

非常好！基于我们此前的深度讨论，现在为你系统梳理一套**可用于工程落地的 AI 简历分析与面试辅助工具方案**。本方案兼顾**功能性、隐私安全、技术可行性与生产可部署性**，并明确推荐**已验证的技术栈与关键组件**。

✅ 一、核心功能列表 (MVP + 扩展)

**1. 简历智能解析 (Resume Parsing) **

- 支持格式: PDF (含扫描件)、DOCX、JPG/PNG (图片简历)
- 输出: 结构化 JSON (姓名、联系方式、教育、工作经历、技能等)
- **强制脱敏**: 自动替换姓名、电话、邮箱等为 `[PERSON]`、`[PHONE]` 等占位符

↴ **2. 能力信号提取 (Signal Extraction) **

- 从简历中识别:
 - 技术栈 (如 Python, Kubernetes)
 - 项目行为动词 (如“主导”、“优化”、“设计”)
 - 经验年限、职级线索
 - 潜在疑点 (时间断层、模糊描述)

? **3. 智能面试问题生成 (Interview Question Generation) **

- 问题类型:
 - 行为问题 (BQ): 基于项目经历
 - 技术问题 (TQ): 匹配技能栈
 - 情景问题 (SQ): 针对空白/高阶能力
- 输出: 问题 + 考察目的 + 目标技能 + 追问建议

⌛ **4. 面试评估要点 (Rubric for Interviewers) **

- 不评估候选人真实回答!
- 仅提供**面试官参考指南**:
 - 优秀回答应包含的关键点
 - 常见缺陷
 - 公平性提示 (避免引导性问题)

↴ **5. 结构化输出**

- JSON (机器可读, 供 HR 系统对接)
- Markdown (人工可读, 含高亮与建议)
- 可选: 嵌入向量 (用于岗位匹配, 未来扩展)

↴ **6. 隐私与安全控制**

- 默认**本地运行**, 原始文件不出设备
- **API 调用需用户显式授权**

- **无数据留存**: 不记录、不缓存、不上传

二、建议技术栈 (生产就绪)

功能模块	推荐技术	说明
核心解析引擎 [**Alibaba SmartResume**](https://github.com/alibaba/SmartResume)		
<input checked="" type="checkbox"/> 开源、支持 OCR/扫描件、专用微调模型、本地部署		
脱敏引擎 **Microsoft Presidio** <input checked="" type="checkbox"/> 支持多语言 PII 识别，符合 GDPR		
语义增强 (可选) **Anthropic Claude API** (`claude-3-5-sonnet`) 仅用于生成问题/rubric，**输入为脱敏数据**		
本地 LLM 备选 **Qwen2.5-7B-Instruct** (通过 Ollama / vLLM) 若禁用网络，可用本地模型生成简单问题		
智能代理框架 **OpenDevin** (替代 Claude Code) <input checked="" type="checkbox"/> 开源、支持 Tools、Docker 沙箱、CLI + Web		
规则与模板引擎 **Jinja2 + YAML** 管理问题模板、提示词		
NLP 辅助 **spaCy** (`zh_core_web_sm`, `en_core_web_sm`) 提取动词短语、技能关键词		
部署 **Docker Compose** 一键启动 SmartResume + OpenDevin + 你的工具		

> **问** **为什么不直接用 Claude Code?**

> 因其**闭源、不可私有部署、无沙箱**，不适合生产环境。**OpenDevin** 是其开源精神继承者。

##^ 三、关键组件 (系统模块分解)

1. **Document Ingestion Layer (文档输入层) **

- 组件: 用户上传文件 (CLI 或 Web)
- 技术: `streamlit` (GUI) / `argparse` (CLI)

2. **Resume Parsing Engine (解析引擎) **

- 组件: `SmartResume` 服务 (Docker)
- 输入: 原始文件
- 输出: 结构化 JSON (含原始 PII)

3. **PII Anonymization Service (脱敏服务) **

- 组件: `presidio-anonymizer` + 自定义规则

- 输入: SmartResume JSON
- 输出: 脱敏 JSON (`[PERSON]`, `[EMAIL]`)

4. **Signal Extractor (信号提取器) **

- 组件: Python 脚本 (基于 spaCy + 规则)
- 输出: 可面试化信号列表 (技能、行为、疑点)

5. **Interview Agent (面试智能体) **

- 组件: OpenDevin Tool (`interview_generator.py`)
- 逻辑:
 - 匹配信号 → 问题模板
 - 调用 Claude (可选) 优化问题
 - 生成 Rubric

6. **Output Generator (输出生成器) **

- 组件: `Jinja2` 模板引擎
- 输出:
 - `analysis.json`
 - `interview_plan.md`

7. **Security & Compliance Guardrails (安全护栏) **

- 组件:
 - 输入文件类型校验
 - PII 过滤器 (输出前两次扫描)
 - 公平性关键词黑名单 (如 “婚育”、“年龄”)
 - API 调用开关 (环境变量控制)

L 四、部署架构 (本地优先)

```
```bash
项目根目录
ai-resume-assistant/
├── docker-compose.yaml # 启动 SmartResume + OpenDevin
├── tools/
| └── resume_analyzer.py # OpenDevin Tool 实现
└── config/
 ├── presidio.yaml # 脱敏规则
 └── question_templates.yaml # 问题模板
└── models/ # (可选) 本地 LLM
└── output/ # 生成结果 (.gitignore)
```
```

```

用户使用:

```
``bash
1. 启动服务
docker-compose up -d

2. 运行分析
opendevin run "分析 ~/resume.pdf, 生成面试问题"
``
```

---

## ## ✅ 五、优势总结

维度   本方案优势
----- -----
**准确性**   SmartResume 专业模型 > 通用 PDF 解析
**格式支持**   ✅ 扫描件、图片、PDF、Word
**隐私安全**   本地运行 + 强制脱敏 + 沙箱执行
**可扩展性**   OpenDevin Tools 机制支持无限扩展
**合规性**   无数据上传, 符合 GDPR/CCPA 基础要求
**成本可控**   本地模式免费, API 按需启用

---

## ## b 下一步行动建议

1. \*\*初始化项目\*\*: 创建上述目录结构
2. \*\*部署 SmartResume\*\*: `docker-compose up`
3. \*\*开发第一个 OpenDevin Tool\*\*: `resume\_analyzer.py`
4. \*\*集成 Presidio 脱敏\*\*
5. \*\*定义问题模板 YAML\*\*

---

这套方案已避开 Claude Code 的闭源限制, 采用\*\*全栈开源、生产就绪\*\*的技术组合, 可立即用于企业内部 HR 工具、招聘平台辅助模块或个人职业分析。

如需, 我可为你生成:

- 完整的 `docker-compose.yaml`
- OpenDevin Tool 模板代码
- Presidio 脱敏配置示例
- 问题模板 YAML 文件

请告诉我你希望优先生成哪个部分？