



# 嘉应学院土木工程学院

## 基于 ThinkPHP 的智慧作业管理系统 的设计和实现

专业班级 : 土木 2005

姓 名 : 沈锐清

学 号 : 201120196

指导老师 : 蔡华龙

日 期 : 2023 年 7 月 5 日

## 摘要

本课题旨在对智慧作业管理系统的研发和设计，主要阐述了智慧作业管理系统中教师后台管理系统和学生前台作业提交系统从 UI 设计、页面布局、各个功能的实现。伴随计算机技术的快速发展，信息化管理已经成为趋势，智慧作业管理系统可以让教师快捷地建立起对多个教学班的学生管理，作业管理，成绩管理，可以让学生通过 Web 网站加入班级门户，即可接收到教师布置的任务，同时学生将完成的成果上传到教师后台管理系统，即可获得作业的成绩，教师通过多次布置作业，获得学生的成绩并导出。信息化管理平台已经成为新时代教学管理的必然趋势。

本课题是依据教师在后台管理和学生在前台门户提交作业过程中的实际需要，在研究了各种 Web 框架之后，前端选用 LayUI 组件技术搭建前台页面、后端选用 ThinkPHP 框架技术和 WAMP 组合框架构建的后端平台、MySQL 作为数据存储的服务器。首先，本文对信息化管理系统的发展进行了分析，介绍了信息化管理平台的优势，进而介绍了系统开发过程中用到的 PHP、MySQL 等技术做了介绍，重点分析了 ThinkPHP 框架技术及 MVC 模式；然后，对系统的功能性需求通过例图的方式进行说明，分析了该系统的各大业务模块的设计，实现了学生作业提交系统提供了学生登录，获取教师布置的作业任务并支持上传提交，查看同班同学作业详情及对其进行评分等功能，教师后台管理系统提供了教师登录、邮件服务、班级创建、学生导入、作业任务布置、成绩自动计算及导出等功能，并对每个功能界面进行说明。

整个系统从符合界面友好，操作简洁，灵活、实用、安全的要求出发，使用了主流的 Ajax 异步提交技术，实现各个页面板块的数据读取，存储，简洁大方的网页设计，满足教师与学生之间的交流；实现方便管理要求、方便教师创建班级、导入学生、发布作业内容和要求、导出成绩等等，满足教师和学生对信息化平台的需求。

关键词：网站管理；ThinkPHP；MySQL

## 目录

摘要	2
目录	3
第一章 绪论	5
1.1 研究背景与意义	5
1.2 本文的章节安排	5
第二章 系统集成开发环境	6
2.1 整体开发环境	6
2.1.1 PHP	6
2.1.2 MySQL	7
2.2 ThinkPHP 开发框架	8
2.2.1 概述	8
2.2.2 体系结构	8
2.3 MVC 设计模式	9
第三章 智慧作业管理系统的需求分析	10
3.1 教师后台管理系统需求分析	10
3.2 学生前台作业提交系统需求分析	11
第四章 智慧作业管理系统的设计	12
4.1 教师后台管理系统设计	12
4.1.1 后台教师登录模块	12
4.1.2 班级管理模块	13
4.1.3 学生管理模块	15
4.1.4 作业布置模块	18
4.1.5 成绩管理模块	21
4.2 学生前台作业提交系统设计	22
4.2.1 系统登陆模块	22
4.2.2 作业上传模块	23
4.2.3 作业评分模块	25
4.3 数据库设计	27
4.3.1 教师账户数据表设计	27
4.3.2 班级数据表设计	28
4.3.3 学生数据表设计	28
4.3.4 作业上传模块数据表设计	29
4.3.5 学生数据表设计	30

4.4 系统架构设计 .....	31
第五章 智慧作业管理系统的实现 .....	32
5.1 系统的环境搭建 .....	32
5.1.1 软件安装目录 .....	32
5.1.2 搭建开发环境 .....	32
5.2 组件库和静态文件的实现 .....	44
5.2.1 Layui 组件库完成前台页面 .....	44
5.2.2 公共文件部署 .....	44
5.3 前端页面模板的实现 .....	47
5.3.1 教师后台管理页面模板 .....	47
5.3.2 学生作业提交页面模板 .....	118
5.4 后端功能模块的实现 .....	154
5.4.1 教师后台管理模块 .....	154
5.4.2 学生作业提交模块 .....	200
第六章 结论与展望 .....	220

# 第一章 绪论

## 1.1 研究背景与意义

随着信息技术和网络技术的不断发展，纸质作业的形式很多被电子作业的形式替代，这对作业的提交和管理提出了新的要求。拥有信息化平台的教师可以通过系统完成作业布置、作业汇总和分析、作业批量下载、作业提交设置和课程相关知识点发布等功能，学生通过系统可以完成作业提交、历史作业下载和查看课程相关知识点等功能，系统比可以使电子作业的管理更加专业，更流程化。

目前市场上常见的系统开发语言主要有：ASP 和 PHP，但由于微软逐渐放弃了对 ASP 的技术支持，所以大多数企业网站管理系统的开发语言都开始转向 PHP。目前，市场上以 PHP 为开发语言的几款主流的后台管理系统，如 SPB、Ucenter home、phpwind、CmsTop、PhpCms，其中 CmsTop 就是专门为大中型网站提供系统解决方案的一种内容管理系统，而 SPB 等都是主要用于 sns 社区网站的管理系统。这其中，大多数后台管理系统为英文界面，并且有很多操作都需要有专业计算机技术，如果管理员没有 html、css 等代码基础就可能无法正常使用。因此，从技术实现来看，在众多的设计方案中，选择使用 ThinkPHP 框架进行开发是最为合理的。考虑到 ThinkPHP 框架在开发成本和可移植性上的优越性，本课题决定使用 ThinkPHP 来开发智慧管理作业管理系统。Ajax 结合 JavaScript 和 xml 技术在 web 应用中可以带来更友好的用户体验，所以在 ThinkPHP 中添加使用 Ajax 技术，在实现所有功能的同时，还处处体现良好的用户交互体验。

## 1.2 本文的章节安排

第一章主要介绍课题的研究背景，另外简单介绍一下本文的章节安排。

第二章主要对智慧作业管理系统中应用到的相关技术和理论进行介绍。

第三章主要对智慧作业管理系统中的学生门户网站及网站后台管理系统进行需求分析，并详细介绍了系统的工作流程。

第四章详细介绍了智慧作业管理系统的门户网站及网站后台管理系统的总体框架设计、模块划分及界面设计和数据库设计等。

第五章主要介绍智慧作业管理系统门户网站的实现。

第六章总结课题的结果和对今后系统的展望。

## 第二章 系统集成开发环境

### 2.1 整体开发环境

WAMP 是指在 Windows 服务器上使用 Apache、MySQL 和 PHP 的集成安装环境，可以快速安装配置 Web 服务器，Windows 操作系统有其优点，就是易用，界面友好，软件丰富，操作起来非常方便；Apache 是最通用的网络服务器；MySQL 是带有基于网络管理附加工具的关系数据库；PHP 是流行的对象脚本语言，它包含了多数其它语言的优秀特征来使得它的网络开发更加有效，本身都是各自独立的程序，但是因为常被放在一起使用，拥有了越来越高的兼容度，共同组成了一个强大的 Web 应用程序平台。

同时需要配置相应的软件，搭建开发环境！如图 1 所示：

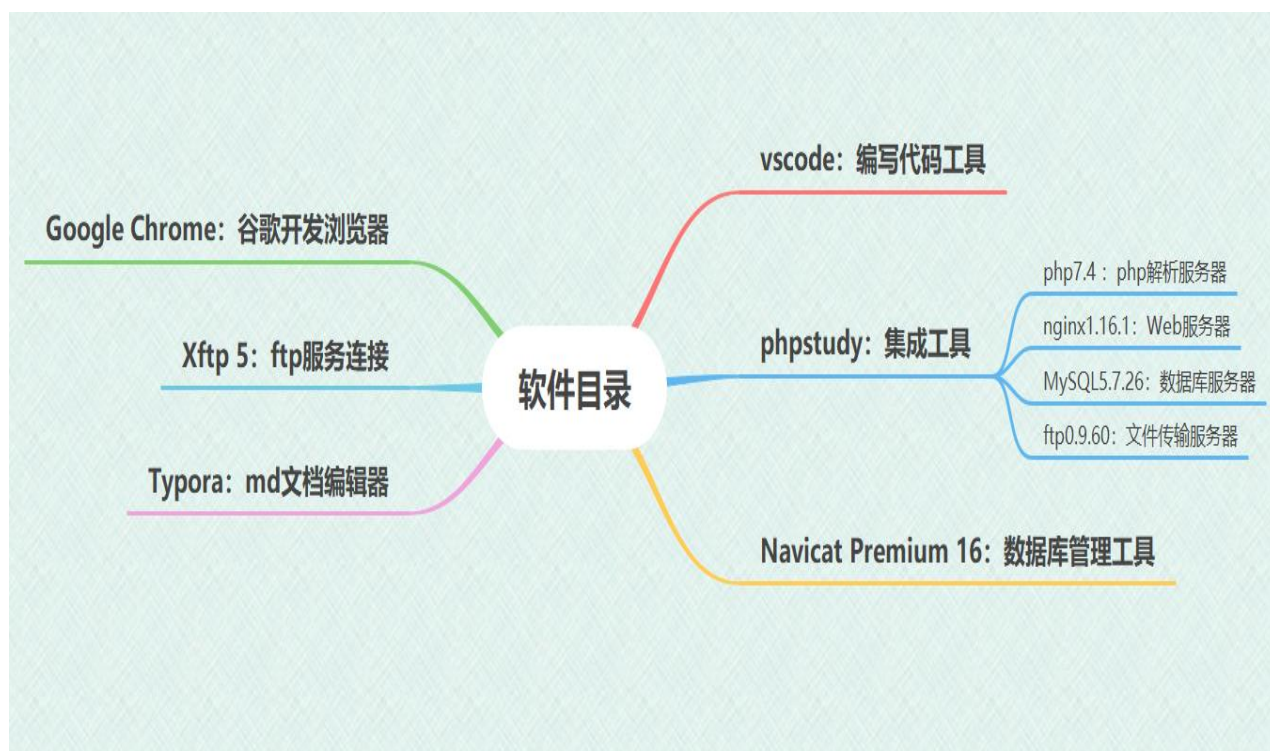


图 2 搭建开发环境软件目录参考

#### 2.1.1 PHP

PHP 是一种 HTML 嵌入式的脚本语言，主要适用于 Web 开发领域。于 1995 年由 Rasmus Lerdorf 创建，最开始是为了统计自己的个人网站的访问量，使用了 Perl 语言进行编写，后来又用 C 语言进行了重新编写。PHP 独具一格的语法混合了 C 语言、Java 语言、Perl 语言以及 PHP 自创的语法。它可以比 CGI 或者 Perl 更快速地执行动态网页。

PHP 之所以应用广泛，受到大众欢迎的原因是因为它有很多特性：

(1) 和其他技术相比 PHP 是开源的，并且可以免费使用，所有的 PHP 源代码都可以免费得到；

(2) PHP 的跨平台性好，方便移植，在 Linux 和 Windows 平台都可以运行；

(3) 由于 PHP 面向对象的特性，使用 PHP 进行 Web 开发时，可以选择使用面向过程或是面向对象方式编程，在 PHP5 以后在面向对象方面都有了很大的改进；

(4) 由于 PHP 支持 ODBC，因此 PHP 可以连接任何支持该标准的数据库，如 MySQL、Oracle、SQL Server 等。其中，PHP 与 MySQL 是最佳搭档，应用广泛。

### 2.1.2 MySQL

MySQL 是一个最为典型的关系型数据库管理系统，并且是一款开源的系统软件。SQL 语句是 MySQL 数据库管理系统最常用的语句，可以用来对数据库进行管理。MySQL 数据库能支持多用户、多线程的服务，这就很好地满足了 C/S(客户机/服务器)结构软件地需求。同时，由于 MySQL 源码的稳定性和开放性，并且能与 PHP 编译语言进行完美结合，所以很多网站都将 MySQL 作为其服务器端的数据库，它也获得了很多好评，得到广泛的推广。

本课题地后台管理系统采用的就是 MySQL 数据库，其主要缘由为：

(1)MySQL 数据库是一种开源免费的轻量级数据库，其操作简单、使用方便，拥有完善的后台日志记录功能，能同时支持 Windows 和 Linux 操作系统，并且在不同数据库之间的可移植性较好。

(2)用户可以通过 Internet 来访问 MySQL 数据库，以此实现数据共享，同时能适应较低的网络带宽，使用者无需花销很多就能够安全稳定的使用这个数据库。

(3)MySQL 采用多层独立模块设计，有助于数据库的稳定性，权限系统能控制登录用户访问数据库表，还可以对数据库表操作设置相应权限。用户的权限完全可以通过 MySQL 服务器来设定，根据用户名和主机名来确定登录用户的身份，是目前常见数据库中最稳定的数据库系统之一。

(4)用于管理 MySQL 的工具特别多，因此数据库的恢复及备份功能都能够轻松完成，整个流程操作简单，容易学习。

(5)MySQL 还包含多种的数据存储引擎，如 InnoDB、MyISAM 等，给用户提供了丰富的选择，根据不同使用场景切换不同引擎，提高检索、排序、存储和搜索数据的性能。

## 2.2 ThinkPHP 开发框架

### 2.2.1 概述

诞生于 2006 初的 ThinkPHP 框架是一个开源免费的框架，原名叫 FCS，于 2007 年改名为 ThinkPHP。以 Apache2 开源协议为基础发布，最开始的设计目标是为了简化企业级应用开发流程，提供一种 WEB 应用的敏捷开发方式。效仿 Java 的 struts 框架思想形成早期的设计思路，后来通过借鉴许多国内外优秀的框架结构，最终形成 ThinkPHP 后续版本。

### 2.2.2 体系结构

ThinkPHP 框架设计的基本思路主要包括以下几方面：

(1)提供友好的 URL 机制 该框架是通过控制名+方法名+参数名的方式进行访问的，整个系统从统一的入口文件 `index.php` 进入，所有的过程实现都要依赖于 URL。它所支持的 URL 模式主要包括普通模式、兼容模式、PATHINFO 模式和 REWRITE 重写模式四种，由于其友好、易用的 URL 模式，给开发人员和使用者提供了良好的使用体验。

(2)框架基于 MVC 模式设计，却不局限于 MVC 模式 ThinkPHP 整体是严格遵循 MVC 模式的，Model 层由 Model 类完成，View 层由 Views 类和模板文件来实现，Controller 层由核心控制器 APP 类和 Aciton 类共同完成。通常情况下，如果用户没有自定义 Action 类和 Model 类，仅通过核心控制器类，系统应用依旧可以完整实现，所以说该框架虽然是基于 MVC 模式，但是不完全拘泥于 MVC 模式的限制。

(3)拥有良好的可移植性，并与多种数据库支持 Think PHP 框架就是基于 LAMP 架构来进行开发的，因此可以使用 Linux 作为操作系统平台，但这并不意味着只能使用 Linux，其也能很好地支持 Windows 平台，所以由 ThinkPHP 开发的系统应用都具有很好的可移植性。与此同时，ThinkPHP 能支持市面上各种主流的数据库，如 MySQL、Pgsql、Oracle 等，只需要通过对配置文件制定不同的设置参数，就可以实现与数据库相连接，进行数据访问交互。

(5)强大的系统自带基类库 ThinkPHP 框架的 Lib 目录下存放了所有的基类库，其中除了一些保证系统框架能正常运行的核心类库外，还包括 ORG、COM 等一些扩展类库。核心类库有数据库类、内置模板引擎类库、系统工具类库、异常处理类库等，这些强大的类库都为框架功能的完整性和健壮性提供了坚实的保证。



## 2.3 MVC 设计模式

MVC (Model-View-Controller) 模式是软件工程中的一种软件架构模式，它把软件系统分为三个基本部分：模型 (Model)、视图 (View) 和控制器 (Controller)。

MVC 的框架设计能很好地促进代码的复用，使程序变得易于扩展，而且随着程序复杂性的增加，这个优点会体现的越来越明显。现在基于 MVC 的模式设计层出不穷，甚至还提出了 MVVM、MVP 等各种组合，但究其本质都是 Model、View、Controller 三层。况且 MVC 模式的包容度比 MVVM 和 MVP 更好。

本文选用的 ThinkPHP 框架就是 MVC 架构，具体实现流程如图 3 所示：

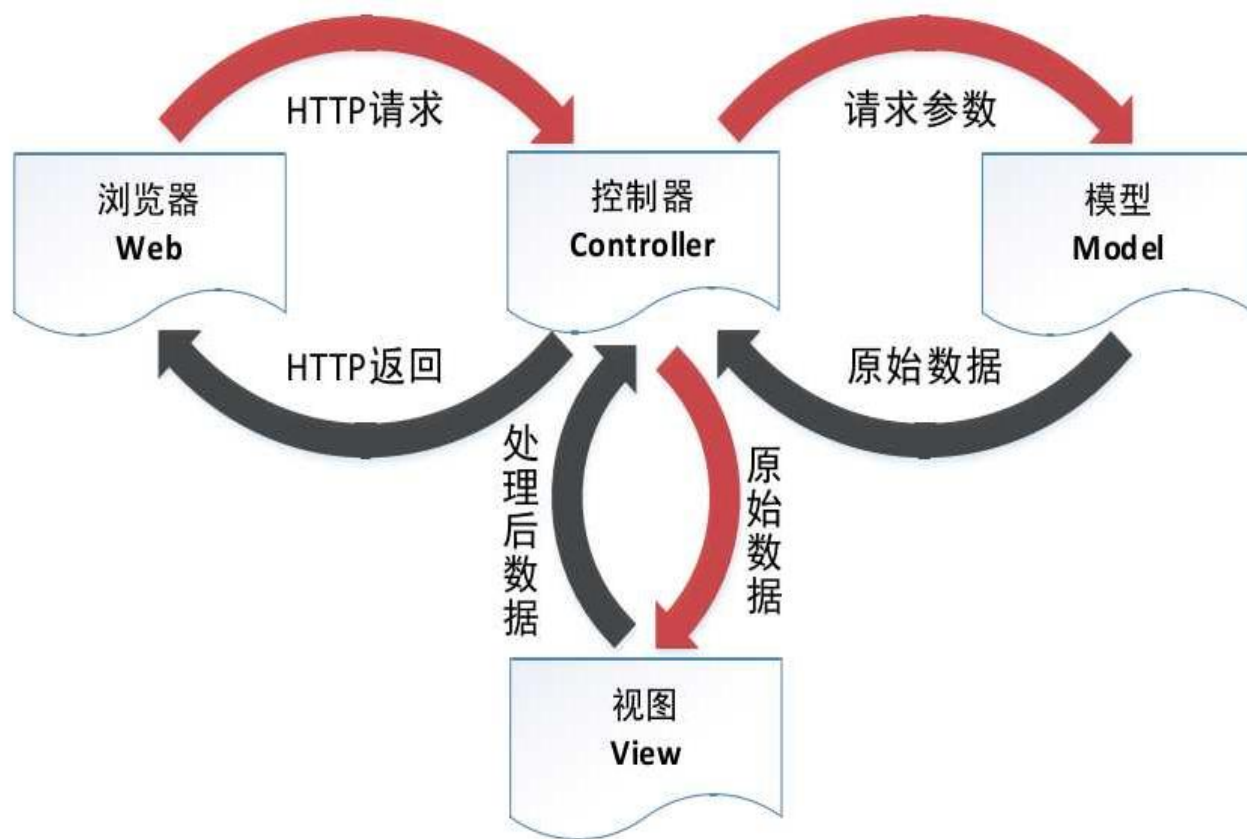


图 3 MVC 架构实现的原理图

## 第三章 智慧作业管理系统的需求分析

### 3.1 教师后台管理系统需求分析

通过实际情况的调查，要求教师后台管理系统应当具有以下功能：

- (1) 后台登录功能；
- (2) 管理教师个人信息，邮箱服务器功能；
- (3) 管理班级权限，维护班级信息，导入导出班级功能；
- (4) 管理学生信息，维护学生信息，导入导出学生功能；
- (5) 管理学生作业，维护学生作业，导出作业文件功能；
- (6) 管理学生成绩，维护学生成绩，导出成绩文件功能。

通过实际情况的调查，应计划设计如下模块：

- (1) 后台教师登录模块：教师通过输入账号密码，方可进入系统；
- (2) 班级管理模块：教师可同时设立多个班级并设立各个班级的状态，开启或关闭班级成员进入作业系统的权限；
- (3) 学生管理模块：教师可通过 excel 实现学生的导入导出功能，自动生成密码，通过邮箱系统一键推送账户信息给学生，同时支持单独维护学生信息；
- (4) 作业布置模块：教师可定时发布作业，上传作业的资料，限制作业上传的截止时间，支持学生作业导入导出；
- (5) 成绩管理模块：教师可根据学生的相互打分，系统通过云计算输出每个学生的成绩，教师可查看并导出最终的成绩，并支持一键录入教务系统。

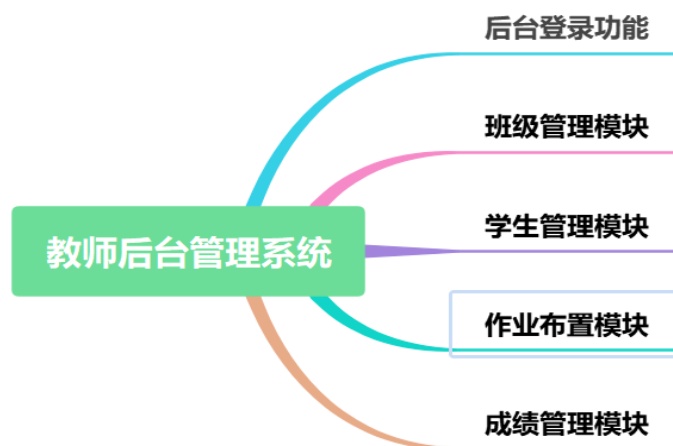


图 4 教师后台管理系统模块

通过实际情况的调查，应达到如下目标：

- (1) 操作简单方便，并且应当附有帮助手册；
- (2) 用户不能重复登录，管理员可修改个人信息；

- (3) 支持发布信息，通知学生完成作业；
- (4) 系统运行稳定、安全可靠。

### 3.2 学生前台作业提交系统需求分析

通过实际情况的调查，要求学生前台作业提交系统应当具有以下功能：

- (1) 友好的交互界面；
- (2) 学生登录功能；
- (3) 学生个人信息，班级信息查看功能；
- (4) 学生提交作业功能，提交文件应支持常见的图片格式，PDF、HTML、CSS、JavaScript、PHP、SQL、txt 等；
- (5) 学生可查看同班同学成绩并相互评分功能。

通过实际情况的调查，应计划设计如下模块：

- (1) 系统登陆模块：教师在后台一键创建学生用户，学生根据教师推送的账户信息，登入系统，方可获得课程的信息；
- (2) 作业上传模块：学生在获得教师推送的作业后，通过文件上传系统和教师的要求，提交对应的作业；
- (3) 作业评分模块：学生在提交完作业后，拥有查看本班同学作业的权限，并可以通过相互评分获得平时成绩，评分有 10 颗星，学生可查看其他同学成绩后做出评分，同时也可以邀请其他同学给自己的作业评分；



图 5 学生前台作业提交系统模块

应达到如下目标：

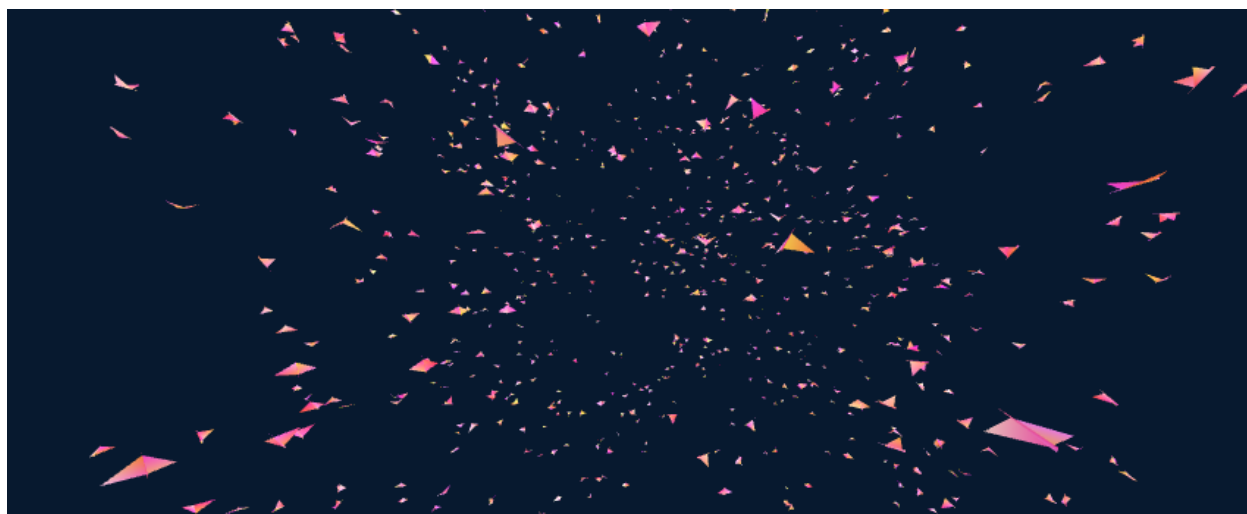
- (1) 采用开放的、动态的系统架构加强网站与学生的动态交互；
- (2) 操作简单方便，并且应当附有帮助手册；
- (3) 学生信息由教师维护，增强系统可维护性；
- (4) 学生上传文件进行校验，过滤非法文件，增强系统安全性。

## 第四章 智慧作业管理系统的设计

### 4.1 教师后台管理系统设计

#### 4.1.1 后台教师登录模块

教师需要在后台 Web 端输入正确的账号密码，以及验证码，才能进入系统，获得作业管理的权限。



### 智慧作业管理系统后台

请输入管理员账户

请输入密码

请输入验证码

登录

图 6 后台教师登陆页面  
第 12 页 / 共 220 页

## 4.1.2 班级管理模块

教师拥有添加、修改、删除、导出班级信息的权限。

### 班级管理门户【ruoshui】

添加班级    个人信息

**注意:**

个人信息的**邮箱**是发送短信的邮箱**请自行修改**

班级代码只支持**数字**命名, 班级一旦生成, 班级代码将**无法修改**

可以设置**班级状态**:

①**正常**表示课程正在进行!

②**结课**学生将无法进入系统!

