作业收集平台PaperStack  
系统设计文档

版本 1

# 版本变更记录

《系统设计》版本1为初始版本。

# 目录

[作业收集平台PaperStack 系统设计文档 1](#_Toc24393275)

[版本变更记录 2](#_Toc24393276)

[目录 3](#_Toc24393277)

[文档目标 4](#_Toc24393278)

[概要设计 4](#_Toc24393279)

[软件架构 4](#_Toc24393280)

[硬件条件 5](#_Toc24393281)

[开发环境及工具 5](#_Toc24393282)

[详细设计 5](#_Toc24393283)

[通用组件设计 6](#_Toc24393284)

[通用界面 6](#_Toc24393285)

[用户界面与路由 8](#_Toc24393286)

[用户会话与会话凭证 8](#_Toc24393287)

[控制器 9](#_Toc24393288)

[服务 10](#_Toc24393289)

[实体 10](#_Toc24393290)

[注册子系统设计 11](#_Toc24393291)

[用户界面 11](#_Toc24393292)

[静态设计 12](#_Toc24393293)

[数据结构 12](#_Toc24393294)

[动态设计 12](#_Toc24393295)

[登录子系统设计 15](#_Toc24393296)

[用户界面 15](#_Toc24393297)

[静态设计 16](#_Toc24393298)

[数据结构 17](#_Toc24393299)

[动态设计 17](#_Toc24393300)

[个人信息子系统设计 20](#_Toc24393301)

[用户界面 20](#_Toc24393302)

[静态设计 21](#_Toc24393303)

[数据结构 22](#_Toc24393304)

[动态设计 22](#_Toc24393305)

[作业收集子系统设计 26](#_Toc24393306)

[用户界面 26](#_Toc24393307)

[静态设计 29](#_Toc24393308)

[数据结构 29](#_Toc24393309)

[动态设计 29](#_Toc24393310)

[平台管理子系统设计 33](#_Toc24393311)

[静态设计 33](#_Toc24393312)

[动态设计 34](#_Toc24393313)

# 文档目标

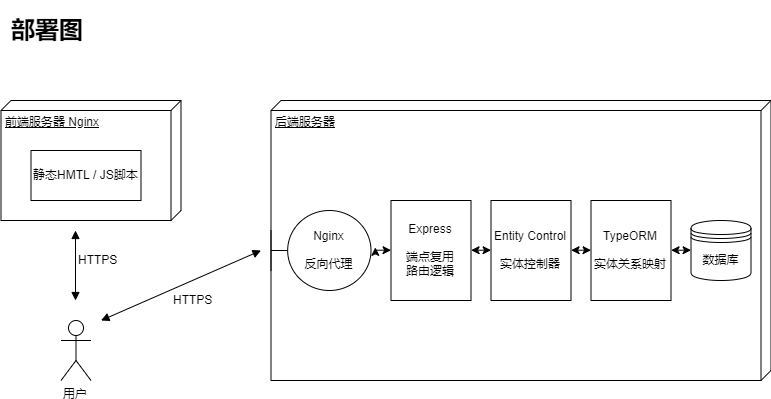
本文档为系统设计文档，目标是在上一阶段所提出的需求分析文档的基础上，进一步对系统进行设计，将系统需求转换为软件表示，描述软件的总体概貌。

本文档分概要设计和详细设计两部分。概要设计主要介绍系统的软件架构和硬件条件等，并将系统划分为子系统和模块，并指出模块所负责的用例；详细设计主要借助UML的类图和顺序图描述模块的静态设计和动态设计，并介绍用户界面和数据表的设计。

# 概要设计

本小节主要介绍系统的软件架构以及系统部署所需的硬件条件，并介绍开发系统所需的环境和工具。

## 软件架构



系统采用B/S架构，分为前端和后端两部分。前端负责呈现图形化用户界面并与用户进行视觉元素交互、将用户的数据发送至后端处理并呈现后端处理后的结果。后端则负责保存用户数据、处理前端请求和返回请求结果。

前端用户界面采用单页面应用程序（SPA）设计，采用TypeScript和React框架开发。代码编译为静态HTML文件和JavaScript文件，由前端HTTP服务器托管。用户通过浏览器访问前端HTTP服务器，获得静态HTML页面和在浏览器上运行的JavaScript脚本。当用户需要执行操作时，前端页面显示视觉元素，浏览器通过HTTP协议向后端服务器发送数据并获取操作结果，不需要经过前端HTTP服务器。

后端通过HTTP协议提供服务。用户需要执行操作时，浏览器根据操作的不同而请求不同的后端HTTP端点。后端服务器根据HTTP的端点和参数来区别用户请求的操作，分配不同的路由来处理用户请求。各路由调用控制器和服务，从数据库中获取数据，处理用户的请求并返回结果。

|  |  |  |
| --- | --- | --- |
| 路由层 | 用户注册路由、用户登录路由等 | |
| 控制器层 | 用户管理器、用户会话管理器等 | |
| 服务层 | 对象维护服务、用户邮箱验证服务等 | |
| NodeJS运行环境 | | PostgreSQL数据库 |

后端服务器路由层采用Express框架进行开发， 控制器层和服务层自行编写，由TypeORM负责对象关系映射。数据库软件使用PostgreSQL。

前端页面包括注册界面、登录界面、个人信息界面、作业收集界面和平台管理界面。后端路由层包括用户注册路由、用户登录路由、用户重置密码路由、用户退出路由、个人信息路由、作业收集组路由、作业收集项路由和管理员路由。控制器包括中央控制器、用户管理器、用户会话管理器、学院管理器、作业收集组管理器。服务层包括对象维护服务、对象持久化服务、用户邮箱验证服务、文件存储服务和作业预览服务等。这些组件的具体功能将在详细设计中介绍。

实际部署时应当使用加密的HTTP协议。前端服务器直接由Nginx托管静态文件。后端服务器运行于Node.js环境中监听非标准端口，并由Nginx反向代理到标准HTTP端口。

## 硬件条件

在系统试运行阶段，为满足需求分析报告提出的“25人同时浏览和下载和容纳 500人的信息存储”要求，若将前端服务器和后端服务器部署于同一台物理机上，则按每人浏览所需10Mbp带宽，产生数据500MiB进行粗略估算，物理机须有20Mbps进出口带宽和至少500GiB的HDD存储空间，处理器不少于2核心，基准速度不低于2.0GHz。

