

用 Cursor 赋能校内开发：让 AI 写基础代码，开发者专注核心逻辑

案例提供部门：管理信息技术与系统办公室

支持部门：西浦学习超市

1. 案例背景

我校信息化建设持续推进，开发小组开发人员（共 4 人）承担多类型开发任务，包括不限于：

- 定制化业务系统开发；
- 零代码/低代码平台的脚本与组件开发；
- HiAgent 智能体平台的脚本编写；
- RPA workflow Python 脚本开发；

面对高频、碎片化的需求，团队长期面临三重挑战：

1. 需求量大但人力有限，常规任务积压严重；
2. 部分小型需求技术栈不统一（如需临时使用 Node.js、Python 、Java），与我们主技术栈（React + Java）不匹配，开发调试成本高；

3. **零、低代码平台扩展能力依赖脚本**，但编写重复性逻辑效率低下。

传统开发模式难以兼顾速度、灵活性与维护性。

2. 解决方案

自 2025 年 3 月起，开发小组引入 **Cursor** 作为核心提效工具，实现“AI 辅助 + 人工主导”的协同开发模式：

- **加速常规开发**：通过自然语言快速生成 CRUD 接口、前端表单、等重复性代码，减少基础编码时间；
- **支持异构技术栈小项目**：对于非主流语言或临时性任务（如编写 Python 脚本批量处理图片文件），直接让 Cursor 生成完整可运行示例，避免查阅文档和环境配置耗时；
- **基于项目上下文智能补全**：Cursor 能理解当前项目结构，在生成代码时自动适配已有命名规范、依赖库和错误处理方式，降低集成成本。

目前由 4 名开发成员按需使用，无需改变现有开发流程，即插即用。

3. 成果与效益

- **整体开发效率提升约 30–40%**：常规功能交付周期明显缩短；

- **异构技术栈任务不再“头疼”**：成功完成多个临时性小项目（如 python 脚本读取 Excel 批量生成图片）；在 Ilead 小程序开发中，基于 Ibss 项目代码进行改造，通过 Cursor 快速理解原有逻辑并精准修改，大幅缩短熟悉和迭代周期；
- **低代码平台扩展更灵活**：行政同事提出的插件扩展需求，可通过 Cursor 快速生成验证脚本并嵌入，响应速度提升；
- **开发者专注度提升**：团队成员将更多时间投入系统架构优化、跨系统集成和业务逻辑等高价值工作；
- **成本极低**：仅 4 个人订阅账号（总计约 ¥7500/年），无基础设施投入。

4. 可复制性与推广价值

- **适用于所有涉及代码开发的部门**；
- 尤其适合处理“小而杂、技术栈不统一、人力紧张”的 IT 场景；
- 不依赖特定平台或私有模型，只需开发者具备**基础编程能力**，即可快速上手；
- 可作为学校“AI+开发”能力提升的试点样板，为后续全校级工具选型提供依据。

5. 下一步计划

- **深入探索 Cursor 高级能力**：研究其对 **MCP (Model Context Protocol)** 的支持,尝试将 Cursor 从“代码生成器”升级为“智能开发代理”，例如自动调用内部 API、查询数据库上下文、生成测试用例等；

- **横向对比其他 AI 编辑器**：持续调研如 **Cdoex**、**Trae**、**Continue (开源)** 等工具，在多语言支持、本地模型兼容性、中文理解等方面进行评估，寻找更适合我校技术生态的替代或补充方案；

