

AI 面试训练 Agent 产品需求文档 PRD

1. 文档信息

产品名称	AI 面试训练 Agent
文档版本	V1.0
文档类型	产品需求文档 PRD
适用场景	技术产品 / 云计算产品 / AI 产品求职训练
当前阶段	已实现 MVP

2. 产品背景

技术产品类岗位通常同时考察候选人的技术理解、产品思维、逻辑表达、问题拆解、数据意识和跨团队推动能力。以腾讯云智技术产品实习生岗位为例，JD 中明确强调云计算、计算能力、全球网络、前沿技术应用、客户需求理解、产品文档梳理和从 0 到 1 的产品建设能力。

传统面试准备方式主要依赖题库、通用 AI 问答或人工辅导，存在以下问题：

1. 训练内容与目标 JD 脱节，难以围绕具体岗位能力要求定制问题。
2. 简历与岗位的匹配度缺少前置诊断，候选人不知道自己的项目应该如何包装。
3. 普通题库只能提供固定问题，无法根据用户回答表现动态追问。
4. 大模型直接问答缺少评分标准、长期记忆和训练闭环，反馈稳定性不足。

5. 候选人完成训练后缺少可复用的复盘报告、简历优化建议和投递材料沉淀。

因此，本项目希望构建一套面向技术产品求职场景的 AI 面试训练 Agent，通过大模型、岗位知识库、评分模型和长期训练画像，帮助用户完成从 JD 分析到面试复盘的完整准备流程。

3. 产品定位

AI 面试训练 Agent 是一款面向技术产品岗位候选人的智能训练工具。产品通过解析目标岗位 JD 和用户简历，自动生成岗位匹配诊断、训练计划和动态面试问题，并在每轮回答后输出结构化评分、参考答案和改进建议，最终形成可导出的复盘报告与简历项目优化建议。

产品核心定位不是单次 AI 问答工具，而是具备规划、检索、记忆、追问、评估和复盘能力的求职训练 Agent。

4. 产品目标

4.1 用户目标

1. 帮助用户理解目标岗位真正考察的能力要求。
2. 帮助用户识别简历与 JD 之间的匹配点和缺口。
3. 通过动态追问提升用户回答的结构性、专业度和岗位匹配度。
4. 通过复盘报告沉淀个人短板、训练重点和简历优化方向。
5. 帮助用户形成可用于真实投递和面试表达的项目叙事。

4.2 产品目标

1. 建立 JD 驱动的训练流程，提升面试训练的岗位针对性。

2. 构建岗位模板驱动的评分体系，提升评价维度的专业性。

3. 引入轻量级 RAG 知识库，使问题生成和反馈更贴近云计算、AI、智能驾驶等技术产品场景。

4. 引入长期训练画像，使系统能够根据历史表现持续调整训练重点。

5. 输出可解释、可导出、可复用的训练报告和投递材料。

5. 目标用户

5.1 核心用户

正在投递技术产品、云计算产品、AI 产品、智能驾驶产品等岗位的学生或初级候选人。

5.2 典型用户画像

用户类型	特征	核心特点
计算机/软件相关专业学生	有技术基础，但产品表达不足	不知道如何把技术项目讲成产品项目
AI/大模型项目实践者	做过模型调用、RAG、Agent 等项目	不知道如何体现产品价值和业务闭环
技术产品岗位求职者	目标岗位明确，准备时间有限	需要快速识别 3D 重点和面试风险
跨方向求职者	有项目经历，但目标岗位跨度较大	需要重新包装项目与岗位匹配点

6. 用户场景

场景一：投递前岗位匹配诊断

用户准备投递腾讯云智技术产品实习生岗位，将 JD 和简历粘贴到系统中。系统基于岗位模板和知识库生成匹配度评分，指出简历中已匹配的技术背景、AI 项目经历和产品能力证据，同时提示云计算认知、客户需求拆

解、产品文档产出等缺口。

场景二：正式面试前专项训练

用户选择腾讯云智模板并开启 Agent 训练。系统先生成训练计划，再根据用户每轮回答动态调整下一题。例如用户回答中缺少云产品场景，系统下一轮会追问“如何将计算/网络能力包装成客户可感知的产品功能”。

场景三：回答复盘与简历优化

用户完成多轮模拟面试后，系统生成能力雷达、短板分析、优先改进动作和 3 天训练计划，并自动生成适合放入简历的项目经历表述，帮助用户将 AI 面试训练 Agent 项目包装为大模型产品项目。

7. 产品范围

7.1 MVP 已覆盖范围

1. JD 输入与岗位模板选择。
2. 简历输入与上传。
3. 训练前 JD/简历匹配诊断。
4. Agent 训练计划生成。
5. 动态面试问题生成。
6. 用户回答输入，支持文字和语音转文字。
7. 回答评分、问题分析、参考答案和改写建议。
8. 最终 Agent 复盘报告。
9. 长期训练画像和历史报告回看。
10. Markdown/PDF/投递材料包导出。

