

桌面性能监控工具 | 终极完整开发文档（ PRD + 架构 + 排期）

一、产品目标与定位

本产品定位为：高颜值、超低占用、长期常驻运行的桌面系统性能监控工具，优先作为个人高质量开源项目打造，同时预留未来商业化扩展空间。

核心价值在于：实时硬件监控 + 轻量网络测速 + 赛博朋克毛玻璃视觉体验 + 极低系统资源占用。

二、完整功能需求（PRD级）

实时监控指标包括：CPU占用率、频率、温度；GPU占用率、温度、显存；内存使用率与容量；磁盘占用与读写速率；网络上下行速率与延迟。

主动测试能力包括：HTTP下载测速、Ping延迟测试、测速结果展示与历史记录（本地）。

设备识别能力包括：CPU型号、GPU型号、内存规格、磁盘型号、主板与整机品牌型号识别。

界面能力包括：主窗口仪表盘、任务栏极简悬浮显示、模块显示开关、自定义刷新率、深色赛博朋克主题。

三、性能与系统指标

常驻CPU占用目标 1%，内存占用 60MB，启动时间小于1秒。

默认采样刷新周期500ms，最小化后自动进入低功耗模式并降低采样频率。

所有后台线程必须支持异常恢复与安全退出，保证长期稳定运行。

四、总体技术架构设计

后端采用Rust实现系统监控、网络测速、设备识别与低功耗调度。

桌面容器使用Tauri实现跨平台能力与原生IPC通信。

前端使用Vue3 + Canvas/ECharts实现实时曲线与赛博朋克毛玻璃界面。

整体采用分层架构：UI层 IPC层 Rust服务层 操作系统接口层。

五、核心模块拆分（工程级）

system_monitor：统一采集CPU/GPU/内存/磁盘/网络数据并定时推送。

network_test：执行HTTP测速、Ping检测、延迟统计。

device_info：解析硬件型号与整机品牌信息。

low_power_scheduler：最小化后动态降低刷新频率与线程优先级。

frontend_dashboard：实时图表、卡片仪表盘与视觉动效。

tray_widget：任务栏极简性能数据显示。

settings模块：主题、刷新率、模块开关与配置持久化。

六、UI设计规范（可直接设计）

整体风格：深色背景 + 高斯毛玻璃 + 霓虹高亮边缘。

主色建议：蓝紫渐变与青色高光组合，强调科技感。

动效规则：曲线平滑过渡、低频闪烁、避免高CPU动画。

布局规则：卡片式网格布局，支持拖拽排序与模块显隐。

七、开发排期（8周落地）

第1周：Rust硬件采集 + Tauri窗口显示CPU数据。

第2~3周：完成全部实时监控与任务栏悬浮功能。

第4~5周：实现网络测速、Ping检测与设备识别。

第6~7周：UI动效、毛玻璃渲染与低功耗优化。

第8周：打包发布、稳定性测试、开源仓库整理。

八、未来扩展路线

增加CPU/GPU/磁盘专业跑分系统。

增加云端排行榜与配置同步。

支持插件系统与专业版功能扩展。

九、AI Code工程生成提示词（完整版）

请生成基于Rust + Tauri + Vue3的完整桌面性能监控项目。

实现全部硬件实时监控、500ms刷新、HTTP测速、Ping延迟检测。

包含任务栏悬浮显示、低功耗调度与清晰工程结构。

项目需支持直接构建、运行与跨平台打包。