班级代码	课程名称	班级创建时间	班级状态	访问班级	操作
2005	管理信息课程	2023-04-02 17:52:50	正常	<a href="#">进入班级</a>	<a href="#">修改</a> <a href="#">删除</a>
202307	公选课第七期	2023-04-03 16:49:16	正常	<a href="#">进入班级</a>	<a href="#">修改</a> <a href="#">删除</a>

ThinkPHP  
以及勤劳的自己

图 7 班级管理主页

班级管理添加

班级代码 例: 2005

课程名称 例: 管理信息课程

班级状态  开启

课程信息 例: 公楼C404 周二上午一二节

课程介绍  
请输入内容课程介绍

保存

ThinkPHP  
以及勤劳的自己

图 8 班级信息添加页面

班级修改 - [ ] X

班级代码	2005
课程名称	管理信息课程
班级状态	<input type="radio"/> 关闭
课程信息	公楼C404第1-2节
课程介绍	本课程旨在于培养学生的计算机兴趣

[保存](#)

图 9 班级信息修改页面

教师可以维护账户的信息，设置邮箱系统的相关配置，获得推送邮件的能力。

个人信息管理 - [ ] X

建议使用163邮箱  
登录网页版163邮箱，点击设置，开启IMAP/SMTP服务  
扫码发送短信后，获取到授权密码并输入  
SMTP服务器: smtp.163.com

### 个人信息维护

用户名	ruoshui
密码	ruoshui123456
邮箱	srq13415151737@163.com
SMTP服务	smtp.163.com
SMTP秘钥	VPQSVJDCJQBCBJQA

[保存](#)

图 10 后台管理员账户维护页面

教师进入班级后，可跳转到系统的核心管理界面，维护班级的信息。



图 11 班级信息展示页面

### 4.1.3 学生管理模块

教师拥有一键导入、一键清空、单独添加、修改、删除和导出学生的权限。同时可以在创建学生信息后，一键或单独推送学生信息到学生的邮箱。



图 12 学生管理主页



图 13 学生一键导入页面

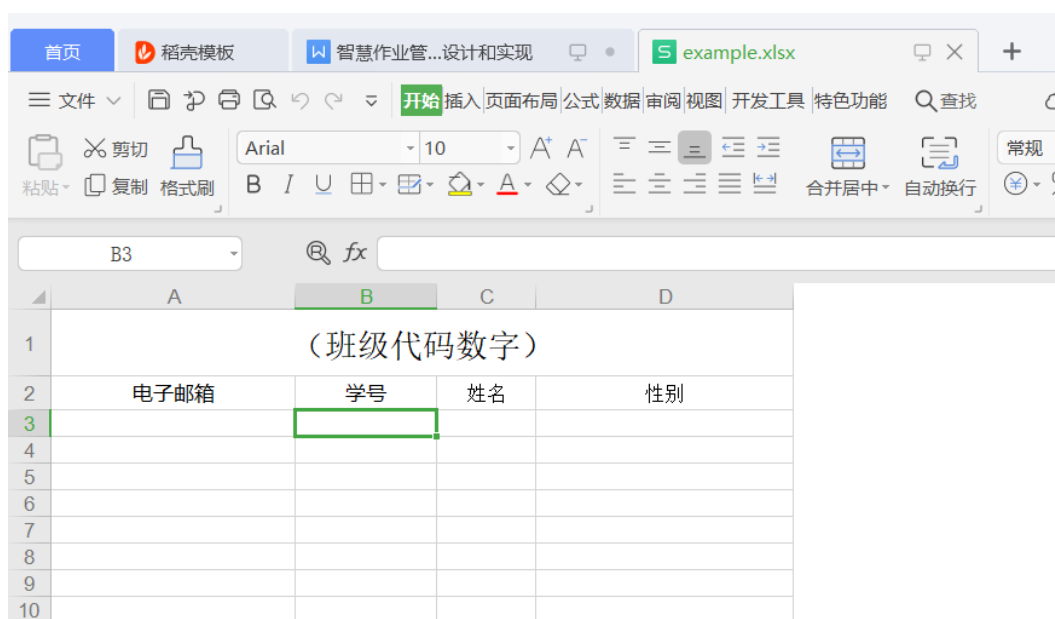


图 14 学生一键导入 Excel 模块页面



图 15 学生单独添加页面





图 16 学生单独修改页面



图 17 学生收到邮件信息

## 4.1.4 作业布置模块

教师拥有布置作业、修改作业状态、删除作业、导出作业、查看作业效果、打分详情的权限。



图 18 作业布置主页

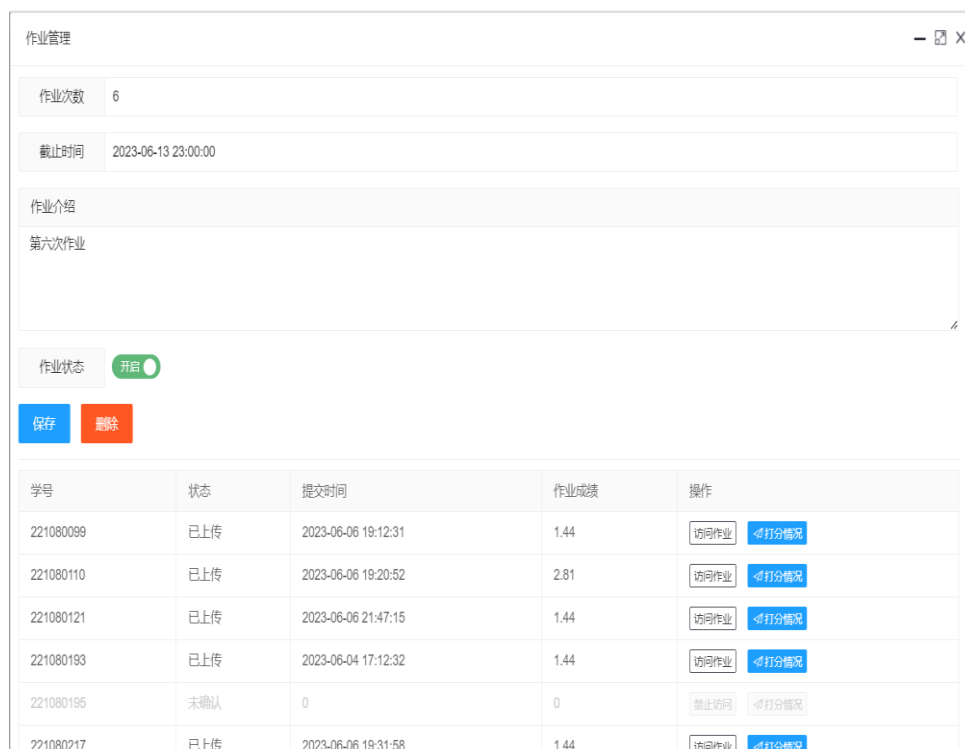


图 19 作业详情和修改页面



图 20 作业布置页面

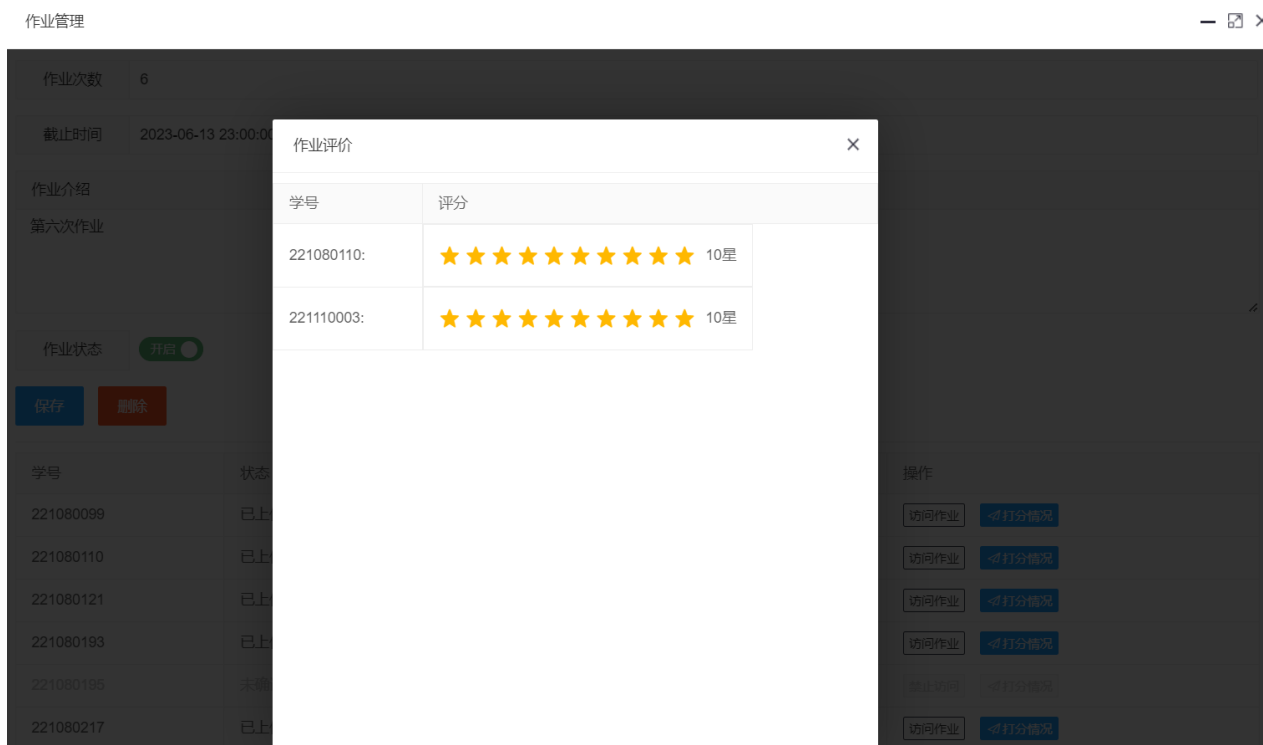


图 21 作业评分详情参考页面

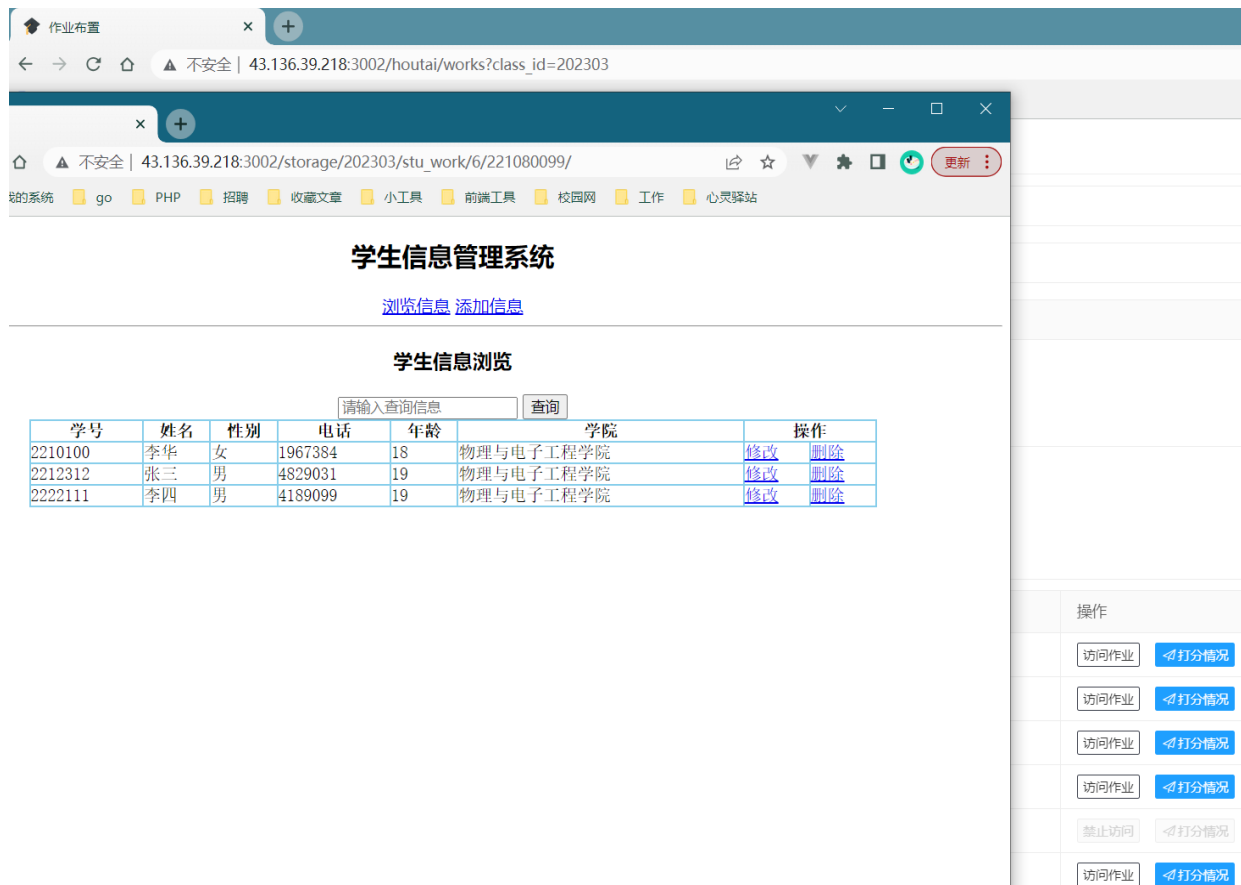


图 22 作业访问详情页面

## 202303班作业管理后台【ruoshui】



图 23 作业导出页面

### 4.1.5 成绩管理模块

教师可获得所有布置完成作业的分数汇总，且可导出成绩。



图 24 成绩自动计算模块

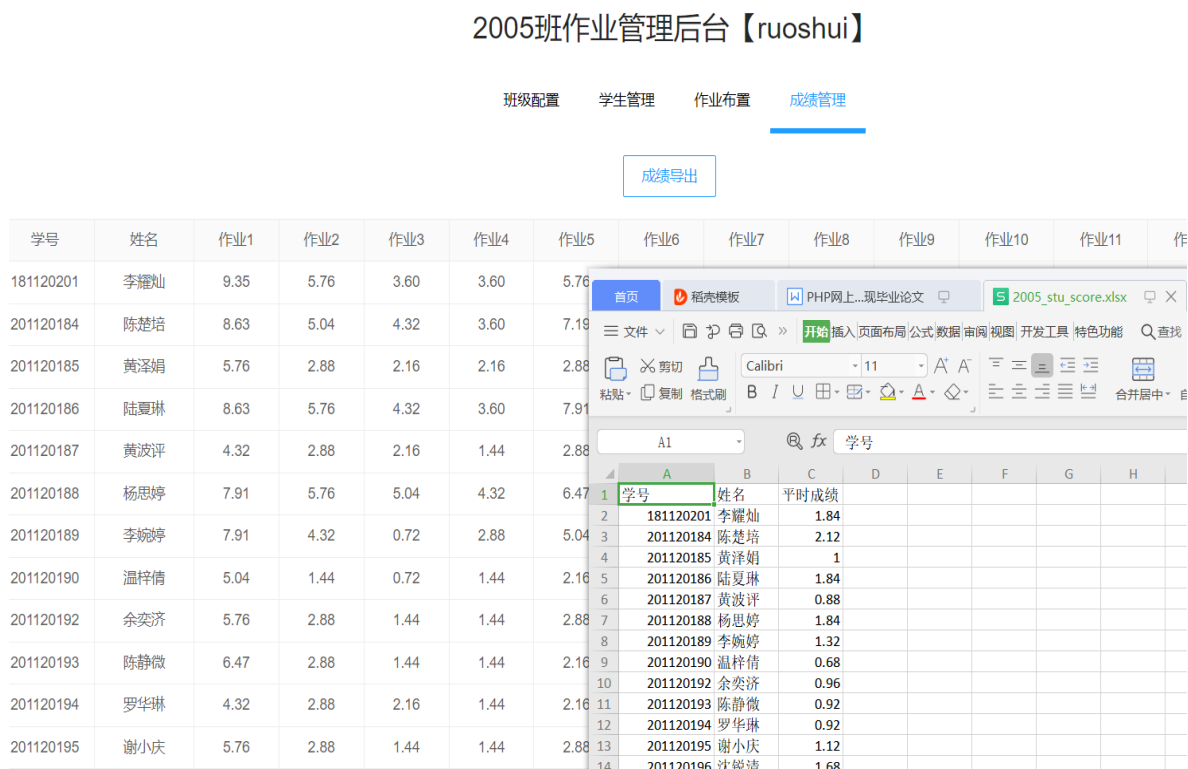
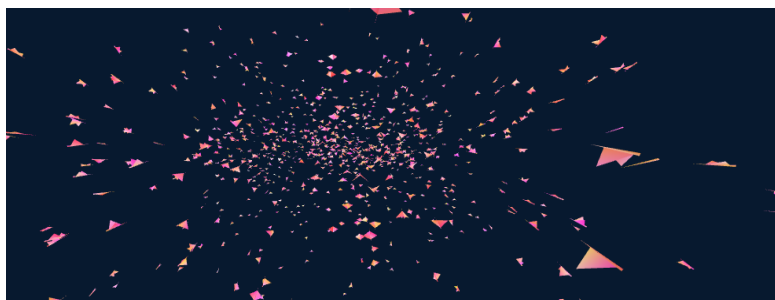


图 25 成绩导出模块

## 4.2 学生前台作业提交系统设计

### 4.2.1 系统登陆模块

学生在获得邮件推送的信息后，根据提示可在 Web 端登录个人页面，登录成功可获得班级的信息，以及上传作业、查看作业、评分作业的权限。



 作业上传评分系统学生门户



图 26 学生系统登陆模块

### 2005班作业上传评分系统【hello~沈锐清】

[学生主页](#)   [作业上传](#)   [作业展示](#)

学号	201120196
姓名	沈锐清
性别	男
登录时间	2023-07-30 14:13:35
课程名称	管理信息课程
课程信息	公楼C404第1-2节
课程介绍	
本课程旨在培养学生的计算机兴趣	

图 27 学生门户主页面

## 4.2.2 作业上传模块

在教师布置完作业后，学生将获得三种作业状态：未确认、未提交、已提交。

学生进入作业后，可以获得作业的布置信息，作业上传的权限；学生只有在截止时间之前才可提交作业；学生只有提交完成作业后，其他同学才可以查看并评分学生作业的效果，学生才可获得对应的成绩；学生提交完成作业后，系统后台根据实际情况赋予分值；学生在作业截止时间之后，无法提交和修改作业，学生的作业将被永久保存。

### 2005班作业上传评分系统【hello~沈锐清】



图 28 作业上传模块主页



图 29 作业上传页面【上传时状态】



图 30 作业上传页面【提交后状态】





图 31 作业访问页面

### 4.2.3 作业评分模块

学生可以访问同班同学作业，并给予一定的评分。未提交作业的同学将不能够被评分。

#### 2005班作业上传评分系统【hello~沈锐清】



图 32 作业评分主页

学号	状态	提交时间	评价人数	评价状态	操作
181120201	已上传	2023-05-30 20:03:30	0	未评价	访问作业 < 打分情况
201120184	已上传	2023-05-30 08:29:19	0	未评价	访问作业 < 打分情况
201120185	已上传	2023-05-30 21:49:32	0	未评价	访问作业 < 打分情况
201120186	已上传	2023-05-30 09:20:02	0	未评价	访问作业 < 打分情况
201120187	已上传	2023-06-04 18:42:39	1	已评价	访问作业 < 打分情况
201120188	已上传	2023-06-01 11:41:35	0	未评价	访问作业 < 打分情况
201120189	已上传	2023-06-11 22:25:32	0	未评价	访问作业 < 打分情况
201120190	已上传	2023-06-05 01:44:47	0	未评价	访问作业 < 打分情况
201120192	未确认	0	0	未评价	禁止访问 < 打分情况
201120193	已上传	2023-06-05 20:36:33	0	未评价	访问作业 < 打分情况
201120194	已上传	2023-06-05 20:15:19	0	未评价	访问作业 < 打分情况
201120195	已上传	2023-06-05 20:33:55	1	未评价	访问作业 < 打分情况
201120196	已上传	2023-05-29 22:25:45	0	未评价	访问作业 < 打分情况
201120197	未确认	0	0	未评价	禁止访问 < 打分情况

图 33 同班作业查看页面

学号	状态	提交时间	评价人数	评价状态	操作
181120201	已上传	2023-05-30 20:03:30	0	未评价	访问作业 < 打分情况
201120184	已上传				
201120185	已上传				
201120186	已上传				
201120187	已上传				
201120188	已上传				
201120189	已上传				
201120190	已上传				
201120192	未确认				
201120193	已上传				
201120194	已上传				
201120195	已上传				
201120196	已上传				
201120197	未确认				

一、单选题

1. MySQL 是一个\_。

A. 关系型数据库管理系统

B. 非关系型数据库管理系统

C. 文件存储系统

D. 操作系统

答案: A

2. 在 MySQL 中, 哪个命令用于创建数据库?

A. CREATE DATABASE

B. NEW DATABASE

C. ADD DATABASE

D. DATABASE CREATE

答案: A

图 34 查看同班同学作业查看页面

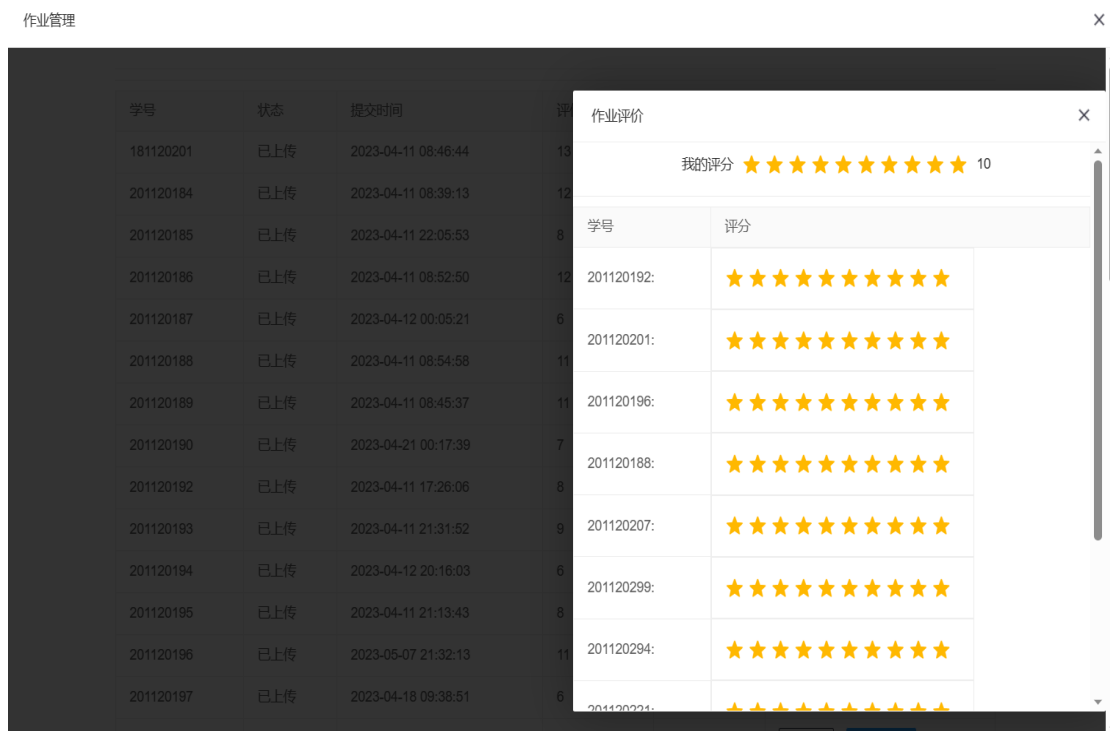


图 35 同班作业评分页面

## 4.3 数据库设计

### 4.3.1 教师账户数据表设计

<b>admin</b>	
管理员信息表	
<b>id: int</b>	主键
<b>uname: varchar(255)</b>	用户名
<b>pwd: varchar(255)</b>	密码
<b>nickname: varchar(255)</b>	管理员名称
<b>email_system: varchar(255)</b>	邮箱系统
<b>email: varchar(255)</b>	邮箱
<b>check_code: varchar(255)</b>	邮箱校验码
<b>add_time: datetime</b>	添加时间
<b>last_time: datetime</b>	修改时间

图 36 教师账户数据表设计

### 4.3.2 班级数据表设计

<b>classes</b> 班级表	
<b>id: int UNSIGNED</b>	主键
<b>class_id: int</b>	班级代码
<b>class_name: varchar(255)</b>	课程名称
<b>class_time: varchar(255)</b>	班级备注信息
<b>class_remark: varchar(255)</b>	课程介绍

图 37 班级数据表设计

### 4.3.3 学生数据表设计

<b>stu</b> 学生表	
<b>id: int</b>	主键
<b>class_id: int</b>	班级代码
<b>stu_no: int</b>	学生学号
<b>stu_pwd: varchar(255)</b>	学生密码
<b>stu_name: varchar(255)</b>	学生姓名
<b>gender: tinyint</b>	性别 <b>0</b> 男 <b>1</b> 女
<b>email: varchar(255)</b>	邮箱
<b>add_time: datetime</b>	添加时间
<b>last_time: datetime</b>	修改时间

图 38 学生数据表设计

### 4.3.4 作业上传模块数据表设计

<b>works</b>	
作业文件路径表	
<b>id: int</b>	主键
<b>class_id: int</b>	班级代码
<b>work_id: int</b>	作业代码
<b>stu_no: int</b>	学生学号
<b>filename: varchar(255)</b>	文件名称
<b>work_url: varchar(255)</b>	文件路径
<b>start_time: datetime</b>	添加时间

图 39 作业文件路径表

<b>is_work</b>	
作业确定上传中间表	
<b>id: int</b>	主键
<b>class_id: int</b>	班级代码
<b>work_id: int</b>	作业代码
<b>stu_no: int</b>	学生学号
<b>is_true: varchar(255)</b>	是否上传 <b>0</b> 未上传 <b>1</b> 已上传
<b>last_time: varchar(255)</b>	上传作业时间

图 40 作业确定上传中间数据表

<b>work</b>	
作业布置表	
<b>id: int</b>	主键
<b>class_id: int</b>	班级代码
<b>work_id: int</b>	作业代码
<b>work_remarks: varchar(255)</b>	作业备注
<b>work_start_time: datetime</b>	作业创建时间
<b>work_last_time: datetime</b>	作业截止时间
<b>status: tinyint</b>	作业状态 <b>0</b> 正常 <b>1</b> 关闭

图 41 作业布置数据表

#### 4.3.5 学生数据表设计

<b>score</b>	
作业评分表	
<b>id: int</b>	主键
<b>class_id: int</b>	班级代码
<b>work_id: int</b>	作业代码
<b>stu_no: int</b>	作业所属的学生学号
<b>to_stu_no: int</b>	参与评价的学生
<b>score: int</b>	评价分数
<b>start_time: datetime</b>	评价时间

图 42 作业评分数据表

## 4.4 系统架构设计

智慧作业管理系统采用 ThinkPHP6.1.1 框架技术，框架整体是严格遵循 MVC 模式设计的，MVC（Model-View-Controller）模式是软件工程中的一种软件架构模式，它把软件系统分为三个基本部分：模型（Model）、视图（View）和控制器（Controller）。课题采用了多应用模块，Admin 为教师后台应用模块，Home 为学生主应用模块。

```
D:\phpstudy_pro\WWW\works>php think version
v6.1.1
```

图 43 ThinkPHP 框架版本

```

├─app 应用目录
│   ├─index                主应用
│   │   ├─controller       控制器目录
│   │   ├─model            模型目录
│   │   ├─view             视图目录
│   │   ├─config           配置目录
│   │   ├─route            路由目录
│   │   └─ ...             更多类库目录
│   └─admin                后台应用
│       ├─controller       控制器目录
│       ├─model            模型目录
│       ├─view             视图目录
│       ├─config           配置目录
│       ├─route            路由目录
│       └─ ...             更多类库目录
├─public                  WEB目录（对外访问目录）
│   ├─admin.php           后台入口文件
│   ├─index.php           入口文件
│   ├─router.php          快速测试文件
│   └─.htaccess           用于apache的重写
├─config                  全局应用配置目录
├─runtime                 运行时目录
│   ├─index               index应用运行时目录
│   └─admin               admin应用运行时目录

```

图 44 多应用下系统工作目录

## 第五章 智慧作业管理系统的实现

### 5.1 系统的环境搭建

#### 5.1.1 软件安装目录

- (1) Google Chrome: 用于开发测试的浏览器;
- (2) vscode: 轻量级的代码编写工具, 拥有大量的插件支持开发;
- (3) Xftp: 轻量级的文件传输服务器, 用于将文件上传到远程服务器;
- (4) phpstudy: 集成后端语言 PHP7.4.3 解析器、数据库服务器 MySQL5.7.26、文件传输服务器 ftp0.9.60、高性能 Web 服务器 nginx1.16.1;
- (5) Typora: md 文档编辑器, 用于形成开发文档;
- (6) Navicat Premium 16: 用于管理数据库服务器;

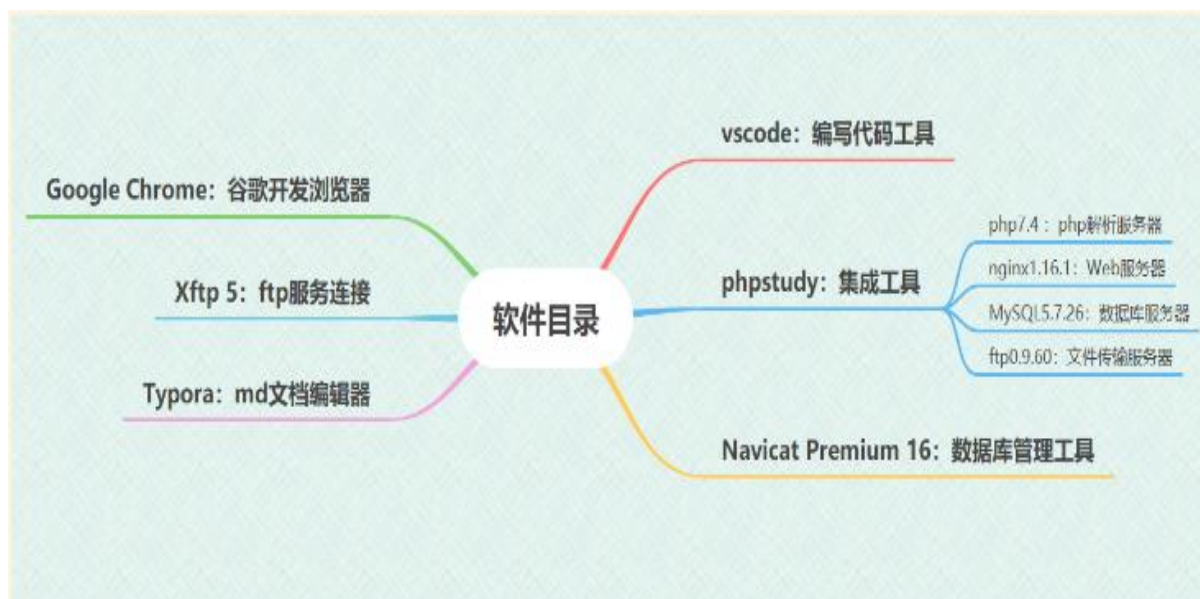


图 45 软件安装目录

#### 5.1.2 搭建开发环境

- (1) 安装 composer 包管理工具: composer 是 PHP 用来管理依赖 (dependency) 关系的工具;
- (2) 安装 ThinkPHP6 框架: 在系统的工作目录打开 cmd 命令行, 输入如下代码;

```
composer create-project tophink/think works
```



(3) 搭建本地虚拟站点，以 public 目录为入口目录

The screenshot shows the '网站' (Website) configuration window in phpstudy. The '基本配置' (Basic Configuration) tab is selected. The configuration fields are as follows:

- 域名 (Domain): works.cn
- 第二域名 (Second Domain): (empty)
- 端口 (Port): http (selected), https (unselected), 80
- 根目录 (Root Directory): D:/phpstudy\_pro/WWW/works/public
- 创建环境 (Create Environment): 创建FTP (unchecked), 创建数据库 (unchecked), 同步hosts (checked), 生产环境 (unchecked)
- 程序类型 (Program Type): PHP (checked)
- PHP版本 (PHP Version): php7.4.3nts
- 到期日期 (Expiry Date): 2028-03-14
- 备注 (Remarks): (empty)

Buttons for '确认' (Confirm) and '取消' (Cancel) are located at the bottom of the window.

