**武汉大学**

**计算机学院实训**

**基于AI Agent 的智能学伴系统需求说明书**

**项目成员： 李超然 丁欣雨 刘子俊**

**撰 写 人： 李超然 丁欣雨 刘子俊**  2025年 6 月 25日

**评 审 人：**  年 月 日

武汉凡诺软件技术有限公司

2025年

**修订记录**

| 日期 | 修订版本 | 修改章节 | 修改描述 | 修订人 |
| --- | --- | --- | --- | --- |
| 2025.6.25 | v1 | 全文 | 初始文档 | 李超然 丁欣雨 刘子俊 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**目 录**

[基于AI Agent 的智能学伴系统项目需求说明书](#_Toc25827)

[1. 简介](#_Toc21947)

[1.1 编写目的](#_Toc18535)

[1.2 范围](#_Toc6556)

[2. 总体概述](#_Toc4674)

[2.1 软件概述](#_Toc27571)

[2.2 项目介绍](#_Toc15526)

[2.3 产品环境介绍](#_Toc31658)

[2.4 软件功能](#_Toc3064)

[3. 功能需求](#_Toc2602)

[3.1 用例图](#_Toc31921)

[3.2 系统模块](#_Toc11895)

[4. 性能需求](#_Toc19204)

[5. 接口需求](#_Toc13807)

[6. 用户接口需求](#_Toc28907)

[7. 总体设计约束](#_Toc21614)

[8. 其他需求](#_Toc5491)

[9. 需求分级](#_Toc21734)

[10. 待确定问题](#_Toc32632)

关键词：AI Agent,知识图谱,ReAct.

摘 要：EduAssistant 是一款基于AI Agent的智能教育辅助系统，通过大语言模型、智能体和知识图谱技术，为学生提供个性化学习体验，为教师提升教学效率。系统核心功能包括智能问答、个性化学习路径规划、知识掌握度评估和学习内容管理，采用分层架构设计，以ReAct Agent为核心，结合知识图谱构建、掌握度评估、学习路径生成等关键算法，实现了教育资源的智能推荐和学习过程的精准指导。EduAssistant 致力于通过技术创新推动教育个性化和智能化发展，为学生提供全方位的学习支持，为实现教育公平和提升教育质量贡献力量。

缩略语清单：

| 缩略语 | 英文全名 | 中文解释 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 

# 简介

## 1.1 编写目的

对软件需求完全理解对于软件开发工作的成功是至关重要的，需求说明的任务是确立、规范软件开发的过程，有益于提高软件开发过程中的能见度，便于软件开发过程中的控制与管理，便于采用工程方法开发软件，提高软件的质量，便于开发人员、维护人员、管理人员之间的交流、协作，并作为工作成果的原始依据，并且在向潜在用户传递软件功能、性能需求，使其能够判断该软件是否与自己的需求相关。

本文档作为所有参与此项目的人共同工作的基础，对《基于AI Agent 的智能学伴系》（以下简称《EduAssistant》）的需求描述。通过此文，让业务人员了解项目组开发人员对《EduAssistant》需求的理解是否正确、到位。

