

福州大学

数据库应用实践报告

小组组长： 梁达毅

小组成员： 林浩然、林炜、宋家锐、吕庆炜

专业班级： 2018 级软件工程 3 班

指导教师： 邓新国

二〇二〇年 8 月 23 日

目录

1 使用说明文档	
1.1 数据库源的建立.....	1
1.2 应用程序使用.....	1
1.3 开发环境简介.....	1
2 需求分析	
2.1 项目功能说明.....	1
2.2 系统流程分析.....	2
3 概念结构设计	
3.1 E-R 图.....	2
3.2 关系模型.....	3
3.3 详细设计.....	3
4 编程设计	
4.1 cookie 约定.....	8
4.2 函数约定.....	8
5 用户使用说明书	
5.1 用户登录模块.....	15
5.2 学生用户.....	17
5.3 教师用户.....	19
5.4 管理员用户.....	24
6 总结.....	26

1 使用说明文档

1.1 数据库源的建立

本设计提供了教师.xlsx、题库 样例.xlsx、学生表.xlsx 三个文件，用来导入数据库。

1.2 应用程序使用

Github:<https://github.com/ycrc123/ycrcSql.git>

初始密码：666

管理员：账号：admin 密码：666

学生：账号：学号 密码：666

教师：账号：教工号 密码：666

1.3 开发环境简介

操作系统：Windows 10、Linux

DBMS：mysql Ver 8.0.20 for Linux on x86_64 (Source distribution)

应用程序开发环境：IntelliJ IDEA Educational Edition 2019.3.1、Visual Studio Code

数据源：Microsoft Office Excel

Java版本：JDK1.8.0_251

2 需求分析

2.1 项目功能说明

项目共有三个角色：管理员、教师、学生。其中管理员可以增、删、改教师账号、重置学生密码、导入题库和开始正式考试。教师可以按照班级从excel文件自动导入学生账号，可以对学生账号增删改查。

本系统有模拟考试和正式考试功能，考试时间60分钟。教师可以管理自己班级的模拟考试时间，可以提前设置考试日期和时间；管理员可以开启正式考试，可以提前设置考试日期和时间。

考试系统的试卷由题库随机生成，有判断题、单选题和多选题3种题型。题库表可以由管理员上传，支持覆盖军事理论4大章节。模拟考试完成后可以显示学生打错的选项和标准答案，辅助学生提高成绩。考完过后会自动判卷，将学生成绩传到对应老师的页面。