图 46 phpstudy 集成根据网站配置

(4) 在工作目录 public/nginx.htaccess 下，隐藏 index.php 入口文件

```
location / {
    if (!-e $request_filename) {
        rewrite ^(.*)$ /index.php?s=/$1 last;
    }
}
```

(5) 使用 composer 包管理，安装第三方扩展类

1.安装 think-view、ThinkTemplate

```
composer require tophink/think-view
```

2.安装多应用模式扩展

```
composer require tophink/think-multi-app
```

3.安装 think-captcha 验证码扩展

```
composer require tophink/think-captcha
```

4.安装 PhpSpreadsheet 扩展（操作 excel 库）

```
composer require phpooffice/phpspreadsheet
```

(6) 在工作目录 extend/file 下，手动添加第三方类库 Zip.php

```
<?php

namespace file;

class Zip
{
    //文件夹打包
    public function zip($class_id)
    {
        $local = app()->getRootPath() . '/public';
        try {
            //文件夹目录
            $dirPath = $local . "/storage" . "/" . $class_id;
            //zip 压缩包保存路径
            $zipPath = $local . "/storage/tmp" . "/" . $class_id . ".zip";
            //创建 zip 实例
            $zip = new \ZipArchive();
            if ($zip->open($zipPath, \ZipArchive::CREATE | \ZipArchive::OVERWRITE)
            === TRUE) {
```

```

        //调用方法，对要打包的根目录进行操作，并将 ZipArchive 的对象传递给方法
        $this->addFileToZip($zip, $dirPath, $dirPath);

        //关闭处理的 zip 文件
        $zip->close();
    }
} catch (\Exception $e) {
    //echo '压缩失败';
    halt($e);
}

$this->downloadZip($zipPath);
}

//下载 function
public function downloadZip($zipPath)
{
    $zipPath = iconv("UTF-8", "GBK", $zipPath); //加这行中文文件夹也 ok 了
    header("Cache-Control: public");
    header("Content-Description: File Transfer");
    header('Content-disposition: attachment; filename=' . basename($zipPath));
//文件名
    header("Content-Type: application/zip"); //zip 格式的
    header("Content-Transfer-Encoding: binary"); //告诉浏览器，这是二进制文件
    header('Content-Length: ' . filesize($zipPath)); //告诉浏览器，文件大小
    @readfile($zipPath); //ob_end_clean();
    @unlink($zipPath); //删除压缩包
}

```

```

//压缩包追加文件

public function addFileToZip($zip, $path, $root)
{
    $handler = opendir($path); //打开当前文件夹

    while (($filename = readdir($handler)) !== false) {
        if ($filename != "." && $filename != "..") { //不操作名字为'.'和'..'的文件
            文件夹或文件

            if (is_dir($path . "/" . $filename)) { // 如果读取的某个对象是文件夹,
                则递归

                $this->addFileToZip($zip, $path . "/" . $filename, $root);
            } else {
                //将文件加入 zip 对象, 第二个参数是 zip 里文件的路径

                $pathFilename = $path . "/" . $filename;

                $zip->addFile($pathFilename, str_replace($root . '/', '',
                $pathFilename));
            }
        }
    }

    @closedir($path);
}
}

```

#### (7) 配置文件修改

①在工作目录 app/AppService.php 下, 开启 debug 模式

```

public function boot()
{
    $this->app->debug(true); // 服务启动
}

```

②在 config/view.php 下，修改 view 模板引擎后缀

```
<?php
// +-----
// | 模板设置
// +-----

return [

    // 模板引擎类型使用 Think
    'type'          => 'Think',

    // 默认模板渲染规则 1 解析为小写+下划线 2 全部转换小写 3 保持操作方法
    'auto_rule'     => 1,

    // 模板目录名
    'view_dir_name' => 'view',

    // 模板后缀
    'view_suffix'   => 'php',

    // 模板文件名分隔符
    'view_depr'     => DIRECTORY_SEPARATOR,

    // 模板引擎普通标签开始标记
    'tpl_begin'     => '{',

    // 模板引擎普通标签结束标记
    'tpl_end'       => '}',

    // 标签库标签开始标记
    'taglib_begin'  => '{',

    // 标签库标签结束标记
    'taglib_end'    => '}',

    //配置全局变量 //配置模板预定义常量
    'tpl_replace_string' => [
        '{__CSS_PATH__}' => "/static/css",
        '{__JS_PATH__}'  => "/static/js",
    ],
];
```

```
'{__IMGAES_PATH__}' => "/static/images",
'{__LAYUI_CSS_PATH__}' => "/static/layui/css/layui.css",
'{__LAYUI_JS_PATH__}' => "/static/layui/layui.js"
]
];
```

③在工作目录 `app/middleware.php` 下开启全局 Session

```
*/ 全局中间件定义文件*
return [
    */ 全局请求缓存*
    */ \think\middleware\CheckRequestCache::class,*
    */ 多语言加载*
    */ \think\middleware\LoadLangPack::class,*
    */ Session 初始化* ， 关闭下面代码的注释
    \think\middleware\SessionInit::class
];
```

④在工作目录 `config/route.php` 下，设置路由规则

```
<?php
// +-----
// | 路由设置
// +-----

return [
    // pathinfo 分隔符
    'pathinfo_depr' => '/',
    // URL 伪静态后缀
    'url_html_suffix' => 'html',
    // URL 普通方式参数 用于自动生成
```

```

'url_common_param'      => true,
// 是否开启路由延迟解析

'url_lazy_route'       => false,
// 是否强制使用路由

'url_route_must'       => false,
// 合并路由规则

'route_rule_merge'    => false,
// 路由是否完全匹配

'route_complete_match' => false,
// 访问控制器层名称

'controller_layer'    => 'controller',
// 空控制器名

'empty_controller'    => 'Error',
// 是否使用控制器后缀

'controller_suffix'   => false,
// 默认的路由变量规则

'default_route_pattern' => '[\w\.]+' ,
// 是否开启请求缓存 true 自动缓存 支持设置请求缓存规则

'request_cache_key'   => false,
// 请求缓存有效期

'request_cache_expire' => null,
// 全局请求缓存排除规则

'request_cache_except' => [],
// 默认控制器名

'default_controller'  => 'Account', //前台账户类为入口类
// 默认操作名

'default_action'      => 'index', //入口方法为 login 方法
// 操作方法后缀

'action_suffix'       => '',

```

```
// 默认 JSONP 格式返回的处理方法
'default_jsonp_handler' => 'jsonpReturn',

// 默认 JSONP 处理方法
'var_jsonp_handler'     => 'callback',

];
```

⑤在工作目录 config/filesystem.php 下，设置文件上传服务器

```
<?php
return [
    // 默认磁盘
    'default' => env('filesystem.driver', 'local'),
    // 磁盘列表
    'disks' => [
        'local' => [
            'type' => 'local',
            'root' => app()->getRuntimePath() . 'storage',
        ],
        'public' => [
            // 磁盘类型
            'type' => 'local',
            // 磁盘路径
            'root' => app()->getRootPath() . 'public/storage', //文件上传的路径
            // 磁盘路径对应的外部 URL 路径
            'url' => '/storage',
            // 可见性
            'visibility' => 'public',
        ],
        // 更多的磁盘配置信息
    ],
];
```



```
],
];
```

⑥在工作目录 config/app.php 下，配置多应用入口文件

```
<?php
// +-----
// | 应用设置
// +-----

return [
    // 应用地址
    'app_host'      => env('app.host', ''),
    // 应用的命名空间
    'app_namespace' => '',
    // 是否启用路由
    'with_route'    => true,
    // 默认应用
    'default_app'   => 'home', //多应用下，默认访问 home 应用，
    // 默认时区
    'default_timezone' => 'Asia/Shanghai',
    // 应用映射（自动多应用模式有效）
    'app_map'       => [
        'houtai' => 'admin', //后台 //在 url 地址上转化 admin 应用的名称，即只能用
        houtai 才能找到 admin 应用
        'www' => 'home' //前台 //在 url 地址上转化 home 应用的名称，即只能用 www 才能找
        到 home 应用
    ],
    // 域名绑定（自动多应用模式有效）
    'domain_bind'   => [],
    // 禁止 URL 访问的应用列表（自动多应用模式有效）
    'deny_app_list' => [],
```

```
// 异常页面的模板文件
'exception_tpl' => app()->getThinkPath() . 'tpl/think_exception.tpl',
// 错误显示信息,非调试模式有效
'error_message' => '页面错误! 请稍后再试~',
// 显示错误信息
'show_error_msg' => false,];
```

⑦在工作目录 `app/admin/route/app.php` 下, 配置 `admin` 应用路由

```
<?php
// +-----
// | ThinkPHP [ WE CAN DO IT JUST THINK ]
// +-----
// | Copyright (c) 2006~2018 http://thinkphp.cn All rights reserved.
// +-----
// | Licensed ( http://www.apache.org/licenses/LICENSE-2.0 )
// +-----
// | Author: liu21st <liu21st@gmail.com>
//      +-----use
think\facade\Route;

Route::get('login', 'account/index'); //重定向到 login 注意: /account 写法是错的
Route::get('welcome', 'home/index'); //重定向到欢迎页面--班级管理页面
Route::get('class_wz', 'class/index'); //重定向到班级设置页面
Route::get('student', 'stu/index'); //重定向到学生管理页面
Route::get('works', 'work/index'); //重定向到作业布置页面
Route::get('scores', 'score/index'); //重定向到成绩管理页面
```

⑧在工作目录 `app/home/route/app.php` 下, 配置 `home` 应用路由

```
<?php
// +-----
```

```
// | ThinkPHP [ WE CAN DO IT JUST THINK ]
// +-----
// | Copyright (c) 2006~2018 http://thinkphp.cn All rights reserved.
// +-----
// | Licensed ( http://www.apache.org/licenses/LICENSE-2.0 )
// +-----
// | Author: liu21st <liu21st@gmail.com>
//      +-----use
think\facade\Route;

Route::rule('login', 'account/index'); //重定向登录页面
Route::rule('welcome', 'home/index'); //重定向欢迎页面
Route::rule('upwork', 'up_work/index'); //重定向上传文件页面
Route::rule('diswork', 'dis_work/index'); //重定向作业评分页面
```

### ⑨配置系统 env 环境

```
APP_DEBUG = true

[APP]

DEFAULT_TIMEZONE = Asia/Shanghai

[DATABASE]

TYPE = mysql

HOSTNAME = 127.0.0.1

DATABASE = test

USERNAME = works

PASSWORD = password

HOSTPORT = 3306

CHARSET = utf8

DEBUG = true

[LANG]
```

```
default_lang = zh-cn
```

## 5.2 组件库和静态文件的实现

### 5.2.1 Layui 组件库完成前台页面

layui（谐音：类 UI）是一套开源的 Web UI 解决方案，采用自身经典的模块化规范，并遵循原生 HTML/CSS/JS 的开发方式，极易上手，拿来即用。其风格简约轻盈，而组件优雅丰盈，从源代码到使用方法的每一处细节都经过精心雕琢，非常适合网页界面的快速开发。Layui 集成了 jQuery 框架、layer 弹出层库，使用 layui 能满足绝大部分的网页动态效果，提高开发效率。直接下载 zip 包，解压即可使用。



图 47 layui 组件库

### 5.2.2 公共文件部署

public 公共文件部署，主要分为 3 个部分：

#### (1) 系统默认文件

public 目录作为系统入口，是对外开放的公共目录，一般会把具有公共属性的文件存放到该目录下，访问权限较低，包括入口文件、路由配置文件、应用配置文件等。

#### (2) 系统静态文件

系统静态文件一般存放在 static 目录下，其他系统可能会存放在 asset 下，主要是

存放一些系统页面布局需要的文件,包括 CSS 样式表、JavaScript 脚本以及存放图片等,也会将组件库等存放到该目录下。在使用静态文件目录时,通常会重定向 `static` 文件的位置,提高开发效率。

```
<?php
// +-----
// | 模板设置
// +-----
return [
    ...code
    //配置全局变量 //配置模板预定义常量
    'tpl_replace_string' => [
        '{__CSS_PATH__}' => "/static/css",
        '{__JS_PATH__}' => "/static/js",
        '{__IMGAES_PATH__}' => "/static/images",
        '{__LAYUI_CSS_PATH__}' => "/static/layui/css/layui.css",
        '{__LAYUI_JS_PATH__}' => "/static/layui/layui.js"
    ]
];
```

### (3) 文件上传路径

当需要文件上传服务时,就会将上传的文件存放在一个仓库 `storage` 里,方便文件管理,同时也有利于前端页面模板的调用。

	favicon.ico	--应用默认ico图标
	index.php	--应用入口文件
	nginx.htaccess	--nginx服务器配置
	robots.txt	--防爬虫配置
	router.php	--路由初始化
	└── example	
	example.xlsx	--导入学生模板示例文件
	└── static	
	└── css	--样式表文件存放路径
	└── images	--图片存放路径
	└── js	--js脚本存放路径
	└── layui	--layui组件库存放路径
	└── storage	--文件上传存放路径
	└── 2005	--班级编号
	└── stu_data	--学生信息实时同步Excel文件
	stu_data.xlsx	
	└── stu_score	--学生成绩实时同步Excel文件
	stu_score.xlsx	
	└── stu_work	--学生作业上传路径
	└── 1	--作业编号
	└── 2011200	--学生学号
	└── index.html	--作业详情
	└── 2011201	
	└── 2011202	
	└── .....	
	└── 2	
	└── 3	
	└── 4	
	└── .....	
	└── 2006	
	└── stu_data	
	└── stu_score	
	└── stu_work	

## 5.3 前端页面模板的实现

### 5.3.1 教师后台管理页面模板

教师后台管理页面，包含登陆页面，班级管理页面、系统主页面、学生管理页面、作业布置页面、成绩管理页面等。

	└─ account	--登陆页面
	login.php	
	└─ home	--班级管理页面
	information.php	--教师个人信息页面
	index.php	--班级主页面
	add.php	--班级添加页面
	edit.php	--班级修改页面
	layout.php	--作业系统主体框架
	└─ public	
	footer.php	--作业系统公共页脚
	header.php	--作业系统公共页眉
	└─ classed	--作业系统主页面
	index.php	
	└─ stu	--学生管理页面
	index.php	--学生主页面
	up_stu.php	--导入学生页面
	add.php	--学生添加页面
	edit.php	--学生修改页面
	└─ work	--作业管理页面
	index.php	--作业主页面
	add.php	--作业添加页面
	edit.php	--作业修改页面
	detail.php	--作业详情页面
	└─ score	--成绩管理页面
	index.php	
	└─	

(1) 登陆页面

```
-----  
<!-- 登录界面视图 -->  
-----  
  
<!DOCTYPE html>  
<html lang="en">  
  
<head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>智慧作业管理系统后台</title>  
    <!-- 引入 login.css 样式文件 -->  
    <link rel="stylesheet" href="/static/css/login.css">  
    <!-- 引入 layui 组件库 -->  
    <link rel="stylesheet" href="/static/layui/css/layui.css">  
    <script src="/static/layui/layui.js"></script>  
    <!-- 引入 vantajs 界面库 -->  
    <script src="/static/js/three.r134.min.js"></script>  
    <script src="/static/js/vanta.birds.min.js"></script>  
    <!-- 设置网站图标 -->  
    <link rel="shortcut icon" href="/static/images/index.ico" type="image/x-icon">  
</head>  
  
<body>  
    <div class="head"></div>  
    <div class="layui-container">  
        <div class="head-title">  
            
```



```

        <span class="title">智慧作业管理系统后台</span>
    </div>
    <div class="layui-form">
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-username"></i></div>
            <div class="input-item">
                <input type="text" name="uname" required lay-verify="required" placeholder="请输入管理员账户"
                    autocomplete="off" class="layui-input" value="">
            </div>
        </div>
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-password"></i></div>
            <div class="input-item">
                <input type="password" name="pwd" required lay-verify="required" placeholder="请输入密码"
                    autocomplete="off" class="layui-input" value="">
            </div>
        </div>
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-verify"></i></div>
            <div class="input-item captcha">
                <input type="text" name="captcha" required lay-verify="required" placeholder="请输入验证码"
                    autocomplete="off" class="layui-input">
                
            </div>
        </div>
    </div>

```

```
        </div>
        <div class="form-item">
            <button class="layui-btn layui-btn-normal" onclick="do_login()">登
录</button>
        </div>
    </div>
</div>

</div>
<script>
VANTA.BIRDS({
    el: ".head",
    mouseControls: true,
    touchControls: true,
    gyroControls: false,
    minHeight: 200.00,
    minWidth: 200.00,
    scale: 1.00,
    scaleMobile: 1.00
})

$ = layui.jquery;

// 刷新验证码
function reloadveriimg(obj) {
    // 加入随机字符，表示验证码图片已经发生改变
    $(obj).attr('src', '/houtai/account/verify?rand=' + Math.random());
}
```

```
// 登录校验

function do_login() {

    var uname = $('input[name="uname"]').val();

    var pwd = $('input[name="pwd"]').val();

    var captcha = $('input[name="captcha"]').val();

    if (uname == '' || pwd == '' || captcha == '') {

        layer.msg('必填项不能为空', {

            icon: 2

        })

    } else {

        $.post('/houtai/account/do_login', {

            uname,

            pwd,

            captcha

        }, function(res) {

            console.log(res)

            if (res.code > 0) {

                layer.msg(res.msg, {

                    icon: 2

                })

                reloadveriimg($('#captcha'));

            } else {

                layer.msg(res.msg, {

                    icon: 1

                })

                setTimeout(function() {

                    window.location.href = '/houtai/welcome'

                }, 1000);

            }

        });

    }

}
```

```
        }
        }, 'json');
    }
}
</script>
</body>
</html>
```

## (2) 班级管理页面

```
-----
<!-- 教师个人信息页面 -->
-----
```

```
<!DOCTYPE html>
<html lang="en">

<head>
    <title></title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="css/style.css" rel="stylesheet">
</head>

<link rel="stylesheet" href="{__LAYUI_CSS_PATH__}">
<script src="{__LAYUI_JS_PATH__}"></script>
<style>
.container {
    margin: 20px auto;

    width: 80%;
    text-align: center;
```

```

}
</style>

<body>
    <p>建议使用 163 邮箱</p>
    <p>登录网页版 163 邮箱，点击设置，开启 IMAP/SMTP 服务</p>
    <p>扫码发送短信后，获取到授权密码并输入</p>
    <p>SMTP 服务器：smtp.163.com</p>
    <div class="container">
        <h1 style="padding: 30px 0;">个人信息维护</h1>

        <div class="layui-form layui-form-pane">
            <div class="layui-form-item">
                <label class="layui-form-label">用户名</label>
                <div class="layui-input-block">
                    <input type="text" autocomplete="on" class="layui-input"
value="{ $admin.uname}" disabled>
                </div>
            </div>
            <div class="layui-form-item">
                <label class="layui-form-label">密码</label>
                <div class="layui-input-block">
                    <input type="text" name="pwd" autocomplete="off" class="layui-
input" value="{ $admin.pwd}">
                </div>
            </div>
            <div class="layui-form-item">
                <label class="layui-form-label">邮箱</label>
                <div class="layui-input-block">
                    <input type="email" name="email" autocomplete="off"

```

```

class="layui-input" value="{ $admin.email}">
        </div>
    </div>
    <div class="layui-form-item">
        <label class="layui-form-label">SMTP 服务</label>
        <div class="layui-input-block">
            <input type="text" name="email_system" autocomplete="off"
class="layui-input"
            value="{ $admin.email_system}">
        </div>
    </div>
    <div class="layui-form-item">
        <label class="layui-form-label">SMTP 秘钥</label>
        <div class="layui-input-block">
            <input type="text" name="check_code" autocomplete="off"
class="layui-input"
            value="{ $admin.check_code}">
        </div>
    </div>
    <div class="layui-form-item">
        <button class="layui-btn layui-btn-fluid layui-btn-normal"
onclick="save('{ $admin.id}')">保存</button>
    </div>
</div>
</div>
<script>
var $ = layui.jquery;

function save(id) {
    var pwd = $('input[name="pwd"]').val();

```

```
var email = $('input[name="email"]').val();

if (email == '') { //非空校验
    layer.msg('必填项不能为空', {
        icon: 2
    })
} else { //邮箱格式校验
    if (!email.match(/^\\w+@\\w+\\.\\w+$/i)) {
        layer.msg('邮箱格式错误', {
            icon: 2
        })
    } else {
        $.post('/houtai/home/information', {
            id,
            pwd,
            email,
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
                })
            } else {
                exit();
            }
        }, 'json');
    }
}

// 退出登录
```

```

function exit() {
    $.post('/houtai/base/exit', {}, function(res) {
        layer.msg('保存成功, 请重新登录', {
            icon: 1
        });
        setTimeout(function() {
            parent.window.location.href = "/houtai/login";
        }, 1000);
    }, 'json')
}
</script>
</body>
</html>

-----

<!-- 班级管理主页 -->
-----

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>班级门户</title>
</head>
<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
<script src="{__LAYOUT_JS_PATH__}"></script>
<style>

```



```
.header {
    text-align: center;
    margin: 10px auto;
}

.container {
    margin: 60px auto;

    width: 60%;
}

.footer {
    position: relative;
    bottom: 10px;
    width: 100%;
    text-align: center;
}

</style>

<body>
    <button class="layui-btn layui-btn-primary layui-border-black exit"
onclick="exit()">退出</button>
    <div class="header">
        <h1 style="margin: 20px auto;">班级管理门户 【{$admin.uname}】 </h1>
    </div>
    <div class="container">
        <div class="layui-form">
            <div class="layui-form-item">
                <!-- 添加、清空数据框 -->
```

```

        <div class="layui-inline">
            <button class="layui-btn layui-btn-primary layui-border-black"
onclick="add()">添加班级</button>
            <button class="layui-btn layui-btn-primary layui-border-black"
onclick="information()">个人信息</button>
        </div>
    </div>
</div>
<div>
    <h5><em><span style="color:red">注意: </span></em></h5>
    <p><em>个人信息的<span style="color:red">邮箱</span>是发送短信的邮箱<span
style="color:red">请自行修改</span></em></p>
    <p><em>班级代码只支持<span style="color:red">数字</span>命名, 班级一旦生成,
班级代码将<span style="color:red">无法修改</span></em></p>
    <p><em>可以设置<span style="color:red">班级状态: </span></em></p>
    <p><em>@正常表示课程正在进行! </em></p>
    <p><em>@结课学生将无法进入系统! </em></p>
    <p><em><span style="color:red">下载文件</span>包括学生的名单、成绩、作
业!</em></p>
    <table class="layui-table">
        <thead>
            <tr>
                <th>班级代码</th>
                <th>课程名称</th>
                <th>班级创建时间</th>
                <th>班级状态</th>
                <th>访问班级</th>
                <th>操作</th>
            </tr>
        </thead>
        <tbody>
            {volist name='classes' id='classes'}

```

```

        <tr>
            <td>{$classes.class_id}</td>
            <td>{$classes.class_name}</td>
            <td>{$classes.start_time}</td>
            <td><?= $classes['status'] == 0 ? "正常" : "<span
style=color:red>关闭</span>" ?></td>
            <td>
                <a href="/houtai/class_wz?class_id={$classes.class_id}">
                    <button class="layui-btn layui-btn-normal layui-btn-
xs">进入班级</button>
                </a>
            </td>
            <td>
                <button class="layui-btn layui-btn-xs"
onclick="get_zip('{$classes.class_id}')">下载</button>
                <button class="layui-btn layui-btn-warm layui-btn-xs"
onclick="edit('{$classes.id}')">修改</button>
                <button class="layui-btn layui-btn-danger layui-btn-xs"
onclick="del('{$classes.id}','{$classes.class_id}')">
删除</button>
            </td>
        </tr>
    </tbody>
</table>
</div>
<div class="footer">
    <p>ThinkPHP</p>