1、本文档作为《EduAssistant》供设计人员使用，作为系统设计的依据。

2、作为项目验收标准之一。

3、软件维护的参考资料。

## 1.2 范围

1、系统核心功能

- 基于大语言模型的智能问答功能

- 个性化学习路径规划与动态调整

- 知识掌握度评估与分析

- 学习内容管理与智能推荐

2、技术架构与实现

- 以ReAct Agent为核心的分层架构设计

- 知识图谱构建与动态更新机制

- 掌握度评估算法与学习路径生成逻辑

- 基于Intranet/Internet的Web应用部署方案

3、目标与特性

- 跨平台、高安全性的智能教育辅助系统

- 支持多终端（PC/移动端）的资源共享与维护

- 通过AI技术实现教育个性化和智能化

- 为教师提供教学效率提升工具，为学生提供精准学习支持

4、文档不涵盖的内容包括：

- 非教育领域的通用AI技术细节（如通用大语言模型训练方法）。

- 硬件基础设施的具体部署方案（如服务器集群配置）。

- 非核心附加功能（如第三方支付、社交模块等）。

- 与教育政策或课程标准直接相关的非技术性内容。

5、系统边界说明：

EduAssistant聚焦于AI技术与教育场景的结合，其范围限于软件系统的功能设计、技术实现及教育应用价值，不涉及线下教学实施或课程内容创作。

# 总体概述

## 2.1 软件概述

EduAssistant是基于AI Agent的双端智能教育系统，整合Flask框架、ReAct 智能体框架，构建教师端AI教研助手与学生端个性化学习引擎。系统具备课程管理、知识库管理、学情分析和智能交互等功能，结合智能语义搜索与个性化推荐，显著提升教学效率与学习体验，助力构建"教-学-评"智能闭环。

## 2.2 项目介绍

EduAssistant是一款基于AI技术的智能教育辅助系统，旨在运用大语言模型、智能体和知识图谱等前沿技术，为教育领域提供个性化学习解决方案。在新时代教育强国建设背景下，我国仍面临教育资源分布不均、优质师资短缺等问题。虽然国家中小学智慧教育平台已初步实现资源共享，但智能化、个性化教育支持仍有提升空间。

本系统作为全新独立项目，通过智能问答、学习路径规划、知识掌握度评估等核心功能，为不同学习能力的学生提供精准辅导，同时帮助教师优化教学方案。系统采用分层架构设计，结合ReAct Agent和知识图谱技术，实现教育资源的智能匹配与推荐。

