

智能学习日用 App

学生学习与日常管理工具 · 选题文档

团队名称：耍起最队 | 课程：202601 福大 · 软件工程实践 W 班 | 团队规模：8 人

一、拟作的团队项目选题描述（一句话）

智能学习日用 App (StudyDay) 是一款把课表、学习任务、AI 学习问答、个人笔记和专注记录连接起来的学生效率工具，让用户从“知道要学什么”顺利走到“完成并记住了什么”。

二、项目背景

大学生需要同时处理课程、作业、复习和生活事务，但课表、待办、问答和笔记通常分散在多个工具中。课表能够展示课程，却不能直接生成作业计划；待办事项缺少课程上下文，需要重复填写课程名称和截止时间；AI 问答产生的有价值内容又难以整理，过一段时间就难以检索。

- ☐ 信息分散：课程安排、任务和知识记录彼此割裂，用户需要反复切换应用。
- ☐ 计划脱节：课程详情不能直接转成作业或复习任务，学习计划缺少上下文。
- ☐ 知识难以沉淀：问答内容缺少结构化整理，重要解释不容易再次找到。
- ☐ 执行难以复盘：用户无法在同一条任务链上查看完成状态和实际专注时间。

智能学习日用 App 针对这些问题，提供从“课程 → 任务 → 学习帮助 → 笔记 → 复盘”的连续流程，让学习安排和实际执行能够在同一个空间中完成。

三、核心功能

- 课表显示与管理：支持周视图、日视图、课程详情、手动添加、课程冲突提示和课前提醒。
- 智能待办与备忘录：支持学习和生活任务、分类、优先级、截止时间、搜索、筛选和完成状态。
- AI 学习助手：支持多轮问答、概念讲解、例题分析、复习提纲、学习计划和错题分析。
- 对话转笔记：选择 AI 对话内容，生成标题、摘要和标签，确认后保存为可编辑笔记。
- 笔记知识库：支持 Markdown 编辑、标签、课程归档、全文搜索、自动保存和版本记录。
- 专注计时：支持番茄钟和自定义时长，结束后将专注记录关联到任务。
- 学习统计与提醒：展示任务完成率、专注时长、笔记数量，并提醒临近课程和截止日期。

四、NABCD 分析

N —— Need（需求）

核心痛点	描述
信息分散	课表、任务、问答和笔记相互分散，用户需要重复录入信息。
计划脱节	课程详情不能直接转成作业或复习任务，学习计划缺少课程上下文。
知识难以沉淀	AI 对话内容难以整理，重要解释不容易再次找到。
执行难以复盘	计划和实际投入缺少关联，用户不知道自己在任务上花了多少时间。

A —— Approach（做法）

关键功能	技术与实现	目标
课程联动	以课程为入口建立课表模块，课程详情一键创建关联任务。	减少重复录入，让任务带有课程上下文。
任务闭环	任务连接截止日期、优先级、提醒和专注计时。	让计划能够被执行和复盘。
AI 学习助手	独立 AI 适配层负责模型调用、上下文裁剪和错误处理。	提供连续、可追问的学习帮助。
知识沉淀	Markdown 笔记支持标签、搜索、版本记录和对话导入。	把一次问答转化为可复习知识。
离线可用	本地缓存最近课表和笔记，网络恢复后再同步。	降低临时无网场景的使用障碍。

B —— Benefit（益处）

目标群体	核心益处	预期指标
在校学生	减少多个工具之间的切换，快速了解课程和待办。	查看当天安排的主要操作控制在 3 次点击内。
复习用户	将 AI 问答和课堂资料沉淀为可搜索笔记。	对话转笔记流程可以在 1 分钟内完成。
效率提升用户	通过专注计时和统计看到实际投入。	每条专注记录都能关联到任务。
团队成员	练习完整的软件工程过程。	覆盖需求、设计、开发、测试、部署和复盘。

C —— Competitors（竞争）

现有工具类型	局限	本项目优势
独立课表工具	能展示课程，不能直接形成任务和笔记。	课程详情可创建关联任务，并支持课前提醒。
独立待办工具	缺少课程上下文和学习内容。	任务与课程、笔记和专注记录关联。
通用 AI 问答工具	对话内容容易流失，缺少学习记录。	AI 对话可以整理为带标签的个人笔记。
独立笔记工具	需要用户手动整理来源和学习计划。	笔记可追溯到课程、任务和对话。

D —— Delivery（交付）

阶段	关键行动	目标
试用	由团队成员和同专业同学试用课表、任务和笔记闭环。	发现主要交互问题，验证核心需求。
迭代	根据试用反馈修复高优先级缺陷，补充提醒和统计。	形成稳定的候选版本。
展示	通过项目仓库、演示视频和班级社区介绍项目。	展示完整的工程过程和可运行功能。
扩展	后续增加图片输入、课程导入和多设备同步。	服务更多校园学习场景。

