

学海伴航——教辅平台设计草案

- 提案人：
 - 郑广懿
 - 朱雄伟
 - 廖鹏飞
- 设计主题：学海伴航——教辅平台
- 提案时间：2024.9.28

设计背景

需求调研

通过积极调研新生需求，了解学院教学需要，同时与辅导员老师沟通，我们发现传统的微信QQ群学习小组的答疑教辅模式已不再高效。建立功能更丰富，交互更便捷，上手更简单，连接更紧密的教辅平台需求迫在眉睫。

以下为传源书院计划开发的微信答疑小程序：

系统部



新设部门！
等你前来！
大展身手！



前端：

- 设计和实现小程序的界面
- 实现响应式布局
- 处理用户输入、动态效果、动画等

中端：

- 设计和开发小程序的API接口
- 与前端、后端对接数据

后端：

- 实现业务逻辑、数据验证、安全控制
- 管理数据库，保证数据安全和一致性
- 处理服务器部署与维护



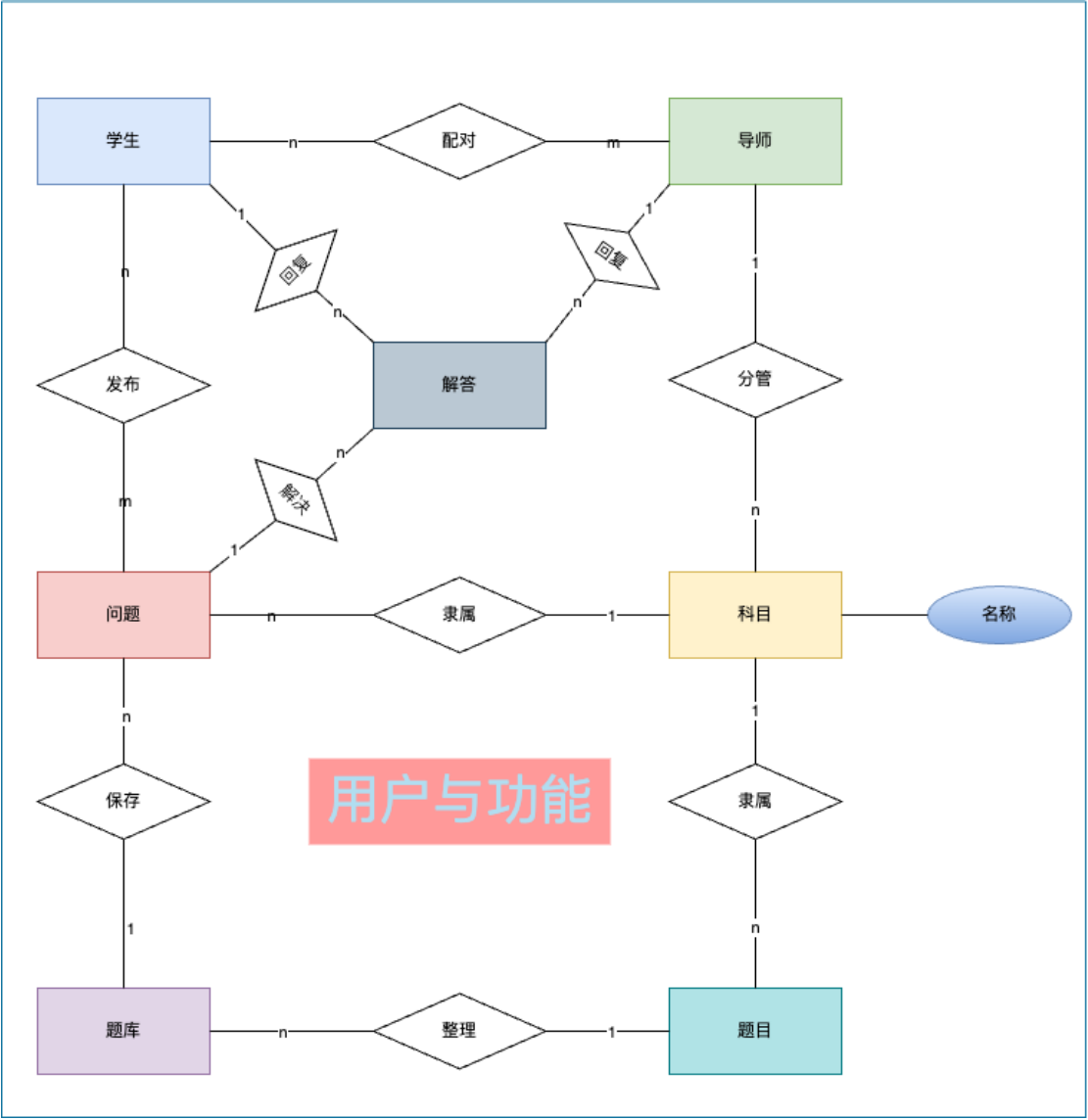
助力全面发展

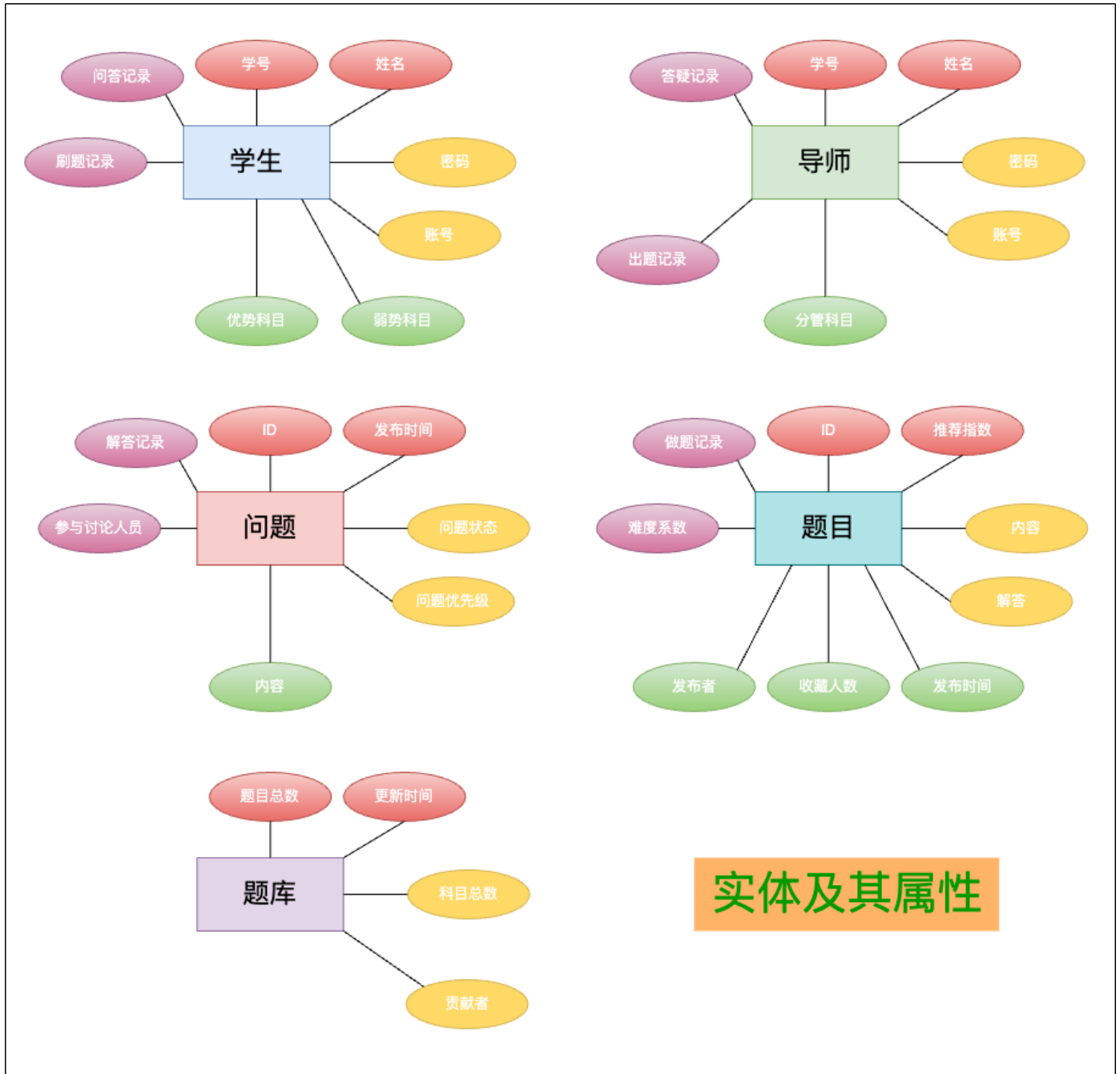
初步设计

我们小组中有两名同学曾担任助教，

通过细化分析学生答疑需求，结合自身辅导学生，整理课程资料，对接老师的相关经验，我们精简提炼了重点功能，通过初步画出 E-R图，将各实体及属性与功能间建立联系，最终完成了草案设计。