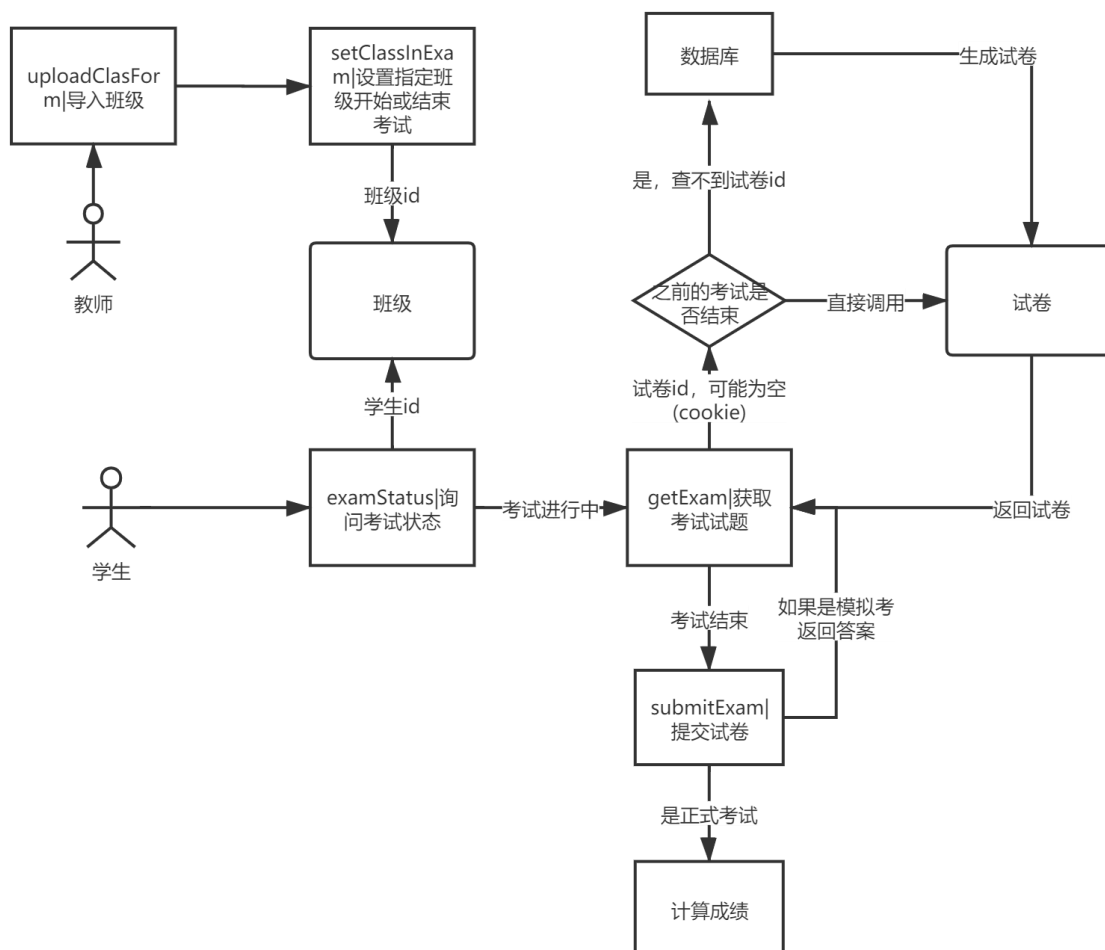
考试到一个小时未点提交，系统将会自动提交试卷。提交试卷后，教师可以立马看到对应学生的成绩。

在教师页面可以提前设置模拟考试的时间周期，自动开启考试自动关闭考试，管理员可以设置正式考试的考试的时间周期，自动开启自动关闭考试。

开始考试前需要学生点击勾选《福州大学考试守则》，才能进入考试。系统会保存学生的考试时间和已达选项，避免学生勿关考试页面导致考试剩余时间清零。用户右上角的个人中心可以显示学生学号和姓名，以便监考老师检查核实。

