

褚昕怡

AI 产品经理 / AI 产品助理

独立 AI 产品实践 · 需求分析 · MVP 设计 · AI workflows



个人概况

- 产品能力：**能够从用户场景中识别核心问题，拆解关键任务与价值假设，制定 MVP 范围并通过真实使用持续迭代。
- AI 理解：**关注大模型的不确定性与能力边界，能围绕结果可追溯、异常兜底、人工确认和数据安全设计 AI 产品机制。
- 复合背景：**拥有 1 年新媒体运营经验，具备用户洞察、转化意识、流程沉淀及结果导向，已独立推进 MyLib 0.1.0。

项目经历

MyLib | 本地优先的 AI 文献管理工作台

- 用户痛点：**面向需要高频阅读英文论文的学生与研究者，发现其在 PDF 解析、翻译、校对和资料归档之间反复切换，存在流程割裂、格式易损坏、历史结果难复用及敏感资料上传顾虑。
- 产品方案：**将“导入—解析—翻译—对照阅读—主题管理”整合为单一 workflow，并确立本地优先原则；减少跨工具搬运与重复整理，使文献处理结果能够持续沉淀为个人知识资产。
- MVP 验证：**围绕最高频的文献导入与翻译阅读链路定义 0.1.0，优先交付元数据管理、全文翻译预览、主题标签与筛选排序；通过真实论文持续试用，迭代信息层级、阅读布局 and 任务反馈。
- AI 机制与价值：**针对双栏、OCR、图表和参考文献等高风险场景设计质量规则、异常提示与人工复核入口，在提升长文献处理效率的同时保证结果可控；目前已形成可稳定使用的 0.1.0 版本。

Game Research Copilot | 游戏竞品研究助手

- 用户痛点：**面向独立游戏开发者与策划，竞品资料分散在商店页、评测和社区内容中，人工检索与横向对比耗时；通用 AI 生成结果又存在缺少来源、维度不统一和难以复用的问题。
- 产品方案：**设计从研究对象输入、资料归集到玩法、美术、叙事、商业化等统一维度分析的工作流，输出带来源依据的结构化报告，并将关键设定保存为可继续编辑的项目资产。
- 产品决策与价值：**将早期“单次设定检索”调整为“竞品分析 + 设定资产沉淀”，把一次性问答转化为可复核、可比较、可持续积累的研究过程，降低前期调研成本并支持后续立项判断。

工作经历

艾珩传媒有限公司 | 新媒体运营

2024.11—2025.11

- 用户与内容洞察：**负责小红书、抖音、视频号多账号运营，结合平台机制、用户反馈与同行动态持续调整内容及账号定位，全平台粉丝累计 1000+，做到行业头部水平。
- 转化路径优化：**梳理公域咨询到微信私域的承接流程，平均每周获得后台咨询 20+、有效线索 10+，累计转化客户 100+，相关销售额累计百万元以上。
- 流程沉淀：**建立从账号冷启动、选题、素材生产到发布复盘的运营 SOP，支持多账号稳定产出与快速复制。

教育背景与专业技能

辽宁师范大学 2020.09—2024.06

文物与博物馆学 | 本科

大学英语六级 | 计算机二级等

产品：用户场景、需求分析、用户流程、竞品分析、MVP、产品文档

AI：能力评估、工作流设计、结果追溯、异常兜底、人工确认

工具：ChatGPT、Codex、Claude、Git、Figma、剪映、可画

MyLib | 本地优先的 AI 科研文献工作台 | Windows 桌面端

项目概述

MyLib 面向研究生、科研人员和个人研究者，将英文论文 PDF 的导入、元数据识别、AI 全文翻译、阅读笔记、研究选题与多维检索整合为一套本地工作流。项目目标是解决原文、译文、笔记分散以及 AI 输出难以长期沉淀的问题，形成“入库—识别—汉化—记录—关联—找回”的科研资料闭环。

