

# AI 行前实战营培训考核项目

## XU-News-AI-RAG：个性化新闻智能知识库

### 一、功能要求：

1. 实现定时任务机制，通过 RSS、网页抓取及智能代理工具获取新闻信息（需遵守网络爬虫规范与安全准则）。
2. 基于 Ollama 部署大模型，推荐采用 qwen2.5::3b 模型。
3. 构建本地知识库系统，配置嵌入模型（推荐使用 all-MiniLM-L6-v2）、重排模型（推荐使用 ms-marco-MiniLM-L-6-v2）及大语言模型。
4. 通过 API 将抓取的信息写入知识库系统，支持结构化数据（可通过 Excel 整理）与非结构化数据类型。
5. 信息入库成功后，自动发送邮件提示（标题及内容自定义）。
6. 提供用户登录功能。
7. 登录后支持知识库内容管理：查看数据列表（可按类型/时间筛选）、执行单条或批量删除操作、编辑元数据（如标签、来源），并支持通过页面上传多种类型数据至知识库。
8. 登录后提供语义查询功能：基于用户提问优先检索知识库内容，结果按相似度排序返回。

9. 若知识库未匹配到相关数据，自动触发联网查询（如调用百度搜索 API），返回前 3 条结果经大语言模型推理后输出。

10. 登录后提供知识库数据聚类分析报告，展示关键词 Top10 分布。

## 二、技术要求：

1. 技术栈选型：前端采用 React 或 Vue 框架，后端采用 SpringBoot 或 Flask 框架，关系型数据库选用 MySQL 或 SQLite，向量数据库采用 FAISS。

2. 数据存储设计：元数据（如数据 ID、数据类型等）存储于关系型数据库，向量数据单独存储于 FAISS 向量数据库，实现数据分类管理与高效检索。

3. 框架集成：核心业务逻辑开发采用 LangChain 框架，支撑知识库构建与检索增强功能。

4. 大模型调用：通过标准化 API 接口实现与大模型服务的交互，确保服务调用的规范性与可扩展性。

5. 身份认证：登录功能优先采用 Spring Security 框架或 JWT 技术方案，保障用户身份验证的安全性与可靠性。

## 三、提交要求：

1. 项目资料提交要求：将本项目全部相关资料上传至 GitHub 仓库，仓库命名为“xu-ai-news-rag”，并将仓库地址提交至 Ida 老师。

2. 需提供产品需求文档 ( PRD )。
3. 需提供概要设计文档与技术架构文档。
4. 需提供产品原型设计文件。
5. 需提供前端与后端完整代码。
6. 需提供单元测试、集成测试及 API 测试相关代码。
7. 若项目涉及关系型数据库操作，需提供对应的 SQL 语句。
8. 需提供项目说明文档 ( README.md )，明确项目部署、运行及使用方式。
9. 可选：准备项目介绍文档进行技术分享。

## 四、PRD 模版

1. 引言 (背景、目标用户、产品愿景)
2. 用户故事与场景描述
3. 产品范围与功能列表 (Scope & Features)
4. 产品特定 AI 需求
  - 4.1. 模型需求 (功能、性能指标)
  - 4.2. 数据需求 (来源、数量、质量、标注)
  - 4.3. 算法边界与可解释性
  - 4.4. 评估标准
  - 4.5. 伦理与合规
5. 非功能性需求 (性能、安全、可用性等)
6. 发布标准与衡量指标
7. 待定项与未来规划