EduAssistant的研发响应了教育数字化转型需求，致力于通过技术创新促进教育公平，提升教学质量，为构建智能化教育生态提供有力支撑。

## 2.3 产品环境介绍

1.硬件环境：

● 服务器端：

支持x86架构的云服务器或物理服务器

推荐配置：8核CPU/16GB内存/500GB存储（根据用户规模可扩展）

GPU加速设备（可选，用于提升AI模型推理性能）

● 客户端：

PC端：支持Windows/macOS系统的个人电脑

移动端：支持Android/iOS系统的智能手机和平板设备

最低配置：双核处理器/4GB内存/50MB可用存储空间

2.软件环境：

● 服务器端：

操作系统：Windows

容器化：Docker + Kubernetes（可选，用于集群部署）

数据库：postgreSQL（用于结构化/非结构化数据存储）

中间件：Redis（缓存服务），RabbitMQ（消息队列）

● 客户端：

浏览器：Chrome/Firefox/Edge等主流浏览器（Web版）

运行环境：Node.js 16+/Python 3.8+（部分本地计算功能）

● AI服务依赖：

大语言模型：支持GPT/GLM等主流模型API接入

知识图谱引擎：Neo4j

● 网络环境：

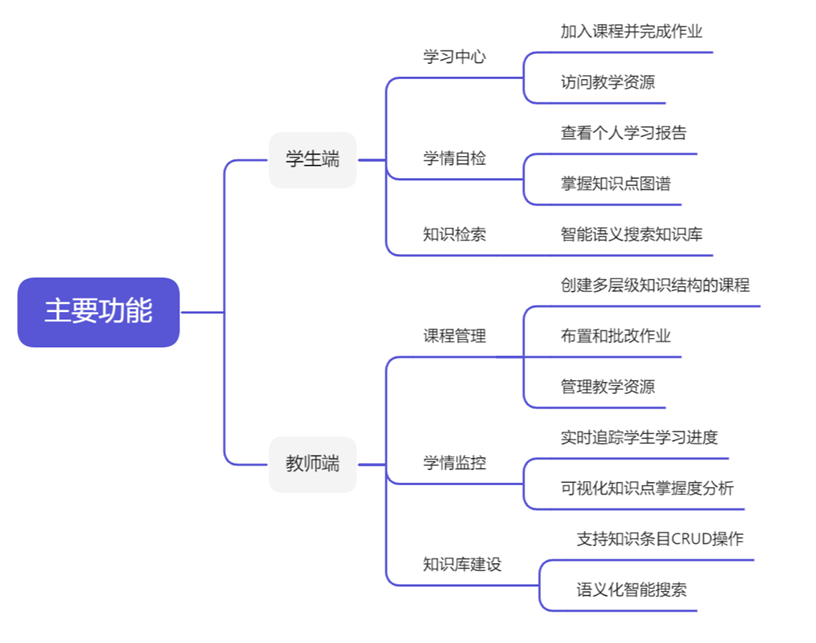
互联网接入：带宽≥10Mbps（建议百兆以上专线）

协议支持：HTTPS/WebSocket（保障通信安全）

注：系统采用B/S架构为主，兼容C/S混合部署模式，所有环境组件均支持国产化替代方案。

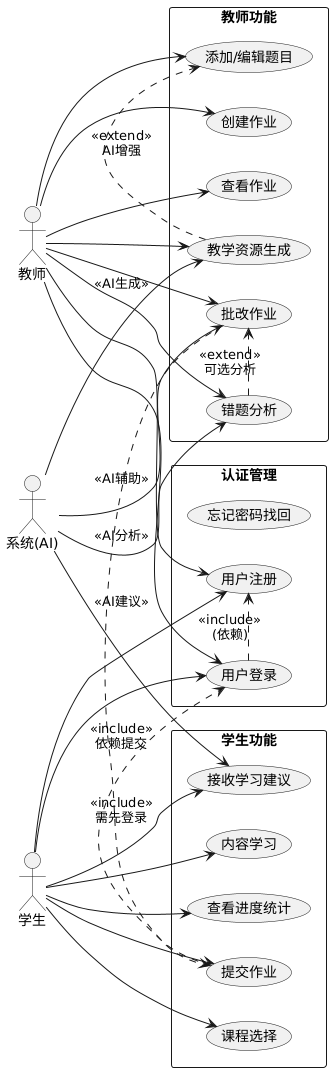
## 2.4 软件功能

EduAssistant是一个基于AI Agent的智能教学辅助系统，主要面向教师和学生两类用户，提供差异化的功能支持，其详细功能如图所示：



# 功能需求

## 3.1 用例图



**Use Case 1: 用户注册**

**Goal in Context**

允许新用户通过填写基本信息完成系统注册，建立个人学习账户。

**Preconditions**

* 用户未注册过相同账号。
* 系统注册功能可用。

**End Conditions**

* Success End Condition：账号创建成功，用户信息加密存储。
* Failed End Condition：注册失败，提示格式错误或账号重复。

**Actors**

学生,教师

**Trigger**

用户点击“注册”按钮并提交表单。

**Description**

* 用户输入用户名、密码、联系方式（需符合格式校验）。
* 系统验证字段格式（如邮箱正则匹配）及账号唯一性。
* 密码通过加密算法（如BCrypt）存储至数据库。
* 返回注册成功提示，跳转至登录页。

**Use Case 2: 用户登录**

**Goal in Context**

验证用户身份并授权访问系统功能**。**

**Preconditions**

* 用户已完成注册。
* 系统会话服务正常。

**End Conditions**

* Success End Condition：生成有效会话Token，跳转至主页。
* Failed End Condition：登录失败，保留未授权状态。

**Actors**

学生,教师

**Trigger**

用户提交登录表单。

**Description**

* 用户输入用户名和密码。
* 系统比对加密存储的密码哈希值。
* 验证通过后生成JWT Token，记录登录时间。
* Token用于后续接口鉴权（如课程访问）。

