MINZHE AI LAB

敏哲AI编程专家

使AI从「会写代码」到「受控地写高质量代码」

六层规范 · 29项工具·9大引擎 · 本地双Hooks · 三线自进化

敏哲AI实验室 · 数智化落地场景

AI 编程全过程保障专家

内容导览

从市场格局到核心优势，从商业模式到落地实践，完整呈现敏哲AI编程专家价值

01 **时代红利**
为什么 AI 编码是必然趋势

02 **失控风险**
质量 / 安全 / 过程 / 成本四大痛点

03 **产品缘起**
从AI「会写代码」到「受控地写代码」

04 **市场格局**
国内外 AI 编程专家包竞争态势

05 **差异化优势**
敏哲AI专家包 vs 第三方方案

06 **产品总览**
面向数智化的 Agent 型编程专家

07 **能力全景**
安全加固 · 上架合规 · 扩展体系

08 **核心能力**
六层跨项目通用规范，29项工具/9大引擎

09 **平台亮点**
确定性判据体系，双 Hooks

10 **四维价值**
质量 · 安全 · 过程 · 成本

11 **版本布局**
免费试用 · 个人订阅 · 企业私有化

12 **结束语**
永远的可靠的助手

时代红利

为什么 AI 编码已成为不可逆转的开发范式

效率跃升
×倍速交付

AI 辅助编码可提升整体开发效率数倍至数十倍，
生产力革命已至，但质量与安全成为新门槛

速度

代码生成与修改速度提升数倍到数十倍，缩短需求到交付的周期

广度

跨语言、跨栈、跨框架的宽泛能力，降低技术栈切换成本

解放

将工程师从重复性样板代码中解放，聚焦架构与业务

普惠

降低编码门槛，让业务与技术更贴近，赋能一线业务团队

失控风险

AI 编码红利之下，质量 / 安全 / 过程 / 成本四大痛点亟待解决

质量不确定

AI 生成的代码看似可用，却暗藏逻辑缺陷、边界遗漏与代码隐患

缺陷密度难控

安全漏洞

密钥硬编码、SQL注入、依赖漏洞、注入攻击等风险频发

安全红线失守

过程不可控

需求漂移、决策无记录、角色越权、流程跳步，黑盒开发

白盒审计缺失

成本与版本

改前不留底、版本漂移、回退失效，返工与数据丢失成本高

回溯能力薄弱

一个普遍困境：AI 越来越会「写」，却越来越难被「信」——谁来为这一切兜底?

产品缘起

从AI「会写代码」到「受控地写代码」

1

问题

AI 生成的代码质量良莠不齐、风险频现

2

思考

能否为 AI 编码注入「确定性约束」

3

答案

敏哲AI编程专家——受控地写高质量代码

核心定位

不是提供「通用的代码补全」，而是为AI辅助开发场景提供「结构化规范指导 + 确定性审查 + 自动化闭环管理」的 AI 编程专家，确保编码产出在企业级质量、安全、可维护性与可追溯性上达标。

使命一句话：让 AI 辅助开发「受控、可审计、可追溯、可进化」

市场格局

国内外开放可扩展的编程专家包 / 规则体系（2026）

Agent 编程进入「规则化」时代：专家包 / Rules / Skill 生态快速爆发，AI 编码助手正由「通用工具」走向「受控专家包」

核心差异：能否把「规范」显式编码为可执行判据，实现可控、可审计、可进化

国际主流专家包 / 体系

Sub-agents / 子代理

Claude Code 等将任务拆解为子代理协作，能力可组合

Rules / 规则文件

Cursor rules、Claude CLAUDE.md，项目级规则注入

Custom Instructions

Copilot 自定义指令，团队级规范模板沉淀

MCP 专家包生态

Agent 通过 MCP 接入企业数据与工具，受控调用

国内主流专家包 / 工程体系

- **工程规范包** — 各厂商沉淀编码规范 / 代码审查规则包
- **私有化知识包** — 企业私有化部署场景下的专属规则沉淀
- **Skill / 技能市场** — 平台级专家包市场，开箱即用的垂类能力
- **行业合规包** — 面向信创 / 等保等行业级合规约束包
- **提示词工程包** — 提示词模板库驱动的受控生成方案