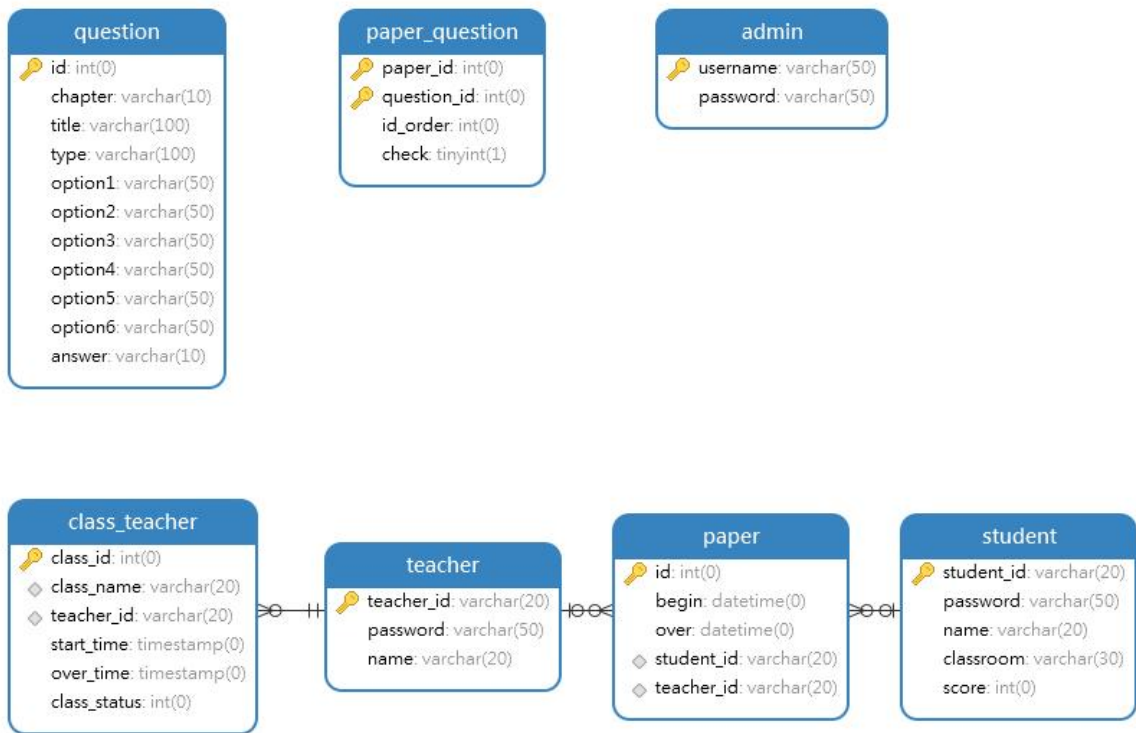
系统采用前后端分离的方式，前端采用nginx部署，通过反向代理实现负载均衡。

2.2 系统流程分析



3 概念结构设计

3.1 E-R图



3.2 关系模型

管理员 (账号, 密码)

学生 (学号, 姓名, 密码, 班级名, 得分)

教师 (教师号, 姓名, 密码)

班级教师表 (班级号, 班级名, 教师号, 考试开始时间, 考试结束时间, 班级考试状况)

试卷 (试卷编号, 开始时间, 结束时间, 学生学号, 教师号)

题目 (题目编号, 题目章节, 题干, 题目类型, 选项信息1, 选项信息2, 选项信息3, 选项信息4, 正确答案)

试卷_试题 (试卷编号, 题目编号, 学生答案, 结果正误判断)

3.3 详细设计

3.3.1 基本表

1. 管理员信息表(admin)

序号	字段名称	字段描述	类型	位数	属性
1	username	管理员用户名	varchar	20	主键、非空唯一
2	password	密码	varchar	50	非空

2. 教师表(teacher)

序号	字段名称	字段描述	类型	位数	属性
1	teacher_id	教师编号	varchar	20	主键、非空唯一
2	password	教师密码	varchar	50	非空
3	name	教师姓名	varchar	20	

3. 学生表(student)

序号	字段名称	字段描述	类型	位数	属性
1	student_id	学生编号	varchar	20	主键、非空唯一
2	password	密码	varchar	50	非空
3	name	学生姓名	varchar	20	
4	classroom	学生所在班级	varchar	30	
5	score	学生最终成绩	int		

4. 试卷表(paper)

序号	字段名称	字段描述	类型	位数	属性
1	id	试卷编号	int		主键、非空唯一、自增
2	student_id	学生编号	varchar	20	外键
3	teacher_id	教师编号	varchar	20	外键
4	begin	考试开始时间	datetime		
5	over	考试结束时间	datetime		
6	score	得分	int		

5. 题目表(question)

序号	字段名称	字段描述	类型	位数	属性
1	id	题目编号	int		主键、非空唯一、自增
2	chapter	题目章节	varchar	10	
3	title	题目题干	varchar	100	
4	type	题目类型	varchar	100	
5	option1	选项1	varchar	50	
6	option2	选项2	varchar	50	
7	option3	选项3	varchar	50	
8	option4	选项4	varchar	50	
9	answer	标准答案	varchar	10	

6. 试卷题目表(paper_question)

序号	字段名称	字段描述	类型	位数	属性
1	paper_id	试卷编号	int		主键
2	question_id	题目编号	int		主键
3	student_answer	学生答案	int		
4	check	答案是否正确	tinyint	1	

7. 班级教师表(class_teacher)