**Use Case 3: 课程选择**

**Goal in Context**

学生浏览并选择课程加入学习计划。

**Preconditions**

* 用户已登录。
* 系统存在可用课程资源。

**End Conditions**

* Success End Condition：课程关联至用户账户，更新可选列表。
* Failed End Condition：选择失败（如课程已满）。

**Actors**

学生

**Trigger**

* 学生点击“加入课程”按钮。Description
* 系统展示课程列表（名称、简介、难度标签）。
* 学生选择目标课程，确认加入。
* 系统关联课程与用户账户，初始化学习进度为0%。

**Use Case 4: 作业提交与AI评估**

**Goal in Context**

学生提交作业后触发AI自动评分与反馈生成。

**Preconditions**

* 学生已登录并选择课程。
* 作业未过截止时间。

**End Conditions**

* Success End Condition：作业状态变更为“已评估”，生成报告。
* Failed End Condition：评估异常（如AI服务超时）。

**Actors**

学生,系统（AI）

**Trigger**

* 学生提交作业文件或文本。Description
* 学生上传作业（支持PDF/DOCX格式）。
* 系统调用AI模型解析内容，对比参考答案。
* AI生成评分（百分制）及错题标记（如“三角函数错误”）。
* 反馈报告存入数据库，学生端可见。

**Use Case 5: 学习进度管理**

**Goal in Context**

系统跟踪学习行为并提供进度提醒。

**Preconditions**

用户至少选择一门课程。

**End Conditions**

Success End Condition：进度数据实时更新，触发提醒（如邮件）。

Failed End Condition：进度同步延迟（网络问题）。

**Actors**

系统

**Trigger**

* 用户完成章节学习或作业。Description
* 系统记录学习时间、章节完成率。
* 计算整体进度（如“65%已完成”）。
* 临近截止时推送提醒（如“作业剩余2天”）。
* 数据可视化展示（折线图/进度条）。

**Use Case 6: 创建作业**

**Goal in Context**

教师通过系统发布新作业，设定题目和要求。

**Preconditions**

* 已登录且具有权限。
* 至少存在一个可用的课程或班级。

**End Conditions**

* Success End Condition：作业创建成功，学生端可见。
* Failed End Condition：作业未保存，系统提示错误原因（如标题为空）。

**Actors**

教师

**Trigger**

* 教师点击“创建作业”按钮。Description
* 教师填写作业标题、描述、截止时间。
* 系统验证必填字段完整性。
* 教师添加题目（可手动输入或从题库选择）。
* 系统保存作业至数据库并关联班级课程。

**Use Case 7: 错题分析**

**Goal in Context**

教师或系统AI分析学生错题分布，生成针对性教学建议。

**Preconditions**

* 至少存在一份已批改的作业。
* 错题数据完整。

**End Conditions**

* Success End Condition：生成可视化报告（如知识点薄弱项）。
* Failed End Condition：分析失败，提示数据不足。

**Actors**

教师,系统（AI）

**Trigger**

教师点击“错题分析”或定期自动任务。

**Description**

* 系统聚合班级错题数据。
* AI识别高频错误知识点。
* 生成图表和建议（如“30%学生未掌握三角函数”）。
* 教师可导出报告或调整教学计划。

## 3.2 系统模块

* **学生端功能**

|  |  |  |  |
| --- | --- | --- | --- |
| 主模块 | 子功能 | 描述 | 输入 |
| 学习中心 | 加入课程 | 绑定学生与课程关系 | 课程ID/课程邀请码 |
| 完成作业 | 提交作业内容 | 作业文件/在线答题数据 |
| 访问教学资源 | 获取课程知识内容 | 资源ID/搜索关键词 |
| 学情自检 | 学习报告 | 多维度学情分析 | 学生ID+时间范围 |
| 知识点图谱 | 可视化知识掌握路径 | 当前学科章节 |
| 知识检索 | 智能语义搜索 | 跨资源库内容检索 | 自然语言查询语句 |