11. 简历项目经历生成。
12. 云计算、AI Agent、智能驾驶等轻量级知识库。

7.2 功能优先级

V1.0 的核心目标为先跑通“岗位理解 - 训练 - 复盘 - 材料沉淀”闭环，因此功能优先级按是否影响闭环成立来划分。

P0 功能必须保留，包括 JD 输入、岗位模板、训练前诊断、Agent 训练计划、回答评分、动态追问、最终复盘报告、历史报告回看和本地数据清空。这些能力共同决定产品是否成立为“AI 面试训练 Agent”，任一核心节点缺失都会让产品退化为普通问答或题库工具。

P1 功能用于提升训练效率和可解释性，包括语音输入、Markdown/PDF 导出、投递材料包导出、简历项目经历生成和 Agent 决策轨迹展示。若开发资源紧张，可先保证文字输入和 Markdown 导出，将语音输入、PDF 导出和材料包生成功能放到 V1.0 后半阶段完善。

P2 功能面向规模化和商业化，包括多岗位模板扩展、报告趋势对比、知识库管理页、账号系统、跨设备同步和人工导师点评。这些能力不影响 MVP 的核心闭环，预计放入 V1.1、V1.2 或 V2.0 规划。

7.3 本期不覆盖范围

1. 多用户账号系统。
2. 在线支付和商业化套餐。
3. 企业级权限管理。
4. 面试视频分析和表情识别。
5. 大规模向量数据库和在线知识库自动更新。

6. 人工导师协同批改。

8. 核心流程

8.1 用户主流程

1. 用户选择岗位模板，例如“腾讯云智技术产品实习生”。
2. 用户输入目标岗位 JD。
3. 用户上传或粘贴个人简历。
4. 用户点击“训练前诊断”。
5. 系统输出岗位匹配度、优势信号、能力缺口、高风险追问和训练顺序。
6. 用户进入 Agent 训练模式。
7. 系统生成训练计划和首轮面试问题。
8. 用户输入或语音录入回答。
9. 系统对回答进行评分、分析和改写。
10. Agent 根据历史回答和薄弱项动态生成下一题。
11. 用户完成多轮训练后生成最终复盘报告。
12. 用户导出报告、投递材料包或生成简历项目经历。

8.2 Agent workflow

1. 任务理解：解析岗位 JD、岗位模板、用户简历和训练目标。
2. 知识检索：从本地岗位知识库中检索云计算、AI Agent、智能驾驶等相关知识。

3. 训练规划：生成能力差距、训练目标、问题序列和评分重点。
4. 动态追问：根据用户上一轮回答中的短板生成下一轮问题。
5. 回答评估：基于岗位模板评分维度输出结构化评分。
6. 记忆更新：记录用户平均分、薄弱信号、历史问题和下一阶段训练重点。
7. 复盘输出：生成能力雷达、短板分析、行动建议和简历优化建议。

8.3 用户故事与异常分支

Agent 的核心流程描述的是理想路径，但真实用户经常会以不完整信息开始训练。因此，V1.0 需要允许用户“先练起来”，同时在关键节点明确提示结果可信度和降级方案。

当用户只输入 JD、没有提供简历时，系统仍允许执行训练前诊断，但诊断结果必须明确标记“简历信息不足”。此时系统重点输出岗位能力拆解、通用准备建议、高风险追问和简历待补充信息；后续训练允许继续，但 Agent 计划应进入“无简历模式”，问题以岗位能力理解和项目经历挖掘为主。

当 JD 信息过短，例如用户只输入岗位名称或少量关键词时，系统应提示 JD 信息不足，并建议补充岗位职责、任职要求、技术关键词和公司业务方向。用户仍可继续训练，但诊断结果需要降低可信度标记；如果用户选择腾讯云智模板，系统可用模板内置 JD 作为补充上下文。

语音输入失败时，系统不能清空用户已经输入的文本。前端提示“语音识别失败，可重试或改用文字输入”，并保留当前输入框内容。若浏览器不支持语音识别，则禁用或隐藏语音按钮，提示用户使用文字输入或更换浏览器。语音始终是效率增强能力，而不是训练流程的必要条件。

用户中途退出训练后，V1.0 至少应保留已完成回答、长期画像和历史报告。未完成训练的断点恢复可作为 P1 能力处理：用户再次进入时，系统可以提示“是否基于历史表现重新生成训练计划”。V1.1 再进一步支持完整训练计划断点恢复。

当模型接口异常或网络请求失败时，前端应保留用户输入，停止 loading，并提示“系统处理异常，请稍后重试”。后端记录接口名、错误类型和时间，便于排查。用户可以在不重新输入回答的情况下再次提交。

8.4 反馈节奏与情绪设计

Agent 的反馈需要在不牺牲专业性的前提下提升用户坚持训练的意愿。

当单轮得分低于 60 分时，Agent 应先肯定用户已经完成表达，再指出一个最优先的改进点。例如：“你已经把项目背景说出来了，接下来我们先集中补强结果量化，这会明显提升技术产品岗位的说服力。”

当用户连续两轮得分提升时，Agent 应给出正向激励，并总结具体进步点。例如：“这两轮你的结构化表达更稳定了，尤其是行动和结果开始分开讲。下一步我们继续抠岗位场景。”