```

```
<p>以及勤劳的自己</p>
</div>
<script>
//注意: 选项卡 依赖 element 模块, 否则无法进行功能性操作
var element = layui.element;
var layer = layui.layer;
var $ = layui.jquery;

// 退出登录
function exit() {
    layer.confirm('是否退出登录', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        $.post('/houtai/base/exit', {}, function(res) {
            layer.msg('退出成功!', {
                icon: 1
            });
            setTimeout(function() {
                window.location.href = "/houtai/login";
            }, 1000)
        }, 'json')
    });
}

// 个人信息
function information() {
    layer.open({
        type: 2,
```

```
        title: '个人信息管理',
        maxmin: false, //开启最大化最小化按钮
        area: ['60%', '90%'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/home/information',
        offset: 't',
        shade: 0.3,
        maxmin: true,
        shadeClose: true
    });
}

// 添加
function add() {
    layer.open({
        type: 2,
        title: '班级添加',
        maxmin: false, //开启最大化最小化按钮
        area: ['80%', '500px'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/home/add',
        offset: 't',
        shade: 0.3,
        shadeClose: true
    });
}

// 修改
```

```
function edit(id) {
    layer.open({
        type: 2,
        title: '班级修改',
        maxmin: false, //开启最大化最小化按钮
        area: ['80%', '100%'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/home/edit?id=' + id,
        offset: 't',
        shade: 0.3,
        maxmin: true,
        shadeClose: true
    });
}

// 删除
function del(id, class_id) {
    layer.confirm('请在下载备份数据后删除', {
        btn: ['下载', '确定', '取消'] //按钮
    }, function() {
        window.location.href = '/houtai/home/get_zip?class_id=' + class_id;
    }, function() {
        $.post('/houtai/home/del', {
            id,
            class_id
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
```

```
        icon: 2
    })
} else {
    layer.msg(res.msg, {
        icon: 1
    })
    setTimeout(function() {
        parent.window.location.reload();
    }, 1000)
}
}, 'json')

});
}

function get_zip(class_id) {
    layer.confirm('确定要下载吗?', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        layer.msg('正在导出作业, 请勿重复点击!', {
            icon: 1,
            time: 2000
        });
        window.location.href = '/houtai/home/get_zip?class_id=' + class_id;
    });
}
</script>
</body>
</html>
```

```
-----  
<!-- 班级添加页面 -->  
-----
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <title>班级添加</title>
```

```
</head>
```

```
<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
```

```
<script src="{__LAYOUT_JS_PATH__}"></script>
```

```
<style>
```

```
    .container {
```

```
        margin: 20px auto;
```

```
        width: 40%;
```

```
    }
```

```
</style>
```

```
<body>
```

```
    <div class="container">
```

```
        <div class="layui-form layui-form-pane">
```

```
            <div class="layui-form-item">
```

```
                <label class="layui-form-label">班级代码</label>
```

```
                <div class="layui-input-block">
```

```
                    <input type="text" name="class_id" autocomplete="on">
```



```

placeholder="例： 2005" class="layui-input">
    </div>
</div>
<div class="layui-form-item">
    <label class="layui-form-label">课程名称</label>
    <div class="layui-input-block">
        <input type="text" name="class_name" autocomplete="off"
placeholder="例： 管理信息课程" class="layui-input">
    </div>
</div>
<div class="layui-form-item">
    <label class="layui-form-label">班级状态</label>
    <div class="layui-input-inline">
        <input type="checkbox" name="status" lay-skin="switch" lay-
text="开启|关闭" checked>
    </div>
</div>
<div class="layui-form-item">
    <label class="layui-form-label">课程信息</label>
    <div class="layui-input-block">
        <input type="text" name="class_time" autocomplete="off"
placeholder="例： 公楼 C404 周二上午一二节" class="layui-input">
    </div>
</div>
<div class="layui-form-item layui-form-text">
    <label class="layui-form-label">课程介绍</label>
    <div class="layui-input-block">
        <textarea placeholder="请输入内容课程介绍" class="layui-
textarea" name="remarks"></textarea>
    </div>
</div>

```

```
<div class="layui-form-item">
    <button class="layui-btn layui-btn-fluid layui-btn-normal"
onclick="save()">保存</button>
</div>
</div>
</div>
<script>
    //注意：选项卡 依赖 element 模块，否则无法进行功能性操作
    var element = layui.element;
    var layer = layui.layer;
    var $ = layui.jquery;
    function save() {
        var class_id = $('input[name="class_id"]').val();
        var class_name = $('input[name="class_name"]').val();
        var status = $('input[name="status"]').is(":checked") ? 0 : 1;
        var class_time = $('input[name="class_time"]').val();
        var remarks = $('textarea[name="remarks"]').val();

        if (class_id == '' || class_name == '') {
            layer.msg('必填项不能为空', {
                icon: 2
            })
        } else {
            $.post('/houtai/home/add', {
                class_id,
                class_name,
                status,
                class_time,
                remarks,
            })
        }
    }
</script>
```

```
    }, function(res) {
        if (res.code > 0) {
            layer.msg(res.msg, {
                icon: 2
            })
        } else {
            layer.msg(res.msg, {
                icon: 1
            })
            setTimeout(function() {
                parent.window.location.reload()
            }, 1000);
        }
    }, 'json');
}
}
</script>
</body>
</html>
-----
<!-- 班级修改页面 -->
-----
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>班级添加</title>
</head>
<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
<script src="{__LAYOUT_JS_PATH__}"></script>
<style>
.container {
    margin: 20px auto;

    width: 40%;
}
</style>
<body>
    <div class="container">
        <input type="hidden" name="id" value="{class.id}">
        <input type="hidden" name="class_id" value="{class.class_id}">
        <div class="layui-form layui-form-pane">
            <div class="layui-form-item">
                <label class="layui-form-label">班级代码</label>
                <div class="layui-input-block">
                    <input type="text" autocomplete="on" placeholder="例: 2005"
class="layui-input"
                    value="{class.class_id}" disabled>
                </div>
            </div>
            <div class="layui-form-item">
                <label class="layui-form-label">课程名称</label>
                <div class="layui-input-block">
```

```

        <input type="text" name="class_name" autocomplete="off"
placeholder="例：管理信息课程" class="layui-input"
        value="{ $class.class_name }">
    </div>
</div>
<div class="layui-form-item">
    <label class="layui-form-label">班级状态</label>
    <div class="layui-input-inline">
        <input type="checkbox" name="status" lay-skin="switch" lay-
text="开启|关闭" { $class.status==0? "checked"
        : "" }>
    </div>
</div>
<div class="layui-form-item">
    <label class="layui-form-label">课程信息</label>
    <div class="layui-input-block">
        <input type="text" name="class_time" autocomplete="off"
placeholder="例：公楼 C404 周二上午一二节"
        class="layui-input" value="{ $class.class_time }">
    </div>
</div>
<div class="layui-form-item layui-form-text">
    <label class="layui-form-label">课程介绍</label>
    <div class="layui-input-block">
        <textarea placeholder="请输入内容课程介绍" class="layui-
textarea" name="remarks">{ $class.remarks }</textarea>
    </div>
</div>
<div class="layui-form-item">
    <button class="layui-btn layui-btn-fluid layui-btn-normal"
onclick="save()">保存</button>

```

```
        </div>
    </div>
</div>
<script>
//注意：选项卡 依赖 element 模块，否则无法进行功能性操作
var element = layui.element;
var layer = layui.layer;
var $ = layui.jquery;
function save() {
    var id = $('input[name="id"]').val();
    var class_id = $('input[name="class_id"]').val();
    var class_name = $('input[name="class_name"]').val();
    var status = $('input[name="status"]').is(":checked") ? 0 : 1;
    var class_time = $('input[name="class_time"]').val();
    var remarks = $('textarea[name="remarks"]').val();
    if (class_name == '') {
        layer.msg('必填项不能为空', {
            icon: 2
        })
    } else {
        $.post('/houtai/home/edit', {
            id,
            class_id,
            class_name,
            status,
            class_time,
            remarks,
        }, function(res) {
            if (res.code > 0) {
```

```
        layer.msg(res.msg, {
            icon: 2
        })
    } else {
        layer.msg(res.msg, {
            icon: 1
        })
        setTimeout(function() {
            parent.window.location.reload()
        }, 1000);
    }
}, 'json');
}
</script>
</body>
</html>
```

### (3) 作业系统主体框架

```
-----
<!-- 公共主体页面框架 -->
-----
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>{$class.title}</title>

<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">

<script src="{__LAYOUT_JS_PATH__}"></script>
</head>
<style>
.pagination {
    display: inline-block;
    padding-left: 0;
    margin: 20px 0;
    border-radius: 4px;
}

.pagination>li {
    display: inline;
}

.pagination>li>a,
.pagination>li>span {
    position: relative;
    float: left;
    padding: 6px 12px;
    margin-left: -1px;
    line-height: 1.42857143;
    color: #337ab7;
    text-decoration: none;
    background-color: #fff;
    border: 1px solid #ddd;
}
```



```
.pagination>li:first-child>a,  
.pagination>li:first-child>span {  
    margin-left: 0;  
    border-top-left-radius: 4px;  
    border-bottom-left-radius: 4px;  
}  
  
.pagination>li:last-child>a,  
.pagination>li:last-child>span {  
    border-top-right-radius: 4px;  
    border-bottom-right-radius: 4px;  
}  
  
.pagination>li>a:hover,  
.pagination>li>span:hover,  
.pagination>li>a:focus,  
.pagination>li>span:focus {  
    z-index: 2;  
    color: #23527c;  
    background-color: #eee;  
    border-color: #ddd;  
}  
  
.pagination>.active>a,  
.pagination>.active>span,  
.pagination>.active>a:hover,  
.pagination>.active>span:hover,  
.pagination>.active>a:focus,  
.pagination>.active>span:focus {
```

```
z-index: 3;

color: #fff;

cursor: default;

background-color: #337ab7;

border-color: #337ab7;
}

.pagination>.disabled>span,
.pagination>.disabled>span:hover,
.pagination>.disabled>span:focus,
.pagination>.disabled>a,
.pagination>.disabled>a:hover,
.pagination>.disabled>a:focus {
    color: #777;
    cursor: not-allowed;
    background-color: #fff;
    border-color: #ddd;
}
</style>

<body>
    {include file="public/header"}
    {__CONTENT__}
    {include file="public/footer"}
</body>

</html>
```

(4) 作业系统公共页眉、页脚

```
-----  
<!-- 作业系统公共页眉 -->  
-----
```

```
<style>
```

```
.header {
```

```
    text-align: center;
```

```
    margin: 10px auto;
```

```
}
```

```
.layui-nav {
```

```
    display: flex;
```

```
    justify-content: center;
```

```
    align-items: center;
```

```
    background-color: white;
```

```
}
```

```
.layui-nav .layui-nav-item a {
```

```
    color: #000;
```

```
}
```

```
.layui-nav .layui-nav-item a:hover,
```

```
.layui-nav .layui-this a {
```

```
    color: #1E9FFF;
```

```
}
```

```
.layui-nav .layui-this:after,
```

```
.layui-nav-bar {
```

```
    background-color: #1E9FFF;
```

```

}

.exit {
    position: absolute;
    top: 20px;
    right: 10px;
}

.exit2 {
    position: absolute;
    top: 20px;
    right: 100px;
}
</style>
<button class="layui-btn layui-btn-primary layui-border-black exit2"
    onclick="window.location.href='/houtai/welcome'">返回</button>
<button class="layui-btn layui-btn-primary layui-border-black exit"
    onclick="exit()">退出</button>
<div class="header">
    <h1 style="margin: 20px auto;">{$class.class_id}班作业管理后台【{$admin.uname}】
</h1>
    <ul class="layui-nav" lay-filter="">
        <li class="layui-nav-item {$class.title=='班级配置'? 'layui-this' : ''}"><a
            href="/houtai/class_wz?class_id={$class.class_id}">班级配置
        </a></li>
        <li class="layui-nav-item {$class.title=='学生管理'? 'layui-this' :
        ''}"><a
            href="/houtai/student?class_id={$class.class_id}">学生管理
        </a></li>

```

```

        <li class="layui-nav-item {$class.title=='作业布置'? 'layui-this' :
''}""><a href="
            /houtai/works?class_id={$class.class_id}">作业布置</a></li>
        <li class=" layui-nav-item {$class.title=='成绩管理' ? 'layui-this' :
'' }""><a href="
            /houtai/scores?class_id={$class.class_id}">成绩管理</a></li>
    </ul>
</div>
<script>
//注意: 选项卡 依赖 element 模块, 否则无法进行功能性操作
var element = layui.element;
var layer = layui.layer;
var $ = layui.jquery;
// 退出登录
function exit() {
    layer.confirm('是否退出登录', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        $.post('/houtai/base/exit', {}, function(res) {
            layer.msg('退出成功!', {
                icon: 1
            });
            setTimeout(function() {
                parent.window.location.href = "/houtai/login";
            }, 1000)
        }, 'json')
    });
}
</script>

```

```
-----  
<!-- 作业系统公共页脚 -->  
-----
```

```
<style>  
.footer {  
    position: relative;  
    bottom: 10px;  
    width: 100%;  
    text-align: center;  
}
```

```
</style>  
<div class="footer">  
    <p>ThinkPHP</p>  
    <p>以及勤劳的自己</p>  
</div>
```

#### (5) 作业系统主页面

```
-----  
<!-- 作业系统主界面 -->  
-----
```

```
<style>  
    .container {  
        margin: 20px auto;  
  
        width: 40%;  
        text-align: center;  
    }
```

```
</style>  
<div class="container">
```

```

<div class="layui-form layui-form-pane">
  <div class="layui-form-item">
    <label class="layui-form-label">班级代码</label>
    <div class="layui-input-block">
      <input type="text" autocomplete="on" placeholder="例：2005"
class="layui-input" value="{ $class.class_id}" disabled>
    </div>
  </div>
  <div class="layui-form-item">
    <label class="layui-form-label">课程名称</label>
    <div class="layui-input-block">
      <input type="text" name="class_name" autocomplete="off"
placeholder="例：管理信息课程" class="layui-input" value="{ $class.class_name}">
    </div>
  </div>
  <div class="layui-form-item">
    <label class="layui-form-label">课程信息</label>
    <div class="layui-input-block">
      <input type="text" name="class_time" autocomplete="off"
placeholder="例：公楼 C404 周二上午一二节" class="layui-input"
value="{ $class.class_time}">
    </div>
  </div>
  <div class="layui-form-item layui-form-text">
    <label class="layui-form-label">课程介绍</label>
    <div class="layui-input-block">
      <textarea placeholder="请输入内容课程介绍" class="layui-textarea"
name="remarks">{ $class.remarks}</textarea>
    </div>
  </div>
  <div class="layui-form-item">

```

```
                <button class="layui-btn layui-btn-fluid layui-btn-normal"
onclick="save('{class.class_id}')">保存</button>

            </div>
        </div>
</div>
<script>
    function save(class_id) {
        var class_name = $('input[name="class_name"]').val();
        var class_time = $('input[name="class_time"]').val();
        var remarks = $('textarea[name="remarks"]').val();

        if (class_name == '') {
            layer.msg('必填项不能为空', {
                icon: 2
            })
        } else {
            $.post('/houtai/Classed/save', {
                class_id,
                class_name,
                class_time,
                remarks,
            }, function(res) {
                if (res.code > 0) {
                    layer.msg(res.msg, {
                        icon: 2
                    })
                } else {
                    layer.msg(res.msg, {
                        icon: 1
                    })
                }
            })
        }
    }
}
```



```
        })
        setTimeout(function() {
            window.location.reload()
        }, 1000);
    }
    }, 'json');
}
}
</script>
```

#### (6) 学生管理页面

```
-----
<!-- 学生主页面 -->
-----

<style>
.container {
    margin: 20px auto;

    width: 90%;
    text-align: center;
}
</style>
<div class="container">
    <div class="layui-form">
        <div class="layui-form-item">
            <!-- 搜索学生 -->
            <div class="layui-inline">
                <input type="text" name="search" placeholder="搜索学生"
autocomplete="off" class="layui-input">
```

```

        value="{ $search? $search: '' }">
    </div>

    <!-- 搜索按钮 -->

    <div class="layui-inline">
        <button class="layui-btn layui-btn-normal"
onclick="search('{ $class.class_id }')"><i
        class="layui-icon layui-icon-search" style="font-
size:24px"></i></button>
    </div>

    <!-- 添加、清空数据框 -->

    <div class="layui-inline">
        <button
            class="layui-btn layui-btn-primary layui-border-black <?=
count($stu) == 0 ? "layui-btn-disabled" : "" ?>"
            onclick="add('{ $class.class_id }') " <?= count($stu) == 0 ?
"disabled" : "" ?>>添加</button>
        <button
            class="layui-btn layui-btn-primary layui-border-black <?=
count($stu) != 0 ? "layui-btn-disabled" : "" ?>"
            onclick="up_stu('{ $class.class_id }') " id="upfile"
            <?= count($stu) != 0 ? "disabled" : "" ?>>一键导入</button>
        <button
            class="layui-btn layui-btn-primary layui-border-black <?=
count($stu) == 0 ? "layui-btn-disabled" : "" ?>"
            onclick=" clear_all('{ $class.class_id }') " <?= count($stu) ==
0 ? "disabled" : "" ?>>一键清空</button>
        <button
            class="layui-btn layui-btn-primary layui-border-black <?=
count($stu) == 0 ? "layui-btn-disabled" : "" ?>"
            onclick="
send_all_email('{ $class.class_id }', '{ $class.class_name }', is_all=1)"
            <?= count($stu) == 0 ? "disabled" : "" ?>>密码群发</button>
    </div>

```

```

        <button
            class="layui-btn layui-btn-primary layui-border-blue <?=  
count($stu) == 0 ? "layui-btn-disabled" : "" ?>"
            onclick=" get_zip('{ $class.class_id}')" <?=  
count($stu) == 0 ?  
"disabled" : "" ?>>学生名单导出</button>
    </div>
</div>
</div>
<table class=" layui-table">
    <thead>
        <tr>
            <td>班级</td>
            <td>学号</td>
            <td>姓名</td>
            <td>性别</td>
            <td>邮箱</td>
            <td>登录密码</td>
            <td>最近登录时间</td>
            <td>操作</td>
        </tr>
    </thead>
    <tbody>
        {volist name='stu' id='stu'}
        <tr>
            <td>{$stu.class_id}</td>
            <td>{$stu.stu_no}</td>
            <td>{$stu.stu_name}</td>
            <td>{$stu.gender==0 ? "男":"女"}</td>
            <td>{$stu.email}</td>
            <td>{$stu.stu_pwd}</td>
        </tr>
    </tbody>
</table>

```

```

        <td>{$stu.last_time}</td>
        <td>
            <button class="layui-btn layui-btn-xs"
                onclick="send_email('{ $class.class_id }', '{ $class.class_name }', '{ $stu.stu_no }')">密码分发</button>
            <button class="layui-btn layui-btn-warm layui-btn-xs"
                onclick="edit('{ $class.class_id }', '{ $stu.stu_no }')">修改
</button>
            <button class="layui-btn layui-btn-danger layui-btn-xs"
                onclick="del('{ $class.class_id }', '{ $stu.stu_no }')">删除
</button>
        </td>
    </tr>
</volist>
</tbody>
</table>
{$page|raw}
</div>

<script>
var upload = layui.upload; //上传文件功能
// 预定义变量

// 学生名单导出
function get_zip(class_id) {
    layer.confirm('确定要下载吗? ', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        window.location.href = '/houtai/stu/get_zip?class_id=' + class_id;
    });
}

```

```
        setTimeout(function() {
            window.location.reload();
        }, 1000)
    });
}

// 添加
function add(class_id) {
    layer.open({
        type: 2,
        title: '学生添加',
        maxmin: false, //开启最大化最小化按钮
        area: ['80%', '250px'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/stu/add?class_id=' + class_id,
        offset: 't',
        shade: 0.3,
        shadeClose: true
    });
}

// 学生一键导入页面
function up_stu(class_id) {
    layer.open({
        type: 2,
        title: '学生一键导入',
        skin: 'layui-layer-rim', //加上边框
        area: ['520px', '240px'], //宽高
    });
}
```

```
        shade: 0.8,
        content: '/houtai/stu/up_stu?class_id=' + class_id
    });
}

//学生密码群发
function send_all_email(class_id, class_name, is_all) {
    layer.confirm('确定要群发吗? ', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        var index = layer.load(1); //换了种风格
        $.post('/houtai/stu/send_email', {
            class_id,
            class_name,
            is_all
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
                })
                layer.close(index);
            } else {
                layer.msg(res.msg, {
                    icon: 1
                })
                layer.close(index);
                setTimeout(function() {
                    window.location.reload();
                }, 1000)
            }
        })
    })
}
```

```
    }
    }, 'json')

});
}
//学生密码单发
function send_email(class_id, class_name, stu_no) {
    layer.confirm('确定要发送吗? ', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        $.post('/houtai/stu/send_email', {
            class_id,
            class_name,
            stu_no
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
                })
            } else {
                layer.msg(res.msg, {
                    icon: 1
                })
                setTimeout(function() {
                    window.location.reload();
                }, 1000)
            }
        }, 'json')
    }
}
```

```
});  
}  
  
// 修改  
function edit(class_id, stu_no) {  
    layer.open({  
        type: 2,  
        title: '学生修改',  
        maxmin: false, //开启最大化最小化按钮  
        area: ['80%', '250px'],  
        shadeClose: false,  
        anim: 5,  
        content: '/houtai/stu/edit?class_id=' + class_id + '&stu_no=' + stu_no,  
        offset: 't',  
        shade: 0.3,  
        shadeClose: true  
    });  
}  
  
// 删除  
function del(class_id, stu_no) {  
    layer.confirm('确定要删除?', {  
        btn: ['确定', '取消'] //按钮  
    }, function() {  
        $.post('/houtai/stu/del', {  
            class_id,  
            stu_no  
        }, function(res) {  
            if (res.code > 0) {
```



```
        layer.msg(res.msg, {
            icon: 2
        })
    } else {
        layer.msg(res.msg, {
            icon: 1
        })
        window.location.reload();
    }
}, 'json')

});
}
// 搜索
function search(class_id) {
    var search = $.trim($('input[name="search"]').val()); //搜索内容
    window.location.href = '/houtai/student?class_id=' + class_id + '&search=' + search;
}
// 清空
function clear_all(class_id) {
    layer.confirm('确定要清空学生信息吗?', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        $.post('/houtai/stu/del', {
            class_id,
            is_clear_all: 1 //传递该值表示清空数据表
        }, function(res) {
            if (res.code > 0) {
```

```
        layer.msg(res.msg, {
            icon: 2
        })
    } else {
        layer.msg(res.msg, {
            icon: 1
        })
        window.location.reload();
    }
}, 'json')
});
}
</script>

-----

<!-- 导入学生页面 -->

-----

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>学生一键导入</title>
    <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
    <script src="{__LAYOUT_JS_PATH__}"></script>
</head>
```

```

<body style="padding: 10px;">
    <p style="color:red;">为确保文件导入正常，请使用该学生表样表导入：<a href="/example/example.xlsx" style="text-decoration: underline;">example.xlsx</a></p>
    <hr>
    <button type="button" class="layui-btn" id="upfile">
        <i class="layui-icon">&#xe67c;</i>导入学生信息
    </button>
    <button type="button" class="layui-btn" id="to_upfile">开始上传</button>
    <script>
        var upload = layui.upload; //得到 upload 对象
        var $ = layui.jquery;
        var layer = layui.layer;

        //创建一个上传组件
        upload.render({
            elem: '#upfile',
            url: '/houtai/stu/upfile',
            data: {
                class_id: '{$class_id}'
            },
            accept: 'file', //允许上传的文件类型
            size: 1000, //最大允许上传的文件大小
            auto: false,
            bindAction: '#to_upfile',
            before: function(obj) { //obj 参数包含的信息，跟 choose 回调完全一致，可参见上文。
                layer.load(); //上传 loading
            },
            done: function(res, index, upload) { //上传后的回调

```

```
if (res.code > 0) {
    layer.msg(res.msg, {
        icon: 2
    })
    setTimeout(function() {
        window.location.reload()
    }, 1000);
    layer.closeAll('loading'); //关闭 loading
} else {
    // layer.msg(res.msg);
    layer.msg(res.msg, {
        icon: 1
    })
    setTimeout(function() {
        parent.window.location.reload()
    }, 1000);
    layer.closeAll('loading'); //关闭 loading
}
},
error: function() {
    layer.msg('上传失败');
    layer.closeAll('loading'); //关闭 loading
    //请求异常回调

    setTimeout(function() {
        window.location.reload();
    }, 1000)
}
```

```
    })
  </script>
</body>

</html>

-----

<!-- 学生添加页面 -->

-----

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>学生添加</title>
  <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
  <script src="{__LAYOUT_JS_PATH__}"></script>
</head>

<body style="padding:10px">

  <table class="layui-table">
    <thead>
      <tr>
        <td>学号</td>
        <td>姓名</td>
      </tr>
    </thead>
  </table>

```

```

        <td>性别</td>

        <td>邮箱</td>

        <td>登录密码</td>

    </tr>
</thead>
<tbody>
    <tr>
        <td><input type="text" name="stu_no" class="layui-input"></td>
        <td><input type="text" name="stu_name" class="layui-input"></td>
        <td>
            男<input type="radio" name="gender" value="0" title="男"
checked>
            女 <input type="radio" name="gender" value="1" title="女">
        </td>
        <td><input type="email" name="email" class="layui-input"></td>
        <td><input type="text" name="stu_pwd" class="layui-input"
placeholder="密码不用填写, 随机生成" disabled></td>
    </tr>
</tbody>
</table>
<button class="layui-btn layui-btn-normal" onclick="save('{class_id}')">保存
</button>
<script>
    var $ = layui.jquery;

    function save(class_id) {
        var stu_no = $('input[name="stu_no"]').val();
        var stu_name = $('input[name="stu_name"]').val();
        var email = $('input[name="email"]').val();
        var gender = $('input[name="gender"]:checked').val();
    }

```

```
if (stu_no == '' || stu_name == '' || email == '') {
    layer.msg('必填项不能为空', {
        icon: 2
    })
} else {
    $.post('/houtai/stu/add', {
        class_id,
        stu_no,
        stu_name,
        email,
        gender,
    }, function(res) {
        if (res.code > 0) {
            layer.msg(res.msg, {
                icon: 2
            })
        } else {
            layer.msg(res.msg, {
                icon: 1
            })
            setTimeout(function() {
                parent.window.location.reload()
            }, 1000);
        }
    }, 'json');
}
}
```

```
</script>
</body>

</html>

-----

<!-- 学生修改页面 -->

-----

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>学生修改</title>
    <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
    <script src="{__LAYOUT_JS_PATH__}"></script>
</head>

<body style="padding:10px">
    <input type="hidden" name="stu_no" class="layui-input" value="{ $stu_no }">
    <input type="hidden" name="class_id" class="layui-input" value="{ $class_id }">
    <table class="layui-table">
        <thead>
            <tr>
                <td>学号</td>
                <td>姓名</td>
                <td>性别</td>
            </tr>
        </thead>
    </table>

```



```

        <td>邮箱</td>

        <td>登录密码</td>

    </tr>

</thead>

<tbody>

    <tr>

        <td><input type="text" name="stu_no" class="layui-input"
value="{ $stu_no}" disabled></td>

        <td><input type="text" name="stu_name" class="layui-input"
value="{ $stu_name}"></td>

        <td>

            男<input type="radio" name="gender" value="0" title="男"
{ $gender==0?"checked":""}>

            女 <input type="radio" name="gender" value="1" title="女"
{ $gender==1?"checked":""}>

        </td>

        <td><input type="email" name="email" class="layui-input"
value="{ $email}"></td>

        <td><input type="text" name="stu_pwd" class="layui-input"
value="{ $stu_pwd}"></td>

    </tr>

</tbody>

</table>

<button class="layui-btn layui-btn-normal" onclick="save()">保存</button>

<script>

var $ = layui.jquery;

function save() {

    var stu_no = $('input[name="stu_no"]').val();

    var class_id = $('input[name="class_id"]').val();

```

```
var stu_name = $('input[name="stu_name"]').val();
var gender = $('input[name="gender"]:checked').val();
var stu_pwd = $('input[name="stu_pwd"]').val();
var email = $('input[name="email"]').val();

if (stu_name == '' || email == '' || stu_pwd == '') {
    layer.msg('必填项不能为空', {
        icon: 2
    })
} else {
    $.post('/houtai/stu/edit', {
        stu_no,
        class_id,
        stu_name,
        gender,
        stu_pwd,
        email,
    }, function(res) {
        if (res.code > 0) {
            layer.msg(res.msg, {
                icon: 2
            })
        } else {
            layer.msg(res.msg, {
                icon: 1
            })
            setTimeout(function() {
                parent.window.location.reload()
            }, 1000);
        }
    });
}
```

```
        }
        }, 'json');
    }

}
</script>
</body>

</html>
```

### (7) 作业管理页面

```
-----
<!-- 作业主页面 -->
-----

<style>
    .container {
        margin: 20px auto;

        width: 60%;
        text-align: center;
    }

    .layui-panel {
        padding: 10px;
    }

    .panel-head {
        text-align: start;
```

```

    }

    .panel-main {
        margin: 20px 0;
        text-align: start;
    }

    .panel-foot {
        display: flex;
        justify-content: space-between;
        align-items: center;
    }
</style>
<div class="container">
    <div class="layui-form">
        <div class="layui-form-item">
            <!-- 添加、清空数据框 -->
            <div class="layui-inline">
                <button class="layui-btn layui-btn-primary layui-border-black"
onclick="add('{${class.class_id}')">布置作业</button>
                <button class="layui-btn layui-btn-primary layui-border-blue <?=
count($work) == 0 ? "layui-btn-disabled" : "" ?>" onclick="
get_zip('{${class.class_id}')" <?= count($work) == 0 ? "disabled" : "" ?>>作业导出
</button>
            </div>
        </div>
    </div>
</div>
<div class="layui-row layui-col-space15">
    {volist name='work' id='work'}