序号	字段名称	字段描述	类型	位数	属性
1	class_id	班级编号	int		主键、非空唯一、自增
2	class_name	班级名称	varchar	20	非空、唯一
3	teacher_id	老师编号	varchar	20	外键
4	start_time	考试开始时间	timestamp		
5	over_time	考试结束时间	timestamp		
6	class_status	班级考试状况	int		

3.3.2 基本表的建立

1、管理员信息表

```

1 CREATE TABLE `admin` (
2   `username` VARCHAR(50) NOT NULL,
3   `password` VARCHAR(50) DEFAULT NULL,
4   PRIMARY KEY (`username`),
5   UNIQUE KEY `admin_username_uindex` (`username`)
6 ) ENGINE=INNODB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

2、教师表

```

1 CREATE TABLE `teacher` (
2   `teacher_id` VARCHAR(20) NOT NULL,
3   `password` VARCHAR(50) DEFAULT NULL,
4   `name` VARCHAR(20) DEFAULT NULL,
5   PRIMARY KEY (`teacher_id`),
6   UNIQUE KEY `teacher_teacher_id_uindex` (`teacher_id`)
7 ) ENGINE=INNODB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

3、学生表

```

1 CREATE TABLE `student` (
2   `student_id` VARCHAR(20) NOT NULL,
3   `password` VARCHAR(50) DEFAULT NULL,
4   `name` VARCHAR(20) DEFAULT NULL,
5   `classroom` VARCHAR(30) DEFAULT NULL,
6   `score` INT DEFAULT NULL,
7   PRIMARY KEY (`student_id`),
8   UNIQUE KEY `student_student_id_uindex` (`student_id`)
9 ) ENGINE=INNODB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

4、试卷表

```

1 CREATE TABLE `paper` (
2   `id` INT NOT NULL,
3   `begin` DATETIME DEFAULT NULL,
4   `over` DATETIME DEFAULT NULL,
5   `student_id` VARCHAR(20) DEFAULT NULL,
6   `teacher_id` VARCHAR(20) DEFAULT NULL,
7   PRIMARY KEY (`id`),
8   UNIQUE KEY `paper_id_uindex` (`id`),
9   KEY `paper_student_student_id_fk` (`student_id`),
10  KEY `paper_teacher_teacher_id_fk` (`teacher_id`),
11  CONSTRAINT `paper_student_student_id_fk` FOREIGN KEY (`student_id`)
12  REFERENCES `student` (`student_id`) ON DELETE CASCADE ON UPDATE CASCADE,
13  CONSTRAINT `paper_teacher_teacher_id_fk` FOREIGN KEY (`teacher_id`)
14  REFERENCES `teacher` (`teacher_id`) ON DELETE CASCADE ON UPDATE CASCADE
15 ) ENGINE=INNODB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

5、题目表


```

1 CREATE TABLE `question` (
2   `id` INT NOT NULL AUTO_INCREMENT,
3   `chapter` VARCHAR(10) DEFAULT NULL,
4   `title` VARCHAR(100) DEFAULT NULL,
5   `type` VARCHAR(100) DEFAULT NULL,
6   `option1` VARCHAR(50) DEFAULT NULL,
7   `option2` VARCHAR(50) DEFAULT NULL,
8   `option3` VARCHAR(50) DEFAULT NULL,
9   `option4` VARCHAR(50) DEFAULT NULL,
10  `option5` VARCHAR(50) DEFAULT NULL,
11  `option6` VARCHAR(50) DEFAULT NULL,
12  `answer` VARCHAR(10) DEFAULT NULL,
13  PRIMARY KEY (`id`),
14  UNIQUE KEY `question_id_uindex` (`id`)
15 ) ENGINE=INNODB AUTO_INCREMENT=961 DEFAULT CHARSET=utf8mb4
  COLLATE=utf8mb4_0900_ai_ci;

