# 教学直播平台概要设计

## 平台设计要求和目标

教学直播平台概主要用于教学直播，课程管理等多个方面，平台设计的特点是：

### 平台功能多样性

平台作为一个软件服务平台，决定了平台功能的多样性。平台对学生，老师，管理员提供服务，并为不同学院部门的学生和老师提供针对性的功能服务。

### 用户操作便利性

直播平台需要实现传输信息的有效性和及时性，提高并发数和在线人数，这对实际教学具有实际意义。同时平台提供打卡功能保证了学生的实时签到，最大程度的减少人工参与的步骤，平台通过系统内部实现替代人工操作，提高用户使用的便捷性。最终目的是为学校教育机构提供技术支持，提高教学效率，降低远程教学的难度，降低教育行业的信息化门槛。

### 屏幕共享的便利性

直播平台提供了屏幕共享功能，可以保证学生和老师在实际教学过程中可以畅通无阻地进行交流和展示，提高教学效率。同时平台提供了屏幕共享的接口，可以让学生和老师在实际教学过程中可以通过网页或者其他方式观看共享的视频，提高教学效率。

### 聊天的及时性

在直播过程中有时候需要通过聊天框聊天来进行沟通交流。直播平台提供了聊天框，可以让学生和老师在实际教学过程中可以通过聊天框进行交流，提高教学效率。与此同时，运用先进技术实现了聊天信息的实时传输，提升用户体验，同时聊天记录也将保存留在以后查询使用。

## 平台软件结构设计

软件结构设计是指从宏观的角度对系统进行提纲挈领的总体设计，理清系统总体架构中各业务对象之间的区别与联系，为之后详细功能设计奠定基础。

整个系统采用前后端分离设计的框架，前端由web浏览器和桌面客户端构成。

后端采用Fastapi为基础框架，mysql数据库存储数据。

其中后端采用分层架构，分为视图层，业务逻辑层，模型层，也就是典型的MVC架构设计。

在这其中，MVC 模式将系统分为三个核心部分：模型(Model)、视图(View)和控制器(Controller)，使得软件代码按照数据、界面显示、业务逻辑分离的方式组织呈现，从而降低不同部分之间的依赖程度，提高模块的内聚性。

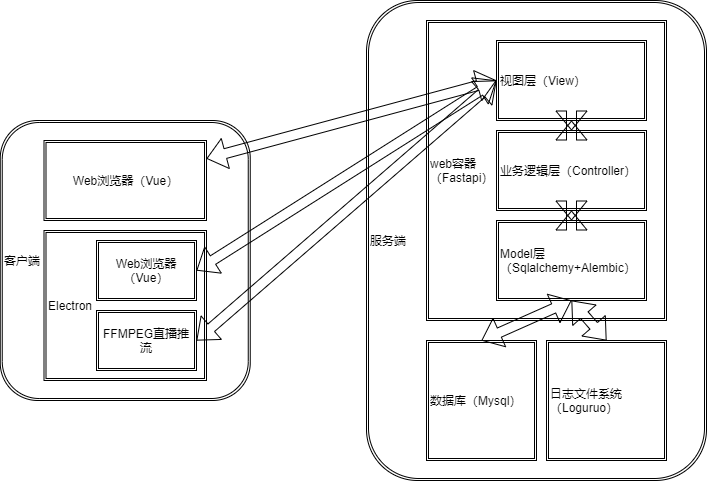
（1）模型（Model）封装了数据和对数据的操作，是实际进行数据处理和计算的地方。一个模型可以同时为多个视图提供数据，从而提高了代码的可重用性。Spring 通常用HashMap 存储模型的处理结果。

数据访问层Model 业务逻辑层Controller 用户交互界面层View 用户交换界面层数据持久化服务实现用户交互逻辑服务

（2）视图（View）是用户看到并与之交互的界面，它负责将应用显现给用户和显示模型的状态。

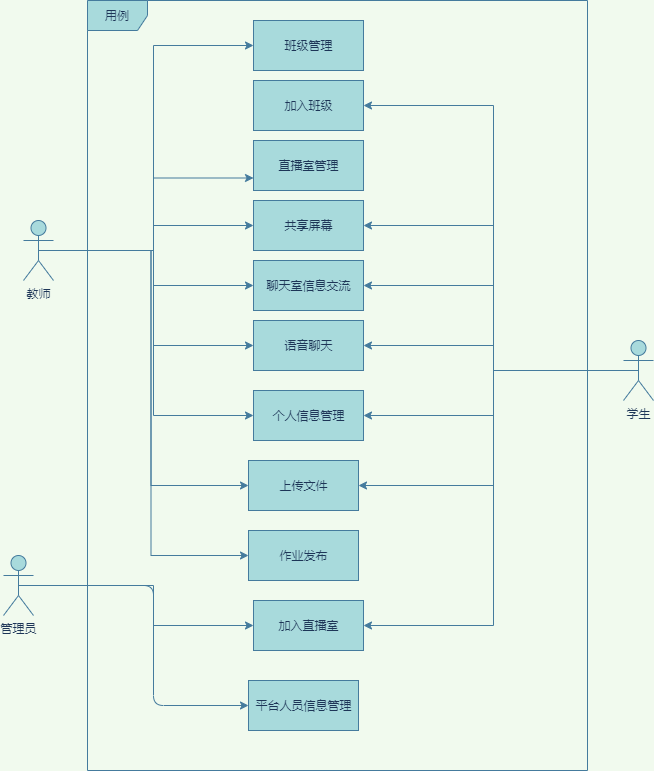
（3）控制器（Controller）负责视图和模型之间的交互，控制对用户输入的响应方式和流程，它主要负责两方面的动作：把用户的请求分发到相应的模型；将模型的改变及时反应到视图上。