## 开发环境及工具

系统使用TypeScript语言开发，开发环境运行Node.js。受益于TS和Node.js的跨平台特性，系统可在任意主流操作系统中开发。编译器为标准TSC，代码格式化工具为ESLint，代码协作工具为Git。后端接口调试使用yAPI。

团队使用Draw.io进行UML建模。

团队中推荐使用的代码编辑器为Visual Studio Code。

团队使用Adobe XD和Zepplin.io进行用户界面设计。

# 详细设计

本小节主要借助UML的类图和顺序图描述模块的静态设计和动态设计，并介绍用户界面和数据表的设计。

## 通用组件设计

### 通用界面



图 1 首页界面设计

首页界面视觉设计如图。页面中央是广州大学的校徽校训和PaperStack的logo，下方是登录按钮和注册按钮，背景是带有毛玻璃特效的广州大学图书馆。登录按钮和注册按钮分别连接到登录子系统的登录页面和注册子系统的注册页面。首页位于路由树的根部，任何未登录而尝试浏览其他路径的用户都会被重定向至这个页面。



图 2 教师和学生用户登陆后的页面-左侧菜单已展开

教师和学生用户登录之后页面会跳转到图示的个人信息页面。个人信息页面相当于登陆用户的主页、系统功能的索引页面。页面的左侧菜单展开以后用户可以点击菜单上的按钮跳转使用作业收集子系统的功能。点击页面左下角的头像可以进入个人信息子系统的个人信息详情页面。

### 用户界面与路由

图片包含 文字

描述已自动生成

图 3 前端用户界面与后端路由

前端用户界面与后端路由是系统中重要的边界类。前端用户界面直接与用户交互，负责呈现视觉元素和后端处理结果。前端用户界面采用React框架和ReactRouter进行开发。当用户在图形界面上进行操作时，用户的浏览器向后端对应的HTTP路径发送请求和请求参数。在后端，HTTP请求由各个路由进行处理。

后端所有路由都继承于公共的路由基类。基类成员展示了路由处理一个前端请求的基本步骤：首先是为请求生成一个请求ID，这样有助于在日志中区分同一个路径的不同请求。然后是验证路由参数。根据用户请求的不同，每个路由完成请求需要不同的参数。后端路由要验证前端路由发送的参数是否完整和有效，以避免前端编程时出错，以及过滤掉恶意用户构造的攻击。接下来是获取会话用户，这与下一小节描述的“用户会话与会话凭证”有关。然后是执行对应路由的控制逻辑，如调用控制器和服务，获取和修改数据库中数据等。最后将路由处理的结果封装在HTTP响应中发送回前端。

### 用户会话与会话凭证

用户使用用户名和密码登录系统后，系统为用户创建一个用户会话，并为这个用户分配一个会话凭证。系统通过会话凭证来识别用户会话。当用户需要操作一些需要特定权限才能进行的路由时，必须发送会话凭证。路由只有根据凭证获取到具备合适权限的用户才会继续进行操作，否则向前端报告权限不足，拒绝执行。

一个用户可以拥有多个用户会话，拥有多个会话凭证，因此可以在多个浏览器中登录。

识别用户具体过程在登录子系统中介绍。

图片包含 屏幕截图

描述已自动生成

顺序图 1 获取用户会话

### 控制器

图片包含 屏幕截图

描述已自动生成

类图 1 系统中的控制器

控制器是系统中的控制类，控制器通常与系统中的实体相关联。中央控制器是系统程序运行时的入口点，负责读取配置文件和启动根路由。使用中央控制器可以启动或停止系统对外提供的服务。学院管理器负责创建学院结构。用户管理器负责创建、查询和删除用户结构。用户会话管理器提供根据会话凭着查询用户会话的功能。作业收集组管理器提供创建作业收集组和查询作业收集组的功能。

### 服务

图片包含 屏幕截图

描述已自动生成

类图 2 系统中的服务

服务是系统中特殊的一类控制器，通常不与特定类型的实体发生关联。用户邮箱验证服务可以生成邮箱地址验证码，提供检查邮箱地址是否被注册、校验邮箱地址验证码等功能。对象维护服务随系统启动时运行，负责删除系统中过期的实体，包括过期的邮箱地址验证码和过期的用户会话等。邮件服务提供发送邮件的功能。对象持久化服务提供将系统中的实体持久化保存到数据库的功能，从而向上层的调用者隐藏数据库的具体实现。

邮件服务和对象持久化服务在使用前，需要提供正确的邮箱账号配置和数据库配置。

### 实体

系统中的是实体指用户信息、班级结构、作业信息等数据。它们可以被对象持久化服务通过数据库进行存取。实体的数据结构在关联实体的子系统设计中进行说明。

## 注册子系统设计

### 用户界面



图 4 注册页面

图示为用户注册界面，未注册的用户通过点击首页上的“注册”按钮可以跳转到这个页面。用户在注册界面中输入自己的邮箱地址和密码。当邮箱地址的输入框失去焦点时，页面判断用户输入的邮箱地址是否符合邮箱地址的一般规则，若不符合则在输入框下方显示一行红字提示用户“输入的邮箱地址不合法”，并高亮邮箱地址输入框；若检查无问题，则通过后端接口检查邮箱地址是否已经注册，若已注册则在输入框下方显示一行红字提示用户“邮箱地址已被注册，若您忘记密码，可以尝试找回密码”。

用户点击注册按钮后，页面向后端服务器“提交注册邮箱并获取邮箱地址验证码”接口发送用户的注册信息。并显示在密码输入框下方显示先前隐藏的邮箱验证码输入框。用户输入邮箱验证码后，页面通过后端“验证邮箱地址并成为注册用户”接口发送用户的注册消息，若后端接口返回成功，则在注册按钮上方显示一行绿字“注册成功”。否则显示红字“验证码输入错误”并高亮验证码输入框。

用户完成注册后，页面显示注册成功消息，并在3秒后跳转至登录子系统的登录页面。

### 类图

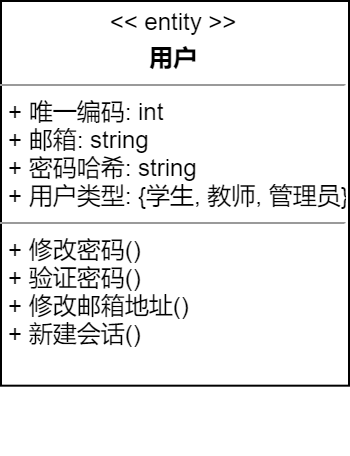
图片包含 屏幕截图

描述已自动生成

类图 3 注册子系统静态设计

注册子系统包括注册页面类和用户注册路由类，使用了通用组件中的用户邮箱验证服务、邮件服务和用户会话管理器。

### 数据模型



类图 4 注册子系统数据结构

注册子系统中的路由可以调用用户管理器创建用户结构，如上图所示。用户管理器创建用户时，要为用户生成唯一编码，保存注册邮箱地址。在系统中保存密码时，并不是直接保存用户的明文密码，而是使用密码哈希算法生成密码的哈希，并在系统中保存密码哈希。系统支持学生、教师、管理员三种用户类型，创建用户时还要记录用户类型。