当同一维度连续两轮偏低时，Agent 不应重复批评，而应把问题转化为专项训练目标。例如：“这个问题连续出现，说明它不是临场失误，而是需要单独训练。下一题我会只追问数据指标和结果口径。”

当用户完成最终报告时，Agent 应给出阶段性总结和下一步任务，例如提醒用户先修改报告中的简历补充点，再进入下一轮训练。这类 soft touch 不改变评分标准，但会提升用户持续训练的意愿。

9. 功能需求

9.1 岗位模板管理

功能描述

系统内置多个岗位模板，用于驱动 JD 示例、训练重点、评分权重和知识库检索方向。

当前模板

1. 通用技术产品实习生。
2. 腾讯云智技术产品实习生。
3. 小米智能驾驶产品实习生。

验收标准

1. 用户选择模板后，系统自动填充对应岗位 JD。
2. 前端展示模板核心能力、推荐问题方向、知识重点和简历包装建议。
3. 后续训练计划和评分维度能够读取当前模板上下文。

9.2 训练前 JD/简历匹配诊断

功能描述

用户在正式训练前，可输入 JD 和简历，系统生成岗位匹配诊断。

输入

1. 目标岗位名称。
2. 岗位 JD。
3. 用户简历或项目背景。

4. 当前岗位模板。

输出

1. **matchScore**: 岗位匹配度, 0-100。
2. **summary**: 一句话诊断结论。
3. **matchedSignals**: 已匹配岗位的能力证据。
4. **gaps**: 当前能力或简历表达缺口。
5. **riskQuestions**: 面试官可能追问的问题。
6. **recommendedTrainingOrder**: 推荐训练顺序。
7. **resumeMissingInfo**: 建议补充到简历中的信息。

验收标准

1. 用户点击“训练前诊断”后, 页面展示结构化诊断结果。
2. 无简历时, 系统可以基于 JD 给出准备建议, 并提示简历信息不足。
3. 不同岗位差异度, 如腾讯云智模板下, 诊断内容应覆盖云计算、技术产品、客户需求、产品文档等关键词。

9.3 Agent 训练计划生成

功能描述

系统根据 JD、简历、岗位模板、历史画像和知识库生成训练计划。

计划内容

1. 当前候选人的能力差距。
2. 本轮训练目标。
3. 推荐问题序列。

4. 每轮问题的考察意图。

5. 评分关注点。

验收标准

1. 训练计划以结构化方式展示。

2. 计划能够体现岗位差异，例如腾讯云智关注云计算和客户需求，小米智能驾驶关注安全边界和数据闭环。

3. 计划生成后自动进入第一轮面试。

9.4 动态追问

功能描述

Agent 根据用户历史回答、当前得分、薄弱维度和训练计划动态生成下一题。

追问策略

1. 如果回答缺少 STAR 结构，下一题加强项目拆解。

2. 如果回答缺少量化数据，下一题追问指标和结果。

3. 如果回答缺少岗位匹配度，下一题追问岗位场景。

4. 如果回答缺少技术产品深度，下一题追问技术方案、边界和产品化判断。

动态追问决策规则

动态追问不采用固定题库顺序，而是从上一轮回答中识别最影响面试表现的短板，并生成一条更聚焦的追问。

1. 当回答 STAR 不完整时，Agent 优先追问项目叙事链路。例如用户讲了做法但没有讲结果时，下一题可以追问：“你刚才讲到了做法，但结果还

不够清楚。能不能补充这个项目最终带来了什么变化，以及你个人贡献是什么？”

2. 当回答缺少量化数据时，Agent 优先追问指标口径和业务影响。例如：“如果要衡量这个功能是否有效，你会选择哪些指标？这些指标分别反映效率、质量还是用户体验？”

3. 当回答岗位匹配弱时，Agent 需要把问题拉回目标岗位场景。腾讯云智模板下，可追问：“如果你在腾讯云智负责一个云计算产品能力，你会如何把这段项目经验迁移到客户需求分析和产品文档设计中？”

4. 当回答技术/产品深度不足时，Agent 追问方案边界、技术取舍和产品化判断。例如：“这个 Agent 功能背后的技术方案有什么边界？如果模型输出不稳定，你会从产品和技术两侧怎么兜底？”

5. 当回答缺少推动执行时，Agent 追问协作对象、优先级和验收标准。例如：“如果这个需求要上线，你需要和前端、后端、算法或运营分别对齐什么？你会如何定义优先级和验收标准？”

6. 当回答缺少反思时，Agent 追问下一步迭代方向。例如：“如果重新做一版，你会优先改哪一个环节？为什么它对用户价值或系统稳定性最关键？”

验收标准

1. 下一题不是固定题库顺序，而是结合上一轮表现生成。
2. 页面展示 Agent 决策轨迹，说明系统为什么选择该追问方向。
3. 每轮问题应包含考察意图和回答提示。