```

6、试卷题目表

```

1 CREATE TABLE `paper_question` (
2   `paper_id` INT NOT NULL,
3   `question_id` INT NOT NULL,
4   `id_order` INT DEFAULT NULL,
5   `check` TINYINT(1) DEFAULT NULL,
6   PRIMARY KEY (`paper_id`, `question_id`)
7 ) ENGINE=INNODB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

7、班级教师表

```

1 CREATE TABLE `class_teacher` (
2   `class_id` INT NOT NULL AUTO_INCREMENT,
3   `class_name` VARCHAR(20) NOT NULL,
4   `teacher_id` VARCHAR(20) NOT NULL,
5   `start_time` TIMESTAMP NULL DEFAULT NULL,
6   `over_time` TIMESTAMP NULL DEFAULT NULL,
7   `class_status` INT DEFAULT NULL,
8   PRIMARY KEY (`class_id`),
9   UNIQUE KEY `class_teacher_class_uindex` (`class_name`),
10  UNIQUE KEY `class_teacher_class_id_uindex` (`class_id`),
11  KEY `class_teacher_fk` (`teacher_id`),
12  CONSTRAINT `class_teacher_fk` FOREIGN KEY (`teacher_id`) REFERENCES
  `teacher` (`teacher_id`) ON DELETE CASCADE ON UPDATE CASCADE
13 ) ENGINE=INNODB AUTO_INCREMENT=20 DEFAULT CHARSET=utf8mb4
  COLLATE=utf8mb4_0900_ai_ci;

```

4 编程设计

- 后端在收到操作时要对操作用户的权限进行判断（过滤器）

- 对于没登录的用户转跳到登录界面
 - 对于权限不够的用户转跳到权限不足页面。
- 第一次登录时必须强制修改密码，不允许其他操作，在用户操作其他方法时强制下线(删除userId,username,character,注销userId) (过滤器)
- 考试分为模拟考和真实考试
 - 模拟考在:一段时间内都可以考,考试时间按照考生点击进入考试的时间计算，并且在考试结束后直接返回做题情况(细节待补充)。
 - 正式考试:考试开始时间在管理员按下考试开始时统一开始计算，考试结束后返回提交成功页面。
- 要调用的函数路径为 `./函数名`

4.1 cookie约定

- userId:用户登录后获取的用户id
- username:用户登录后获取的用户的教师号/学号/admin
- character:用户角色student/teacher/admin

4.2 函数约定

login

- 参数

```

1  {
2      "username": "用户名",
3      "password": "md5加密后的用户密码"
4  }
```

- username: 代表用户名
- password: 代表md5加密后的用户密码

- 返回值

```

1  {
2      "status": 0|1|2|3
3  }
```

- status: 代表登录状态，0代表登录失败，1代表登录成功，2代表第一次登录未修改密码，3代表重复登录

- 功能

- 用户登入
- 设置userId，每次用户登录后获得的userId都是随机的
- 设置username
- 设置character
- 依据用户组转跳页面
- 对于第一次登录的用户，其只能进行修改密码的操作，其他的操作可认为其未登录

logout

- 参数

- void

- 返回值

- void

- 功能
 - 删除userId
 - 删除username
 - 删除character
 - 后端注销userId
 - 转跳到index.html

changePassword

- 参数

- ```
1 {
2 "newPassword": "md5加密后的新密码"
3 }
```

- newPassword: 代表用户md5加密后的新密码

- 返回值

- void

- 功能

- 设置新密码

## examStatus

- 参数

- ```
1 {
2     "examType": 0|1
3 }
```

- examType代表考试类型，0代表正式考试，1代表模拟考试

- 返回值

- ```
1 {
2 "examStart": 0|1
3 }
```

- examStart: 代表考试状态，0代表未开始，1代表开始

- 功能

- 根据考试类型，返回对应考试状态

## getPaper

- 参数

- void

- 返回值

- ```
1 {
2     "paperId": string,
3     "date": string,
4     "Qtype": [ number, number, number ],
5     "data": [
6         {
7             // 单选
8             "title": string,
```

