–500 files dragged; throttle to 25 concurrent uploads; queue remainder; render sliding window and per-file progress cards (QUEUED → STRIKING → SUCCESS).
- FR-10 → FR-12: Batch, Export, Archive

5. NON-FUNCTIONAL REQUIREMENTS & DATA

- NFR-01 → NFR-02: Performance & Security
  - NFR-01 Performance — First paint ≤500 ms on localhost; UI feedback (click→visual change) ≤200 ms; batch throughput ≥5 files/s on 4-core, 8 GB machine.
  - NFR-02 Security — API keys stored only in localStorage; network calls only over HTTPS; all file parsing in Web Workers; no eval(); confirm header "### SPARK RESPONSE ###" before clipboard write.
- NFR-03 → NFR-04: Reliability & Visual Fidelity
  - NFR-03 Reliability — Graceful degradation if IndexedDB unavailable (fallback to RAM + sessionStorage); retry failed API calls with exponential back-off (max 3); crash recovery restores last known state machine state.
  - NFR-04 Visual Fidelity — CRT scanline overlay at 60% opacity; 8 px text glow; strict color accuracy ΔE < 3 vs spec.
- Data Requirements
  - Entities — User, Stage, File, Strike, Log, API_Key, Preference.
  - Persistence — All entities except API_Key live in IndexedDB; survive refresh and browser restart; API_Key handled separately and not exported.

6. CRITICAL USER FLOWS, GAP ANALYSIS & ASSUMPTIONS

- Critical User Flows
  - Flow A: Manual Strike — User clicks NEXUS → selects MANUAL → drops file → clicks WRAP → edits → clicks STRIKE → watches CLI → sees Homing Retraction → auto-advances to SPARK.
  - Flow B: Batch Factory — User drags 500 files → system throttles to 25 concurrent → per-file cards update → audio pings on completion → [DOWNLOAD .ZIP] dossier appears for user save.
- Gap Analysis & Assumptions
  - Missing Info — Exact API rate limits, max payload size, and precise retry policy are not specified and require clarification.
  - Assumptions — Groq/Google/Mistral/DeepSeek caps approximated at 8 MB payload and 60 req/min; retry stops after 3 attempts; target accessibility WCAG 2.1 AA and English-only for V21.3.
  - SPARK analysis complete. READY FOR FALCON ARCHITECTURE handoff.