注：专家包 / 规则 / Skill 生态为各厂商公开披露能力，形态持续演进，敏哲AI编程专家以工程化系统化体系化差异化为切入点

差异化优势

不是零散的 规范 / 提示词，而是企业级受控、可审计、可进化的工程化专家包

第三方专家包侧重「配置即用」：单点规范 / 提示词 / 技能；敏哲AI专家包侧重「工程化系统」：结构化规范 × 确定性判据 × 闭环治理

维度	第三方 AI 编程专家包	敏哲AI专家包
形态	单一规范 / 提示词 / 技能	六层规范 + 29工具•9引擎 + 双Hooks 系统化
判据	靠模型自由发挥，结果随机难复现	六层确定性判据：零幻觉、可回归、可验证
治理	配置零散，缺流程与权限约束	规范/审查/知识/待办备份全程白盒可审计
闭环	静态规则，落地即固化的知识	待办六态闭环 + 三色备份 + 三线自进化
落地	通用导向，企业合规与私有化难	本地/私有化可用，面向合规与信创场景

敏哲AI编程专家 的核心差异：把「专家规则」从静态配置升级为「受控、可审计、可追溯、可进化的工程化体系」

产品总览

一个面向数智化的 Agent 型编程专家



六层规范

跨项目确定性判据体系

六步闭环

分析→方案→执行→验证→交付→复盘

29项工具·9大引擎

29工具统一入口 · 即开即用

双Hooks

待办闭环 + 备份三色确定性兜底

三线自进化

质量+规范+项目三线并行演进

可审计可追溯

全程白盒审计 · 决策可追溯

能力全景

从安全到合规，从核心引擎到扩展体系，已为生产与交付做好充分准备

安全加固

密钥轮换 · 鉴权审计 · 敏感信息扫描

生产级安全基线

上架合规

hooks 双包拆分 · 元数据合规

开放平台规范对齐完成

一致性治理

五副本防漂移 · git 门禁

md5 逐文件复验

配置热刷新

后台可视化调整额度/限流/判据阈值

无需重启，约 3 秒生效

CMHub 扩展

快照 / 回滚 / 基线 / 审计

24 个 MCP 配置管理工具

自进化机制

质量 / 规范 / 项目三线演进

内部版 / 企业版专属能力

当前能力底座：707 专项用例 • 30+ 运行时 Hooks • 29 MCP 工具 • 13+ 安全护栏 • 6 大数据文件一致性校验

四维价值

质量 · 安全 · 过程 · 成本 四维并重，可度量、可验证

01 质量

确定性规范判据 + 工程审查，降低缺陷密度，代码可维护、可回归

707 用例 / 100% 通过

02 安全

密钥/SQL注入/依赖/注入攻击四重防线 + 安全纵深规范

30+ 运行时 Hooks + 多防线

03 过程

待办闭环六态 + 白盒化三件套，开发全过程可审计可追溯

todo/backup 双 Hooks

04 成本

改前留底 + 三色触发 + 版本管理，减少返工与数据丢失

backup_tricolor 绿灯合规

核心能力 · 六层跨项目通用规范

确定性判据体系，为 AI 编码划定「硬边界」

H 系

Harness 开发全流程

H-01~H-12：实践→需求→方案→评审→复盘→缺陷→迭代→产品化→多轮交付→环境→质量→时间成本

L•E•D 系

LLM 内容生产判据

L-01~09 / E-01~05 / D-01~04：法学·工程学·数据学判据

软件工程质量

工程保障

性能铁律 / 安全鉴权密钥 / 测试验收 / 数据判据 / 合并准入 / 三问边界

安全纵深

安全编码

威胁建模 / 密钥检测 / SQL注入 / 依赖扫描 / 注入变异 / 自检清单 / 幽灵引用

待办闭环

过程闭环

进场回顾 / 单一事实源 / 六态生命周期 / 收尾归零 / 周期体检

版本备份

数据安全

改前留底 / 分层回退 / 三色触发 / 数据库强制备份 / 配置密钥分离

核心能力 · 29项工具/9大引擎

确定性判据29 项工具，个人版开箱即用，企业版能力服务端化，一个统一入口 + 一把密钥

29个审查工具归并为9大引擎

个人版开箱即用，企业版：1 个统一入口 · 1 把钥匙（MCP_MASTER_KEY） · 29 个 MCP 工具全通

规范决策

get_specs 系列

工程审查

review_code 系列