```

9         "choice":[ string, string, string, string ]
10     },
11     ...
12     {
13         //多选
14         "title":string,
15         "choice":[ string, string, string, string ]
16     },
17     ...
18     {
19         //判断
20         "title":string
21     },
22     ...
23 ]
24 }

```

- examId:考卷的唯一id, 为随机值
- date: 代表考试剩余的时间,返回结束时的时间string "YYYY/MM/DD hh:mm:ss"
- Qtype: 分别代表单选, 多选, 判断的题目数量
- 不同类型的题目有不同格式, data里必须严格按照单选, 多选, 判断的顺序排列
- 功能
 - 返回考试试题

submitPaper

- 参数

```

1  [
2      [ number, number, number],
3      [ number, number, ...]
4  ]

```

- 第一个数组: 分别代表单选, 多选, 判断的题目数量
- 第二个数组: 代表考生选的选项, 范围为0-15, 使用二进制表示:
 - 0代表未选
 - 1代表A选项或正确
 - 2代表B选项或错误
 - 4代表C选项
 - 8代表D选项
 - 多选题则是选项的累加

- 返回

```

1  [
2      [ number, number, number],
3      [ number, number, ...]
4  ]

```

- 格式同上

- 功能

- 提交学生选项
- 如果是模拟考则返回答案, 正式考试返回null

- 一定要用服务器数据来判断是否是模拟考，不能用cookie

getClassInExam

- 参数

- void

- 返回

- ```
1 [
2 {
3 "className":string,
4 "classId":sting,
5 "classStatus":0|1|2,
6 "examTime":string
7 }
8 ...
9]
```

- className: 班级名称，如"软件工程三班"
- classId: 班级唯一id
- classStatus: 班级状态，是否在考试中，0代表不在考试中，1代表在模拟考试，2代表在正式考试
- examTime: 如果班级在考试，返回班级考试时间段，格式"YYYY/MM/DD HH:mm:ss ~ YYYY/MM/DD HH:mm:ss"

- 功能

- 查询对应教师账号的所有班级考试状态

### setClassInExam

- 参数

- ```
1  [
2      {
3          "className":string,
4          "classId":sting,
5          "classStatus":0|1|2,
6          "examTime":string
7      }
8      ...
9  ]
```

- className: 班级名称，如"软件工程三班"
- classId: 班级唯一id
- classStatus: 班级状态，是否在考试中，0代表不在考试中，1代表在模拟考试，2代表在正式考试
- examTime: 如果设置班级在考试，返回班级考试时间段，格式"YYYY/MM/DD HH:mm:ss ~ YYYY/MM/DD HH:mm:ss"

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功
- 功能
  - 设置指定班级的考试时间范围
  - 如果该班级考试已开放, 再次设置考试时间应当返回设置失败

### doImportStudentExcel

- 参数

- ```
1 //受到layui限制, 数据格式为FormData
2 "file":上传的文件
3 "className":上传的班级名称
```

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功
- 对于参数不够返回失败

- 功能

- 上传班级学生名单, 为xls或xlsx

### delectClass

- 参数

- ```
1 [string,string,...]
```

- 一个包含删除班级classId的数组

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功

- 功能

- 删除指定班级

### getStudentInfo

- 参数

- ```
1 {
2     "classId":string,
3 }
```

- classId: 要获取的学生信息的班级id,"ALL"代表获取所有学生信息

- 返回

- ```
1 [
2 {
3 "studentName":string,
4 "studentNo":number,
5 "studentScore":number,
6 }
7]
```

- studentName:代表学生姓名
- studentNo: 代表学生学号
- studentScore: 代表学生成绩
- 功能
  - 获取指定班级的学生信息

### getTeacherList

- 参数
  - void
- 返回

- ```
1 [
2   {
3     "teacherId":string,
4     "teacherName":sting
5   }
6   ...
7 ]
```

- teacherId: 教师唯一id
- teacherName:教师姓名
- teacherStatus: 教师状态, false表示教师被删除
- 功能
 - 查询所有的教师账号

resetPassword

- 参数