### 用例的动态设计

|  |  |
| --- | --- |
| **用例号** | UC001 |
| **用例名称** | 提交注册邮箱并获取邮箱地址验证码 |

图片包含 屏幕截图

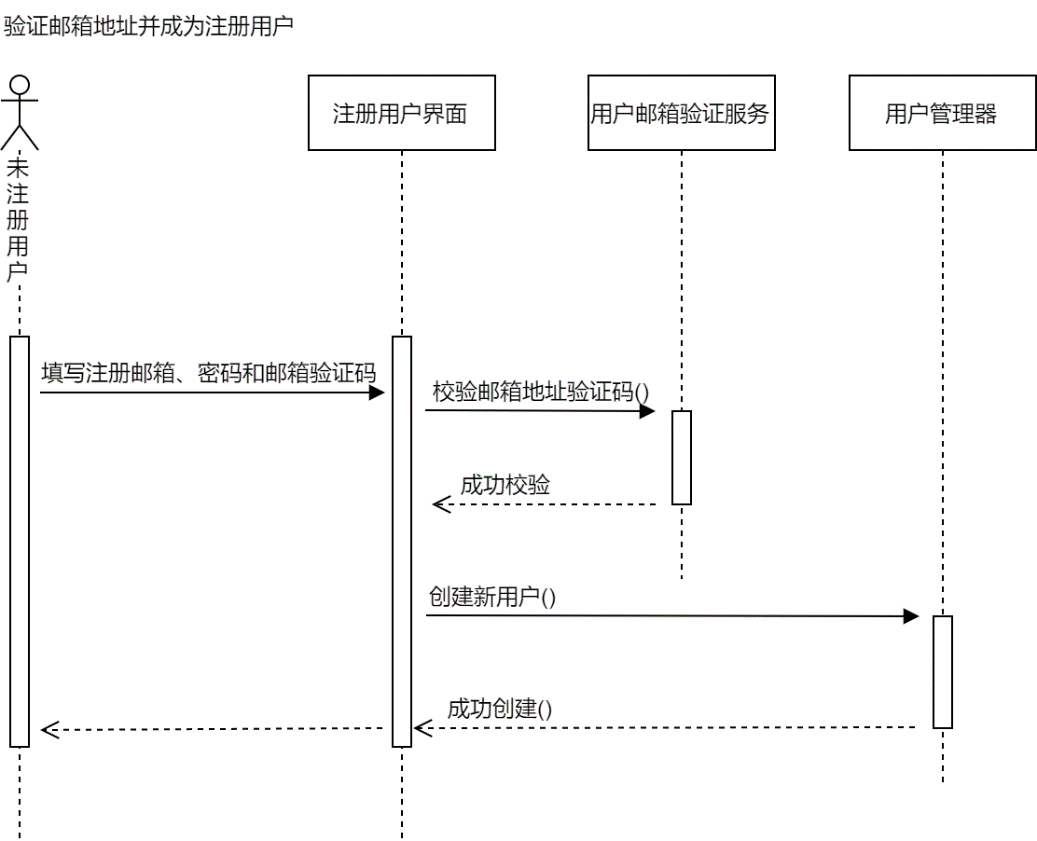
描述已自动生成

顺序图 2 提交注册邮箱并获取邮箱地址验证码

当注册路由收到前端发送的注册邮箱后，调用邮件用户邮箱验证服务检查用户的邮箱地址是否被占用，并返回占用信息。

当注册路由收到前端发送的邮箱和密码消息后，检查邮箱地址是否被占用，若未被占用则调用用户邮箱验证服务，生成对应邮箱地址的验证码。用户邮箱验证服务相应地生成验证码，并将验证码和邮箱地址对使用对象持久化服务保存（顺序图中未标出此步骤）。注册路由取得验证码后，生成一封包含验证码和问候消息的邮件，随后调用邮件服务将邮件发送往注册邮箱地址，并向前端返回成功消息。若过程中用户邮箱地址已被注册，或邮件服务发送失败，则要相应地向前端返回错误消息。

|  |  |
| --- | --- |
| **用例号** | UC002 |
| **用例名称** | 验证邮箱地址并成为注册用户 |



顺序图 3 验证邮箱地址并成为注册用户

当路由收到注册邮箱、密码和邮箱地址验证码后，调用用户邮箱验证服务检查邮箱地址与验证码是否相符。若相符则调用用户管理器创建用户结构。若验证码与邮箱地址不符，或创建用户过程中出错（比如系统中已经存在同邮箱地址的用户），则向前端返回错误消息。

## 登录子系统设计

### 用户界面



图 5 登录页面

图示为用户登录界面。用户可以通过首页的“登录”按钮跳转至这个页面。用户在输入框中输入注册邮箱和密码。若用户输入的邮箱地址格式不正确，则用红色高亮邮箱地址输入框，并在输入框下方显示“邮箱地址不正确，请重新输入”。

当用户输入用户名和密码，并点击登录按钮。系统通过后端“使用邮箱和密码登录”接口尝试获取会话凭证。若成功获取到会话凭证则表明登录成功。页面跳转到个人信息页面。

当用户输入错误的密码超过3次时，页面在登录按钮上方显示找回密码功能的横幅。



图 6 找回密码页面

图示为找回密码界面。用户可以使用找回密码功能来重置密码。用户在页面中输入待找回密码的邮箱地址。页面检查邮箱地址的格式无误后向后端接口“使用邮箱地址获取重置密码验证码”发送请求。后端路由返回重置密码邮件发送成功的消息后，在页面中的验证码输入框上方显示“请登录邮箱查看验证码”等字样。

当用户输入验证码后，页面在验证码输入框下方显示原本隐藏的密码输入框，供用户输入新的密码。用户点击确认按钮之后，页面请求后端的“验证邮箱并重置密码”接口。若后端返回成功，则在确认按钮下方实现一行绿色文字，提示用户可以使用新的密码登录了。若后端返回失败，则提示用户验证码输入错误。