```

```

<div class="layui-col-md4">
    <div class="layui-panel" style="color:<?= (($recent_date >
$work['work_last_time']) || ($work['status'] == 1)) ? "#ccc" : "" ?> ;">
        <div class="panel-head">
            <h2>作业{$work.work_id}</h2>
            <hr>
        </div>
        <div class="panel-main" style="height: 80px;">
            <div class="">
                介绍: <?= (mb_strlen($work['work_remarks']) > 100) ?
mb_substr($work['work_remarks'], 0, 100) . "...": $work['work_remarks'] ?>
            </div>
        </div>
        <div class="panel-foot">
            <i class="">截止时间: {$work.work_last_time|date='Y-m-d
H:i'}</i>
            <button type="button" class="layui-btn layui-btn-sm layui-
btn-warm" onclick='edit("{class.class_id}", "{$work.id}", "{$work.work_id}")'>
                <i class="layui-icon">&#xe602;</i>
            </button>
        </div>
    </div>
</div>
</volist>
</div>
</div>
<script>
    // 预定义变量

```

```
// 作业导出
function get_zip(class_id) {
    layer.confirm('确定要下载吗? ', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        layer.msg('正在导出作业, 请勿重复点击!', {
            icon: 1,
            time: 2000
        });
        window.location.href = '/houtai/work/get_zip?class_id=' + class_id;
    });
}

// 添加
function add(class_id) {
    layer.open({
        type: 2,
        title: '作业布置',
        maxmin: false, //开启最大化最小化按钮
        area: ['80%', '100%'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/work/add?class_id=' + class_id,
        offset: 't',
        shade: 0.3,
        shadeClose: true
    });
}
```

```
// 修改
function edit(class_id, id, work_id) {
    layer.open({
        type: 2,
        title: '作业管理',
        maxmin: false, //开启最大化最小化按钮
        area: ['80%', '100%'],
        shadeClose: false,
        anim: 5,
        content: '/houtai/work/edit?class_id=' + class_id + '&id=' + id + '
&work_id=' + work_id,
        offset: 't',
        shade: 0.3,
        maxmin: true,
        shadeClose: true
    });
}
</script>
-----
<!-- 作业添加页面 -->
-----
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>作业布置</title>
```

```

<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
<script src="{__LAYOUT_JS_PATH__}"></script>
</head>

<body style="padding: 10px;">
  <div class="layui-form layui-form-pane">
    <div class="layui-form-item">
      <label class="layui-form-label">作业次数</label>
      <div class="layui-input-block">
        <input type="text" name="work_id" autocomplete="on" placeholder="
例: 1、2、3" class="layui-input">
      </div>
    </div>
    <div class="layui-form-item">
      <label class="layui-form-label">截止时间</label>
      <div class="layui-input-block">
        <input type="text" name="work_last_time" autocomplete="off"
class="layui-input" id="date">
      </div>
    </div>
    <div class="layui-form-item layui-form-text">
      <label class="layui-form-label">作业介绍</label>
      <div class="layui-input-block">
        <textarea placeholder="请输入作业要求" class="layui-textarea"
name="work_remarks"></textarea>
      </div>
    </div>
    <div class="layui-form-item">
      <label class="layui-form-label">作业状态</label>
      <div class="layui-input-block">

```



```

        <input type="checkbox" name="status" lay-skin="switch" lay-text="
开启|关闭" checked>
    </div>
</div>
<div class="layui-form-item">
    <button class="layui-btn layui-btn-fluid layui-btn-normal"
onclick="save('{class_id}')">保存</button>
</div>
</div>
<script>
    var laydate = layui.laydate;
    var $ = layui.jquery;
    var form = layui.form;
    var layer = layui.layer;

    //执行一个 laydate 实例
    laydate.render({
        elem: '#date', //指定元素
        type: 'datetime',
    });

    //保存数据
    function save(class_id) {
        var work_id = $('input[name="work_id"]').val();
        var work_last_time = $('input[name="work_last_time"]').val();
        var work_remarks = $('textarea[name="work_remarks"]').val();
        var status = $('input[name="status"]').is(":checked") ? 0 : 1;

        // if (work_id == '' || work_last_time == '' || work_remarks == '') {
        //     layer.msg('必填项不能为空', {

```

```
        //      icon: 2
        //    })
        // } else {
        // $.post('/houtai/work/add', {
        //   class_id,
        //   work_id,
        //   work_last_time,
        //   work_remarks,
        //   status,
        // }, function(res) {
        //   if (res.code > 0) {
        //     layer.msg(res.msg, {
        //       icon: 2
        //     })
        //   } else {
        //     layer.msg(res.msg, {
        //       icon: 1
        //     })
        //     setTimeout(function() {
        //       parent.window.location.reload()
        //     }, 1000);
        //   }
        // }, 'json');
        // }
    }
</script>
</body>
</html>
```

```
-----
<!-- 作业修改页面 -->
-----
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <title>作业布置</title>
```

```
    <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
```

```
    <script src="{__LAYOUT_JS_PATH__}"></script>
```

```
</head>
```

```
<body style="padding: 10px;">
```

```
    <input type="hidden" name="id" value="{work.id}">
```

```
    <div class="layui-form layui-form-pane">
```

```
        <div class="layui-form-item">
```

```
            <label class="layui-form-label">作业次数</label>
```

```
            <div class="layui-input-block">
```

```
                <input type="text" name="work_id" autocomplete="on" placeholder="例：1、2、3" class="layui-input"
```

```
                value="{work.work_id}" disabled>
```

```
            </div>
```

```
        </div>
```

```
        <div class="layui-form-item">
```

```
            <label class="layui-form-label">截止时间</label>
```

```

        <div class="layui-input-block">
            <input type="text" name="work_last_time" autocomplete="off"
class="layui-input" id="date"
            value="{ $work.work_last_time }">
        </div>
    </div>
    <div class="layui-form-item layui-form-text">
        <label class="layui-form-label">作业介绍</label>
        <div class="layui-input-block">
            <textarea placeholder="请输入作业要求" class="layui-textarea"
            name="work_remarks"> { $work.work_remarks }</textarea>
        </div>
    </div>
    <div class="layui-form-item">
        <label class="layui-form-label">作业状态</label>
        <div class="layui-input-block">
            <input type="checkbox" name="status" lay-skin="switch" lay-text="
开启|关闭" { $work.status==0?"checked":"" }>
        </div>
    </div>
    <div class="layui-form-item">
        <button class="layui-btn layui-btn-normal"
onclick="save('{ $work.class_id }')">保存</button>
        <button class="layui-btn layui-btn-danger"
            onclick='del("{ $work.class_id }","{ $work.id }","{ $work.work_id }")'>
删除</button>
    </div>
    <hr>
    <table class="layui-table">
        <thead>

```

```

        <tr>
            <td>学号</td>
            <td>状态</td>
            <td>提交时间</td>
            <td>作业成绩</td>
            <td>操作</td>
        </tr>
    </thead>
    <tbody>
        {volist name='works' id='works'}
        <tr style='color:<?= $works["is_true"] != 1 ? "#ccc" : "" ?>'>
            <td>{$works.stu_no}</td>
            <td><?= $works['is_true'] == 1 ? "已上传" : ($works['is_true']
== 0 ? "未上传" : "未确认") ?></td>
            <td>{$works.save_time}</td>
            <td>{$works.all_score}</td>
            <td>
                <button
                    class="layui-btn layui-btn-primary layui-border-black
layui-btn-xs <?= $works["is_true"] != 1 ? "layui-btn-disabled" : "" ?>"
                    <?= $works["is_true"] != 1 ? "disabled" : "" ?>>
                    {if ( $works.is_true != 1) } <span>禁止访问</span>
                    {else /} <a
href="/storage/{$works.class_id}/stu_work/{$works.work_id}/{$works.stu_no}"
                    target="_blank">访问作业</a>
                {/if}
            </button>
            <button
                class="layui-btn layui-btn-normal layui-btn-xs <?=
$works["is_true"] != 1 ? "layui-btn-disabled" : "" ?>"

```

```
        <?=$works["is_true"] != 1 ? "disabled" : "" ?>
        onclick="detail('{works.class_id}','{works.stu_no}',
'{$works.work_id}')">
        <i class="layui-icon">&#xe609;</i>打分情况
    </button>
</td>
</tr>
{/volist}
</tbody>
</table>
</div>
<script>
var laydate = layui.laydate;
var $ = layui.jquery;

//执行一个 laydate 实例
laydate.render({
    elem: '#date', //指定元素
    type: 'datetime',
});

function save(class_id) {
    var id = $('input[name="id"]').val();
    var work_last_time = $('input[name="work_last_time"]').val();
    var work_remarks = $('textarea[name="work_remarks"]').val();
    var status = $('input[name="status"]').is(":checked") ? 0 : 1;

    if (work_last_time == '' || work_remarks == '') {
        layer.msg('必填项不能为空', {
```

```
        icon: 2
    })
} else {
    $.post('/houtai/work/edit', {
        id,
        class_id,
        work_last_time,
        work_remarks,
        status,
    }, function(res) {
        if (res.code > 0) {
            layer.msg(res.msg, {
                icon: 2
            })
        } else {
            layer.msg(res.msg, {
                icon: 1
            })
            setTimeout(function() {
                parent.window.location.reload()
            }, 1000);
        }
    }, 'json');
}

function del(class_id, id, work_id) {
    layer.confirm('确定要删除?', {
        btn: ['确定', '取消'] //按钮
```

```
    }, function() {
        $.post('/houtai/work/del', {
            id,
            class_id,
            work_id
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
                })
            } else {
                layer.msg(res.msg, {
                    icon: 1
                })
                parent.window.location.reload();
            }
        }, 'json')

    });
}

// 评论方法
function detail(class_id, stu_no, work_id) {
    layer.open({
        type: 2,
        title: '作业评价',
        shadeClose: true,
        shade: 0.8,
        area: ['580px', '90%'],
        anim: 5,
```



```
        offset: 't',
        content: '/houtai/work/detail?stu_no=' + stu_no + '&work_id=' + work_id
+ '&class_id=' + class_id,
    });
}
</script>
</script>
</body>

</html>

-----

<!-- 作业评分详情页面 -->

-----

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>作业评价</title>
    <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
    <script src="{__LAYOUT_JS_PATH__}"></script>
</head>

<style>
.container {
    margin: 0.01px auto;

```

```
}

.box {
  display: flex;
  justify-content: center;
  align-items: center;
}
</style>

<body>
  <div class="container">
    <table class="layui-table">
      <thead>
        <tr>
          <th>学号</th>
          <th>评分</th>
        </tr>
      </thead>
      <tbody>
        {volist name='score' id='score'}
        <tr>
          <td>{$score.to_stu_no}</td>
          <td id="{ $score.id}">{$score.score / 10}</td>
        </tr>
        <script>
        var rate = layui.rate;

        rate.render({
          elem: "#{$score.id}",
```

```
        value: "{$score.score}" / 10,
        text: true,
        length: 10,
        readonly: true
    })
</script>
{/volist}
</tbody>
</table>
</div>
</body>
</html>
```

### (8) 成绩管理页面

```
-----
<!-- 作业成绩主页面 -->
-----

<style>
.container {
    margin: 20px auto;

    width: 90%;
    text-align: center;
}
}
```

```

</style>
<div class="container">
  <div class="layui-form">
    <div class="layui-form-item">
      <!-- 添加、清空数据框 -->
      <div class="layui-inline">
        <button
          class="layui-btn layui-btn-primary layui-border-blue <?=
count($work) == 0 ? "layui-btn-disabled" : "" ?>"
          onclick=" get_zip('{ $class.class_id}')" <?= count($work) ==
0 ? "disabled" : "" ?>>成绩导出</button>
        </div>
      </div>
    </div>
  </div>
  <table class="layui-table">
    <colgroup>
      <col width="60">
      <col width="100">
    </colgroup>
    <thead>
      <tr>
        <td>学号</td>
        <td>姓名</td>
        {volist name='work' id='work'}
        <td>作业{$work.work_id}</td>
        {/volist}
        <td>作业成绩</td>
      </tr>
    </thead>
    <tbody>

```

```
{volist name='stu' id='stu'}
<tr>
    <td>{$stu.stu_no}</td>
    <td>{$stu.stu_name}</td>
    <?php foreach ($stu['work'] as $k => $v) : ?>
    <td><?= $v['score_all'] ?></td>
    <?php endforeach; ?>
    <td><?= number_format($stu['score_all'] / $work_count, 2) ?></td>
</tr>
{/volist}

</tbody>
</table>
<script>
// 作业导出
function get_zip(class_id) {
    layer.confirm('确定要下载吗?', {
        btn: ['确定', '取消'] //按钮
    }, function() {
        window.location.href = '/houtai/score/get_zip?class_id=' + class_id;

        setTimeout(function() {
            window.location.reload();
        }, 1000)
    });
}
</script>
</div>
```

### 5.3.2 学生作业提交页面模板

学生作业上传页面，包含登陆页面，班级信息展示页面、作业上传页面、作业评分页面。

——account	--学生登录认证页面
login.php	
——layout.php	--作业上传系统主框架
——public	--作业上传系统公共文件
header.php	--页脚
footer.php	--页眉
——home	--主页
index.php	
——up_work	--作业上传页面
index.php	--作业查看页面
details.php	--作业上传页面
——dis_work	--作业评分页面
index.php	--作业查看页面
display.php	--作业评分详情页面
remarks.php	--作业评分页面

#### (1) 学生登录认证页面

```
-----
<!-- 学生登录认证页面 -->
-----
```

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>作业上传评分系统学生门户</title>

  <!-- 引入 login.css 样式文件 -->

  <link rel="stylesheet" href="/static/css/login.css">

  <!-- 引入 layui 组件库 -->

  <link rel="stylesheet" href="/static/layui/css/layui.css">

  <script src="/static/layui/layui.js"></script>

  <!-- 引入 vantajs 界面库 -->

  <script src="/static/js/three.r134.min.js"></script>

  <script src="/static/js/vanta.birds.min.js"></script>

  <!-- 设置网站图标 -->

  <link rel="shortcut icon" href="/static/images/index.ico" type="image/x-icon">

</head>

<body>

  <div class="head"></div>

  <div class="layui-container">

    <div class="head-title">

      <span class="title">作业上传评分系统学生门户</span>

    </div>

    <div class="layui-form">

      <div class="form-item">

        <div class="label"><i class="layui-icon layui-icon-
location"></i></div>

        <div class="input-item">
```

```

                <input type="text" name="class_id" required lay-
verify="required" placeholder="请输入班级代码"
                autocomplete="off" class="layui-input" value="">
            </div>
        </div>
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-
username"></i></div>
            <div class="input-item">
                <input type="text" name="stu_no" required lay-
verify="required" placeholder="请输入学号"
                autocomplete="off" class="layui-input" value="">
            </div>
        </div>
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-
password"></i></div>
            <div class="input-item">
                <input type="password" name="stu_pwd" required lay-
verify="required" placeholder="请输入密码"
                autocomplete="off" class="layui-input" value="">
            </div>
        </div>
        <div class="form-item">
            <div class="label"><i class="layui-icon layui-icon-
vercode"></i></div>
            <div class="input-item captcha">
                <input type="text" name="captcha" required lay-
verify="required" placeholder="请输入验证码"
                autocomplete="off" class="layui-input">
                

```



```
        </div>
    </div>
    <div class="form-item">
        <button class="layui-btn layui-btn-normal" onclick="do_login()">登
录</button>
    </div>
</div>
</div>
</div>
<script>
VANTA.BIRDS({
    el: ".head",
    mouseControls: true,
    touchControls: true,
    gyroControls: false,
    minHeight: 200.00,
    minWidth: 200.00,
    scale: 1.00,
    scaleMobile: 1.00
})

$ = layui.jquery;

// 刷新验证码
function reloadveriimg(obj) {
    // 加入随机字符，表示验证码图片已经发生改变
    $(obj).attr('src', '/www/account/verify?rand=' + Math.random());
}
```

```
// 登录校验
function do_login() {
    var class_id = $('input[name="class_id"]').val();
    var stu_no = $('input[name="stu_no"]').val();
    var stu_pwd = $('input[name="stu_pwd"]').val();
    var captcha = $('input[name="captcha"]').val();

    if (class_id == '' || stu_no == '' || stu_pwd == '' || captcha == '') {
        layer.msg('必填项不能为空', {
            icon: 2
        })
    } else {
        $.post('/www/account/do_login', {
            class_id,
            stu_no,
            stu_pwd,
            captcha
        }, function(res) {
            console.log(res)
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
                })
                reloadveriiimg($('#captcha'));
            } else {
                layer.msg(res.msg, {
                    icon: 1
                })
            }
        })
    }
}
```

```
        setTimeout(function() {  
            window.location.href = '/www/welcome'  
        }, 1000);  
    }  
    }, 'json');  
}  
}  
</script>  
</body>  
</html>
```

## (2) 作业上传系统主框架

```
-----  
<!-- 作业上传系统主框架 -->  
-----  
<!DOCTYPE html>  
<html lang="en">  
  
<head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>{$class.title}</title>  
    <link rel="stylesheet" href="{__LAYUI_CSS_PATH__}">  
    <script src="{__LAYUI_JS_PATH__}"></script>  
</head>  
<style>  
.pagination {  
    display: inline-block;
```

```
padding-left: 0;
margin: 20px 0;
border-radius: 4px;
}

.pagination>li {
    display: inline;
}

.pagination>li>a,
.pagination>li>span {
    position: relative;
    float: left;
    padding: 6px 12px;
    margin-left: -1px;
    line-height: 1.42857143;
    color: #337ab7;
    text-decoration: none;
    background-color: #fff;
    border: 1px solid #ddd;
}

.pagination>li:first-child>a,
.pagination>li:first-child>span {
    margin-left: 0;
    border-top-left-radius: 4px;
    border-bottom-left-radius: 4px;
}
```

```
.pagination>li:last-child>a,  
.pagination>li:last-child>span {  
    border-top-right-radius: 4px;  
    border-bottom-right-radius: 4px;  
}  
  
.pagination>li>a:hover,  
.pagination>li>span:hover,  
.pagination>li>a:focus,  
.pagination>li>span:focus {  
    z-index: 2;  
    color: #23527c;  
    background-color: #eee;  
    border-color: #ddd;  
}  
  
.pagination>.active>a,  
.pagination>.active>span,  
.pagination>.active>a:hover,  
.pagination>.active>span:hover,  
.pagination>.active>a:focus,  
.pagination>.active>span:focus {  
    z-index: 3;  
    color: #fff;  
    cursor: default;  
    background-color: #337ab7;  
    border-color: #337ab7;  
}
```

```
.pagination>.disabled>span,  
.pagination>.disabled>span:hover,  
.pagination>.disabled>span:focus,  
.pagination>.disabled>a,  
.pagination>.disabled>a:hover,  
.pagination>.disabled>a:focus {  
    color: #777;  
    cursor: not-allowed;  
    background-color: #fff;  
    border-color: #ddd;  
}  
</style>  
  
<body>  
    {include file="public/header"}  
    {__CONTENT__}  
    {include file="public/footer"}  
</body>  
</html>
```

### (3) 作业上传系统公共页眉、页脚

```
-----  
<!-- 作业上传系统公共页眉 -->  
-----
```

```
<style>  
    .header {  
        text-align: center;  
        margin: 10px auto;  
    }
```

```
.layui-nav {
    display: flex;
    justify-content: center;
    align-items: center;

    background-color: white;
}

.layui-nav .layui-nav-item a {
    color: #000;
}

.layui-nav .layui-nav-item a:hover,
.layui-nav .layui-this a {
    color: #1E9FFF;
}

.layui-nav .layui-this:after,
.layui-nav-bar {
    background-color: #1E9FFF;
}

.exit {
    position: absolute;
    top: 20px;
    right: 10px;
}
}
```

```

</style>
<button class="layui-btn layui-btn-primary layui-border-black exit"
onclick="exit()">退出</button>
<div class="header">
    <h1 style="margin: 20px auto;">{$class.class_id} 班作业上传评分系统
【hello~{$stu.stu_name}】 </h1>
    <ul class="layui-nav" lay-filter="">
        <li class="layui-nav-item {$class.title=='学生主页'? 'layui-this' : ''}"><a
href="/www/welcome">学生主页</a></li>
        <li class="layui-nav-item {$class.title=='作业上传'? 'layui-this' :
''}"><a href="/www/upwork">作业上传</a></li>
        <li class="layui-nav-item {$class.title=='作业展示'? 'layui-this' :
''}"><a href=" /www/diswork">作业展示</a></li>
    </ul>
</div>
<script>
    //注意: 选项卡 依赖 element 模块, 否则无法进行功能性操作
    var element = layui.element;
    var layer = layui.layer;
    var $ = layui.jquery;

    // 退出登录
    function exit() {
        layer.confirm('是否退出登录', {
            btn: ['确定', '取消'] //按钮
        }, function() {
            $.post('/www/base/exit', {}, function(res) {
                layer.msg('退出成功!', {
                    icon: 1
                });
                window.location.href = "/www/login";
            });
        });
    }

```



```
        }, 'json')

    });
}
</script>
```

-----

```
<!-- 作业上传系统公共页脚 -->
```

-----

```
<style>
.footer {
    position: relative;
    bottom: 10px;
    width: 100%;
    text-align: center;
}
</style>
<div class="footer">
    <p>ThinkPHP</p>
    <p>以及勤劳的自己</p>
</div>
```

#### (4) 作业上传系统主页

-----

```
<!-- 作业上传系统主页 -->
```

-----

```
<style>
.container {
    margin: 20px auto;
}
```

```

width: 40%;
text-align: center;
}
</style>
<div class="container">
  <div class="layui-form layui-form-pane">
    <div class="layui-form-item">
      <label class="layui-form-label">学号</label>
      <div class="layui-input-block">
        <input type="text" name="class_id" autocomplete="on" class="layui-
input" value="{ $stu.stu_no}" disabled>
      </div>
    </div>
    <div class="layui-form-item">
      <label class="layui-form-label">姓名</label>
      <div class="layui-input-block">
        <input type="text" name="class_id" autocomplete="on" class="layui-
input" value="{ $stu.stu_name}"
          disabled>
      </div>
    </div>
    <div class="layui-form-item">
      <label class="layui-form-label">性别</label>
      <div class="layui-input-block">
        <input type="text" name="class_id" autocomplete="on" class="layui-
input"
          value="{ $stu.gender==0?"男":"女"}" disabled>
      </div>
    </div>
  </div>
</div>

```

```

<div class="layui-form-item">
    <label class="layui-form-label">登录时间</label>
    <div class="layui-input-block">
        <input type="text" name="class_id" autocomplete="on" placeholder="
例: 2005" class="layui-input"
            value="{ $stu.last_time}" disabled>
        </div>
    </div>

<div class="layui-form-item">
    <label class="layui-form-label">课程名称</label>
    <div class="layui-input-block">
        <input type="text" name="class_name" autocomplete="off"
placeholder="例: 管理信息课程" class="layui-input"
            value="{ $class.class_name}" disabled>
        </div>
    </div>

<div class="layui-form-item">
    <label class="layui-form-label">课程信息</label>
    <div class="layui-input-block">
        <input type="text" name="class_time" autocomplete="off"
placeholder="例: 公楼 C404 周二上午一二节"
            class="layui-input" value="{ $class.class_time}" disabled>
        </div>
    </div>

<div class="layui-form-item layui-form-text">
    <label class="layui-form-label">课程介绍</label>
    <div class="layui-input-block">
        <span style="text-align: start;" placeholder="请输入内容课程介绍"
class="layui-textarea"
            name="remarks">{ $class.remarks}</span>
    </div>
</div>

```

```
        </div>
    </div>
</div>
</div>
```

### (5) 作业上传页面

```
-----
<!-- 作业查看页面 -->
-----
```

```
<style>
    .container {
        margin: 60px auto;

        width: 60%;
        text-align: center;
    }

    .layui-panel {
        padding: 10px;
    }

    .panel-head {
        text-align: start;
    }

    .panel-main {
        margin: 20px 0;
        text-align: start;
    }
}
```

```

.panel-foot {
    display: flex;
    justify-content: space-between;
    align-items: center;
}
</style>
<div class="container">
    <div class="layui-row layui-col-space15">
        {volist name='work' id='work'}
        <div class="layui-col-md4">
            <div class="layui-panel">
                <div class="panel-head">
                    <h2>作业{$work.work_id} 【<?= $work['is_true'] == 1 ? '已提交' :
($work['is_true'] == 0 ? '<span style="color: red;">未提交</span>' : '<span
style="color: red;">请确认</span>') ?>】
                    <?= $work['is_true'] == 1 ? $work['all_score'] . '分' :
"" ?>
                    </h2>
                    <hr>
                </div>
                <div class="panel-main" style="height: 80px;">
                    <div class="">
                        介绍: <?= (mb_strlen($work['work_remarks']) > 100) ?
mb_substr($work['work_remarks'], 0, 100) . "...": $work['work_remarks'] ?>
                    </div>
                </div>
                <div class="panel-foot">
                    <em style="color:<?= ($recent_date > $work['work_last_time']) ?
"#ccc" : "" ?>">截止时间: {$work.work_last_time|date='Y-m-d

```



```
        window.location.reload();
    }
});
}
</script>
-----
<!-- 作业上传页面 -->
-----
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>作业管理</title>
    <link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
    <script src="{__LAYOUT_JS_PATH__}"></script>
</head>
<style>
    .container {
        margin: 60px auto;
        width: 80%;
    }
</style>

<body>
    <hr>
```

```

<hr>

<div style="position: absolute;top:10px;right:10px;">

    {if $is_work.is_true==0}<button class="layui-btn layui-btn-normal <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "layui-btn-disabled" : "" ?>" onclick="to_do(1,{ $work_id },{ $stu_no })" <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "disabled" : "" ?>>确认提交
</button>

    {else /}<button class="layui-btn layui-btn-danger <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "layui-btn-disabled" : "" ?>" onclick="to_do(0,{ $work_id },{ $stu_no })" <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "disabled" : "" ?>>取消提交
</button>

    {/if}

</div>

<div class="container">

    <hr>

    <h2>作业详情</h2>

    <p class="layui-elem-field layui-field-title">{$work.work_remarks}</p>

    <hr>

    {if $is_work.is_true==1}

    <p style="padding-bottom:10px">

        <a style="font-size: 30px;text-decoration: underline;" href="/storage/{ $class_id }/stu_work/{ $work_id }/{ $stu_no }" target="_blank"><i style="font-size: 30px;" class="layui-icon layui-icon-website"></i><em>访问作业
</em></a>

    </p>

    {/if}

    <div class="layui-upload">

        <p>(* 请将文件全部放在同一目录 *)</p>

        <p>(* 请在上传文件时确保有且仅有<em style="color: red;">一个 index</em>文件，默认访问 index 文件 *)</p>

        <p>(* 暂不支持文件夹上传 *)</p>

```



```

<p>>(* 单文件上传限制为 3MB *)</p>

<p>>(* 支持文件格式 php、sql、html、css、js、jpg、png、gif、bmp、jpeg、ico、
svg、webp、pdf *)</p>

<hr class="layui-border-red">

<hr class="layui-border-red">

    {if $is_work.is_true==0}<button type="button" class="layui-btn layui-
btn-normal <?= (($recent_date > $work['work_last_time']) || ($work['status'] ==
1)) ? "layui-btn-disabled" : "" ?>" id="testList" <?= (($recent_date >
$work['work_last_time']) || ($work['status'] == 1)) ? "disabled" : "" ?>>选择多文
件</button>

        {else /}<button class="layui-btn layui-btn-normal layui-btn-disabled"
disabled>选择多文件</button>

    {/if}

<div class="layui-upload-list" style="max-width: 1000px;">

    <table class="layui-table">

        <colgroup>

            <col>

                <col width="150">

                    <col width="260">

                        <col width="150">

                    </col>

                </colgroup>

            <thead>

                <tr>

                    <th>文件名</th>

                    <th>大小</th>

                    <th>上传进度</th>

                    <th>操作</th>

                </tr>

            </thead>

            <tbody id="demoList"></tbody>

```

```

        </table>

    </div>

    {if $is_work.is_true==0}<button type="button" class="layui-btn <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "layui-btn-disabled" : "" ?>" id="testListAction" <?=(($recent_date > $work['work_last_time']) || ($work['status'] == 1)) ? "disabled" : "" ?>>开始上传
</button>

