AGENTIC ENGINEERING

Brand rules the AI can't break

Six agent-skills auto-enforce the locked palette, voice, page rhythm, FAQ rules, and conversion psychology on every change, across Claude Code, Cursor, Codex, and Gemini.

Cognilium · agentic engineering

DISCIPLINE	Agentic engineering
STACK	Claude Code Agent Skills • file-pattern activation • runtime rule enforcement • multi-tool (Cursor / Codex CLI / Gemini CLI / Antigravity) • persistent auto-memory

MY ROLE

My work across the Paralegent and Cognilium builds: I authored the agent-skills and the auto-memory architecture that keep every AI agent on-brand.

0	6	5
BRAND DRIFT SHIPPED	RUNTIME SKILL CONTRACTS	AI TOOLS ENFORCED ACROSS

THE PROBLEM

When you build at speed with AI, the AI drifts: the wrong shade of blue, off-brand wording, a thin page with no schema. Documentation does not stop it, because the agent does not reliably read the docs.

WHAT I DID

- Authored 6 agent-skills that activate on file patterns and enforce the rules at edit time: design-system (locked palette), voice and vocabulary, page rhythm, conversion psychology, FAQ builder, and visual-design bans (no gradient-mesh, no glassmorphism).
- Made them runtime contracts, not docs: the rules fire on every relevant change instead of waiting to be read.
- Made them portable across agents, so the same rules hold whether the work runs through Claude Code, Cursor, Codex CLI, Gemini CLI, or Antigravity.
- Paired them with a curated persistent auto-memory (role, scope, brand independence, vocabulary drift) so context survives across sessions and projects.

THE RESULT

Brand and quality rules that enforce themselves on every change, so building fast stopped meaning drifting off-brand. The standards are encoded, not hoped for.

THE JUDGMENT CALL · WHAT THE AI COULDN'T DO

Everyone writes a brand guide and hopes the AI follows it. It doesn't. I encoded the rules as skills that fire on every edit, so the agent can't ship the wrong blue or a schema-less page even when it tries. Making the standard a runtime contract instead of a document is the difference between fast-and-sloppy and fast-and-right.

PROOF

Note: 6 agent-skills + auto-memory architecture (private repos).

On request: Walk-through of the skills and how they fire.