* **教师端功能**

|  |  |  |  |
| --- | --- | --- | --- |
| 主模块 | 子模块 | 描述 | 输入 |
| 课程管理 | 创建课程 | 构建课程知识体系 | 课程大纲/章节层级结构 |
| 布置作业 | 发布学习任务 | 题库选择/自定义题目 |
| 批改作业 | 人工复核AI批改结果 | 学生提交的作业ID |
| 管理教学资源 | 维护课程资源库 | 资源文件/外部链接 |
| 学情监控 | 学习进度追踪 | 实时监控班级学习状态 | 班级ID/时间区间 |
| 知识点掌握度分析 | 可视化班级知识短板 | 知识点ID/测评数据集 |
| 知识库建设 | 知识条目CRUD | 维护底层知识图谱 | 知识实体/关系/属性 |
| 语义化智能搜索 | 精准定位知识库内容 | 多条件复合查询 |

# 性能需求

系统的核心性能需求主要有：

|  |  |  |
| --- | --- | --- |
| **操作类型** | 要求响应时间 | 说明 |
| **常规操作** | ≤3秒 |  |
| 用户登录认证 | ≤3秒 | 身份验证及会话初始化 |
| 作业列表/错题本加载 | ≤3秒 | 包含基础数据渲染 |
| 资源库检索 |  | 关键词匹配与分页加载 |
| **AI核心功能** |  | *需异步队列处理* |
| 作业智能批改 | ≤15秒 | NLP解析+标记 |
| 错题深度分析 | ≤20秒 | 知识图谱推理+错因报告 |
| 个性化资源生成 | ≤30秒 | 生成式AI模型推理 |
| 复杂知识图谱查询 | ≤8秒 | 跨节点语义检索 |

## 4.1 静态需求

静态的量化需求包括：

1. 最大注册用户数：系统应支持至少 5000 个注册用户账户。(理由：适用于学校或中等规模培训机构用户群体，预留合理增长空间)
2. 最大活动知识库条目：系统底层知识图谱及资源库应能有效管理至少 100,000 道覆盖核心学科的习题及其关联知识点、错题标签。(理由：支撑精准错题分析所需的知识基础规模)
3. 用户历史记录容量：单个活跃用户账户应能存储至少 6个月 的作业记录（包括原始文件、批改结果、分析报告）、错题集（包含详细的错因分析）及系统生成的个性化资源历史。(理由：保证学生有足够历史数据用于学习分析跟踪)
4. 文件上传限制：

* 单个作业上传文件（支持PDF, DOCX, JPG, PNG等常见格式）最大大小：10MB (理由：平衡用户体验与存储成本，大多数作业文档/图片在此限制内)
* 单用户总存储空间配额（包含作业文件、错题集附件、生成资源等）：1GB (理由：基于用户活跃度与内容积累速度设定合理阈值)

## 4.2 动态需求

动态的量化需求包括：

**响应性能**

1. 用户界面核心操作响应时间：

* 页面加载/导航 (静态内容)：< 2 秒
* 用户登录认证：< 3 秒
* 作业列表加载、错题本浏览、资源库预览：< 3 秒

2. 核心智能处理操作响应时间： (在典型云端推理资源环境下) 智能作业批改 (含基础错题标记)： 对于标准单页作业（约10-20道题目），处理时间应 < 15 秒。(理由：分析需要时间，需及时反馈)

* 精准错题分析 (基于知识图谱推理)： 对单次作业中标记的错题（一次处理≤10题），生成详细错因报告时间 < 20 秒。(理由：涉及知识节点关联、错因模式匹配等复杂推理)
* 个性化学习资源自动生成 (文本/题目)： 生成单一类型资源（如一组5道针对性练习、一个知识点总结文档草稿）时间 < 30 秒。(理由：生成式AI模型推理时间开销较大，需设定合理期望)
* 复杂查询响应：知识图谱高级检索、资源库组合筛选： < 8 秒。