    {else /}<button class="layui-btn layui-btn-disabled" disabled>开始上传
</button>

    {/if}

</div>

<hr class="layui-border-red">
<hr class="layui-border-red">
<table class="layui-table">
    <thead>
        <tr>
            <td>文件名称</td>
            <td>添加时间</td>
            <td>操作</td>
        </tr>
    </thead>
    <tbody>
        {volist name='works' id='works'}
        <tr>
            <td>{$works.filename}</td>
            <td>{$works.start_time}</td>
            <td>
                {if $is_work.is_true==0}<button class="layui-btn layui-
                btn-danger layui-btn-xs <?=(($recent_date > $work['work_last_time']) ||
                ($work['status'] == 1)) ? "layui-btn-disabled" : "" ?>"
                onclick="del('{$works.id}')" <?=(($recent_date > $work['work_last_time']) ||

```

```

($work['status'] == 1)) ? "disabled" : "" ?>>删除</button>
                {else /}<button class="layui-btn layui-btn-xs layui-btn-
disabled" disabled>删除</button>
                {/if}

            </td>
        </tr>
    </volist>
</tbody>
</table>

</div>
<hr>
<hr>
<script>
    var upload = layui.upload; //上传
    var $ = layui.jquery; //上传
    var element = layui.element; //上传
    var layer = layui.layer; //上传

    // 删除方法
    function del(id) {
        layer.confirm('确定要删除?', {
            btn: ['确定', '取消'] //按钮
        }, function() {
            $.post('/www/up_work/delfile', {
                id
            }, function(res) {
                if (res.code > 0) {
                    layer.msg(res.msg, {

```

```
        icon: 2
    })
} else {
    layer.msg(res.msg, {
        icon: 1
    })
    setTimeout(function() {
        window.location.reload();
    }, 100)
}
}, 'json')

});
}

// 确认提交方法
function to_do(is_true, work_id, stu_no) {

    layer.confirm("确定要" + (is_true == 1 ? "提交" : "取消") + "吗?", {
        btn: ['确定', '取消'] //按钮
    }, function() {
        $.post('/www/up_work/is_true', {
            is_true,
            work_id,
            stu_no
        }, function(res) {
            if (res.code > 0) {
                layer.msg(res.msg, {
                    icon: 2
```

```
        })
        setTimeout(function() {
            window.location.reload();
        }, 500)
    } else {
        layer.msg(res.msg, {
            icon: 1
        })
        setTimeout(function() {
            window.location.reload();
        }, 500)
    }
}, 'json')

});
}

//演示多文件列表
var uploadListIns = upload.render({
    elem: '#testList',
    elemList: $('#demoList') //列表元素对象
    ,
    url: '/www/up_work/upfile' //此处用的是第三方的 http 请求演示，实际使用时
改成您自己的上传接口即可。
    ,
    accept: 'file',
    multiple: true,
    auto: false,
    size: 3000, //文件上传尺寸设置
```

```
bindAction: '#testListAction',
data: {
    stu_no: "{$stu_no}",
    work_id: "{$work_id}"
},
choose: function(obj) {
    var that = this;

    var files = this.files = obj.pushFile(); //将每次选择的文件追加到文件队列

    //读取本地文件

    obj.preview(function(index, file, result) {
        var tr = $(['<tr id="upload-' + index + '">', '<td>' +
file.name + '</td>', '<td>' +
        (file.size / 1024).toFixed(1) + 'kb</td>',
        '<td><div class="layui-progress" lay-filter="progress-
demo-' + index +
        '"><div class="layui-progress-bar" lay-
percent=""></div></div></td>',
        '<td>',
        '<button class="layui-btn layui-btn-xs demo-reload layui-
hide">重传</button>',
        '<button class="layui-btn layui-btn-xs layui-btn-danger
demo-delete">删除</button>',
        '</td>', '</tr>']
        ].join(''));

        //单个重传
        tr.find('.demo-reload').on('click', function() {
            obj.upload(index, file);
        });
    });
}
```

```
//删除
tr.find('.demo-delete').on('click', function() {
    delete files[index]; //删除对应的文件
    tr.remove();
    uploadListIns.config.elem.next()[0].value =
        ''; //清空 input file 值，以免删除后出现同名文件不可选
});

that.elemList.append(tr);
element.render('progress'); //渲染新加的进度条组件
});
},
done: function(res, index, upload) { //成功的回调
    var that = this;
    //if(res.code == 0){ //上传成功
    var tr = that.elemList.find('tr#upload-' + index),
        tds = tr.children();
    tds.eq(3).html(''); //清空操作
    delete this.files[index]; //删除文件队列已经上传成功的文件
    if (res.code == 0) {
        layer.msg(res.msg, {
            icon: 1
        });
    } else {
        layer.msg(res.msg, {
            icon: 2
        });
        setTimeout(function() {
            window.location.reload();
        }, 1000);
    }
}
```

```

        }, '500')
    }

    return;
    //}

    this.error(index, upload);

},
allDone: function(obj) { //多文件上传完毕后的状态回调
    setTimeout(function() {
        window.location.reload()
    }, 1000);
},
error: function(index, upload) { //错误回调
    var that = this;
    var tr = that.elemList.find('tr#upload-' + index),
        tds = tr.children();
    tds.eq(3).find('.demo-reload').removeClass('layui-hide'); //显示重
传
},
progress: function(n, elem, e, index) { //注意: index 参数为 layui 2.6.6
新增
        element.progress('progress-demo-' + index, n + '%'); //执行进度条。
n 即为返回的进度百分比
    }
});
</script>
</body>
</html>

```



(6) 作业评分页面

```
-----  
<!-- 作业查看页面 -->  
-----
```

```
<style>
```

```
  .container {  
    margin: 60px auto;  
  
    width: 60%;  
    text-align: center;  
  }
```

```
  .layui-panel {  
    padding: 10px;  
  }
```

```
  .panel-head {  
    text-align: start;  
  }
```

```
  .panel-main {  
    margin: 20px 0;  
    text-align: start;  
  }
```

```
  .panel-foot {  
    display: flex;  
    justify-content: space-between;  
    align-items: center;
```

```

    }
</style>
<div class="container">
    <div class="layui-row layui-col-space15">
        {volist name='work' id='work'}
        <div class="layui-col-md4">
            <div class="layui-panel">
                <div class="panel-head">
                    <h2> 作 业  {$work.work_id}
【{$work.is_true_stu_num}/{$work.stu_num}】 </h2>
                    <hr>
                </div>
                <div class="panel-main" style="height: 80px;">
                    <div class="">
                        介绍: <?=(mb_strlen($work['work_remarks']) > 100) ?
mb_substr($work['work_remarks'], 0, 100) . "...": $work['work_remarks'] ?>
                    </div>
                </div>
                <div class="panel-foot">
                    <i class="">截止时间: {$work.work_last_time|date='Y-m-d
H:i'}</i>
                    <button type="button" class="layui-btn layui-btn-sm layui-btn-
normal" onclick='to_remarks("{$work.work_id}")'>
                        <i class="layui-icon">&#xe66e;</i>
                    </button>
                </div>
            </div>
        </div>
    </div>
    {/volist}

```

```
</div>
</div>

<script>
    // 评论
    function to_remarks(work_id) {
        layer.open({
            type: 2,
            title: '作业管理',
            maxmin: false, //开启最大化最小化按钮
            area: ['80%', '100%'],
            shadeClose: false,
            anim: 5,
            content: '/www/dis_work/display?work_id=' + work_id,
            offset: 't',
            shade: 0.3,
            shadeClose: true
        });
    }
</script>

-----

<!-- 作业评分详情页面 -->

-----

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>作业展示</title>
<link rel="stylesheet" href="{__LAYOUT_CSS_PATH__}">
<script src="{__LAYOUT_JS_PATH__}"></script>
</head>
<style>
    .container {
        margin: 20px auto;
        width: 80%;
    }
</style>
<body>
    <div class="container">

        <table class="layui-table">
            <thead>
                <tr>
                    <td>学号</td>
                    <td>状态</td>
                    <td>提交时间</td>
                    <td>评价人数</td>
                    <td>评价状态</td>
                    <td>操作</td>
                </tr>
            </thead>
            <tbody>
```

```

{volist name='works' id='works'}
|
|  |

```

```
</div>

<hr>

<hr>

<script>

    var upload = layui.upload; //上传
    var $ = layui.jquery; //上传
    var element = layui.element; //上传
    var layer = layui.layer; //上传

    // 评论方法
    function to_remarks(stu_no, work_id) {
        layer.open({
            type: 2,
            title: '作业评价',
            shade: 0.8,
            area: ['580px', '90%'],
            anim: 5,
            content: '/www/dis_work/remarks?stu_no=' + stu_no + '&work_id=' +
work_id,
            cancel: function(index, layero) {
                window.location.reload();
            }
        });
    }
</script>
</body>
</html>
```

---

```
<!-- 作业评分页面 -->
-----
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>作业评价</title>
    <link rel="stylesheet" href="{__LAYUI_CSS_PATH__}">
    <script src="{__LAYUI_JS_PATH__}"></script>
</head>
<style>
.container {
    margin: 0.01px auto;
}

.box {
    display: flex;
    justify-content: center;
    align-items: center;
}
</style>

<body>
    <div class="container">
        <div class="box">
            <span style="padding: 10px;">我的评分</span>
        </div>
    </div>
</body>
</html>
```

```
<div id="test6"></div>
</div>
<hr>
<table class="layui-table">
  <thead>
    <tr>
      <th>学号</th>
      <th>评分</th>
    </tr>
  </thead>
  <tbody>
    {volist name='remarks' id='remarks'}
    <tr>
      <td>{$remarks.to_stu_no}</td>
      <td id="{ $remarks.id}">{$remarks.score / 20}</td>
    </tr>
    <script>
    var rate = layui.rate;
    rate.render({
      elem: "#{$remarks.id}",
      value: "{$remarks.score}" / 10,
      length: 10,
      readonly: true
    })
    </script>
    {/volist}
  </tbody>
</table>
</div>
```



```
<script>

var $ = layui.jquery; //上传

var rate = layui.rate;

var layer = layui.layer;

rate.render({

  elem: '#test6',

  value: "{$score}" / 10,

  text: true,

  length: 10,

  readonly: "{$is_true}",

  setText: function(value) {

    this.span.text(value);

  },

  choose: function(value) {

    layer.confirm('评分后不能修改!', {

      btn: ['确定', '取消'] //按钮

    }, function() {

      $.post('/www/dis_work/remarks', {

        score: value * 10,

        work_id: "{$work_id}",

        stu_no: "{$stu_no}",

      }, function(res) {

        if (res.code > 0) {

          layer.msg(res.msg, {

            icon: 2

          })

        }

      })

    })

  }

});
```

```
        } else {
            layer.msg(res.msg, {
                icon: 1
            })
            setTimeout(function() {
                window.location.reload();
            }, 500)
        }
    }, 'json');
}, function() {
    layer.msg('你已取消评分');
    setTimeout(function() {
        window.location.reload();
    }, 500)
});
}
})
</script>
</body>
</html>
```

## 5.4 后端功能模块的实现

### 5.4.1 教师后台管理模块

教师模块，包含系统登陆模块，系统权限校验模块，班级管理模块、学生管理模块、作业布置模块、成绩管理模块。

	controller	
	Account.php	--系统登陆模块
	Base.php	--系统权限校验模块
	Classed.php	--班级管理模块
	Home.php	--主页模块
	Score.php	--成绩管理模块
	Stu.php	--学生管理模块
	Work.php	--作业布置模块

### (1) 系统登陆模块

```
<?php

namespace app\admin\controller; //定义多应用下命名空间

use app\BaseController;
use think\captcha\facade\Captcha;
use think\facade\Db;
use think\facade\Session;
use think\facade\Request;
use think\facade\Cookie;
use think\facade\Cache;

/**
 * 后台管理系统登录校验类
 * index:      生成登录页面方法
 * verify:     生成验证码方法
 * do_login:   登录校验方法
 *
 */
```

```
*/  
class Account extends BaseController  
{  
    // 登录界面  
    public function index()  
    {  
        return view('/account/login'); //加载视图页面  
    }  
  
    // 注册验证码  
    public function verify()  
    {  
        return Captcha::create(); //创建验证码  
    }  
  
    // 登录校验  
    public function do_login()  
    {  
        // 获取登录信息  
        $uname = trim(Request::post('uname', ''));  
        $pwd = Request::post('pwd', '');  
        $captcha = trim(Request::post('captcha', ''));  
  
        // 非空校验  
        if (!$uname) exit(json_encode(['code' => 1, 'msg' => '账户不能为空'],  
true));  
        if (!$pwd) exit(json_encode(['code' => 1, 'msg' => '密码不能为空'],  
true));  
        if (!$captcha) exit(json_encode(['code' => 1, 'msg' => '验证码不能为空'],  
true));  
    }  
}
```

```

        if (!captcha_check($captcha)) exit(json_encode(['code' => 1, 'msg' =>
'验证码错误'], true)); // 验证码校验

        $user = Db::table('admin')->where('uname', $uname)->find(); // 用户名
是否存在校验

        if (!$user) exit(json_encode(['code' => 1, 'msg' => '账户不存在'],
true));

        if ($user['pwd'] != $pwd) exit(json_encode(['code' => 1, 'msg' => '密
码错误'], true)); // 密码校验

// 通过登录

$data['last_time'] = date('Y-m-d H:i:s', time()); //记录登录时间

Db::table('admin')->where('uname', $user['uname'])->update($data); //
修改登录时间

// 分别保存用户信息到 Session【服务端记录用户信息】、cookie、cache【浏览器端
记录用户信息】

Session::set('admin', $user); //将登录信息保存到 session 中
Session::delete('captcha'); //删除验证码 session

$randstr_houtai = time() . rand(10, 99); // 存用户 id-userkey（随机字符）
Cache::set('houtai_' . $user['id'], $randstr_houtai); //保存到服务器端
Cookie::Set('randstr_houtai', $randstr_houtai); //保存到 cookie 前端

echo json_encode(['code' => 0, 'msg' => '登录成功'], true); // exit 会
导致 session 失效

    }

}

```

## (2) 系统权限校验模块

```
<?php
```

```
namespace app\admin\controller;
```

```
use app\BaseController;

use think\facade\Session;

use think\facade\Request;

use think\facade\Db;

use think\facade\Cookie;

use think\facade\Cache;

class Base extends BaseController
{
    public $admin; //登入用户

    // 初始化: 未登录的用户不允许进入
    protected function initialize()
    {
        $this->admin = Session::get('admin');

        $admin = Session::get('admin');

        if (!$admin) {
            if (Request::isAjax()) {
                exit(json_encode(['code' => 1, 'msg' => '你还未登录, 请登录']));
            }
            exit('你还未登录, 请登录
            <script>
                setTimeout(function(){window.parent.location.href="/houtai/login"},2000)
            </script>
            ');
        }
    }
}
```

```
// 2.取出 cookie 中的 randstr，做登录校验，不允许多人登录同一账号
$randstr_houtai = Cookie::get('randstr_houtai');
$cache_randstr = Cache::get('houtai_' . $admin['id']);
if ($randstr_houtai != $cache_randstr) {
    Session::delete('admin');
    if (Request::isAjax()) {
        exit(json_encode(['code' => 1, 'msg' => '你的账号在其他地方登录，
请重新登录']));
    }
    exit('你的账号在其他地方登录，请重新登录
<script>setTimeout(function(){window.parent.location.href="/houtai/login"},2000)<
/script>');
}
}

// 退出登录
public function exit()
{
    Session::delete('admin');
}
}
```

### (3) 班级管理模块

```
<?php

namespace app\admin\controller;

use think\facade\Db;
```

```

use think\facade\Request;

use think\facade\View;

use file\File;

/**
 * 后台页面类
 * index 班级主页页面
 * save 班级信息保存
 */
class Classed extends Base
{
    public function index()
    {
        $data['admin'] = $this->admin;
        $class_id = (int)Request::get('class_id', '');
        if (!$class_id) {
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        $data['class'] = Db::table('classes')->where('class_id',
$class_id)->find();

        $data['class']['title'] = '班级配置'; //动态设置网站名称

        //使用模板
        View::engine()->layout('layout');
        return View::fetch('Classed/index', $data);
    }

    public function save()
    {

```



```
$class_id = trim(Request::post('class_id', ''));
$data['class_name'] = trim(Request::post('class_name', ''));
$data['class_time'] = trim(Request::post('class_time', ''));
$data['remarks'] = trim(Request::post('remarks', ''));

// 非空验证
if ($class_id == '') {
    exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
}
if ($data['class_name'] == '') {
    exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'], true));
}

//表单长度填写限制
if (mb_strlen($data['class_name']) > 30 ||
mb_strlen($data['class_time']) > 30) {
    exit(json_encode(['code' => 1, 'msg' => '表单填写长度超出限制!'],
true));
}
if (mb_strlen($data['remarks']) > 150) {
    exit(json_encode(['code' => 1, 'msg' => '备注填写长度超出限制!'],
true));
}

// 修改数据
Db::table('classes')->where('class_id', $class_id)->update($data);
exit(json_encode(['code' => 0, 'msg' => '保存成功'], true));
}
}
```

#### (4) 学生管理模块

```
<?php

namespace app\admin\controller;

use think\facade\Config;
use think\facade\Db;
use think\facade\Request;
use think\facade\Session;
use think\facade\View;
use file\File;
use file\Zip1;
use PhpOffice\PhpSpreadsheet\IOFactory; //用于载入已有的 xlsx 文件
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx; //保存 xlsx 文件
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\SMTP;
use PHPMailer\PHPMailer\Exception;

/**
 * 后台页面类
 * 学生信息管理
 * index 学生信息主页
 * up_stu 学生一键导入页面
 * upfile 执行文件导入
 * send_email 发送邮件，支持单独发送和群发
 * add 支持单独添加学生信息
```

```
* edit 学生信息修改
* del 学生信息删除
*/
class Stu extends Base
{
    // 主页
    public function index()
    {
        $class_id = (int)Request::get('class_id', '');
        if (!$class_id) {
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        $data['class'] = Db::table('classes')->where('class_id',
$class_id)->find();

        $data['admin'] = $this->admin;
        $data['class']['title'] = '学生管理';

        // 根据条件获取数据
        $data['search'] = trim(Request::get('search', '')); //搜索内容
        $data['page'] = Request::get('page', '1'); //页数

        $where = [];
        if (!empty($data['search'])) {
            $where[] = ['stu_no', 'like', '%' . $data['search'] . '%'];
            $where[] = ['stu_name', 'like', '%' . $data['search'] . '%'];
            ($data['search'] == '男') ? $where[] = ['gender', '=', 0] : '';
            ($data['search'] == '女') ? $where[] = ['gender', '=', 1] : '';
        }
    }
}
```

```

// $sql = Db::table('stu')->where(function ($query) use ($where) {
//     $query->whereOr($where);
//     })->where('class_id', $class_id)->order('stu_no',
'asc')->fetchsql(true)->select();

// dump($where);
// dump($sql);
// 获取学生数据

        $data['stu'] = Db::table('stu')->where('class_id',
$class_id)->where(function ($query) use ($where) {
            $query->whereOr($where);
        })->order('stu_no', 'asc')->paginate([
            'list_rows' => 8,
            'query' => request()->param(),
        ]);
$data['page'] = $data['stu']->render();

// 清空文件-----
// 删除原文件
$file = new file();

    if (file_exists('storage' . '/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx')) {
        $file->unlink_file('storage' . '/' . $class_id . '/stu_data' .
 '/' . $class_id . '_stu_data.xlsx');
    }

        $data_stu =
Db::table('stu')->field('stu_no,email,stu_name,gender')->where('class_id',
$class_id)->order('stu_no', 'asc')->select()->toArray();

// 创建新的 stu_data.xlsx 文件

```

```

        $spreadsheet_stu_data = new Spreadsheet();
        $spreadsheet_stu_data->getActiveSheet()->mergeCells('A1:D1');
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('A')->set
Width(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('B')->set
Width(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('C')->set
Width(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('D')->set
Width(10);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A1')->setValue($cla
ss_id);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A2')->setValue('电子
邮箱');
        $spreadsheet_stu_data->getActiveSheet()->getCell('B2')->setValue('学号
');
        $spreadsheet_stu_data->getActiveSheet()->getCell('C2')->setValue('姓名
');
        $spreadsheet_stu_data->getActiveSheet()->getCell('D2')->setValue('性别
');

        $worksheet_stu_data = $spreadsheet_stu_data->getActiveSheet(); //读取
excel 文件
        $worksheet_stu_data->getCell('A1')->setValue($class_id);
        $worksheet_stu_data->getCell('A2')->setValue('电子邮箱');
        $worksheet_stu_data->getCell('B2')->setValue('学号');
        $worksheet_stu_data->getCell('C2')->setValue('姓名');
        $worksheet_stu_data->getCell('D2')->setValue('性别');

        for ($i = 0; $i < count($data_stu); $i++) { //循环出 excel 所有的数据,
并进行校验
                $worksheet_stu_data->setCellValue('A' . ($i + 3),
$data_stu[$i]['email']);
                $worksheet_stu_data->setCellValue('B' . ($i + 3),

```

```
$data_stu[$i]['stu_no']);
        $worksheet_stu_data->setCellValue('C' . ($i + 3),
$data_stu[$i]['stu_name']);
        $worksheet_stu_data->setCellValue('D' . ($i + 3),
$data_stu[$i]['gender'] == 0 ? "男" : "女");
    }
    $writer_stu_data = new Xlsx($spreadsheet_stu_data);
    $writer_stu_data->save('storage/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx');

    View::engine()->layout('layout');
    return View::fetch('stu/index', $data);
}

// 文件上传页面
public function up_stu()
{
    $data['class_id'] = (int)Request::get('class_id', '');

    return View::fetch('stu/up_stu', $data);
}

//学生 excel 维护和 mysql 数据表同步接口
public function upfile() //导入学生的学号不能重复、班级代码要和登入的系统一致
{
    $class_id = (int)Request::param('class_id', '');
    $fileclass = new File(); //调用 extend 内的 file 类（文件操作）
    $exts = ['xlsx']; //限制文件格式

    // //从前端获取文件上传信息
```

```
$file = Request::file('file');
$file_ext = $file->extension();

// 校验文件格式
if (in_array($file_ext, $exts) != 1) {
    exit(json_encode(['code' => 1, 'msg' => "仅支持:xls、xlsx 格式"]));
}

$spreadsheet = IOFactory::load($file); //载入 xlsx 文件
$worksheet = $spreadsheet->getActiveSheet(); //读取 excel 文件
$excel_class_id = $worksheet->getCell('A1')->getvalue(); //获取班级代

码

if ($excel_class_id != $class_id) {
    exit(json_encode(['code' => 1, 'msg' => "导入文件的班级代码不匹配，请
重试！"]));
}

$row_count = $worksheet->getHighestRow(); //读取 excel 最大行数，即学生的
数量 = row_Count - 前两个标题行

if ($row_count < 3) {
    exit(json_encode(['code' => 1, 'msg' => "表格无数据"]));
}

for ($i = 3; $i <= $row_count; $i++) { //循环出 excel 所有的数据，并进行
校验

    if ($worksheet->getCell('A' . $i)->getvalue() == "") {
        exit(json_encode(['code' => 1, 'msg' => "邮箱出现空行，请重新上
传"]));
    }

    $patt_email = "/^[A-Za-z0-9\u4e00-\u9fa5]+@[a-zA-Z0-9_-]+(\.[a-zA-
```

```

Z0-9_-]+)$/";

        if (!(preg_match($patt_email, $worksheet->getCell('A' .
$i)->getvalue())) { //如果不匹配邮箱

            exit(json_encode(['code' => 1, 'msg' => "第 $i 行的邮箱格式错误
"]););

        }

        if ($worksheet->getCell('B' . $i)->getvalue() == "") {

            exit(json_encode(['code' => 1, 'msg' => "学号出现空行, 请重新上
传"]););

        }

        if ($worksheet->getCell('C' . $i)->getvalue() == "") {

            exit(json_encode(['code' => 1, 'msg' => "姓名出现空行, 请重新上
传"]););

        }

        if ($worksheet->getCell('D' . $i)->getvalue() == "") {

            exit(json_encode(['code' => 1, 'msg' => "性别出现空行, 请重新上
传"]););

        }

        if ($worksheet->getCell('D' . $i)->getvalue() != "男" &&
$worksheet->getCell('D' . $i)->getvalue() != "女") {

            exit(json_encode(['code' => 1, 'msg' => "第 $i 行的性别格式错误
"]););

        }

        $data['stu'][$i - 3]['class_id'] = $class_id;

        $data['stu'][$i - 3]['email'] = $worksheet->getCell('A' .
$i)->getvalue();

        $data['stu'][$i - 3]['stu_no'] = $worksheet->getCell('B' .
$i)->getvalue();

        $data['stu'][$i - 3]['stu_name'] = $worksheet->getCell('C' .
$i)->getvalue();

        $data['stu'][$i - 3]['gender'] = ($worksheet->getCell('D' .

```



```
$i)->getvalue() == "男") ? "0" : "1";

        // 密码随机生成 8 位
        $characters = '1234567890abcdefghijklmnopqrstuvwxyz';
        $data['stu'][$i - 3]['stu_pwd'] = '';
        for ($j = 0; $j < 8; $j++) {
            $index = rand(0, strlen($characters) - 1);
            $data['stu'][$i - 3]['stu_pwd'] .= $characters[$index];
        }
        $data['stu'][$i - 3]['add_time'] = date('Y-m-d H:i:s', time());
        $data['stu'][$i - 3]['last_time'] = date("Y-m-d H:i:s", time());
    }

    //根据数据库的数据将学生更新到 excel 表格中
    $spreadsheet_stu_data = IOFactory::load('storage/' . $class_id .
'/stu_data' . '/' . $class_id . '_stu_data.xlsx');
    $worksheet_stu_data = $spreadsheet_stu_data->getActiveSheet(); //读取
excel 文件
    $worksheet_stu_data->getCell('A1')->setValue($class_id);
    $worksheet_stu_data->getCell('A2')->setValue('电子邮箱');
    $worksheet_stu_data->getCell('B2')->setValue('学号');
    $worksheet_stu_data->getCell('C2')->setValue('姓名');
    $worksheet_stu_data->getCell('D2')->setValue('性别');
    for ($i = 3; $i <= $row_count; $i++) { //循环出 excel 所有的数据, 并进行
校验
        $worksheet_stu_data->setCellValue('A' . $i, $data['stu'][$i -
3]['email']);
        $worksheet_stu_data->setCellValue('B' . $i, $data['stu'][$i -
3]['stu_no']);
        $worksheet_stu_data->setCellValue('C' . $i, $data['stu'][$i -
3]['stu_name']);
```

```

        $worksheet_stu_data->setCellValue('D' . $i, $data['stu'][$i -
3]['gender'] == 0 ? "男" : "女");
    }

    $writer_stu_data = new Xlsx($spreadsheet_stu_data);

    $writer_stu_data->save('storage/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx');

    Db::table('stu')->insertAll($data['stu']); // 添加新的学生数据

    exit(json_encode(['code' => 0, 'msg' => "学生导入成功"]));
}

//邮箱发送接口
public function send_email()
{
    $class_id = (int)Request::post('class_id', '');
    $class_name = Request::post('class_name', '');
    $stu_no = (int)Request::post('stu_no', ''); // 单发标识
    $is_all = (int)Request::post('is_all', ''); //群发标识
    $admin = $this->admin; //用户信息

    if ($class_id == '') {
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }
    if ($class_name == '') {
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }

    if ($stu_no) { //单独发送密码
        $stu = Db::table('stu')->where(['class_id' => $class_id, 'stu_no'

```

```

=> $stu_no])->find());

        $mail = new PHPMailer(true); //开启邮箱发送

        try {
            //服务器配置

            $mail->CharSet = "UTF-8"; //设定邮件编码

            $mail->SMTPDebug = 0; // 调试模式输出

            $mail->isSMTP(); // 使用 SMTP

            $mail->Host = $admin['email_system']; // SMTP 服务器

            $mail->SMTPAuth = true; // 允许 SMTP 认证

            $mail->Username = $admin['email']; // SMTP 用户名 即邮箱的用户名

            $mail->Password = $admin['check_code']; // SMTP 密码 部分邮箱是授权码(例如 163 邮箱)