需求分析与产品取舍

核心痛点：① 手工录入论文信息效率低；② 长篇英文阅读门槛高；③ 翻译过程耗时且缺少反馈；④ 文献、译文与笔记相互割裂；⑤ 资料积累后难以按选题找回。

MVP 优先打通一篇真实论文的完整处理链路，暂不加入云同步、多人协作、PDF 批注、完整引用管理和知识图谱，以控制范围并验证核心价值。

产品方案拆解

- **智能入库：**上传 PDF 后由 AI 识别标题、作者、摘要、关键词等信息，用户修改确认后入库；识别失败时支持手动录入。
- **AI 汉化：**长论文分块处理，以异步任务返回进度，支持停止、失败提示与重复任务复用，降低等待焦虑和 API 浪费。
- **知识沉淀：**生成中文及双语 Markdown，将原文、译文、笔记、分类和研究选题持续关联。
- **快速找回：**支持标题、作者、摘要、关键词、分类、阅读状态、汉化状态和研究选题组合筛选。

设计亮点

- AI 结果采用“模型建议—人工校正—确认执行”，兼顾效率与科研信息准确性；
- GROBID、Poppler、pdf-parse 多级解析回退，提升 PDF 兼容性；
- 本地优先并允许用户自配模型 API，兼顾隐私、数据归属和成本透明；
- 将不稳定的模型调用包装为可观察、可停止、可追踪的任务；
- 原版 PDF 与生成译文分离保存，失败或取消不破坏源文件。

个人职责与能力体现

独立完成用户问题定义、目标用户分析、产品定位、MVP 范围、用户流程、功能优先级、异常场景与验收标准设计，并借助 AI 辅助完成 React + Express + Electron 产品落地。项目重点体现了 AI 能力边界判断、人机协作设计、长任务体验、成本控制、失败兜底以及产品与技术协同能力。

成果验收与质量保障

已交付可安装的 Windows 0.1.0 版本，核心链路可完成“PDF 导入—元数据确认—异步汉化—译文生成—笔记保存—选题关联—筛选找回”。通过单元、API、组件、集成与端到端测试覆盖核心流程，并对 API Key 缺失、PDF 解析失败、重复翻译、任务停止、应用重启遗留状态和非法文件路径等异常场景进行处理。现阶段以链路可用性和异常覆盖验收。

Game Research Copilot | 本地优先的

AI 游戏竞品研究工作台

项目简介

Game Research Copilot 面向游戏策划、独立开发者和立项团队，将自有游戏设定、公开竞品情报与 AI 分析整合为本地工作流。用户维护项目设定库后，可输入竞品名称和市场范围，系统生成中文竞品报告，并保留公开来源、内部设定命中与历史记录，形成“设定入库—证据检索—竞品分析—设计复盘”的闭环。

需求与拆分

核心痛点：①竞品资料分散，检索和归纳成本高；②竞品结论难与自身设定建立关联；③AI 输出易缺少依据、难以复盘。

MVP 聚焦单用户本地研究场景，优先验证“公开证据 + 内部设定 + 可执行建议”的分析链路；暂不提供登录协作、云同步、文档上传解析、定时监控和聊天追问，以控制范围。

产品实现

- **设定库与索引：**基于 Prisma + SQLite 保存游戏项目、完整设定和报告历史；按标题、段落切片，调用可配置 Embedding 服务生成向量，并支持索引状态、预览与手动重建。
- **证据驱动分析：**Tavily 检索玩法、口碑、商业化等公开资料；语义召回相关内部切片；DeepSeek 按严格 JSON 生成执行摘要、核心循环、系统、市场、差异化与行动建议。
- **可信与可用性：**前端分开展示公开来源、内部证据和生成结果；证据不足时隐藏可选模块并提示限制；Embedding 不可用时降级为公开资料分析，JSON 解析失败自动重试一次。