平台采用MVC 开发模式，极大的提高了平台组件的可复用性和开发效率。

整个系统的架构设计如下图：

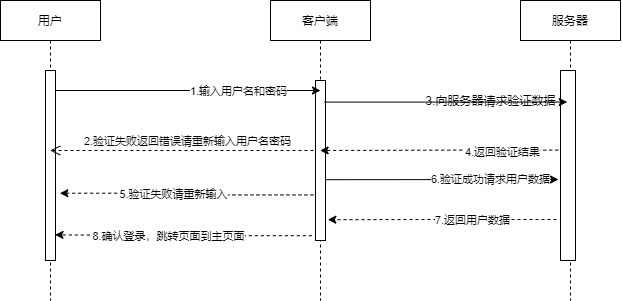
## 平台功能模块设计

根据教育平台的基本需求，结合各个角色的功能，讲平台功能按照角色进行划分，分为学生模块，教师模块，管理员模块，每一个模块的功能如下图所划分展示的：



## 登录模块设计

登录模块的具体的流程图如下所示：



## 平台数据库设计

数据结构描述数据的存储结构、组织结构等信息，设计数据结构的目的是更好的为平台提供有效的数据。教学直播平台涉及的数据结构主要是对数据库的设计。数据库设计（DatabaseDesign）是指对于一个给定的应用环境，构造最优的数据模式，建立数据库及其应用系统，使之能够有效的存储数据，满足各种不同用户的应用需求（信息要求和处理要求）。本平台采用的是Mysql 数据库，主要涉及的表有：用户表，角色表，权限表，课程表，直播室表等

数据库设计原则

数据库是系统数据存储的服务单元，是系统主要业务数据的集合，要求具有规范性和严格性。数据库设计是平台功能模块实现和运行的基础，数据库设计的好坏直接影响平台运行速度、效率和用户体验。因此设计数据库应该遵守良好的数据库设计原则。数据库设计原则如下：

（1）降低数据冗余。适当的数据冗余可以减少数据查询的频率，但过多的数据冗余不仅会占用过多的存储空间，也会对数据库的维护和扩展带来诸多不便，因此数据库设计时应该适当的使用冗余数据。

（2）高效性。系统的后期瓶颈一般都是IO，数据库读写效率直接影响用户体验，高效的数据操作可以带来良好的用户体验。

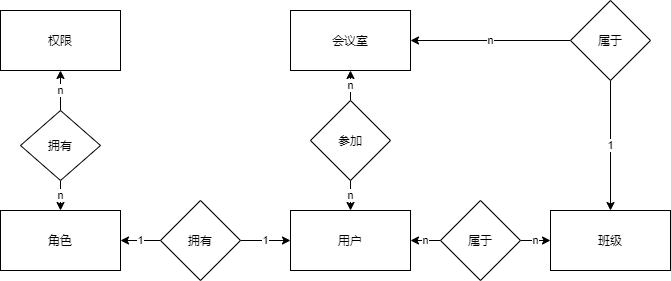
（3）扩展性。平台的可扩展性不仅是系统的可扩展性，更是数据库的可扩展性，数据库的可扩展性是评价一个系统可扩展性的重要因素，数据库设计应具备良好的可扩展性。

（4）安全性。数据库中存储所有信息的安全性是系统平台安全、稳定、高效运行的基础，非法使用数据库，使数据遭到破坏、篡改或者非法删除，将给平台带来严重影响。因此，采取必要的安全策略，保障数据库安全性。

数据库E-R图

E-R图也称实体-联系图(EntityRelationshipDiagram)，提供了表示实体类型、属性和联系的方法，它能够将错综复杂的业务对象通过简单明了图形和符号表示，是数据库概念设计阶段建立数据模型的有力工具，它已经成为一种普遍的、标准的数据库模式建立范型(paradigm)。根据第二章需求分析，已经明确平台的主要业务对象、属性和表之间的联系。依据这些信息，将设计整个系统的Ｅ－R表

整体系统的Ｅ－R表设计如下图所示：



## 运行环境

硬件环境

采用阿里云服务器具体配置如表：

|  |  |
| --- | --- |
| 产品型号 | 阿里云共享型服务器 |
| CPU | ２核（Intel志强处理器） |
| 内存 | 2G |
| 网络带宽 | 峰值5M |
| 硬盘 | 40G |

软件环境

|  |  |
| --- | --- |
| 操作系统 | Ubuntu20.04LTS |
| 容器 | Docker |
| 数据库 | MYSQL8.0（Docker部署） |
| Python版本 | 3.9.0以上 |
| 负载均衡 | Nginx |

开发环境

|  |  |
| --- | --- |
| 操作系统 | Win11 |
| 开发语言 | Python，Javascript |
| 数据库版本 | Mysql8.0以上 |
| Python版本 | 3.9以上 |
| 开发工具 | Vscode |
| Sqlalchemy版本 | 1.4.26以上 |

平台部署

计划初期使用nginx进行负载均衡，同时直接部署在服务器上。同时学习docker，尝试使用docker进行容器化部署。