E-R图：





设计分析

我们采用软件工程中用于项目分析的NABCD五个维度，分析我们的设计提案。

N (Need, 需求)

现有问题：

1. 答疑信息集中在微信聊天记录中，高度序列化、相对不公开，增加了问题检索的工作量。
2. 学生出于各种原因不愿意实名提问。
3. 微信答疑可能会打扰辅导员日常生活。
4. 活动组织者不方便统计辅导员的工作量、发布志愿时长。

对于需要获取学业帮扶的学生

他们需要一款产品，能够：

- 提出课业上的问题，并获得及时准确的回答。
- 看见其它同学的答疑内容。
- 允许一定次数的匿名提问。
- 支持针对答疑内容的检索。
- 有条理地组织答疑数据。
- 对辅导师的回答给予反馈。

对于小导师

他们需要一款产品，能够：

- 认领同学提出的问题并作出回答。
- 即时地了解自己的答疑工作量。
- 获得同学们对回答的反馈。
- 避免对日常生活的过度打扰。
- 避免过度同质化的提问。

对于学业支持中心/导员

他们需要一款产品，能够：

- 直观了解辅导师工作情况，方便监督。
- 自动统计辅导师的工作量并换算为志愿时长。
- 批量注册学生的账号。
- 开展学习打卡等分享活动。

A（Approach，做法）

为了满足上述的需求，我们需要实现如下功能：

1. 通过类似 **CO OO OS** 讨论区提问的形式管理每个问题，同时支持问题认领，以提供“一对一”的答疑环境
2. 搭建基于标签的多粒度问题检索（例如以学科为第一级、以课本章节为第二级）
3. 题库：由学习部管理，整理上传历年真题，并每年更新。
4. 招募志愿者的栏目。
5. 加入小导师的介绍。
6. 学生评价小导师的功能。

B（Benefit，好处）

通过我们的平台，三类目标用户可以在多种场景下获得利好

从学生的角度

- 共享学习资料：平台统一整合学习资料，方便同学查阅下载。
- 共享问题与解答：所有问题均公开共享，同学可以搜索自己困惑的相似的问题，从其他人的已解决问题和回答得到启发。
- 匿名提问：参加答疑的部分同学不主动提问，不主动寻求帮助，这部分同学可以选择匿名提问减少提问的心理压力与负担，提升提问的主动性，积极性。
- 激励学习：可视化自己提出的问题数量，其他同学对这个问题的点赞数，学习更有动力，更有积极性。
- 减少微信私聊数量：学生反馈群聊数量过多的问题，一个线下答疑活动需要建立很多的群聊，大群、小组群、互助群，影响日常的消息接收，带来不必要的麻烦。通过平台进行信息整合，交流与信息统一化。

从小导师角度

- 辅导积极性更高：基于issue的解决数分配志愿时长，奖励机制更加公平合理，增强小导师的解答积极性。
- 引入值班制，合理化时间安排：鼓励有能力的小导师分时间段进行值班，每个时间段由一个或多个固定的小导师来解答问题，避免占用小导师过多的个人时间。

从学院组织者角度

- 活动推进：可视化活动活跃度情况，学生参与度情况，有效推进活动开展。
- 志愿时长分配：主要基于issue的解决问题数量进行志愿时长分配，时长分配更加公平合理，对辅导员起到激励作用。
- 氛围营造：通过平台统一管理活动，定期发布相关学习活动公众号推文，营造良好的学习氛围。

C（Competitors，竞争）

通过调研目前的学院书院答疑平台，得到具体内容见设计背景与 **Need** 部分。

D（Delivery，推广）

暂无

设计实现

根据我们需要实现的功能和团队成员的开发经验，我们初步考虑选取的技术栈包括：

Web 前端：

- **Vue.js**：Vue 是一款流行的轻量级 JavaScript 前端框架，采用组件化的开发方式，兼具易用性和灵活性，并且拥有丰富的生态。
- **Element UI Plus**：EUI+ 是一个基于 Element UI 的 Vue 的组件库扩展，提供了一些新的、实用的组件和功能，可以方便地扩展 Element UI 的能力。

后端：

- **Django**：**Django** 是一个基于 Python 的免费开源的 Web 应用程序框架，团队成员对此框架较熟悉。
- **MySQL**：**MySQL** 是一种关系型数据库管理系统，是最流行的开源数据库之一。

预计目标

通过不断深入学习数据库相关知识并练习掌握前端设计技巧，在数据库大作业验收时，完成 **Web App** 的发布。