9.5 回答评分与参考答案

功能描述

系统对用户回答进行多维度评分，输出问题分析、改进建议、参考答案和改写版本。

评分维度

通用评分维度包括：

1. 岗位匹配度。
2. STAR 完整度。
3. 技术/产品深度。
4. 数据量化与结果意识。
5. 结构化表达。
6. 反思与成长。

评分权重

评分维度	通用	技术产品	智驾产品	说明
岗位匹配度	20	20	20	是否贴合目标岗位职责、业务场景和关键词
STAR 完整度	20	15	15	是否完整交代背景、任务、行动、结果和个人贡献
技术/产品深度	20	20	20	腾讯云智侧重云计算/AI 产品理解，智能驾驶侧重安全边界和数据闭环
客户需求/场景拆解	10	15	15	是否能从客户或用户场景出发拆解问题
数据量化与结果意识	15	15	15	是否有指标、规模、前后对比、验证方式
结构化表达与沟通推动	15	15	15	是否表达清晰，并体现协作推进和复盘意识
总分	100	100	100	V1.0 先采用固定权重，V1.1 支持模板配置化

评分校准规则

1. 回答过短时设置总分上限。
2. 缺少结果或量化数据时限制高分。

3. 缺少岗位匹配信号时限制岗位匹配维度得分。

4. STAR 结构不完整时限制结构化表达和完整度得分。

评分校准阈值：

V1.0 先采用一组可配置的初始阈值，用于降低大模型评分漂移。回答少于 30 字时，视为信息不足，总分上限为 40 分；回答在 30-80 字之间时，总分上限为 60 分，但如果当前问题本身是简短追问，可由模型标记为豁免。

如果回答缺少结果描述，例如没有出现结果、影响、变化、产出、上线或验证方式，总分上限为 75 分。若回答缺少量化数据，例如没有数字、比例、时间、规模、频次或指标名称，数据量化维度最高为 8/15；对于早期项目，系统不要求用户编造数据，而应提示“可补充指标”。

如果回答缺少岗位匹配信号，岗位匹配度最高按该维度 60% 计算。如技术产品模板下重点检查云计算、客户需求、产品文档和技术产品化表达。若回答只讲团队成果、没有说明个人贡献，STAR 完整度最高按该维度 70% 计算，并引导用户补充“我负责、我推动、我设计”的具体动作。

如果模型识别到明显夸大或无法验证的数据，例如虚构用户量、收入或上线范围，系统应降低可信度并提示风险，不应鼓励用户直接使用该表述。

验收标准

1. 输出总分和分维度评分。
2. 输出 2-4 条主要问题。
3. 输出可直接学习的参考答案。
4. 输出更贴近岗位的改写版本。

9.6 长期训练画像

功能描述

系统在浏览器本地记录用户训练画像，用于后续训练个性化。

画像字段

1. 累计回答次数。
2. 完成训练场次。
3. 历史平均分。
4. 高频薄弱信号。
5. 最近问题。
6. 下一阶段训练重点。
7. 最近更新时间。

验收标准

1. 每次回答评分后更新长期画像。
2. 最终报告生成后更新下一阶段训练重点。
3. 用户可以一键清空本地画像和历史记录。

9.7 最终复盘报告

功能描述

用户完成多轮训练后，系统生成 Agent 复盘报告。

报告内容

1. 总体总结。
2. readinessScore: 面试准备度评分。

3. capabilityRadar: 能力雷达。
4. strengths: 优势。
5. weaknesses: 短板。
6. recommendedActions: 优先改进动作。
7. nextTrainingPlan: 三日训练计划。
8. resumeAdvice: 简历优化建议。
9. agentMemory: 长期画像更新建议。

验收标准

10. 报告可在页面中完整展示。
11. 报告支持 Markdown 导出。
12. 报告支持打印或保存 PDF。
13. 报告可重新从历史记录中打开。

9.8 简历项目经历生成

功能描述

系统基于用户项目背景、目标岗位和训练复盘结果，生成可放入简历的项目经历。

输出内容

1. 项目名称。
2. 项目角色。
3. 一句话简介。
4. 3-5 条简历 bullet。

5. 项目关键词。
6. 60 秒面试介绍话术。
7. 风险提示，指出哪些数据需要用户确认真实性。

验收标准

1. 针对不同岗位，简历经历应突出相应所需能力。
2. 不编造真实用户量、收入、上线数据等无法验证的信息。
3. 缺少数据时，用“可补充指标”提示用户后续完善。

9.9 语音输入

功能描述

系统支持在面试问题输入、复盘回答和模拟面试回答中使用语音转文字，提高训练效率和真实面试感。

技术路线与边界



语音输入流程图

V1.0 至少定义以下状态：空闲、权限请求中、正在聆听、识别中、识别完成、识别失败、浏览器不支持、权限拒绝。状态变化应通过按钮文案、动画或提示语传达给用户，如：空闲态显示“语音输入”；正在聆听时显示“正在聆听...”并高亮麦克风；识别中显示转圈动画和“识别中”；识别完成后短暂提示“已填入文本”；失败时提示“识别失败，可重试或改用文字输入”。任何失败状态都不得清空用户已输入内容。

V1.0 采用浏览器 Web Speech API 实现语音识别，原因是接入成本低、无需额外上传音频文件，适合个人训练工具的轻量化场景。

该方案的优点是实现轻、成本低，也不需要应用侧保存音频文件；限制是识别能力依赖浏览器和系统环境，Chrome 系浏览器通常更稳定。对于不支持 Web Speech API 的浏览器，产品直接降级为文字输入。