五、本期范围（做什么 / 不做什么）

本期必须完成（MVP）：

- ☐ 移动端基础导航、用户登录和个人数据隔离。
- ☐ 课表手动添加、周视图、日视图、详情查看和课程提醒。
- ☐ 待办增删改查、分类、优先级、截止时间和完成状态。
- ☐ 笔记新建、编辑、Markdown 预览、标签和关键词搜索。
- ☐ AI 文字对话、上下文管理和对话转笔记。
- ☐ 可关联任务的专注计时，以及统一的加载、空状态、错误提示和重试交互。

本期明确不做：

- ☐ 图片搜题和语音输入。
- ☐ 多设备实时同步。
- ☐ 学习社区、公开笔记和多人协作。
- ☐ 复杂的自动课表解析和第三方教务系统对接。

六、技术方案

层级	技术方案
前端	移动端优先的 Web / 跨平台方案，负责页面、导航、表单、状态和适配。
后端	Java 后端服务，提供用户、课表、任务、笔记和会话 API。
数据库	MySQL，保存用户、课程、任务、笔记、标签、会话和专注记录。
缓存 / 中间件	Redis，按实际部署需要缓存热点课表、会话状态和提醒任务。
AI 接入	独立适配层封装模型调用、上下文和错误处理。
文件与导出	支持 Markdown 文件导出，图片和附件预留对象存储接口。
部署与协作	云服务器和 Nginx；采用 Git 分支、任务看板、代码评审和周版本。

七、执行计划（学期）

阶段	周次	阶段目标	主要交付	主责
立项与原型	第 1—2 周	需求梳理、用户故事、原型和技术选型	需求文档、原型图、接口初稿	PM / 全体
基础框架	第 3—4 周	登录、导航、课程和任务数据模型	可运行的基础框架	前端 / 后端
核心功能	第 5—7 周	课表、待办、提醒和专注计时	学习计划闭环	前端 / 后端
知识沉淀	第 8—9 周	笔记编辑、搜索、标签和版本记录	知识库模块	前端 / 后端
AI 助手	第 10—11 周	多轮对话、上下文和对话转笔记	AI 学习助手候选版本	算法 / 后端 / 前端
测试与交付	第 12—14 周	联调、兼容性测试、性能优化和部署	测试报告、部署版本	测试 / 全体
展示与复盘	第 15—16 周	试用反馈、缺陷修复和答辩准备	演示材料、项目总结	全体

阶段硬门槛：第 7 周

如果“课表—任务—专注计时”的闭环尚未跑通，则暂停扩展功能，优先保证核心流程可用。

八、风险分析与应对

风险	可能影响	应对措施
AI 服务不稳定或额度受限	对话功能无法演示	独立封装 AI 接口，准备模拟回答和本地示例数据。
课程数据格式不统一	导入结果错误	首期提供手动录入和模板导入，导入后由用户确认。
功能范围过大	核心功能无法按期交付	冻结 MVP 范围，新增功能需要团队确认。
团队成员时间冲突	任务延期	任务拆小，设置替补负责人，提前同步阻塞。
用户隐私风险	个人数据泄露	最小化采集、权限隔离、日志脱敏和数据删除入口。
移动端适配不足	不同设备体验不一致	尽早在真实手机尺寸上进行连续回归测试。
网络异常	页面或 AI 服务不可用	提供离线状态、重试按钮和本地演示数据。

九、团队愿景

我们希望把“智能学习日用 App”做成大学生每天愿意打开的学习伙伴，而不是功能堆叠的工具集合。项目初期将扎实完成课表、任务、AI 学习助手、笔记和专注记录的闭环，让用户能够从查看课程开始，完成计划、获得帮助并留下可复习的知识。随着使用反馈积累，我们计划支持更多课程导入方式、跨设备同步和个性化学习统计，在保证隐私和可控性的前提下，让应用服务于真实校园场景。团队也希望通过这次项目形成规范的需求、开发、测试和复盘习惯，把每一次迭代都变成可验证、可改进的工程实践。

十、团队分工

团队成员先根据职责自选表填写第一、第二意向，最终分工以团队讨论确认为准。下表列出当前技术意向，便于安排协作。

成员	当前技术 / 角色意向	可参与方向
黄思哲	Python、PyTorch、AI 算法	AI 学习助手、模型评估
胡加乘	编码、后端开发	后端接口、业务逻辑
郑宇澄	Java 后端、数据库、UML	后端、数据库、系统设计
范敬翔	测试、用例、缺陷记录	测试与质量保障
杨伟烨	前端、查缺补漏	前端页面、交互检查
黄思扬	前端、AI 辅助编程	前端页面、公共组件
赵子龙	前端	前端页面、移动端适配
柯镇海	前端、PM，组长	项目管理、产品、前端协作