### 类图

图片包含 屏幕截图

描述已自动生成

类图 5 登录子系统静态设计

登录子系统包括登录页面类、用户登录路由类、用户重置密码路由类和用户退出路由类，使用了通用组件中的邮件服务、用户邮箱验证服务、用户管理器和用户会话管理器。

### 数据模型

图片包含 屏幕截图

描述已自动生成

类图 6 登录子系统数据结构

登陆子系统所使用的数据结构如图。用户与用户会话是一对多的关系。一个用户可以拥有多个会话，一个会话只归属于一个用户。用户会话相应地存在一个用户标志用户身份的会话凭证，并存在一个有效期限。当一个用户会话被创建和被使用时，会话的上次会话时间属性就会被更新。当一个会话长期处于不活动状态后，现在时间与上次会话时间的间隔如果超过对象维护服务所设定的会话过期时间间隔，对象维护服务将从系统中移除这个用户会话。用户也不能凭借过期的用户会话凭证登录。

### 用例的动态设计

|  |  |
| --- | --- |
| **用例号** | UC003 |
| **用例名称** | 使用邮箱地址获取重置密码验证码 |

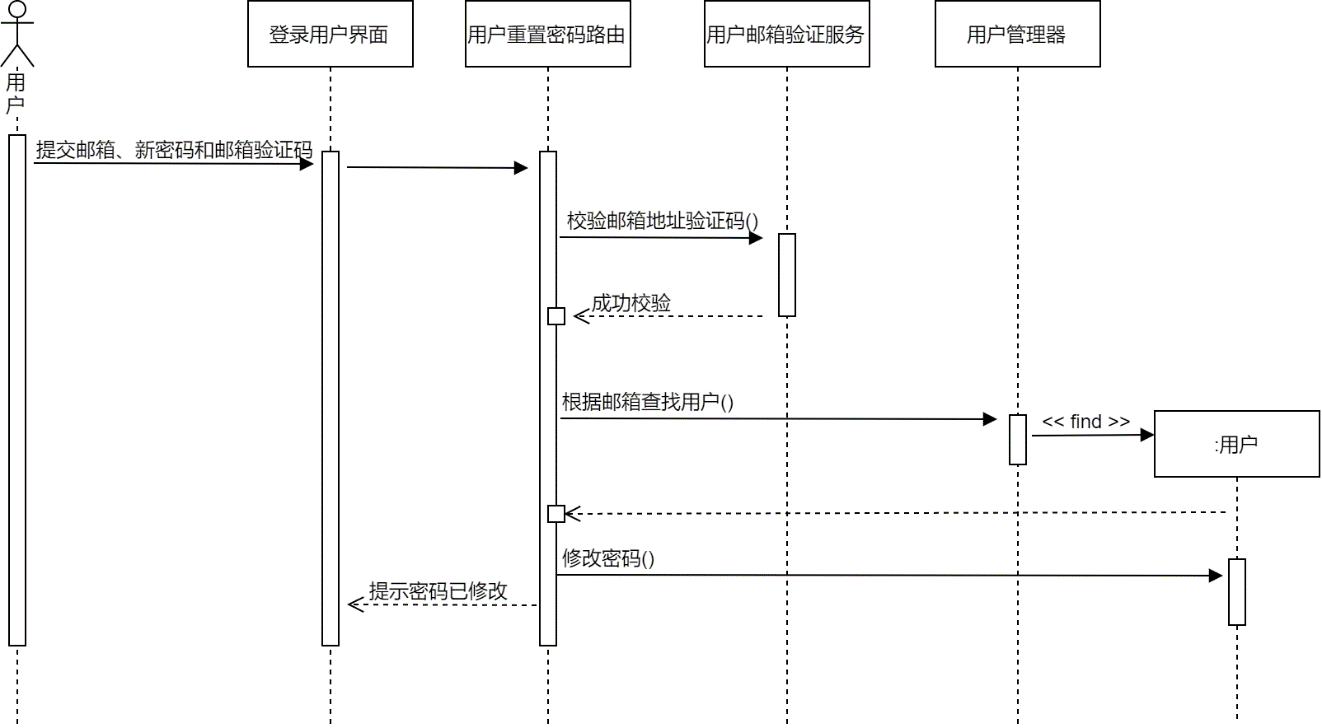
图片包含 屏幕截图

描述已自动生成

顺序图 4 使用邮箱地址获取重置密码验证码

路由收到前端的请求后，调用用户邮箱验证服务生成邮箱地址验证码，然后编写一封带有问候消息和验证码的邮件，通过邮件服务将邮件发送到待重置密码的账户的邮箱地址。若邮件发送成功，则向前端返回成功消息；若邮件发送失败，则向前端返回失败消息。

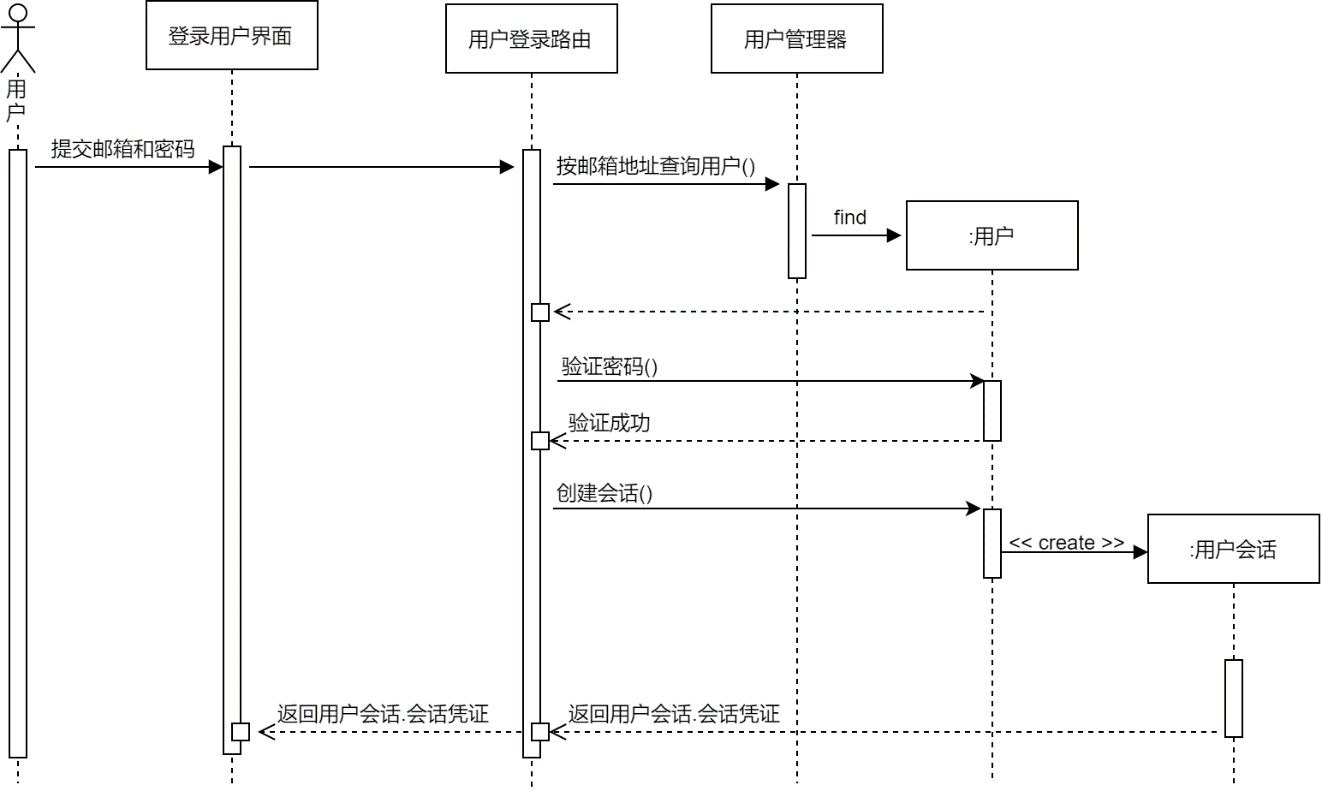
|  |  |
| --- | --- |
| **用例号** | UC004 |
| **用例名称** | 验证邮箱并重置密码 |