后续如果需要更稳定的跨端识别效果，可以接入模型厂商语音转文字 API，但这会引入音频上传、识别延迟、模型成本和更严格的隐私说明。该能力预计放到 V1.1 或 V1.2，不作为 V1.0 核心依赖。

权限与错误提示

1. 首次点击语音输入时，浏览器请求麦克风权限。
2. 用户拒绝授权时，提示“无法访问麦克风，请开启浏览器录音权限，或直接使用文字输入”。
3. 浏览器不支持时，提示“当前浏览器不支持语音识别，请使用文字输入或更换浏览器”。
4. 识别中断或失败时，保留已有文本，不清空输入框。

验收标准

1. 用户可点击语音按钮开始录入。
2. 识别结果自动填入对应文本框。

3. 浏览器不支持语音识别时给出提示。

9.10 隐私与本地数据管理

功能描述

系统需要向用户说明数据处理方式，并提供本地数据清理能力。

数据说明

1. 训练画像和历史报告保存在当前浏览器 `localStorage`。
2. JD、简历和回答内容会发送到用户配置的模型 API 进行分析。
3. 当前本地应用不在服务端持久化用户内容。

本地存储策略

长期训练画像保存在 `localStorage` 中，采用持续覆盖更新策略。为避免画像无限膨胀，`weakSignals`、`recentProblems`、`nextSessionFocus` 等字段只保留最近或最高频的 8-10 条。

历史复盘报告默认保留最近 10-12 份，超出后自动删除最旧记录。未完成训练草稿在 V1.0 可只保存在页面状态或短期本地状态中，不强制持久化；完整断点恢复放入 V1.1。导出文件由用户下载到本地保存，导出前需要提示其中可能包含 JD、简历和个人经历等敏感信息。

当浏览器 `localStorage` 写入失败或容量不足时，系统应保留当前页面状态，并提示用户导出历史报告或清空本地数据。

验收标准

1. 页面展示隐私与本地数据说明。
2. 用户可一键清空本地训练画像和历史报告。
3. 清空后历史面板和画像状态同步刷新。

9.11 知识库维护

功能描述

V1.0 知识库面向内部维护，不面向普通用户开放编辑。知识条目用于增强 Agent 对云计算、AI Agent、智能驾驶等岗位场景的理解。

知识条目字段

知识条目应采用结构化字段，至少包含唯一标识、标题、正文、标签、关联岗位模板、检索优先级和最近更新时间。正文建议控制在 200-500 字，便于模型在训练计划、追问和复盘阶段引用。

示例：

```
1. {  
2.   "id": "cloud-product-metrics",  
3.   "title": "云计算产品指标设计",  
4.   "content": "云计算产品评估通常关注可用性、稳定性、资源利用率、成本、交付效率和客户问题解决效率。",  
5.   "tags": ["cloud", "product", "metrics"],  
6.   "relatedTemplates": ["tencent-cloud"],  
7.   "priority": 5,  
8.   "updatedAt": "2026-05-04"  
9. }
```

维护方式

1. V1.0: 由开发/产品手工维护 `api/_knowledge.js` 中的结构化条目。
2. V1.1: 支持用 Markdown/JSON 文件维护知识条目，降低代码修改成本。
3. V1.2: 支持简易管理页或向量知识库，允许导入公司业务资料、岗位 JD 和优秀答案样本。

验收标准

1. 新增知识条目后，Agent 训练计划、动态追问和复盘报告能命中相关知识。
2. 知识条目应避免编造公司内部信息，只沉淀公开、通用、可验证的

方法论和岗位知识。

3. 每个岗位模板至少配置 3-5 条高优先级知识。

9.12 产品调性与反馈体验

产品整体调性应保持专业、克制、鼓励和行动导向。系统可以指出问题，但不能用打击式、评判人格式或过度夸张的表达。所有反馈都应围绕“下一步怎么改”展开，避免只给用户压力。

在单轮回答分析中，反馈结构建议遵循“先定位表现 - 再指出关键问题 - 最后给出下一步动作”的顺序。低分回答也需要包含至少一句具体、真实的正向引导；高分回答则应避免泛泛夸奖，而要指出用户做对了什么，以及如何进一步贴近目标岗位。

验收标准

1. 单轮反馈不只输出分数，还应包含一句自然语言总结。
2. 低于 60 分的反馈需要先给出具体鼓励，再指出优先级最高的问题。
3. 连续两轮得分提升时，系统应提示用户进步点。
4. 同一短板连续出现时，系统应将其转化为专项训练目标，而不是重复批评。
5. 反馈文案不得出现人格否定、绝对化打击或无依据夸大。

10. 非功能需求

10.1 可用性

1. 核心训练流程应在一个页面内完成，减少跳转成本。
2. 用户无需理解 Prompt 或模型参数即可使用。

3. 关键操作需要有 loading 状态和错误提示。

10.2 稳定性

1. API 返回必须采用 JSON 结构，降低前端解析失败概率。
2. 对模型输出解析失败、网络失败、API Key 缺失等情况提供明确错误提示。
3. 本地存储异常不应阻断核心训练流程。

10.3 可解释性

1. 评分结果应展示维度和原因，而不是只给总分。
2. Agent 决策轨迹应展示训练规划、知识检索、动态追问和画像更新逻辑。
3. 复盘报告应给出证据和下一步行动建议。

