ARCHITEKTUR-NOTIZ · VERTRAULICH

StudyWithMe

Referenz-Implementierung souveräner, auditierbarer KI — am eigenen Produktivsystem.

LEON KÖLLERWIRTH HLIHEL
AGENTIC AI · SECURITY · ARCHITECTURE
STAND · JUNI 2026

> **Ein Modell ist der Anfang.** StudyWithMe ist ein curriculum-geführter KI-Studiencoach für Universitäten: ein Schwarm zusammenarbeitender Agenten koppelt den Studienplan dynamisch an die geführte Lernstrecke jedes Studierenden, bindet aktuelle Forschung ein und erzeugt didaktische Reasoning-Pfade — Medizin als Pilot. Aufgebaut wie regulierte KI: lokal, isoliert, belegbar. Die Pipeline unten läuft bereits in Produktion und beweist genau die Prinzipien, die ich verkaufe — Souveränität, Isolation, Evidenz und Audit-Trail.

DIE AGENTISCHE PIPELINE

01 **Gap-Detection** Deterministischer Scan: was fehlt, wird aus den Dateien abgeleitet — nie aus einem Snapshot.	02 **Lokale Übersetzung** aya-expanse:8b über Ollama, Temperatur 0, SQLite-Checkpoint. 0 € pro Aufruf, on-device.	03 **Citation-Gate** Jedes kyrillische Token muss verbatim überleben — sonst wird die Übersetzung verworfen.
04 **Idempotenter Apply** Chirurgischer JSON-Patch mit JSON.parse-Guard — die Datei wird nie ungültig geschrieben.	05 **Eskalation** Nur harte Fälle gehen an ein gehostetes Modell (Haiku) — und durch dasselbe Gate.	06 **Lint-Gates + Ledger** Maschinelle Gates und ein append-only Ledger pro Welle; Commit erst, wenn alles grün ist.

PRINZIP → BELEG

Souveränität Übersetzung läuft lokal (aya-expanse:8b über Ollama). Bulgarische Inhalte verlassen die Maschine nicht — 0 € Inferenz statt Cloud-Tokens.	**Isolation** API, App und Daten getrennt (Docker); der API-Container kann das Content-Verzeichnis bewusst nicht lesen. Secrets via gitleaks in der CI.
Evidenz Jeder KI-Aufruf wird geloggt (Modell, Tokens, Latenz). Wissensgraph-Kanten tragen Quelle, Confidence und Soft-Delete — KI-Kanten sind low-trust.	**Audit-Trail** KI-Funktionen einzeln schaltbar, standardmäßig aus; jede Admin-Aktion und jeder Tier-Wechsel landet unveränderlich (DB-Trigger) im Audit-Log.

STACK

Frontend: Vite · React 19 · TypeScript · Tailwind v4. **Backend:** PHP 8 (eigener Router + Middleware-Pipeline) · MySQL 8 (81 Tabellen, nummerierte idempotente Migrationen). **KI:** lokal Ollama (aya-expanse:8b, 0 €) für Content; gehostete Modelle (gpt-4o-mini / Whisper) laufzeit-gegated, standardmäßig aus. **Qualität:** Ratchet-Gates (Token-/LOC-/Typecheck-Budgets), gitleaks, ~2.400 Vitest- + ~500 PHPUnit-Tests, Pre-Commit-Hooks, CI.

Leon Köllerwirth Hlihel · Bielefeld (DE) · **leonkoellerwirth.de**
info@leonkoellerwirth.de · +49 152 3367 2982

Eigenes Vorhaben in Entwicklung — keine fertige Kundenreferenz. Alle Angaben am eigenen Produktivsystem belegt.