**并发与吞吐量**

1. 最大在线用户数：系统支持300位用户同时在线。(理由：估算班级或年级同时使用的峰值)
2. Web端并发用户数：

* 常规操作（浏览、简单查询）平均并发：支持100用户。
* 峰值时段（如晚间作业提交/考试后分析高峰）关键操作并发：
* 同时处理作业上传与提交操作：至少支持50个并发请求。
* 同时进行错题分析请求：至少支持20个并发分析任务。
* 同时请求资源生成：至少支持10个并发生成任务。(理由：AI处理是计算密集型资源消耗大项，需重点限制并发以保障整体稳定性和单个任务性能)

1. AI任务队列吞吐量: 在峰值负载时，系统处理队列中的作业批改任务速率应不低于20份作业/分钟，错题分析任务不低于15份分析报告/分钟。(理由：量化系统在高负载下的整体处理能力)

# 接口需求

## 5.1 软件接口

(1)数据库接口

a.系统使用 PostgreSQL 作为主数据库，通过 Peewee ORM 实现数据操作触发条件

b.数据库连接配置包括主机地址、端口、用户名、密码及数据库名称，通过环境变量管理。

c.支持 ChromaDB 用于增强检索能力，实现教学资源的语义搜索功能。

(2)AI 模型接口

a.集成大语言模型（如 Deepseek 系列）作为 AI Agent 的核心推理引擎，通过 API 调用实现自然语言交互。

b.支持 ReAct 框架，动态调用外部工具（如 Google 搜索 API）获取实时数据。

(3)前端接口

a.后端基于 Flask 框架提供 RESTful API，前端通过 HTTP 请求与后端交互。

b.使用 Jinja2 模板引擎动态渲染 HTML 页面，实现前后端数据无缝衔接。

(4)第三方服务接口

a.支持 OAuth 2.0 协议，未来可扩展集成第三方登录（如微信、Google 账号）。

b.预留接口支持与学校教务系统对接，同步课程和学生数据。

## 5.2 硬件接口

(1)服务器配置：

a.最低配置：4 核 CPU，8GB 内存，100GB 存储（支持 SSD）。

b.推荐配置：8 核 CPU，16GB 内存，200GB 存储（SSD）。

(2)客户端要求：

a.支持主流浏览器（Chrome、Firefox、Safari），无需额外硬件支持。

b.网络协议：使用 HTTPS 协议保障数据传输安全，支持 WebSocket 实现实时通知功能。

# 用户接口需求

(1)教师端界面

a.课程管理：支持创建、搜索、分类和删除课程（优化点 1）。

操作示例：点击“创建课程”按钮 → 填写表单 → 提交后显示成功提示。

b.资源上传：支持上传文本、PDF、视频等课程资源（优化点 2）。

操作示例：拖拽文件至上传区域 → 系统显示上传进度 → 完成后资源列表更新。

c.智能批改：在作业详情页提供一键批改功能（优化点 3）。

操作示例：点击“智能批改” → AI Agent 分析答案 → 显示批改结果与评分。

1. 学生端界面

a.错题分析：通过对话形式调用 AI Agent 生成分析报告（如图 3.10）。

操作示例：输入“分析我的错题” → 系统返回薄弱知识点及学习建议。

b.资源推荐：结合语义搜索与全文搜索展示学习资源（优化点 4）。

操作示例：输入“推荐微积分资料” → 返回图文、视频等资源链接。

(3)通用界面

知识图谱可视化：以图数据库（如 Neo4j）展示课程知识点关联。

操作示例：点击“查看知识图谱” → 显示交互式图谱，支持节点展开与搜索。

# 总体设计约束

(1)技术栈限制

a.后端：后端必须使用 Python 3.10+ 和 Flask 框架，确保与现有代码兼容。

b.数据库：数据库仅支持 PostgreSQL，禁止使用非关系型数据库存储核心业务数据。

(2)性能约束

a.智能体：AI Agent 响应时间需控制在 5 秒内，复杂任务可异步处理。

b.并发访问系统需支持 50 名用户并发访问，平均负载低于 70%。

(3)安全约束