知识库检索

search_kb 系列

待办备份

todo/backup

安全扫描

secret/security

写前门禁

lint/precommit

一致性校验

consistency/contract

跨文件审查

cross_file/debug

交付验证

delivery/commitment

平台亮点 · 确定性判据体系

零 LLM 参与、零幻觉、零延迟、可回归 —— 判据逻辑全部可确定性验证

确定性优先

「判据优先于模型」

- 六层规范以「确定性判据」而非「模型直觉」驱动
- 每次检查输出可复现，不依赖模型随机性
- 零 LLM 参与 → 零幻觉、零延迟、成本可控
- 全流程可回归：同一输入必得同一结果
- 自动化校验、可单元测试、可自动化审计

707+ 累计回归用例全部通过

30+ 运行时代码 Hooks（安全/密钥/注入/门禁）

20+18 待办/备份判据用例 + Hook 专项全部通过

100% L4 集成 / L2 Hook 功能测试通过率

平台亮点 · 双 Hooks

待办闭环 + 备份三色触发，确定性兜底不遗漏

待办闭环六态

待筛选 → 已登记 → 已确认 → 进行中 → 已闭环 → 已废弃

todo_sync_hook：进场回顾检查未闭合待办 + 收尾归零兜底，单一事实源（TODO.md / MEMORY.md）

备份三色触发（backup_tricolor_hook）

绿灯

合规

快照新鲜，正常执行

黄灯

首次违规

提示补快照，允许继续

红灯

二次违规

阻断破坏性操作 + 审计留痕

确定性执行 · 零依赖（仅标准库）· 非阻塞 · 副本对齐防漂移

可信审查 · 防假绿体系

从「测试通过」到「判据真跑通」，全链路防伪、防假绿、防漂移

防假绿门禁

G1-G8 八道闸 · 8 金丝雀反向验证

篡改即翻红，拒绝假绿

覆盖清单

运行覆盖清单 · 声明≠执行识别

未决项强制拦截

防空转引擎

连击熔断 · 冻结拒注册

工具真执行不空转

模型质量签核

工程等价 · 模型质量分离

过期签核自动回退

代码审查

Open Code Review 批判者校验

无证据四态降级可追溯

可信判定闭环

哈希锁定 · 反向注入验证

每次结论有据可依

当前可信底座：G1-G8 全绿 • 8 金丝雀 • 707 专项用例全通过 • 双写一致 • 反向翻红验证

过程规范 · 全周期模板体系

敏哲AI

从策划到验收全过程标准化，模板即规范、规范即门禁

模板资产

19 → 34 篇 · 全生命周期覆盖

策划/需求/设计/测试/验收

契约标识

C-ID/C-STATE/C-EVID/C-REVIEW

四要素单一真源

验收门禁

V1-V10 判据 · --gate 强校验

仅认 status=pass

真源+3副本

一处定义 · 逐字节一致

防漂移可审计

自检机制

--self-test 内置 · 反向注入翻红

门禁自身可信

过程闭环

策划→设计→测试→验收全链路

模板即规范可执行

过程底座：34 篇模板 • C 契约四要素 • V1-V10 门禁 • 真源+3副本逐字节一致可追溯

版本布局

同一核心、三种封装，按部署位置与数据主权选择

同一套 29 工具纯函数逻辑，三种封装形态，适配不同部署位置与数据主权需求

内部版连自有引擎 · 个人版即装即用 · 企业版代码不出本地

内部版

自有引擎

内部使用 · 团队协作

v3.0.0 • 29 工具全量 • 远程 MCP 引擎

自进化机制 · 数据自有

个人版

即装即用

个人开发 · 开源项目

v2.0.0 • 29 工具内置 • 云端沙箱运行

免连引擎 · 平台托管

企业版

本地部署

企业内网 · 保密代码

v2.0.0 • 29 工具满配 • 本地单文件 EXE

代码不出本地 · 自进化机制

数据主权三级梯度：内部版数据在自有服务器 • 个人版在平台云 • 企业版代码不出本地 —— 面向合规与信创场景

结束语

你永远的可靠的AI编程助手

受控 • 可审计 • 可追溯 • 可进化

敏哲AI编程专家 —— 让每一次 AI 辅助，都有据可依 · 有迹可循 · 有备无患