测试验证

使用 Vitest 对外部服务进行 Mock，覆盖文档切片、向量相似度与降级、项目 CRUD/重建索引、分析请求校验与报告解析，以及分析台、设定库页面交互。已执行测试：8 个测试文件、29 项用例全部通过，核心流程和异常场景具备可验证性。

Seekey | 一次失败的 AI MVP 验证

用户问题

购买游戏捆绑包的玩家需要在多个网站之间查找包内游戏、Key 回收价和风险信息，再手工计算是否回本。这个问题真实存在：信息分散、名称不统一、价格变化快，判断一次礼包需要反复查价和核对。

核心假设

最初假设 AI 可以识别不同网站中的同一款游戏，整理价格与风险信息，并给出购买建议。成功标准是：用户导入礼包后，几分钟内得到有来源、可检查的结论，而且人工核对时间明显少于手工查价。

MVP 设计

为低成本验证，只保留核心流程：同步 Fanatical 礼包，手动导入 Humble 礼包，使用 ITAD 和 Steam 辅助匹配游戏，读取 Steamy Key 价，计算预计收入与利润，并在看板中显示建议和可信度。历史趋势、复杂筛选等功能暂不开发。

测试结果

测试任务是完成“导入真实礼包—自动匹配游戏—取得报价—查看建议”，重点检查价格覆盖、匹配准确、来源可追溯、人工修正时间和接口稳定性。工程测试全部通过，但产品验证没有通过：用户仍需大量手工核对，无法直接依赖结果做购买决定。

失败原因

主要问题不是单一的模型能力，而是数据和产品机制。部分游戏没有稳定的 Key 报价，名称匹配仍会出错；接口会出现 403、429 或验证码，手动导入又打断了自动流程；报价也不能完整反映手续费、成交速度、锁区和价格波动。AI 可以整理已有信息，却无法补足缺失的实时数据。增加模型调用和更新频率只会提高成本，不能解决可信度问题。

产品决策

因此停止继续投入“由 AI 自动给出购买结论”的方案。用户需求可能真实存在，但“用 AI 解决”这个方案并不一定成立。

MotionVault | 本地优先的前端动效资产管理工具

Windows 桌面端

项目简介

MotionVault 面向个人前端开发者与设计工程师，将 React Bits、CodePen 等站点的动效源码或 AI Prompt 集中到本地工作流。用户可完成源码入库、实时预览、分类检索与一键复制，形成“发现动效—收藏源码—运行验证—快速找回—项目复用”的个人动效资产闭环。

需求与拆分

核心痛点：①动效分散在不同网站，截图无法保留可复用源码；②依赖、多文件和素材差异导致复制后难以直接运行；③网页收藏增多后检索、预览和备份成本上升。

MVP 先验证“粘贴即预览”，再交付源码 CRUD、搜索筛选、React/HTML 双运行器和本地备份。

产品实现

- **粘贴即运行：**独立 iframe 内通过 react-runner 编译 TSX，注入 React、Motion、GSAP、Three.js 等白名单依赖，并由 Tailwind 浏览器运行时生成样式；编译失败仍可保存源码并显示错误提示。
- **管理与复用：**React + Express + SQLite 实现分类、标签、技术栈、收藏和搜索；支持多文件、本地素材、自动截图、Shiki 高亮复制、书签预填及 AI Prompt 双向导入导出。
- **本地与稳定性：**用户数据统一保存在 vault 目录并可导出 ZIP；IntersectionObserver 延迟挂载预览，错误边界与 postMessage 隔离单条动效异常，Electron 配置支持 Windows 桌面交付。

测试验证

已执行 TypeScript 类型检查与 Vite 生产构建，均通过；并以 React Bits 单文件示例、HTML 片段及 Motion-Primitives 多文件 + 图片素材场景验证预览链路，覆盖缺失依赖、编译失败、素材导入和截图兜底。当前测试以构建校验和典型场景验收为主，后续可补充自动化单元与端到端测试。