10.4 隐私与安全

1. 不在前端硬编码 API Key。
2. API Key 通过环境变量读取。
3. 本地数据清空功能可用。
4. 对简历、JD 等敏感信息进行清晰提示。

10.5 可扩展性

1. 岗位模板可继续扩展到更多公司和岗位。
2. 知识库可从静态条目升级为向量数据库。
3. 评分模型可根据真实用户反馈继续校准。

4. 后续可接入账号系统，实现跨设备训练历史同步。

10.6 模型输出校验与失败兜底

模型输出必须经过后端校验后再返回前端。若 JSON 解析失败，后端先尝试从代码块或纯文本中提取 JSON；若仍失败，可使用同样上下文重试 1 次，并在前端展示“系统正在重新整理结果，请稍候”。

如果重试后仍失败，后端返回统一结构化错误，前端保留用户已输入的 JD、简历和回答，并提示“系统处理异常，请重试，刚才的回答已保留”。Debug 模式下可展示原始模型返回摘要，便于开发排查，但默认不暴露给普通用户。

模型接口超时，前端停止 loading 并允许用户重试。API Key 缺失或无效时，系统阻断模型调用，并提示检查模型服务配置。若模型输出字段缺失，前端使用默认值兜底，同时标记结果不完整并允许用户重新生成。

V1.0 后端应至少保证即使模型输出异常，前端也不会清空用户已输入的 JD、简历或回答。

10.7 模型版本与配置方式

1. V1.0 默认使用通义千问 OpenAI-Compatible API，通过 `DASHSCOPE_API_KEY`、`AI_BASE_URL`、`AI_MODEL` 环境变量配置。

2. 所有核心 API 默认使用同一模型，便于控制成本和排查问题。

3. V1.1 可按任务拆分模型，例如回答评分使用稳定低温模型，复盘报告使用更强生成模型。

4. 最小模型能力要求：支持中文理解与生成、稳定输出 JSON、长上下文输入、遵循系统提示词。

5. 如果后续切换 GPT、Claude、DeepSeek 等模型，应保持 OpenAI-Compatible 调用层或增加 provider adapter，避免前端改动。

11. 数据与指标设计

11.1 用户行为指标

1. JD 填写率。
2. 简历上传/粘贴率。
3. 训练前诊断点击率。
4. Agent 训练启动率。
5. 单场训练完成率。
6. 最终报告生成率。
7. 报告导出率。
8. 简历项目生成率。

11.2 训练效果指标

1. 用户回答平均分变化。
2. STAR 完整度提升幅度。
3. 岗位匹配度得分提升幅度。
4. 数据量化与结果意识得分提升幅度。
5. 技术/产品深度得分提升幅度。
6. 高频薄弱信号减少数量。

11.3 模型质量指标

1. JSON 输出解析成功率。
2. 报告生成成功率。
3. 评分一致性。
4. 参考答案采纳率。
5. 用户对反馈有用性的主观评分。
6. 幻觉或不可信建议出现率。

11.4 MVP 埋点与数据回收方案

V1.0 先采用轻量方案，不引入复杂数据平台，优先保证隐私透明和实现成本可控。

第一阶段建议先采用本地事件日志：在浏览器 `localStorage` 中记录匿名事件，例如模板选择、训练前诊断完成、报告生成和导出行为。该方案实现成本低、隐私风险较小，但只能在本机查看，需要用户主动导出后才能做汇总分析。

如果后续需要观察整体转化漏斗，可以新增 `\api/event-log` 匿名日志接口，只上报事件名、时间、岗位模板、结果状态等信息，不上传简历正文、JD 原文和回答原文。引入后台日志前，需要同步补充隐私说明和日志保留策略。

此外，报告页可以增加一个轻量反馈入口，让用户评价“本次反馈是否有用”。这类主观反馈样本量可能有限，但对判断参考答案、复盘报告和简历建议的质量很有帮助。

V1.0 推荐埋点

1. `template_selected`: 岗位模板选择。
2. `preflight_completed`: 训练前诊断完成。
3. `agent_plan_completed`: 训练计划生成完成。
4. `analysis_completed`: 单轮回答分析完成。
5. `next_question_generated`: 动态追问生成完成。
6. `report_generated`: 最终报告生成。
7. `report_exported`: 报告或材料包导出。
8. `local_data_cleared`: 用户清空本地数据。

埋点不得上传用户简历正文、JD 原文、回答原文和 API Key。

12. 产品架构

12.1 前端模块

1. 岗位模板选择模块。
2. JD 与简历输入模块。
3. 训练前诊断展示模块。
4. 模拟面试交互模块。
5. 语音输入模块。
6. 回答评分展示模块。
7. Agent 决策轨迹模块。
8. 最终复盘报告模块。
9. 历史报告管理模块。
10. 导出与投递材料生成模块。