            $mail->SMTPSecure = 'ssl'; // 允许 TLS 或者 ssl 协议

            $mail->Port = 465; // 服务器端口 25 或者 465 具体要看邮箱服务器支持

            $mail->setFrom($admin['email'], $class_name); //发件人

            $mail->addAddress($stu['email'], $stu['stu_name']); // 收件人

            // $mail->addAddress('ellen@example.com'); // 可添加多个收件人

            $mail->addReplyTo($admin['email'], $class_name); //回复的时候回复给哪个邮箱 建议和发件人一致

            //Content

            $mail->isHTML(true); // 是否以 HTML 文档格式发送 发送后客户端可直接显示对应 HTML 内容

            $mail->Subject = $stu['stu_name'] . '你好! 智慧作业上传评分系统门户【' . $class_name . '】';

            $mail->Body = '<h1>智慧作业上传评分系统门户</h1><p>网站地址:

```

```

http://43.138.189.65:3002/班级代码: ' . $stu['class_id'] . '账号: ' .
$stu['stu_no'] . '密码: ' . $stu['stu_pwd'] . '
        $mail->AltBody = '<p>网站地址: http://43.138.189.65:3002/班级代码: ' . $stu['class_id'] . '账号: ' . $stu['stu_no'] . '密码: ' . $stu['stu_pwd'] . '
        $mail->send();
        exit(json_encode(['code' => 0, 'msg' => "邮件发送成功"]));
    } catch (Exception $e) {
        exit(json_encode(['code' => 1, 'msg' => '邮件发送失败' .
$mail->ErrorInfo]));
    }
}
if ($is_all == 1) {
    $stu_data = Db::table('stu')->where('class_id',
$class_id)->select()->toArray();
    foreach ($stu_data as $v) {
        $mail = new PHPMailer(true); //开启邮箱发送

        try {
            //服务器配置
            $mail->CharSet = "UTF-8"; //设定邮件编码
            $mail->SMTPDebug = 0; // 调试模式输出
            $mail->isSMTP(); // 使用 SMTP
            $mail->Host = $admin['email_system']; //
SMTP 服务器
            $mail->SMTPAuth = true; // 允许 SMTP 认
证
            $mail->Username = $admin['email']; // SMTP 用户名 即邮箱的
用户名
            $mail->Password = $admin['check_code']; // SMTP 密
码 部分邮箱是授权码(例如 163 邮箱)

```

```

        $mail->SMTPSecure = 'ssl'; // 允许 TLS 或者 ssl 协议

        $mail->Port = 465; // 服务器端口
25 或者 465 具体要看邮箱服务器支持

        $mail->setFrom($admin['email'], $class_name); //发件人
        $mail->addReplyTo($admin['email'], $class_name); //回复的时候回复给哪个邮箱 建议和发件人一致

        $mail->addAddress($v['email'], $v['stu_name']); // 收件人
        $mail->isHTML(true); // 是否以 HTML 文档格式发送 发送后客户端可直接显示对应 HTML 内容

        $mail->Subject = $v['stu_name'] . '你好! 智慧作业上传评分系统门户【' . $class_name . '】';

        $mail->Body = '<h1>智慧作业上传评分系统门户</h1><p>班级代码: ' . $v['class_id'] . '</p><p>账号: ' . $v['stu_no'] . '</p><p>密码: ' . $v['stu_pwd'] . '</p><p>网站地址: http://43.138.189.65:3002/</p>';

        $mail->AltBody = '<p>网站地址: http://43.138.189.65:3002/</p><p>班级代码: ' . $v['class_id'] . '</p><p>账号: ' . $v['stu_no'] . '</p><p>密码: ' . $v['stu_pwd'] . '</p>';

        $mail->send();

    } catch (Exception $e) {
        exit(json_encode(['code' => 1, 'msg' => '邮件发送失败' . $mail->ErrorInfo]));
    }

    exit(json_encode(['code' => 0, 'msg' => "邮件发送成功"]));
}

//学生单独添加
public function add()

```

```
{
    if (Request::isPost()) {
        $class_id = (int)Request::post('class_id', '');
        $data['class_id'] = (int)Request::post('class_id', '');
        $data['stu_no'] = (int)Request::post('stu_no', '');
        $data['stu_name'] = trim(Request::post('stu_name', ''));
        $data['email'] = trim(Request::post('email', ''));
        $data['gender'] = (int)Request::post('gender', '');

        if ($data['stu_no'] == '') {
            exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
        }
        if ($data['stu_name'] == '') {
            exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
        }
        if ($data['class_id'] == '') {
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }
        if ($data['email'] == '') {
            exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
        }
        $patt_email = "/^[A-Za-z0-9\u4e00-\u9fa5]+@[a-zA-Z0-9_-]+(\.[a-zA-Z0-9_-]+)+$/";
        if (!(preg_match($patt_email, $data['email']))) { //如果不匹配邮箱
            exit(json_encode(['code' => 1, 'msg' => "邮箱格式错误"]));
        }
    }
}
```

```

        $res = Db::table('stu')->where(['stu_no' => $data['stu_no'],
'class_id' => $data['class_id']])->find();

        if ($res) {
            exit(json_encode(['code' => 1, 'msg' => "学号已经存在"]));
        }

        // 密码随机生成 8 位

        $characters = '1234567890abcdefghijklmnopqrstuvwxyz';

        $data['stu_pwd'] = '';

        for ($i = 0; $i < 8; $i++) {
            $index = rand(0, strlen($characters) - 1);
            $data['stu_pwd'] .= $characters[$index];
        }

        $data['add_time'] = date('Y-m-d H:i:s', time());
        $data['last_time'] = date("Y-m-d H:i:s", time());

        // 清空文件-----
        // 删除原文件

        $file = new file();

        if (file_exists('storage' . '/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx')) {
            $file->unlink_file('storage' . '/' . $class_id . '/stu_data' .
'/' . $class_id . '_stu_data.xlsx');
        }

        $data_stu =
        Db::table('stu')->field('stu_no,email,stu_name,gender')->where('class_id',
$class_id)->order('stu_no', 'asc')->select()->toArray();

        // 创建新的 stu_data.xlsx 文件
    
```

```

        $spreadsheet_stu_data = new Spreadsheet();
        $spreadsheet_stu_data->getActiveSheet()->mergeCells('A1:D1');
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('A')->
setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('B')->
setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('C')->
setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('D')->
setWidth(10);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A1')->setValue($
class_id);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A2')->setValue('
电子邮箱');
        $spreadsheet_stu_data->getActiveSheet()->getCell('B2')->setValue('
学号');
        $spreadsheet_stu_data->getActiveSheet()->getCell('C2')->setValue('
姓名');
        $spreadsheet_stu_data->getActiveSheet()->getCell('D2')->setValue('
性别');

        $worksheet_stu_data = $spreadsheet_stu_data->getActiveSheet(); //
读取 excel 文件
        $worksheet_stu_data->getCell('A1')->setValue($class_id);
        $worksheet_stu_data->getCell('A2')->setValue('电子邮箱');
        $worksheet_stu_data->getCell('B2')->setValue('学号');
        $worksheet_stu_data->getCell('C2')->setValue('姓名');
        $worksheet_stu_data->getCell('D2')->setValue('性别');

        for ($i = 0; $i < count($data_stu); $i++) { //循环出 excel 所有的数
据，并进行校验
            $worksheet_stu_data->setCellValue('A' . ($i + 3),
$data_stu[$i]['email']);
            $worksheet_stu_data->setCellValue('B' . ($i + 3),

```



```
$data_stu[$i]['stu_no']);
                $worksheet_stu_data->setCellValue('C' . ($i + 3),
$data_stu[$i]['stu_name']);
                $worksheet_stu_data->setCellValue('D' . ($i + 3),
$data_stu[$i]['gender'] == 0 ? "男" : "女");
        }
        $writer_stu_data = new Xlsx($spreadsheet_stu_data);
        $writer_stu_data->save('storage/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx');

        $res = Db::table('stu')->insert($data);
        if ($res) {
            exit(json_encode(['code' => 0, 'msg' => '保存成功'], true));
        }
        exit(json_encode(['code' => 1, 'msg' => '保存失败'], true));
    } else {
        $data['class_id'] = (int)Request::get('class_id', '');
        return View::fetch('stu/add', $data);
    }
}

// 修改
public function edit()
{
    if (Request::isPost()) {
        $stu_no = (int)Request::post('stu_no', ''); //学号默认不能修改
        $class_id = (int)Request::post('class_id', '');
        $data['stu_name'] = trim(Request::post('stu_name', ''));
        $data['gender'] = (int)Request::post('gender', '');
        $data['stu_pwd'] = Request::post('stu_pwd', '');
    }
}
```

```

$data['email'] = trim(Request::post('email', ''));

if (empty($stu_no)) { //如果没有 get 参数, 则返回
    exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
}

if (empty($class_id)) { //如果没有 get 参数, 则返回
    exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
}

if ($data['stu_name'] == '') {
    exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
}

if ($data['stu_pwd'] == '') {
    exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
}

if ($data['email'] == '') {
    exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
}

$patt_email = "/^[A-Za-z0-9\u4e00-\u9fa5]+@[a-zA-Z0-9_-]+(\.[a-zA-Z0-9_-]+)+$/";

if (!(preg_match($patt_email, $data['email']))) { //如果不匹配邮箱
    exit(json_encode(['code' => 1, 'msg' => "邮箱格式错误"]));
}

Db::table('stu')->where(['stu_no' => $stu_no, 'class_id' =>
$class_id])->update($data);

// 清空文件-----

```

```

        // 删除原文件

        $file = new file();

        $file->unlink_file('storage' . '/' . $class_id . '/stu_data' .
        '/' . $class_id . '_stu_data.xlsx');

        $data_stu =
        Db::table('stu')->field('stu_no,email,stu_name,gender')->where('class_id',
        $class_id)->order('stu_no', 'asc')->select()->toArray();

        // 创建新的 stu_data.xlsx 文件

        $spreadsheet_stu_data = new Spreadsheet();

        $spreadsheet_stu_data->getActiveSheet()->mergeCells('A1:D1');

        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('A')->
        setWidth(20);

        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('B')->
        setWidth(20);

        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('C')->
        setWidth(20);

        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('D')->
        setWidth(10);

        $spreadsheet_stu_data->getActiveSheet()->getCell('A1')->setValue($
        class_id);

        $spreadsheet_stu_data->getActiveSheet()->getCell('A2')->setValue('
        电子邮箱');

        $spreadsheet_stu_data->getActiveSheet()->getCell('B2')->setValue('
        学号');

        $spreadsheet_stu_data->getActiveSheet()->getCell('C2')->setValue('
        姓名');

        $spreadsheet_stu_data->getActiveSheet()->getCell('D2')->setValue('
        性别');

        $worksheet_stu_data = $spreadsheet_stu_data->getActiveSheet(); //
        读取 excel 文件
    
```

```

        $worksheet_stu_data->getCell('A1')->setValue($class_id);
        $worksheet_stu_data->getCell('A2')->setValue('电子邮箱');
        $worksheet_stu_data->getCell('B2')->setValue('学号');
        $worksheet_stu_data->getCell('C2')->setValue('姓名');
        $worksheet_stu_data->getCell('D2')->setValue('性别');
        for ($i = 0; $i < count($data_stu); $i++) { //循环出 excel 所有的数
数据, 并进行校验
                $worksheet_stu_data->setCellValue('A' . ($i + 3),
$data_stu[$i]['email']);
                $worksheet_stu_data->setCellValue('B' . ($i + 3),
$data_stu[$i]['stu_no']);
                $worksheet_stu_data->setCellValue('C' . ($i + 3),
$data_stu[$i]['stu_name']);
                $worksheet_stu_data->setCellValue('D' . ($i + 3),
$data_stu[$i]['gender'] == 0 ? "男" : "女");
        }
        $writer_stu_data = new Xlsx($spreadsheet_stu_data);
        $writer_stu_data->save('storage/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx');

        exit(json_encode(['code' => 0, 'msg' => '保存成功'], true));
    } else {
        $stu_no = (int)Request::get('stu_no', '');
        $class_id = (int)Request::get('class_id', '');
        if (empty($stu_no)) { //如果没有 get 参数, 则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }
        if (empty($class_id)) { //如果没有 get 参数, 则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }
        $data = Db::table('stu')->where(['stu_no' => $stu_no, 'class_id'

```

```
=> $class_id]->find();

        return View::fetch('stu/edit', $data);
    }
}

// 学生导出
public function get_zip()
{
    $class_id = (int)Request::get('class_id', '');

    $zip1 = new Zip1();
    $zip1->zip($class_id, 'stu_data');
}

// 删除
public function del()
{
    if (Request::isPost()) {
        //删除学生的 stu、score、is_work、works
        $stu_no = (int)Request::post('stu_no', ''); //获取到学号
        $class_id = (int)Request::post('class_id', ''); //获取到学号
        $is_clear_all = (int)Request::post('is_clear_all', ''); //获取到清空权限，则删除整个数据表里的数据

        if (empty($class_id)) { //如果没有 get 参数，则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        if ($is_clear_all) { //清空所有数据
            // 清空文件-----

```

```
// 删除原文件

$file = new file();

    if (file_exists('storage' . '/' . $class_id . '/stu_data' .
'/' . $class_id . '_stu_data.xlsx'))

        $file->unlink_file('storage' . '/' . $class_id .
'/stu_data' . '/' . $class_id . '_stu_data.xlsx');

// 创建新的 stu_data.xlsx 文件

$spreadsheet_stu_data = new Spreadsheet();
$spreadsheet_stu_data->getActiveSheet()->mergeCells('A1:D1');
$spreadsheet_stu_data->getActiveSheet()->getColumnDimension('A
')->setWidth(20);
$spreadsheet_stu_data->getActiveSheet()->getColumnDimension('B
')->setWidth(20);
$spreadsheet_stu_data->getActiveSheet()->getColumnDimension('C
')->setWidth(20);
$spreadsheet_stu_data->getActiveSheet()->getColumnDimension('D
')->setWidth(10);
$spreadsheet_stu_data->getActiveSheet()->getCell('A1')->setVal
ue($class_id);
$spreadsheet_stu_data->getActiveSheet()->getCell('A2')->setVal
ue('电子邮箱');
$spreadsheet_stu_data->getActiveSheet()->getCell('B2')->setVal
ue('学号');
$spreadsheet_stu_data->getActiveSheet()->getCell('C2')->setVal
ue('姓名');
$spreadsheet_stu_data->getActiveSheet()->getCell('D2')->setVal
ue('性别');

$worksheet = $spreadsheet_stu_data->getActiveSheet();
$writer_stu_data = new Xlsx($spreadsheet_stu_data);
    $writer_stu_data->save('storage' . '/' . $class_id .
'/stu_data' . '/' . $class_id . '_stu_data.xlsx');
```

```

        Db::table('stu')->where('class_id', $class_id)->delete();
        Db::table('works')->where('class_id', $class_id)->delete();
        Db::table('is_work')->where('class_id', $class_id)->delete();
        Db::table('score')->where('class_id', $class_id)->delete();

        exit(json_encode(['code' => 0, 'msg' => '清空数据成功'], true));
    } else { //删除单条数据

        if (empty($stu_no)) { //如果没有 get 参数, 则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        Db::table('stu')->where(['stu_no' => $stu_no, 'class_id' =>
$class_id])->delete();
        Db::table('score')->where(['stu_no' => $stu_no, 'class_id' =>
$class_id])->delete();
        Db::table('is_work')->where(['stu_no' => $stu_no, 'class_id'
=> $class_id])->delete();
        Db::table('works')->where(['stu_no' => $stu_no, 'class_id' =>
$class_id])->delete();

        // 清空文件-----
        // 删除原文件
        $file = new file();
        $file->unlink_file('storage' . '/' . $class_id . '/stu_data' .
'/' . $class_id . '_stu_data.xlsx');

        $data_stu =
        Db::table('stu')->field('stu_no,email,stu_name,gender')->where('class_id',
$class_id)->order('stu_no', 'asc')->select()->toArray();
    }

```

```

        // 创建新的 stu_data.xlsx 文件
        $spreadsheet_stu_data = new Spreadsheet();
        $spreadsheet_stu_data->getActiveSheet()->mergeCells('A1:D1');
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('A')->setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('B')->setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('C')->setWidth(20);
        $spreadsheet_stu_data->getActiveSheet()->getColumnDimension('D')->setWidth(10);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A1')->setValue($class_id);
        $spreadsheet_stu_data->getActiveSheet()->getCell('A2')->setValue('电子邮箱');
        $spreadsheet_stu_data->getActiveSheet()->getCell('B2')->setValue('学号');
        $spreadsheet_stu_data->getActiveSheet()->getCell('C2')->setValue('姓名');
        $spreadsheet_stu_data->getActiveSheet()->getCell('D2')->setValue('性别');

        $worksheet_stu_data = $spreadsheet_stu_data->getActiveSheet();
//读取 excel 文件
        $worksheet_stu_data->getCell('A1')->setValue($class_id);
        $worksheet_stu_data->getCell('A2')->setValue('电子邮箱');
        $worksheet_stu_data->getCell('B2')->setValue('学号');
        $worksheet_stu_data->getCell('C2')->setValue('姓名');
        $worksheet_stu_data->getCell('D2')->setValue('性别');

        for ($i = 3; $i <= count($data_stu); $i++) { //循环出 excel 所有的数据，并进行校验
            $worksheet_stu_data->setCellValue('A' . $i, $data_stu[$i - 3]['email']);
        }
    }
}

```



```

        $worksheet_stu_data->setCellValue('B' . $i, $data_stu[$i -
3][ 'stu_no' ]);
        $worksheet_stu_data->setCellValue('C' . $i, $data_stu[$i -
3][ 'stu_name' ]);
        $worksheet_stu_data->setCellValue('D' . $i, $data_stu[$i -
3][ 'gender' ] == 0 ? "男" : "女");
    }
    $writer_stu_data = new Xlsx($spreadsheet_stu_data);
    $writer_stu_data->save('storage/' . $class_id . '/stu_data' .
 '/' . $class_id . '_stu_data.xlsx');
    exit(json_encode([ 'code' => 0, 'msg' => '删除成功'], true));
}
} else {
    exit(json_encode([ 'code' => 1, 'msg' => '非法请求'], true));
}
}

// 生成 excel 表
// public function reset_excel()
// {
//     $fileclass = new File(); //调用 extend 内的 file 类（文件操作）

//     //读取 excel 表格的内容
//     $file = "storage/2005/stu_data/2005 班录入名单.xls";

//     $spreadsheet = IOFactory::load($file); //载入 xlsx 文件

//     $worksheet = $spreadsheet->getActiveSheet();

//     $data[ 'class_id' ] = $worksheet->getCell('A1')->getvalue();

```

```

//      $row_count = $worksheet->getHighestRow(); //最大行数,数据量

//      for ($i = 3; $i <= $row_count; $i++) { //循环出所有的数据
//          if ($worksheet->getCell('A' . $i)->getValue() == "") {
//              $fileclass->unlink_file($file); //删除文件
//              exit(json_encode(['code' => 1, 'msg' => "邮箱出现空行,请重新
上传"])));
//          }

//          $patt_email = "/^[A-Za-z0-9\u4e00-\u9fa5]+@[a-zA-Z0-9_-]+(\.[a-
zA-Z0-9_-]+)+$/";

//          if (!(preg_match($patt_email, $worksheet->getCell('A' .
$i)->getValue())) { //如果不匹配邮箱
//              exit(json_encode(['code' => 1, 'msg' => "第 $i 行的邮箱格式错
误"])));
//          }

//          if ($worksheet->getCell('B' . $i)->getValue() == "") {
//              $fileclass->unlink_file($file); //删除文件
//              exit(json_encode(['code' => 1, 'msg' => "学号出现空行,请重新
上传"])));
//          }

//          if ($worksheet->getCell('C' . $i)->getValue() == "") {
//              $fileclass->unlink_file($file); //删除文件
//              exit(json_encode(['code' => 1, 'msg' => "姓名出现空行,请重新
上传"])));
//          }

//          if ($worksheet->getCell('D' . $i)->getValue() == "") {
//              $fileclass->unlink_file($file); //删除文件
//              exit(json_encode(['code' => 1, 'msg' => "性别出现空行,请重新
上传"])));

```

```
//      }

//      if ($worksheet->getCell('D' . $i)->getValue() != "男" &&
$worksheet->getCell('D' . $i)->getValue() != "女") {
//      exit(json_encode(['code' => 1, 'msg' => "第 $i 行的性别格式错
误]));
//      }

//      $data['stu'][$i - 3]['email'] = $worksheet->getCell('A' .
$i)->getValue();
//      $data['stu'][$i - 3]['stu_no'] = $worksheet->getCell('B' .
$i)->getValue();
//      $data['stu'][$i - 3]['stu_name'] = $worksheet->getCell('C' .
$i)->getValue();
//      $data['stu'][$i - 3]['gender'] = ($worksheet->getCell('D' .
$i)->getValue() == "男") ? "0" : "1";

//      // 密码随机生成 8 位
//      $characters = '1234567890abcdefghijklmnopqrstuvwxyz';
//      $data['stu'][$i - 3]['stu_pwd'] = '';
//      for ($j = 0; $j < 8; $j++) {
//          $index = rand(0, strlen($characters) - 1);
//          $data['stu'][$i - 3]['stu_pwd'] .= $characters[$index];
//      }
//      $data['stu'][$i - 3]['add_time'] = date('Y-m-d H:i:s', time());
//      $data['stu'][$i - 3]['last_time'] = date("Y-m-d H:i:s", time());
//      }

//      Db::table('stu')->insertAll($data['stu']);
// }
}
```

(5) 作业布置模块

```
<?php

namespace app\admin\controller;

use think\facade\Config;
use think\facade\Db;
use think\facade\Request;
use think\facade\Session;
use think\facade\View;
use file\File;
use file\Zip1;

/**
 * 后台页面类
 * 作业权限开放方法
 * index 作业开放主页
 * add 作业布置
 * edit 作业信息修改
 */
class Work extends Base
{
    // 主页
    public function index()
    {
        $class_id = (int)Request::get('class_id', '');
        if (!$class_id) {
```

```

        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }

    $data['class'] = Db::table('classes')->where('class_id',
$class_id)->find();

    $data['admin'] = $this->admin;
    $data['recent_date'] = date("Y-m-d H:i", time());
    $data['class']['title'] = '作业布置';

    $data['work'] = Db::table('work')->where('class_id',
$class_id)->order('work_id', 'desc')->select();

    View::engine()->layout('layout');
    return View::fetch('work/index', $data);
}

// 添加
public function add()
{
    if (Request::isPost()) {
        $class_id = (int)Request::post('class_id', '');
        $data['class_id'] = (int)Request::post('class_id', '');
        $data['work_id'] = (int)Request::post('work_id', '');
        $data['work_remarks'] = trim(Request::post('work_remarks', ''));
        $data['work_last_time'] = Request::post('work_last_time', '');
        $data['status'] = (int)Request::post('status', '0');

        if ($data['work_id'] == '' || $data['work_id'] <= 0) {
            exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
        }

        if ($data['work_remarks'] == '') {

```

```
        exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
    }
    if (mb_strlen($data['work_remarks']) > 150) {
        exit(json_encode(['code' => 1, 'msg' => '备注填写长度超出限制!
'], true));
    }
    if ($data['work_last_time'] == '') {
        exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
    }

    $res = Db::table('work')->where(['class_id' => $class_id, 'work_id'
=> $data['work_id']])>find();
    if ($res) {
        exit(json_encode(['code' => 1, 'msg' => '作业号已存在'], true));
    }

    $data['work_start_time'] = date('Y-m-d H:i:s', time());
    $res = Db::table('work')->insert($data);

    if ($res) {
        exit(json_encode(['code' => 0, 'msg' => '保存成功'], true));
    }
    exit(json_encode(['code' => 1, 'msg' => '保存失败'], true));
} else {
    $data['class_id'] = (int)Request::get('class_id', '');
    $data['exts'] = Db::table('exts')->where('status', '<',
2)->select()->toArray();

    // dump($data);
```

```

        return View::fetch('work/add', $data);
    }
}

// 修改
public function edit()
{
    if (Request::isPost()) {
        $id = (int)Request::post('id', '');
        $class_id = (int)Request::post('class_id', '');

        if (empty($id)) { //如果没有 get 参数, 则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }
        if (empty($class_id)) { //如果没有 get 参数, 则返回
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        // $data['work_id'] = (int)Request::post('work_id', ''); //不修改

        $data['work_remarks'] = trim(Request::post('work_remarks', ''));
        $data['work_last_time'] = Request::post('work_last_time', '');
        $data['status'] = (int)Request::post('status', '0');

        if ($data['work_remarks'] == '') {
            exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
        }
        if (mb_strlen($data['work_remarks']) > 150) {

```

作业名称

```
        exit(json_encode(['code' => 1, 'msg' => '备注填写长度超出限制!',
'], true));
    }
    if ($data['work_last_time'] == '') {
        exit(json_encode(['code' => 1, 'msg' => '必填项不能为空'],
true));
    }

    Db::table('work')->where(['class_id' => $class_id, 'id' =>
$id])->update($data);
    exit(json_encode(['code' => 0, 'msg' => '保存成功'], true));
} else {
    $id = (int)Request::get('id', '');
    $class_id = (int)Request::get('class_id', '');

    if (empty($id)) { //如果没有 get 参数, 则返回
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }
    if (empty($class_id)) { //如果没有 get 参数, 则返回
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }

    $data['work_id'] = (int)Request::get('work_id', '');
    if (empty($id)) {
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }

    $data['work'] = Db::table('work')->where(['id' =>
$id])->find(); //根据唯一 id 查到作业
```



```

        $data['works'] = Db::table('stu')->where('class_id',
$class_id)->select()->toArray();

        foreach ($data['works'] as $k => $v) {

            if (Db::table('is_work')->where(['stu_no' => $v['stu_no'],
'work_id' => $data['work_id']])->find()) { //如果学生已点击

                if (Db::table('is_work')->where(['stu_no' => $v['stu_no'],
'work_id' => $data['work_id'], 'is_true' => 1])->find()) { //如果学生点击作业且确认
上传

                    $data['works'][$k]['is_true'] = 1;

                    $data['works'][$k]['work_id'] = $data['work_id'];

                    $data['works'][$k]['save_time'] =
Db::table('is_work')->where(['stu_no' => $v['stu_no'], 'work_id' =>
$data['work_id'], 'is_true' => 1])->find()['last_time']; //提交作业时间

                    $score_sum = Db::table('score')->where(['stu_no' =>
$v['stu_no'], 'work_id' => $data['work_id']])->sum('score'); //获取的总分

                    $stu_count = Db::table('stu')->>count(); //学生总人数

                    $data['works'][$k]['all_score'] =
number_format($score_sum / $stu_count, 2);

                } else { //学生点击作业, 但未上传

                    $data['works'][$k]['is_true'] = 0;

                    $data['works'][$k]['work_id'] = $data['work_id'];

                    $data['works'][$k]['save_time'] = 0; //提交作业时间

                    $data['works'][$k]['all_score'] = 0; //分数为 0

                }

            } else { //如果学生未点击作业

                $data['works'][$k]['is_true'] = -1;

                $data['works'][$k]['work_id'] = $data['work_id'];

                $data['works'][$k]['save_time'] = 0; //提交作业时间

                $data['works'][$k]['all_score'] = 0; //分数为 0

            }

        }

    }

```

```
        return View::fetch('work/edit', $data);
    }
}

// 删除
public function del()
{
    if (Request::isPost()) {
        $id = (int)Request::post('id', ''); //删除单条作业
        $class_id = (int)Request::post('class_id', '');
        $work_id = Request::post('work_id', ''); //删除作业的 score、
is_work、works 记录

        $file = new file();

        if (file_exists("storage" . "/" . $class_id . '/stu_work' . '/' .
$work_id)) { //判断文件是否存在

            //删除单条数据

            $file->remove_dir("storage" . "/" . $class_id . '/stu_work' .
 '/' . $work_id, true); //清空作业目录

        }

        Db::table('work')->where('id', $id)->delete();

        Db::table('is_work')->where(['class_id' => $class_id, 'work_id' =>
$work_id])->delete();

        Db::table('works')->where(['class_id' => $class_id, 'work_id' =>
$work_id])->delete();

        Db::table('score')->where(['class_id' => $class_id, 'work_id' =>
$work_id])->delete();