顺序图 5 验证邮箱并重置密码

后端接收到前端请求后，调用用户邮箱验证服务检查邮箱地址验证码是否与邮箱地址相符。若不符，则直接返回邮箱地址验证失败消息。若校验成功，则调用用户管理器，根据邮箱地址在系统中找到用户实体，并修改用户的密码。若用户不存在，则向前端返回用户不存在消息。若修改成功，则向前端发送密码重置成功消息。

|  |  |
| --- | --- |
| **用例号** | UC005 |
| **用例名称** | 使用邮箱和密码登录 |



顺序图 6 使用邮箱和密码登录

后端收到前端请求后。后端路由调用用户管理器，按邮箱地址查询用户。若根据邮箱地址查询不到用户，则像前端返回用户不存在消息。若查询到用户，则尝试比对用户的密码，若验证失败，则向前端返回密码错误消息。若密码正确，则为用户创建一个新的用户会话，并向前端返回会话的用户会话的凭证。

## 个人信息子系统设计

### 用户界面

图片包含 屏幕截图

描述已自动生成

图 7 个人信息页面

图示为个人信息页面的视觉效果图。顶端栏目展示头像、姓名、性别、学号、班级等个人信息。顶栏右侧有编辑个人信息的按钮，点击按钮后跳转到修改个人信息的页面。顶栏下方是主面板，展示用户所参与的作业收集项目。并附有新建作业项目和导出按钮。若为教师用户则可以点击按钮跳转到作业收集子系统的创建作业收集项目页面。用户可以使用导出按钮跳转到导出页面，导出个人数据。主面板右侧提供了账号绑定、修改邮箱地址、修改密码、退出登录等辅助功能。



图 8 个人信息页面：修改个人信息

修改个人信息页面的视觉效果图如图所示。用户可以在页面上自助修改昵称、性别、学号等、学院和班级信息。用户必须设置符合格式的学号。用户不能自由设置任意的学院和班级，只能将自己的学院和班级设置为系统管理员在平台管理子系统中创建的学院和班级的其中一个。