a.密码加密：用户密码需加密存储（如 bcrypt），禁止明文传输。

b.接口：所有 API 接口需实现 JWT 鉴权，防止未授权访问。

(4)扩展性约束

a.预留接口：预留接口支持未来接入更多 AI 模型（如 GPT-4）。

b.知识图谱：知识图谱模块需兼容主流图数据库（Neo4j、NebulaGraph）。

# 其他需求

(1)数据库需求

a.新增 knowledge\_graph 表存储知识点关联（优化点 5），字段包括 node\_id、node\_name、relation\_type。

b.新增 resource\_upload 表记录课程资源（优化点 2），字段包括 file\_id、course\_id、file\_type（文本/PDF/视频）。

(2)算法需求

a.知识点掌握度算法（优化点 6）：

公式：掌握度 = (正确题数 / 总题数) \* 权重 + (学习时长 / 标准时长) \* 权重。

实现：通过 Peewee 聚合查询计算，结果写入 knowledge\_mastery 表。

(3)错误处理

a.AI 服务异常：若模型调用超时，自动降级返回缓存结果或提示“系统繁忙”。

b.文件上传失败：限制文件类型和大小（如 ≤100MB），错误时提示具体原因。

(4)测试需求

a.语义搜索测试：验证向量搜索与全文搜索的混合排序效果（优化点 4）。

b.知识图谱导入测试：检查图数据库中的数据一致性，确保节点和边关系正确（优化点 5）。

(5)部署需求

a.环境依赖：服务器需安装 Python 3.12、PostgreSQL 14、ChromaDB 及 Neo4j（可选）。

b.使用 Docker 容器化部署，配置文件包含 docker-compose.yml。

# 需求分级

|  |  |  |  |
| --- | --- | --- | --- |
| 需求ID | 需求名称 | 需求分级 | 说明 |
| REQ-001 | 用户登录与注册 | A | 系统核心功能，无此功能则无法使用 |
| REQ-002 | 课程管理（创建、删除、搜索） | A | 教师端核心功能，影响教学流程 |
| REQ-003 | 作业管理（布置、提交、批改） | A | 教学核心环节，直接影响学习体验 |
| REQ-004 | AI Agent 错题分析与学习建议 | B | 提升个性化学习，但可手动替代 |
| REQ-005 | 教学资源生成（文案、大纲、题目） | c | 提高教师效率，但非必需 |
| REQ-006 | 课程资源上传（文本、PDF、视频） | B | 优化资源管理，但基础功能仍可用 |
| REQ-007 | 智能批改作业 | B | 提高效率，但教师可手动批改 |
| REQ-008 | 知识图谱与图数据库集成 | C | 增强知识关联，但非核心功能 |
| REQ-009 | 混合搜索（语义+全文） | C | 优化搜索体验，但基础搜索仍可用 |
| REQ-010 | 知识点掌握度算法 | B | 提升个性化推荐，但可简化计算 |

# 待确定问题

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 需求ID | 问题描述 | 影响  (H/M/L) | 风险 | 责任人 | 解决日期 | 状态(Open/Close) |
| PRO-001 | 作业批改AI准确率基准值（需明确学科的最小可接受阈值） | H | 准确率<90%将导致用户流失 | 研发组长 | 2025/7/10 | Open |
| PRO-002 | 与学校教务系统（ClassIn/Moodle）的API对接规范 | M | 未获授权将导致数据孤岛 | 对接项目经理 | 2025/7/10 | Open |
| PRO-003 | 错题分析的人工修正数据如何回流训练模型？ | M | 缺失修正功能将降低AI进化能力 | 研发组长 | 2025/7/10 | Open |