Javari Backend Dashboard — Living Spec (v0.2)

This document is the living specification for the Javari Backend Dashboard. It breaks down the wishlist into modules, defines MVP endpoints and data flows, and orders them by dependency. Items are marked as ACTIVE (implemented), PLACEHOLDER (scaffold only), or BLOCKED (future).

# Core Module Groups

|  |  |
| --- | --- |
| Module | Scope |
| System Health & Operations | Disk/Memory/CPU/Docker/UFW/Nginx, health checks, logs, runbooks with double-click safety, certificate tracker. |
| Learning & Knowledge Intake | Learning dashboard, hourly new material intake, source & relevance logging, document upload portal. |
| CRM & Vendor Management | Customer info, subscriptions, visits, password resets, vendor accounts with contacts & AI integration. |
| Internal AI & Avatar World | Interface to talk with Javari, Kairo, crawlers, and bots; dispatch crawlers with natural language tasks. |
| Project Management | Live/closed project tracker, categories, docs/logs, who worked on what, portal for new projects. |
| To‑Do & Scheduling | Tasks (daily/weekly/monthly), calendar integration, auto-rotation reminders (e.g. passwords). |
| Feeds & Comms | RSS/feeds intake, social/content bots, newsletter campaigns with monetization. |
| Payments & Monetization | Stripe/PayPal dashboards, KPIs, reset/reconnect scripts. |

# Build Order / Dependencies

1. Auth + RBAC (users, roles, audit trail) — foundation.  
2. System Health + Logging (health endpoints, /stack-health).  
3. Certs tracker + Secrets vault integration.  
4. Knowledge Intake + Learning timeline.  
5. CRM + Vendor connectors.  
6. Internal AI/Avatars + crawler dispatch.  
7. Project Mgmt + To-Do + Scheduler.  
8. Feeds/Comms + Campaign bots.  
9. Payments & Monetization dashboards.

# Visual Dashboard Concept

The backend dashboard will use a \*\*two-color placeholder scheme\*\*:  
- PLACEHOLDER modules: Gray  
- ACTIVE/live modules: Blue  
  
Layout:  
- Left sidebar: Navigation (System, Learning, CRM, Vendors, Avatars, Projects, To‑Do, Feeds, Payments, Admin).  
- Top bar: User/session info, search, kill-switch (with confirmation).  
- Main panels: Each module opens as a grid of cards.  
 - Each card = sub-tool (e.g., /health, /metrics, /logs).  
 - Cards show status color (blue=live, gray=placeholder).  
 - Cards with script buttons require double‑click confirmation.

# Next Steps

We will now proceed module by module, starting with System Health & Operations. Each module will:  
- Define DB schema (Postgres).  
- Define API endpoints (FastAPI).  
- Expose minimal frontend card (blue if live, gray if placeholder).  
- Tie into logging + audit trail.  
  
Snapshots (`snapshot-stack.sh`) and rollbacks (`rollback-stack.sh`) ensure safety.