### 类图

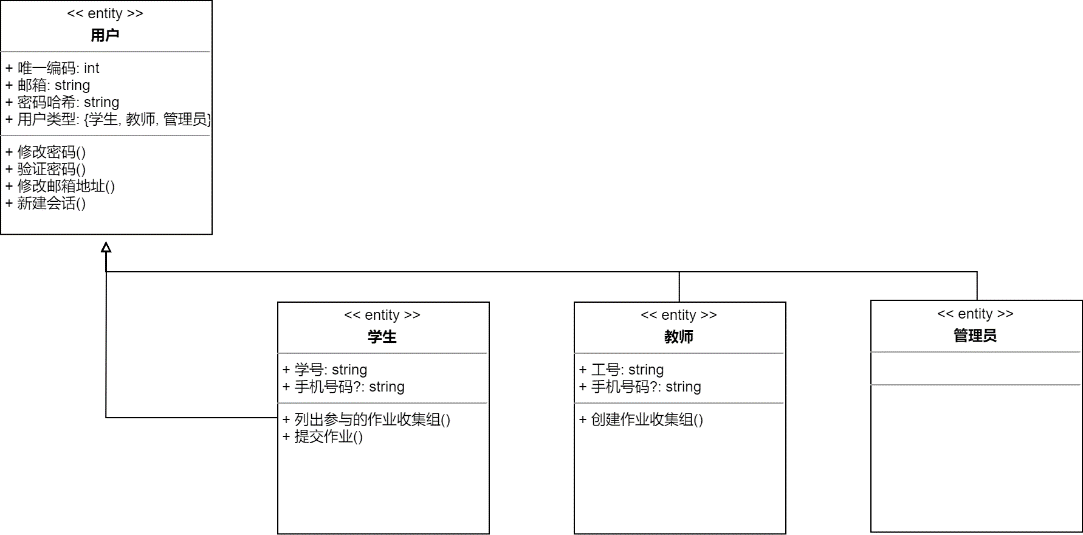
图片包含 屏幕截图

描述已自动生成

类图 7 个人信息子系统静态设计

个人信息子系统包括个人信息界面类和个人信息路由类，使用了通用组件中的用户邮箱验证服务、用户管理器和用户会话管理器。

### 数据模型

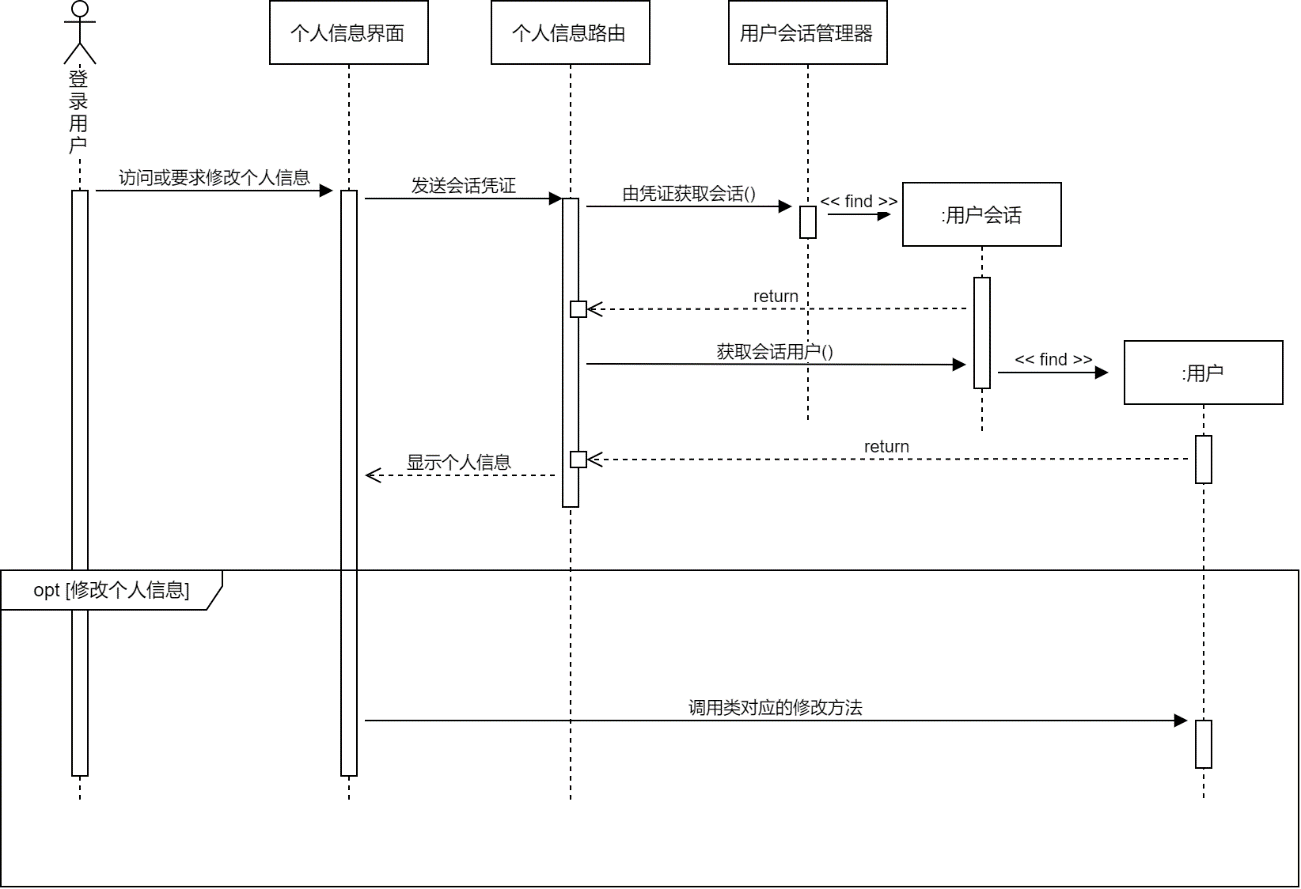


类图 8 个人信息子系统数据结构

个人信息子系统涉及的类结构为用户结构。系统中存在学生、教师、管理员三类用户。用户的公共信息存放在基类中，而各个类型的独有信息（如学生的学号或教师的工号）存放在子类中。

### 用例的动态设计

|  |  |
| --- | --- |
| **用例号** | UC007 |
| **用例名称** | 访问和修改个人信息 |



顺序图 7 访问和修改个人信息

后端路由接收到前端的请求时，调用用户会话管理器，通过会话凭证来甄别用户会话，并通过用户会话定位到用户，读取用户的信息并返回前端。若前端传递的参数中要求修改用户信息，则相应地修改用户信息，并将修改后的信息返回前端。

|  |  |
| --- | --- |
| **用例号** | UC008 |
| **用例名称** | 修改密码 |

图片包含 屏幕截图

描述已自动生成

顺序图 8 修改密码

后端路由收到前端发送的修改密码请求后，根据用户凭证甄别定位到用户。随后验证用户输入的旧密码与系统中用户的密码是否相符。若密码不相符，则向前端返回旧密码错误消息。若密码相符，则将系统中用户的密码修改为新密码，并前端返回修改密码成功消息。

|  |  |
| --- | --- |
| **用例号** | UC009 |
| **用例名称** | 修改邮箱 |

图片包含 文字, 地图

描述已自动生成

顺序图 9 修改邮箱

当后端路由收到修改邮箱请求后，先根据会话凭证定位到会话用户。随后调用用户邮箱验证服务生成新邮箱地址的验证码，编写附有验证码的邮件并调用邮件服务发送到用户的新邮箱，并前端返回成功消息。这时前端提示用户打开邮箱查收验证码；若发送邮件失败，则向前端返回失败消息。

当后端路由收到修改邮箱的请求和验证码后，定位到当前会话的用户。接下来调用用户邮箱验证服务，校验用户的新邮箱与邮箱地址验证码是否相对应。若邮箱验证码与邮箱地址不符，则向前端发送验证码错误消息。此时前端应当提示用户验证码输入错误，要求重新输入。若验证码校验成功，则将用户的邮箱地址修改为新邮箱地址，并向前端返回成功消息。

## 作业收集子系统设计

### 用户界面



图 9 创建作业收集项目页面

图示为教师用户创建新作业收集项目的视觉效果图。教师用户在新建收集项目时，可以输入作业的名称（作业收集项），选择或输入作业的科目（作业收集组）。为本次作业安排截止日期。教师可以为作业添加文字形式的作业要求，作业要求会显示在学生用户的作业详情界面。教师用户可以从行政班级列表中选择需要提交作业的学生的名单。

图片包含 屏幕截图

描述已自动生成

图 10 浏览作业收集组

图示为学生用户或教师用户浏览作业收集组页面。同一个科目的作业被聚合到一起，科目称为“作业收集组”，作业称为“作业收集项”。



图 11 浏览作业收集项

图示为学生用户的浏览作业收集项页面。



图 12 浏览作业收集项：某次作业提交情况

图示为学生或教师用户浏览作业的页面。学生在本页面中可以看到自己提交的作业和其他同学的提交作业概况。点击每个作业的“进入”按钮可以作业详情页面。点击预览作业按钮可以进入作业预览页面，点击下载作业按钮可以下载已经提交的作业。