12.2 后端 API 模块

1. /api/analyze: 回答评分与反馈。
2. /api/generate-questions: 普通模拟面试问题生成。
3. /api/agent-plan: Agent 训练计划生成。
4. /api/agent-next-question: 动态追问生成。
5. /api/agent-report: 最终复盘报告生成。
6. /api/preflight-diagnosis: 训练前岗位匹配诊断。
7. /api/resume-project: 简历项目经历生成。
8. /api/health: 健康检查。

12.3 核心 API 请求/响应约定

/api/analyze: 回答评分

请求示例:

```
1. {
2.   "position": "腾讯云智技术产品实习生",
3.   "question": "请介绍一个你将技术能力产品化的项目。",
4.   "answer": "用户回答正文",
5.   "templateContext": {
6.     "key": "tencent-cloud",
7.     "label": "腾讯云智 · 技术产品实习生"
8.   }
9. }
```

响应示例:

```
1. {
2.   "success": true,
3.   "result": {
4.     "score": 76,
5.     "summary": "回答结构较清楚, 但云计算客户场景和量化结果不足。",
6.     "dimensionScores": {
7.       "roleFit": 16,
8.       "technicalProductDepth": 15,
9.       "starCompleteness": 12,
10.      "quantification": 9,
11.      "structuredExpression": 13
12.     },
13.     "problems": ["缺少客户场景", "缺少量化指标"],
14.     "suggestions": ["补充目标用户和使用场景", "增加训练完成率或报告生成成功率等指标"],
```

```
15.   "referenceAnswer": "参考答案正文",
16.   "revisedAnswer": "岗位化改写版本"
17. }
18. }
```

/api/agent-next-question: 动态追问

请求示例:

```
1. {
2.   "position": "腾讯云智技术产品实习生",
3.   "jd": "岗位 JD",
4.   "resume": "简历或项目背景",
5.   "history": [
6.     {
7.       "question": "上一轮问题",
8.       "answer": "上一轮回答",
9.       "analysis": { "score": 68, "weakSignals": ["缺少量化数据"] }
10.    },
11.  ],
12.  "profile": { "averageScore": 70, "nextSessionFocus": ["数据量化"] },
13.  "templateContext": { "key": "tencent-cloud" }
14. }
```

响应示例:

```
1. {
2.   "success": true,
3.   "next": {
4.     "text": "如果要衡量这个 AI 面试训练 Agent 是否有效, 你会设计哪些指标?",
5.     "intent": "考察数据量化与技术产品评估能力",
6.     "focusArea": "数据量化与结果意识",
7.     "answerHints": ["说明指标口径", "区分用户行为指标和模型质量指标"]
8.   },
9.   "knowledge": []
10. }
```

/api/agent-report: 最终复盘报告

响应核心字段:

```
1. {
2.   "success": true,
3.   "report": {
4.     "summary": "总体复盘结论",
5.     "readinessScore": 78,
6.     "capabilityRadar": [
7.       { "name": "云计算产品理解", "score": 70, "evidence": "评分依据" }
8.     ],
9.     "strengths": ["优势"],
10.    "weaknesses": ["短板"],
11.    "recommendedActions": [
12.      { "priority": "high", "action": "补充云计算客户场景", "reason": "岗位匹配度不足" }
13.    ],
14.    "nextTrainingPlan": [
15.      { "day": "Day 1", "focus": "云计算产品理解", "tasks": ["梳理计算/网络/存储基础"] }
16.    ],
17.    "resumeAdvice": ["简历优化建议"],

```

```

18.   "agentMemory": {
19.     "persistentProfileUpdate": "长期画像更新摘要",
20.     "nextSessionFocus": ["下次训练重点"]
21.   }
22. }
23. }

```

所有 API 错误响应统一为:

```

1. {
2.   "error": "错误说明",
3.   "code": "OPTIONAL_ERROR_CODE"
4. }

```

核心状态码约定如下:

HTTP 状态码	适用场景	前端处理方式
400	请求参数缺失、回答过短、JSON 请求体无效	提示用户补充或修改输入，不自动重试
401	API Key 缺失、无效或无权限	提示“模型服务未配置或密钥无效”，引导检查环境变量
408	请求超时或用户网络中断	保留用户输入，允许用户重试
429	模型服务限流或额度不足	提示稍后再试，必要时建议切换模型或检查额度
500	服务端处理异常、模型返回解析失败且重试后仍失败	提示系统处理异常，保留用户输入并允许重新生成
503	模型服务不可用、上游服务维护或网络不可达	提示模型服务暂时不可用，建议稍后重试；前端不应误导用户修改回答

其中 401 和 503 的前端处理需要区分：401 代表配置或授权问题，用户反复重试通常无效；503 代表上游服务暂时不可用，用户输入本身没有问题，应保留现场并允许稍后重试。