        // 清空文件-----
```

```

        exit(json_encode(['code' => 0, 'msg' => '删除成功'], true));
    } else {
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }
}

public function detail()
{
    $class_id = Request::get('class_id', '');
    if (empty($class_id)) { //如果没有 get 参数, 则返回
        exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
    }
    $data['stu_no'] = (int)Request::get('stu_no', '');
    $data['work_id'] = (int)Request::get('work_id', '');
    $data['score'] = Db::table('score')->where(['class_id' => $class_id,
'work_id' => $data['work_id'], 'stu_no' => $data['stu_no']])>select();

    return View::fetch('work/detail', $data);
}

// 作业导出
public function get_zip()
{
    $class_id = (int)Request::get('class_id', '');

    $zip1 = new Zip1();
    $zip1->zip($class_id, 'stu_work');
}
}

```

## (6) 成绩管理模块

```
<?php
namespace app\admin\controller;

use think\facade\Config;
use think\facade\Db;
use think\facade\Request;
use think\facade\Session;
use think\facade\View;
use file\Zip1;
use file\File;
use PhpOffice\PhpSpreadsheet\IOFactory; //用于载入已有的 xlsx 文件
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx; //保存 xlsx 文件

/**
 * 成绩管理类
 * 学生信息管理
 */
class Score extends Base
{
    public function index()
    {
        $class_id = (int)Request::get('class_id', '');
        if (!$class_id) {
            exit(json_encode(['code' => 1, 'msg' => '非法请求'], true));
        }

        $data['class'] = Db::table('classes')->where('class_id',
$class_id)->find();
    }
}
```

```

        $data['admin'] = $this->admin;

        $data['class']['title'] = '成绩管理';

        $data['stu'] =
        Db::table('stu')->field('stu_no,stu_name')->where('class_id',
        $class_id)->order('stu_no', 'asc')->select()->toArray(); //获取所有的学生

        $data['work'] = Db::table('work')->where(['class_id' => $class_id,
        'status' => 0])->select()->toArray(); //获取所有的作业

        $data['is_work'] =
        Db::table('is_work')->field('stu_no,work_id,is_true')->where('class_id',
        $class_id)->select()->toArray();

        $data['scores'] =
        Db::table('score')->field('stu_no,class_id,work_id,sum(score) score_all')->where
        ('class_id', $class_id)->group('stu_no,class_id,work_id')->select()->toArray();

        $work_count = Db::table('work')->where('status', 0)->count(); //作业
        次数

        $data['work_count'] = Db::table('work')->where('status',
        0)->count(); //作业次数

        $stu_count = Db::table('stu')->count(); //学生人数

        foreach ($data['stu'] as $k => $v) { //work\is_work\stu\score 四表联查
            if ($data['work']) { //存在作业的情况下，输出作业

                $data['stu'][$k]['work'] = $data['work']; //每个学生都有自己的作
                业

                foreach ($data['work'] as $kk => $vv) {
                    $data['stu'][$k]['work'][$kk]['score_all'] = 0; //默认作业
                    成绩为 0
                }
            }
        }
    
```

```

        if ($data['is_work']) { //存在了 is_work 数据项

            foreach ($data['is_work'] as $kkk => $vvv) {

                if (

                    $data['stu'][$k]['stu_no']      ==      $data['is_work'][$kkk]['stu_no']      &&
                    $data['stu'][$k]['work'][$kk]['work_id'] == $data['is_work'][$kkk]['work_id']
                    && $data['is_work'][$kkk]['is_true'] == 1

                ) { //如果作业号、且学号匹配，则该学生有上传数据

                    foreach ($data['scores'] as $kkkk => $vvvv) {

                        if (

                            $data['stu'][$k]['stu_no']      ==      $data['scores'][$kkkk]['stu_no']      &&
                            $data['stu'][$k]['work'][$kk]['work_id'] == $data['scores'][$kkkk]['work_id']

                        ) {

                            $score_all =
                            number_format($data['scores'][$kkkk]['score_all'] / $stu_count, 2);

                            $data['stu'][$k]['work'][$kk]['score_al
                            l'] = $score_all;

                                }

                            }

                        }

                    }

                }

            } else { //没有作业的情况下，输出一个空数组

                $data['stu'][$k]['work'] = [];

            }

        }

        foreach ($data['stu'] as $k => $v) {

```

```
$data['stu'][$k]['score_all'] = 0; //总成绩平均分

foreach ($v['work'] as $kk => $vv) {
    $data['stu'][$k]['score_all'] =
number_format(($data['stu'][$k]['score_all'] + $vv['score_all']), 2);
}
}

$file = new file();

if (file_exists('storage' . '/' . $class_id . '/stu_data' . '/' .
$class_id . '_stu_data.xlsx')) {
    $file->unlink_file('storage' . '/' . $class_id . '/stu_data' .
 '/' . $class_id . '_stu_data.xlsx');
}

$spreadsheet = new Spreadsheet();
$worksheet = $spreadsheet->getActiveSheet();

$worksheet->getCell('A' . 1)->setvalue('学号');
$worksheet->getCell('B' . 1)->setvalue('姓名');
$worksheet->getCell('C' . 1)->setvalue('平时成绩');

for ($i = 0; $i < count($data['stu']); $i++) { //循环出所有的数据
    $worksheet->getCell(
'A' . ($i + 2))->setvalue($data['stu'][$i]['stu_no']);
    $worksheet->getCell(
'B' . ($i + 2))->setvalue($data['stu'][$i]['stu_name']);
    $worksheet->getCell(
'C' . ($i + 2))->setvalue(number_format($data['stu'][$i]['score_all'] /
$work_count, 2));
}

$writer = new Xlsx($spreadsheet);
```

```
        $writer->save('storage/' . $class_id . '/stu_score' . '/' . $class_id .
'_stu_score.xlsx');

        View::engine()->layout('layout');
        return View::fetch('score/index', $data);
    }

    // 成绩导出
    public function get_zip()
    {
        $class_id = (int)Request::get('class_id', '');

        $zip1 = new Zip1();
        $zip1->zip($class_id, 'stu_score');
    }
}
```

## 5.4.2 学生作业提交模块

学生作业上传系统模块，包含学生登陆模块、学生权限校验模块、作业上传模块、作业评分模块。



	controller	
	Account.php	--学生登陆模块
	Base.php	--学生权限校验模块
	Home.php	--主页
	UpWork.php	--作业上传模块
	DisWork.php	--作业评分模块

### (1) 学生登陆模块

```
<?php

namespace app\home\controller;

use app\BaseController;
use think\captcha\facade\Captcha;
use think\facade\Db;
use think\facade\Session;
use think\facade\Request;
use think\facade\Cookie;
use think\facade\Cache;

class Account extends BaseController
{
    // 登录界面
    public function index()
    {
        return view('/account/login');
    }
}
```

```
}

// 注册验证码
public function verify()
{
    return Captcha::create();
}

// 登录校验
public function do_login()
{
    $class_id = (int)Request::post('class_id', '');
    $stu_no = (int)Request::post('stu_no', '');
    $stu_pwd = Request::post('stu_pwd', '');
    $captcha = Request::post('captcha', '');

    // 非空验证
    if (!$class_id) exit(json_encode(['code' => 1, 'msg' => '班级代码不能为空'], true));
    if (!$stu_no) exit(json_encode(['code' => 1, 'msg' => '学号不能为空'], true));
    if (!$stu_pwd) exit(json_encode(['code' => 1, 'msg' => '密码不能为空'], true));
    if (!$captcha) exit(json_encode(['code' => 1, 'msg' => '验证码不能为空'], true));

    if (!captcha_check($captcha)) exit(json_encode(['code' => 1, 'msg' => '验证码错误'], true));

    //班级是否存在
    $class = Db::table('classes')->where('class_id', $class_id)->find();
```

```

        if (!$class) exit(json_encode(['code' => 1, 'msg' => '班级不存在'], true));

        // 用户名是否存在

        $user = Db::table('stu')->where(['class_id' => $class_id, 'stu_no' =>
        $stu_no])->find();

        if (!$user) exit(json_encode(['code' => 1, 'msg' => '学号不存在'], true));

        // 密码校验

        if ($user['stu_pwd'] != $stu_pwd) exit(json_encode(['code' => 1, 'msg' =>
        '密码错误'], true));

        // 通过登录

        $data['last_time'] = date('Y-m-d H:i:s', time());

        $res = Db::table('stu')->where(['class_id' => $class_id, 'stu_no' =>
        $user['stu_no']])->update($data); // 修改登录时间

        Session::set('stu', $user); //将登录信息保存到 session 中
        Session::delete('captcha'); //删除验证码 session

        $randstr_www = time() . rand(10, 99); // 存用户 id-userkey (随机字符)

        Cache::set(
        'www_' . $user['class_id'] . $user['stu_no'], $randstr_www
        ); //保存到服务器端

        Cookie::Set('randstr_www', $randstr_www); //保存到 cookie 前端

        // exit 会导致 session 失效

        echo json_encode(['code' => 0, 'msg' => '登录成功'], true);
    }
}

```

(2) 学生权限校验模块

```
<?php

namespace app\home\controller;

use app\BaseController;
use think\facade\Session;
use think\facade\Request;
use think\facade\Db;
use think\facade\Cookie;
use think\facade\Cache;

class Base extends BaseController
{
    public $class; //网站常量
    public $stu; //登入学生

    // 初始化：未登录的用户不允许进入
    protected function initialize()
    {
        $this->stu = Session::get('stu');

        $stu = Session::get('stu');
        if (!$stu) {
            if (Request::isAjax()) {
                exit(json_encode(['code' => 1, 'msg' => '你还未登录，请登录']));
            }
            exit('你还未登录，请登录');
        }
    }
}

<script>
```

```

                setTimeout(function(){window.parent.location.href="/www/login"},2
000)

        </script>

        ');

    }

        $this->class    =    Db::table('classes')->where('class_id',
$this->stu['class_id']->find());

        // 2.取出 cookie 中的 randstr, 做登录校验, 不允许多人登录同一账号
        $randstr_www = Cookie::get('randstr_www');
        $cache_randstr = Cache::get('www_' . $stu['class_id'] . $stu['stu_no']);
        if ($randstr_www != $cache_randstr) {
            Session::delete('stu');

            if (Request::isAjax()) {

                exit(json_encode(['code' => 1, 'msg' => '你的账号在其他地方登录, 请
重新登录']));

            }

            exit(' 你的账号在其他地方登录 , 请重新登录
<script>setTimeout(function(){window.parent.location.href="/www/login"},2000)</sc
ript>');

        }

    }

    // 退出登录
    public function exit()
    {
        Session::delete('stu');
    }
}

```

### (3) 作业上传模块

```
<?php
```

```
namespace app\home\controller;

use think\captcha\facade\Captcha;
use think\facade\Db;
use think\facade\Session;
use think\facade\Request;
use think\facade\View;

/**
 * 作业提交页面
 */
class DisWork extends Base
{
    public function index()
    {
        $data['class'] = $this->class;
        $data['stu'] = $this->stu;
        $data['class']['title'] = '作业展示';

        $data['work'] = Db::table('work')->where(['class_id' =>
        $data['class']['class_id'], 'status' => 0])->order('work_id',
        'desc')->select()->toArray();

        foreach ($data['work'] as $k => $v) {
            $data['work'][$k]['stu_num'] = Db::table('stu')->where(['class_id' =>
            $data['class']['class_id']])->count(); //根据学生表查询学生总数

            $data['work'][$k]['is_true_stu_num'] = Db::table('is_work')
            ->where(['class_id' => $data['class']['class_id'], 'work_id' =>
```

```

$data['work'][$k]['work_id'], 'is_true' => 1])
        ->count(); //根据 is_work 的上传记录确定已经上传作业的名单
    }
    // exit(print_r($data)); //测试代码

    View::engine()->layout('layout');
    return View::fetch('dis_work/index', $data);
}

public function display()
{
    $class = $this->class;
    $data['stu'] = $this->stu;
    $data['work_id'] = (int)Request::get('work_id', '');

    $data['works'] = Db::table('stu')->where(['class_id' =>
$class['class_id']])->select()->toArray();
    foreach ($data['works'] as $k => $v) {
        if (Db::table('is_work')->where(['class_id' =>
$class['class_id'], 'stu_no' => $v['stu_no'], 'work_id' =>
$data['work_id']])->find()) { //如果学生已点击
            if (Db::table('is_work')->where(['class_id' => $class['class_id'],
'stu_no' => $v['stu_no'], 'work_id' => $data['work_id'], 'is_true' => 1])->find())
            { //如果学生点击作业且确认上传
                $data['works'][$k]['is_true'] = 1;
                $data['works'][$k]['work_id'] = $data['work_id'];
                $data['works'][$k]['save_time'] =
Db::table('is_work')->where(['class_id' => $class['class_id'], 'stu_no' =>
$v['stu_no'], 'work_id' => $data['work_id'], 'is_true' =>
1])->find()['last_time']; //提交作业时间

                $data['works'][$k]['remark_count'] =

```

```

Db::table('score')->where(['class_id' => $class['class_id'], 'stu_no' =>
$v['stu_no'], 'work_id' => $data['work_id']])->count(); //评价人数为0

        $data['works'][$k]['remark_stu'] =
Db::table('score')->field('to_stu_no')->where(['class_id' => $class['class_id'],
'stu_no' => $v['stu_no'], 'work_id' => $data['work_id']])->select()->toArray(); //
当前评价状态

    } else { //学生点击作业, 但未上传

        $data['works'][$k]['is_true'] = 0;

        $data['works'][$k]['work_id'] = $data['work_id'];

        $data['works'][$k]['save_time'] = 0; //提交作业时间

        $data['works'][$k]['remark_count'] = 0; //评价人数为0

        $data['works'][$k]['remark_stu'] =
Db::table('score')->field('to_stu_no')->where(['class_id' => $class['class_id'],
'stu_no' => $v['stu_no'], 'work_id' => $data['work_id']])->select()->toArray(); //
当前评价状态

    }

} else { //如果学生未点击作业

    $data['works'][$k]['is_true'] = -1;

    $data['works'][$k]['work_id'] = $data['work_id'];

    $data['works'][$k]['save_time'] = 0; //提交作业时间

    $data['works'][$k]['remark_count'] = 0; //评价人数为0

    $data['works'][$k]['remark_stu'] = []; //当前评价状态未知

}

}

return View::fetch('dis_work/display', $data);

}

public function remarks()
{
    $class = $this->class;

    if (Request::isPost()) {

```



```
$stu = $this->stu;
$data['class_id'] = $class['class_id'];
$data['work_id'] = (int)Request::post('work_id', '');
$data['stu_no'] = (int)Request::post('stu_no', ''); //被评分的学号
$data['to_stu_no'] = $stu['stu_no']; //参与评分的学号
$data['score'] = (int)Request::post('score', '');
$data['start_time'] = date("Y-m-d H:i:s", time()); //评分时间

$res = Db::table('score')
    ->where([
        'class_id' => $class['class_id'],
        'work_id' => $data['work_id'],
        'stu_no' => $data['stu_no'],
        'to_stu_no' => $data['to_stu_no']
    ])->find();

if ($res) { //如果已经评分过了，执行修改评分操作
    Db::table('score')->where([
        'class_id' => $class['class_id'],
        'work_id' => $data['work_id'],
        'stu_no' => $data['stu_no'],
        'to_stu_no' => $data['to_stu_no']
    ])->update($data);

    exit(json_encode(['code' => 0, 'msg' => "评分成功"]));
} else { //如果没有评分过，执行添加评分记录
    Db::table('score')->insert($data);

    exit(json_encode(['code' => 0, 'msg' => "评分成功"]));
}
```

```
    }  
    exit(json_encode(['code' => 1, 'msg' => "评分失败"]));  
} else {  
    // 第一次请求页面  
    $data['class_id'] = $class['class_id'];  
    $data['stu'] = $this->stu;  
    $data['work_id'] = Request::get('work_id', '');  
    $data['stu_no'] = Request::get('stu_no', ''); //作业归宿者  
    $data['to_stu_no'] = $data['stu']['stu_no']; //作业评价者  
    $data['score'] = 30;  
    $data['is_true'] = false; //提交状态, 默认为 false  
  
    $data['remarks'] = Db::table('score')->where([  
        'class_id' => $class['class_id'],  
        'work_id' => $data['work_id'],  
        'stu_no' => $data['stu_no'],  
    ])->order('score', "desc")->select();  
  
    $res = Db::table('score')  
        ->where([  
            'class_id' => $class['class_id'],  
            'work_id' => $data['work_id'],  
            'stu_no' => $data['stu_no'],  
            'to_stu_no' => $data['to_stu_no']  
        ])->find();  
  
    if ($res) {  
        $data['is_true'] = true; //已经提交过了  
        $data['score'] = $res['score'];  
    }  
}
```

```
    }  
    // print_r($res);  
    return View::fetch('dis_work/remarks', $data);  
  }  
}  
}
```

#### (4) 作业评分模块

```
<?php  
  
namespace app\home\controller;  
  
use think\facade\Db;  
use think\facade\Session;  
use think\facade\Request;  
use think\facade\View;  
use think\facade\Filesystem;  
use file\File;  
  
/**  
 * 作业上传  
 *  
 *  
 */  
class UpWork extends Base  
{  
    public function index()  
    {  
        $data['class'] = $this->class;  
    }  
}
```

```

$data['stu'] = $this->stu;

$data['recent_date'] = date("Y-m-d H:i", time());

$data['class']['title'] = '作业上传';

        $data['work'] = Db::table('work')->where(['class_id' =>
$data['class']['class_id'], 'status' => 0])->order('work_id',
'desc')->select()->toArray();

    foreach ($data['work'] as $k => $v) {
        if (Db::table('is_work')
            ->where(['class_id' => $data['class']['class_id'], 'stu_no' =>
$data['stu']['stu_no'], 'work_id' => $v['work_id']])
            ->find()
        ) { //可以查到对应的数据，学生已确认
            if (Db::table('is_work')
                ->where(['class_id' => $data['class']['class_id'], 'stu_no' =>
$data['stu']['stu_no'], 'work_id' => $v['work_id'], 'is_true' => 1])
                ->find()
            ) { //学生已提交
                $data['work'][$k]['stu_no'] = $data['stu']['stu_no'];
                $data['work'][$k]['is_true'] = 1;
                $score_sum = Db::table('score')
                    ->where(['class_id' => $data['class']['class_id'],
'stu_no' => $data['stu']['stu_no'], 'work_id' => $v['work_id']])
                    ->sum('score'); //获取的总分
                $stu_count = Db::table('stu')->where(['class_id' =>
$data['class']['class_id']])->count(); //学生总人数

                $data['work'][$k]['all_score'] = number_format($score_sum /
$data['stu_count'], 2);
            } else {
                // 学生未提交

```

```
        $data['work'][$k]['stu_no'] = $data['stu']['stu_no'];
        $data['work'][$k]['is_true'] = 0;
    }
} else { //不能查到对应的数据, 学生未确认
    $data['work'][$k]['stu_no'] = $data['stu']['stu_no'];
    $data['work'][$k]['is_true'] = -1;
}
}

View::engine()->layout('layout');
return View::fetch('up_work/index', $data);
}
```

// 主页面

```
public function details()
```

```
{
```

```
    $class = $this->class;
```

```
    $data['stu'] = $this->stu;
```

```
    $data['class_id'] = $class['class_id'];
```

```
    $data['stu_no'] = (int)Request::get('stu_no', ''); //当前作业号
```

```
    $data['work_id'] = (int)Request::get('work_id', ''); //当前学号
```

```
    $data['recent_date'] = date("Y-m-d H:i", time()); //当前的时间, 与作业时间
```

对比

```
        $data['work'] = Db::table('work')->where(['class_id' =>
$class['class_id'], 'work_id' => $data['work_id']])->find(); //获取当前作业的信息,
包括截止时间和作业的状态, 确保作业在截止时间结束后或状态关闭时, 是无法上传作业的!
```

```
if ($data['work_id'] == "" || $data['stu_no'] == "") { //不允许非法进入
```

```
    exit(json_encode(['code' => 1, "msg" => "非法请求"]));
```

```

    }

    $res = Db::table('is_work')->where(['class_id' => $class['class_id'],
'stu_no' => $data['stu_no'], 'work_id' => $data['work_id']])->find();

    if (!$res) {
        //如果没有查询到对应的 is_work 的数据, 则添加一条数据

        Db::table('is_work')
            ->insert([
                'class_id' => $class['class_id'],
                'work_id' => $data['work_id'],
                'stu_no' => $data['stu_no'],
                'is_true' => '0',
                'last_time' => date("Y-m-d H:i", time())
            ]);
    }

    $data['works'] = Db::table('works') //获取已上传的文件信息
        ->where(['class_id' => $class['class_id'], 'stu_no' =>
$data['stu_no'], 'work_id' => $data['work_id']])
        ->select();

    $data['is_work'] = Db::table('is_work')->where(['class_id' =>
$class['class_id'], 'stu_no' => $data['stu_no'], 'work_id' =>
$data['work_id']])->find();

    return View::fetch('up_work/details', $data);
}

//确认上传作业

public function is_true()

```

```

{
    $class = $this->class;

    $data['class_id'] = $class['class_id'];
    $data['stu_no'] = (int)Request::post('stu_no', '');
    $data['work_id'] = (int)Request::post('work_id', '');
    $data['is_true'] = (int)Request::post('is_true', '0');
    $data['last_time'] = date("Y-m-d H:i:s", time());

    $res = Db::table('work')->where(['class_id' => $class['class_id'],
'work_id' => $data['work_id']])->find();

    if ($res['status'] == 1) {
        exit(json_encode(['code' => 1, 'msg' => "作业已关闭"]));
    }

    // 如果确认提交, 要确保提交文件中至少有一个 index 文件
    if ($data['is_true'] == 1) {
        $res = Db::table('works')
            ->where(
                ['class_id' => $class['class_id'], 'stu_no' => $data['stu_no'],
'work_id' => $data['work_id']],
            )
            ->where('filename', 'like', 'index.' . '%')
            ->find();

        if (!$res) {
            exit(json_encode(['code' => 1, 'msg' => "请上传 index 文件"]));
        }
    }

    if (Db::table('is_work')->where(['class_id' => $class['class_id'],
'stu_no' => $data['stu_no']])->find() == "" ||

```

```

Db::table('is_work')->where(['class_id' => $class['class_id'], 'work_id' =>
$data['work_id']])->find() == "") {
    // 第一次上传, 添加一条数据
    Db::table('is_work')->insert($data);
    exit(json_encode(['code' => 0, 'msg' => "保存成功"]));
} else {
    // 执行修改操作
    Db::table('is_work')->where(['class_id' => $class['class_id'],
'stu_no' => $data['stu_no'], 'work_id' => $data['work_id']])->update($data);
    $data['is_true'] == 0 ? exit(json_encode(['code' => 0, 'msg' => "取消
成功"])) : exit(json_encode(['code' => 0, 'msg' => "保存成功"]));
}
}

// 上传文件
public function upfile()
{
    $class = $this->class;

    $file = Request::file('file');
    $data['class_id'] = $class['class_id'];
    $data['stu_no'] = Request::post('stu_no', '');
    $data['work_id'] = Request::post('work_id', '');
    $data['filename'] = Request::file('file')->getOriginalName();

    $res = Db::table('work')->where(['class_id' => $class['class_id'],
'work_id' => $data['work_id']])->find();

    if ($res['status'] == 1) {
        exit(json_encode(['code' => 1, 'msg' => "作业已关闭"]));
    }

    $count = Db::table('works')->where(['class_id' => $class['class_id'],
'work_id' => $data['work_id'], 'stu_no' => $data['stu_no']])->count(); //当前文件

```



个数

```
if ($count > 15) {
    exit(json_encode(['code' => 1, 'msg' => "最多上传 15 个文件"]));
}

if ($data['stu_no'] == "") {
    exit(json_encode(['code' => 1, 'msg' => "非法请求"]));
}

if ($data['work_id'] == "") {
    exit(json_encode(['code' => 1, 'msg' => "非法请求"]));
}

$data['work_url'] = "storage" . "/" . $class['class_id'] . "/stu_work" .
"/" . $data['work_id'] . "/" . $data['stu_no'] . "/" . $data['filename'];

$data['start_time'] = date("Y-m-d H:i:s", time());

$file_ext = $file->extension();

$exts = ['php', 'sql', 'html', 'css', 'js', 'jpg', 'png', 'gif', 'bmp',
'jpeg', 'svg', 'webp', 'ico', 'pdf'];

// 校验文件格式

if (in_array($file_ext, $exts) != 1) {
    exit(json_encode(['code' => 1, 'msg' => "暂不支持该格式文件上传"]));
}

if (Db::table('works')->where(['class_id' => $class['class_id'],
'filename' => $data['filename'], 'stu_no' => $data['stu_no'], 'work_id' =>
$data['work_id']])>find()) {
    // 修改操作

    $res = Db::table('works')->where(['class_id' => $class['class_id'],
'filename' => $data['filename'], 'stu_no' => $data['stu_no'], 'work_id' =>
$data['work_id']])>update($data);
}
```

```

                                $savename          =
\think\facade\Filesystem::disk('public')->putFileAs($class['class_id']      .
"/stu_work" . "/" . $data['work_id'] . '/' . $data['stu_no'], $file,
$data['filename']);

        exit(json_encode(['code' => 0, 'msg' => "保存成功, 请确认提交", "data"
=> "']));

    } else { //添加操作

        $res = Db::table('works')->insert($data);

        // 上传到本地服务器

                                $savename          =
\think\facade\Filesystem::disk('public')->putFileAs($class['class_id']      .
"/stu_work" . "/" . $data['work_id'] . '/' . $data['stu_no'], $file,
$data['filename']);

        if ($savename) {

            exit(json_encode(['code' => 0, 'msg' => "保存成功, 请确认提交",
"data" => "']));

        }

    }

    exit(json_encode(['code' => 1, 'msg' => "上传失败"]));

}

// 删除文件
public function delfile()
{
    $class = $this->class;

    $id = (int)Request::post('id', '');

    if ($id) {

        $res = Db::table('works')->where(['class_id' => $class['class_id'],
'id' => $id])->find();
    }
}

```

```
        $file = new file();
        $file->unlink_file($res['work_url']);

        Db::table('works')->where(['class_id' => $class['class_id'], 'id' =>
$id])->delete();
        exit(json_encode(['code' => 0, 'msg' => "删除成功"]));
    }
    exit(json_encode(['code' => 1, 'msg' => "非法请求"]));
}
}
```

## 第六章 结论与展望

基于 ThinkPHP 框架的智慧作业管理系统，主要完成了教师端和学生端的交互，在本系统中教师可以通过系统完成作业布置、作业汇总和分析、作业批量下载、作业提交设置和课程相关知识点发布、导出成绩等功能，学生可以通过系统可以完成作业提交、历史作业下载和查看课程相关知识点等功能，系统比可以使电子作业的管理更加专业，更流程化。

在充分分析用户需求后，设计实现了系统功能：登录校验、班级信息管理、学生信息管理、作业管理、作业上传、作业评分、成绩自动计算等功能模块。课题的主要工作如下：

1. 分析信息化高速发展下，教师和学生对于信息化平台管理的需求，针对师生互动需求提出系统开发的目的。

2. 根据业务需求对系统做了可行性分析并研究学习了系统开发的相关技术。如 B/S 构架思想、MVC 设计模型思想、PHP 开发语言、ThinkPHP 框架开发特性、HTML+CSS3+JS 前端技术、jQuery 框架、Ajax 技术、WAMP 开发环境搭建等 Web 开发相关技术。

3. 对系统进行需求分析，规划整理设计目标，对系统进行了概要设计，包括技术架构，应用功能设计、数据库概要设计。根据功能设计采用树状图，对系统每一个功能模块进行了详细设计，采用 E-R 图对数据库进行详细设计并给出对应的数据表。最后对系统界面实现和核心代码实现。

在互联网新技术不断发展的今天，信息化智能化对人们的影响越来越大，给师生互动提出新需求，教师拥有信息化平台，更加快捷的获取课程方面的信息，学生通过交互式的信息化网页，提交课程需要完成的任务，增强师生、学生之间的互动。课题设计实现了基于 ThinkPHP 框架的智慧作业管理系统，但是由于时间和编程水平限制，部分模块设计不够细致，学生与教师的互动弹性设计不够开放。因此，希望未来能够有广阔的视野，更加深刻的剖析师生互动的可扩展性。