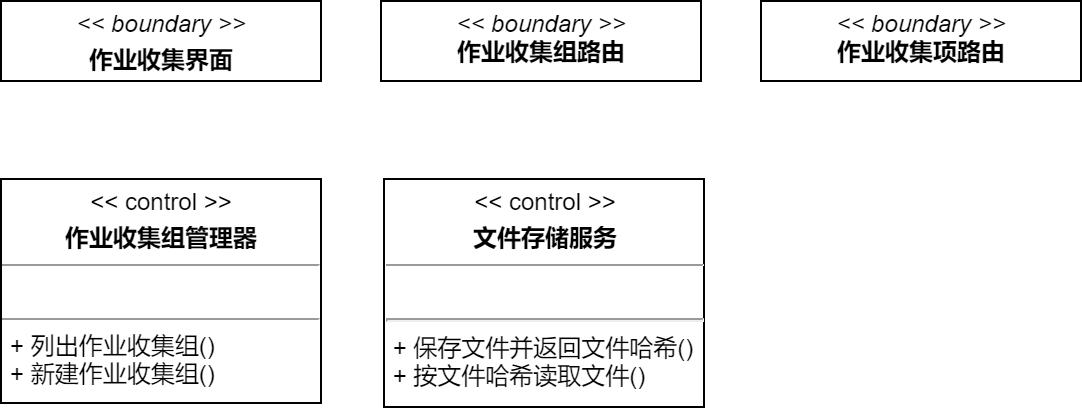
图片包含 屏幕截图

描述已自动生成

图 13 浏览作业详情

图示为作业详情页面。顶部栏目左侧展示作业要求。学生可以将作业拖动到顶栏右侧区域，或者手动选择作业上传。

### 类图



类图 9 作业收集子系统静态设计

作业收集子系统包括作业收集界面类、作业收集组路由类和作业收集项路由，以及作业收集组管理器。

### 数据模型

图片包含 屏幕截图

描述已自动生成

类图 10 作业收集子系统数据结构

类图中，作业收集组在用户界面中为“科目”，作业收集项在用户界面中为“作业”。例如课程面向对象设计与分析可能有第一周、第二周……第N周作业。面向对象设计与分析在用户界面中显示为科目，在类图中类型为作业收集组。第一周、第二周……第N周作业在用户界面中为作业。故一个作业收集组可能包含有多个作业收集项。

在一个作业收集项中可能包含多个“作业”，因为一项作业可能有多于一名同学提交作业。

每份学生提交的作业都可能有多份作业评论。

### 用例的动态设计

|  |  |
| --- | --- |
| **用例号** | UC013 |
| **用例名称** | 创建作业收集项目 |

图片包含 屏幕截图

描述已自动生成

顺序图 10 创建作业收集项目

当后端收到创建作业收集组的请求时，首先通过会话凭证定位到会话用户，并检查当前用户是否为教师用户（只有教师用户才能创建作业收集项目）。若权限检查通过，则调用作业收集组管理器尝试新建作业收集组和作业收集项。若作业收集组已经存在，则直接新建作业收集项。最后向前端返回项目新建成功消息。

|  |  |
| --- | --- |
| **用例号** | UC014 |
| **用例名称** | 下载作业 |

|  |  |
| --- | --- |
| **用例号** | UC015 |
| **用例名称** | 查看作业 |

|  |  |
| --- | --- |
| **用例号** | UC018 |
| **用例名称** | 作业评分和评论 |

图片包含 屏幕截图

描述已自动生成

顺序图 11 下载作业/查看作业/评分和评论

由于下载作业/查看作业/评分和评论都需要从作业收集组-作业收集项-作业的层级结构中定位到某一份作业，因此路由的前半部分中逻辑结构基本相同，均为用户鉴别、定位作业收集组、定位作业收集项和定位作业。查看作业路由将定位到的作业相关信息返回前端；评分和评论路由调用定位到的作业的添加评论或添加评分方法，为作业添加一条评论或评论；下载作业路由则调用文件存储服务，按作业的文件哈希从磁盘中读取文件，并向前端返回文件内容。

|  |  |
| --- | --- |
| **用例号** | UC016 |
| **用例名称** | 提交作业 |

图片包含 屏幕截图

描述已自动生成

顺序图 12 提交作业

后端路由收到上传作业的请求时，鉴别用户，然后调用文件储存服务保存用户上传的文件到磁盘中，并获得上传作业文件的哈希。路由依次通过作业收集组管理器、作业收集组和作业收集项，在作业收集组中创建作业实体对象，并将作文文件哈希保存入作业中，将作业对象使用对象持久化服务持久化到数据库中。最后向前端返回作业上传成功消息。

|  |  |
| --- | --- |
| **用例号** | UC017 |
| **用例名称** | 公开作业 |

图片包含 屏幕截图

描述已自动生成

顺序图 13 公开作业

后端路由收到设置作业公开属性的请求后，依次通过作业收集组管理器、作业收集组和作业收集项定位到作业对象，并将作业对象的公开属性修改为前端所请求的属性。最后向前端返回状态修改成功。

