

Cursor Meetup

感悟总结



基本信息：

地址：浦东软件园-2号楼

时间：08-16-2025 周六下午一点半开幕

流程：六位嘉宾分享（提问答疑）、穿插抽奖和茶歇、观众开放麦+1min投屏产品介绍

嘉宾分享



AWS-Kiro产品经理的分享（非CS出身但是观感是VB）熟练使用者

- 一句话总结：Cursor擅长单节点性任务，长开发与难项目需要用户以软件开发全生命周期管理的方式优化并执行。
- （kiro的产品特性）结合软件开发流程关注VB项目的开发全生命周期管理：需求定义与规划→设计与架构：整体架构搭建、技术栈选择→开发与实现（我也喜欢组队编程+定期交流分享）：子模块MVP、分模块展开→测试质量与保障：测试与跑通；部署与发布→版本更新管理（git）--Tips：变成Cursor 记忆的形式→维护与迭代：吸收用户反馈后再升版本。



个人实践：

- 开发前：多轮直接和chatbox交互，明确需求、路径并识别到必要的踩坑点；
- 版本更新的部分--Tips：变成Cursor 记忆的形式（大版本更新都打上标签、说明更新内容、git push到仓库）；



个人开发：

集成了VB开发小项目的网页（目标是将周期管理和项目管理能力融入VB开发过程）



湾区程序员回归-AI解决方案创始人（个人欣赏信息收集速度、产品使用广度+产品思维）

- 一句话总结：给出VB的很多建议，工程化视角与和业务端接触经验总结

- 推荐：基于其沟通成本高、业务需求变化快的个人业务场景。V0、Bolt结合Cursor IDE开发平台使用。前者AI工程师与前端业务专家聊需求时能将原先3-4天的定义，捏合为半天的可视化沟通。
- 推荐的工作流：需求定义好打包为Swagger来生成后端代码框架→让cursor读取代码框架生成cursor rules
- VB Tips:
 - rules很重要
 - 分模块完成+先让cursor利用测试文件验证功能后 封装起来不动：先写顶层需求框架、写指令的模块功能、分步测试、选代码技术栈、测试模块、分装部署/版本管理
 - 提前想好能够复用的函数

💡 个人开发/灵感：

- 试用基于Aceternity UI，试用V0、Bolt、lovable。感受UI界面生成模拟速度+质量高，且可以和多款UI库联动、上传github很丝滑。进一步加快业务需求的可视化。
- 好产品=AI^(工程师+产品经理+业务专家)×(执行力+品味)⇒可衡量价值

👤 全栈开发的一个哥，有小红书账号“来点AI”

- 总结：Cursor memories非常重要，系统讲解了记忆的机制、生成、魔改建议；
- How ChatGPT memory works
- 观点：Cursor memories生成机制来自-用户主动要求+后台自运行的来自AI的观察。→进一步可以将记忆云端上传，方便团队共享、编辑；
- 指出记忆分成：个人研发能力、偏好+项目进度记忆
- 具体：

👤 个人实践：

- 个人成长：将记忆也沿用到了GPT-5的子项目setting中。如个人开发板块：整理一下我过往问题的方向，把我的几个主要项目存成记忆，这样后续所有的新知识点（比如我问了关于UI库的问题），它会帮我往这个方向上面靠。
- 项目记忆：结合了版本管理的git使用，我增加了rules为主动记录大小版本每次更新的内容。方便直接调取回滚。

💡 个人开发：



孟健：善用MCP、也是一个全栈全周期爱好者

- 总结：用软件生命周期来管理cursor项目（需求分析、方案设计、代码实现、code review、功能发布）。
- 观点：CC发布建议AI渗透全生命周期的开发。Discover-Design-Build-Deploy-Support&Scale
- MCP推荐：飞书MCP、MCP Hub
- 具体：



个人实践：MCP: File的直接放心新建、Github的访问、push&Pull\Browser MCP



个人开发：做了一个VB全生命周期ToDo的网站，旨在让VB helps VB



Tina Huang Refiy创始人（吃到第一波Cursor红利、爱用X 程序员出身的爱思考卷王）

- 总结：此刻是最好的创业时代。
- 观点：创业公司vs大厂：信息差和工作效率；这些能被善用AI最大程度拉平；
- 观点：技术实现边界趋于0后，品味为王。



个人观点：个人影响力（背书）、品味、执行力；



个人实践：彻底打通X，积极互动；