12.4 大模型能力模块

1. JD 解析。
2. 简历匹配诊断。
3. 问题生成。
4. 动态追问。
5. 多维度评分。

6. 参考答案生成。
7. 复盘报告生成。
8. 简历项目经历生成。

12.4 知识库模块

当前知识库以轻量级本地文本条目实现，覆盖：

1. 云计算与技术产品。
2. AI 产品与 Agent workflow。
3. 智能驾驶与产品安全边界。

后续可升级为：

1. 向量数据库检索。
2. 岗位 JD 自动入库。
3. 公司业务资料知识库。
4. 面试题与优秀答案知识库。

13. 竞品与差异化

类型	代表产品	局限	本产品差异化
通用 AI 助手	ChatGPT、通义千问、豆包	需要用户自己设计问题和评分标准	提供岗位模板、训练流程和评分闭环
简历优化工具	简历润色类产品	偏静态文本优化	结合面试表现反向优化简历项目
题库平台	牛客、面试题库	固定题目为主，缺少动态追问	根据回答表现生成下一题
AI 面试工具	模拟面试类产品	多数缺少岗位知识和长期画像	引入 RAG、Agent 记忆和报告回看

14. 风险与应对

14.1 模型幻觉风险

风险：模型可能生成不准确的岗位知识或夸大项目结果。

应对：

1. 在 Prompt 中要求不编造真实数据。
2. 对简历项目生成增加风险提示。
3. 引入知识库检索增强，减少纯模型自由发挥。

14.2 评分主观性风险

风险：大模型评分可能存在波动。

应对：

1. 使用岗位模板驱动的评分维度。
2. 设置评分校准上限。
3. 输出分维度证据，提升可解释性。

14.3 隐私风险

风险：用户会输入简历、JD 和个人经历。

应对：

1. 明确提示数据会发送到模型 API。
2. 本地应用不做服务端持久化。
3. 提供一键清空本地数据能力。

14.4 岗位知识更新风险

风险：岗位要求、公司业务和技术趋势会变化。

应对：

1. 将知识库模块独立维护。
2. 后续支持岗位资料动态更新。
3. 对不同公司和岗位建立模板配置。

15. 版本规划

V1.0 MVP

1. 完成 JD/简历输入。
2. 完成岗位模板。
3. 完成 Agent 训练流程。
4. 完成回答评分和复盘报告。
5. 完成历史报告和本地画像。
6. 完成导出与简历项目生成。

V1.1 体验优化

1. 增加可视化 Agent 工作流图。
2. 增加报告对比功能，展示多次训练进步。
3. 增加更完整的 PRD/产品文档生成模块。
4. 优化移动端体验。

V1.2 能力增强

1. 引入向量知识库。

2. 支持用户自定义岗位模板。
3. 支持公司业务资料上传。
4. 支持更严格的评分一致性评估。

V2.0 商业化版本

1. 增加账号系统。
2. 支持跨设备历史同步。
3. 增加岗位训练课程包。
4. 接入人工导师点评。
5. 增加团队/学校/求职机构管理后台。

V1.0 发版标准

满足以下条件后，才可以认为 V1.0 达到可发布状态：

V1.0 发布前，所有 P0 功能必须通过验收，包括 JD/简历输入、训练前诊断、Agent 计划、回答评分、动态追问、最终报告和本地数据清空。主流程需要连续完成 3 场完整训练，且不出现阻断性 bug。

模型输出质量需要达到内测可用标准：JSON 解析成功率不低于 95%，报告生成成功率不低于 95%。如果出现解析失败，系统必须能保留用户输入并给出可重试提示。

数据安全方面，API Key 不得出现在前端代码、网络响应和部署文件中，`.env.local` 不得被上传。隐私能力需要完成验证：本地数据说明可见，清空本地数据后历史报告和训练画像同步刷新。

线上可用性方面，Vercel 主站和至少一个镜像站需要完成首页、健康接口、训练前诊断接口检查。降级体验方面，语音不支持、模型失败、localStorage 写入失败等情况均应有明确提示，且不丢失用户已输入内容。

文档材料方面，PRD、架构图、指标方案、竞品分析和低保真原型至少完成 Markdown 版本，便于后续导出 Word、图片或作品集材料。

16. 验收标准汇总

1. 用户可以选择岗位并填充 JD。
2. 用户可以基于 JD 和简历生成训练前诊断。
3. 系统可以生成 Agent 训练计划并开始模拟面试。
4. 针对不同岗位，问题和反馈能体现相应的能力维度。
5. 每轮回答后系统可以输出评分、问题分析、参考答案和改写建议。
6. Agent 可以根据上一轮表现生成下一轮追问。
7. 系统可以生成最终复盘报告。
8. 报告可以导出 Markdown、打印 PDF，并生成投递材料包。
9. 历史报告可以重新打开。
10. 本地训练画像可以持续更新，也可以一键清空。
11. 只输入 JD、语音识别失败、模型接口异常、中途退出等异常分支均有明确提示和降级方案。
12. 核心 API 请求/响应字段稳定，模型输出解析失败时不会丢失用户输入。
13. 评分权重、评分校准阈值和动态追问触发逻辑有明确配置或文档说明。
14. 历史报告有保留上限，localStorage 写入失败时有导出或清理提示。
15. 产品反馈符合“专业、克制、鼓励、行动导向”的调性，低分反馈不打击用户，高分反馈不空泛夸奖。

16. 语音输入具备空闲、权限请求、正在聆听、识别中、完成、失败、不支持、权限拒绝等状态提示。

17. API 错误状态码能区分参数错误、鉴权失败、限流、模型服务不可用和服务端异常，前端处理策略不同。