## 平台管理子系统设计

### 用户界面

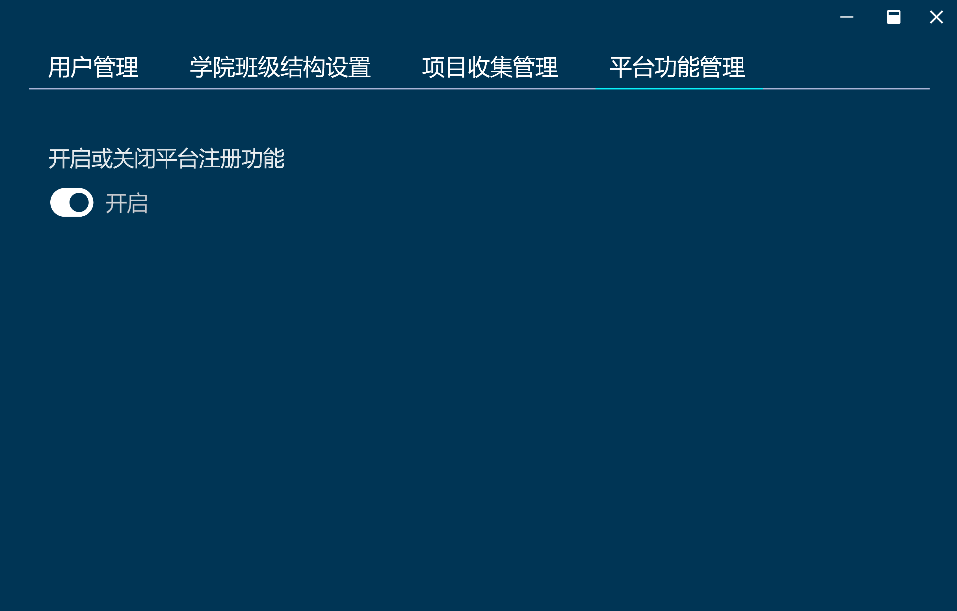


图 14 平台功能管理界面

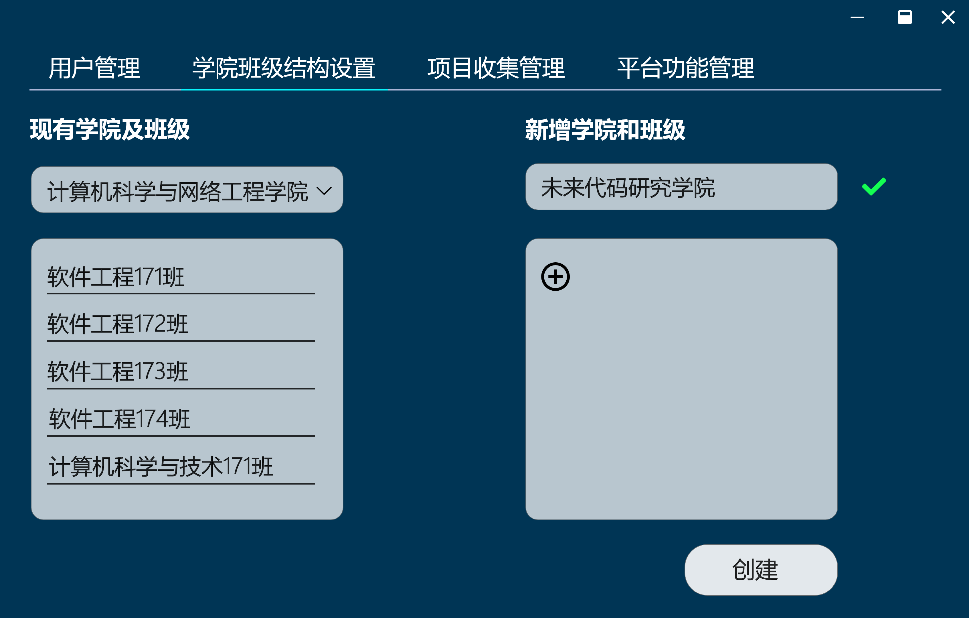


图 15 学院班级结构设置界面



图 16 用户管理界面

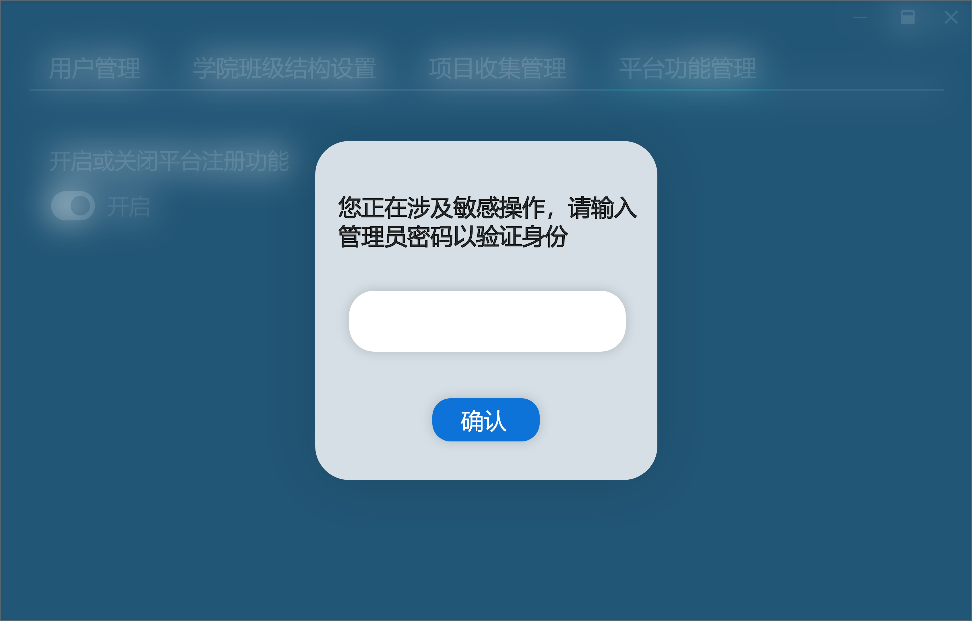


图 17 管理员密码确认界面

### 类图

图片包含 屏幕截图

描述已自动生成

类图 11 平台管理子系统静态设计

平台管理子系统包括平台管理界面类和管理员路由类，一些功能使用了中央控制器和用户管理器。

### 用例的动态设计

|  |  |
| --- | --- |
| **用例号** | UC019 |
| **用例名称** | 增设学院和班级 |

图片包含 屏幕截图

描述已自动生成

顺序图 14 增设学院和班级

路由收到要求增设的学院和班级信息时，通过会话凭证判断当前操作的用户是否为管理员。若为管理员，则调用学院管理器创建学院对象结构。若同名的学院对象已经存在，则不进行创建操作，直接返回已经存在的学院对象。路由获得返回的学院对象后，使用学院对象按照请求的请求信息创建下辖班级。若班级已经存在，向前端返回班级已存在消息；否则向前端返回创建成功消息。

|  |  |
| --- | --- |
| **用例号** | UC021 |
| **用例名称** | 新增用户 |

图片包含 屏幕截图

描述已自动生成

顺序图 15 新增用户

后端路由收到新增用户请求后，鉴别请求发起方的管理员身份，然后调用用户管理器创建新用户。若具有相同注册邮箱的用户已经存在于系统中，则向前端返回用户已存在消息。若用户创建成功，则向前端返回用户创建成功消息。

|  |  |
| --- | --- |
| **用例号** | UC024 |
| **用例名称** | 开启或关闭平台注册功能 |

图片包含 屏幕截图

描述已自动生成

顺序图 16 开启或关闭平台注册功能

管理员可以控制平台是否能接受新用户注册。当后端路由收到变更平台注册功能的请求时，首先验证请求者的身份。验证通过后，从硬盘中加载程序现有的配置文件，根据用户请求修改配置文件中平台注册功能的开关。随后将配置文件写入硬盘。最后调用中央控制器重新加载配置文件，完成平台注册功能的调整。