- ```
1 [
2 String
3 ...
4]
```

- String:代表要重置密码的学生或教师的ID
- 返回
  - void
- 功能
  - 重置密码

### doImportQuestionExcel

- 参数

- ```
1 //受到layui限制，数据格式为FormData
2 "file":上传的文件
```

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功
- 对于参数不够返回失败

- 功能

- 上传题库文件

## setOfficialExam

- 参数

- ```
1 {
2     "examTime":string
3 }
4 ...
```

- examTime: 如果设置班级正式考试, 返回班级考试时间段, 格式"YYYY/MM/DD HH:mm:ss ~ YYYY/MM/DD HH:mm:ss"

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功

- 功能

- 设置正式考试

## deleteTeacher

- 参数

- ```
1 [
2     String
3 ]
```

- String:将要删除的教师ID

- 返回

- void

- 功能

- 删除教师

doImportTeacherExcel

- 参数

- - 1 //受到layui限制，数据格式为FormData
 - 2 "file":上传的文件

- 返回

- ```
1 {
2 "status":0|1
3 }
```

- status: 0设置失败, 1设置成功
- 对于参数不够返回失败

- 功能

- 上传教师文件

## getServerIP

- 参数

- void

- 返回

- ```
1 {
2     "IP":String,
3     "port":String
4 }
```

- 功能

- 返回当前服务器的IP和端口号

getClientIP

- 参数

- void

- 返回

- ```
1 {
2 "IP":String,
3 "port":String
4 }
```

- 功能

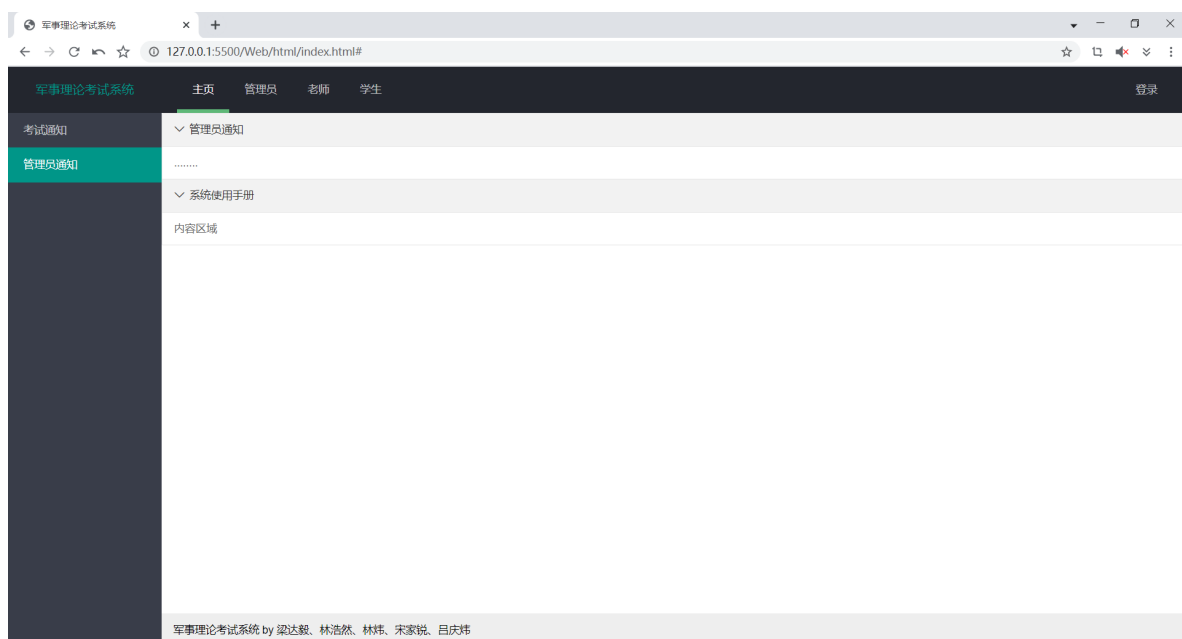
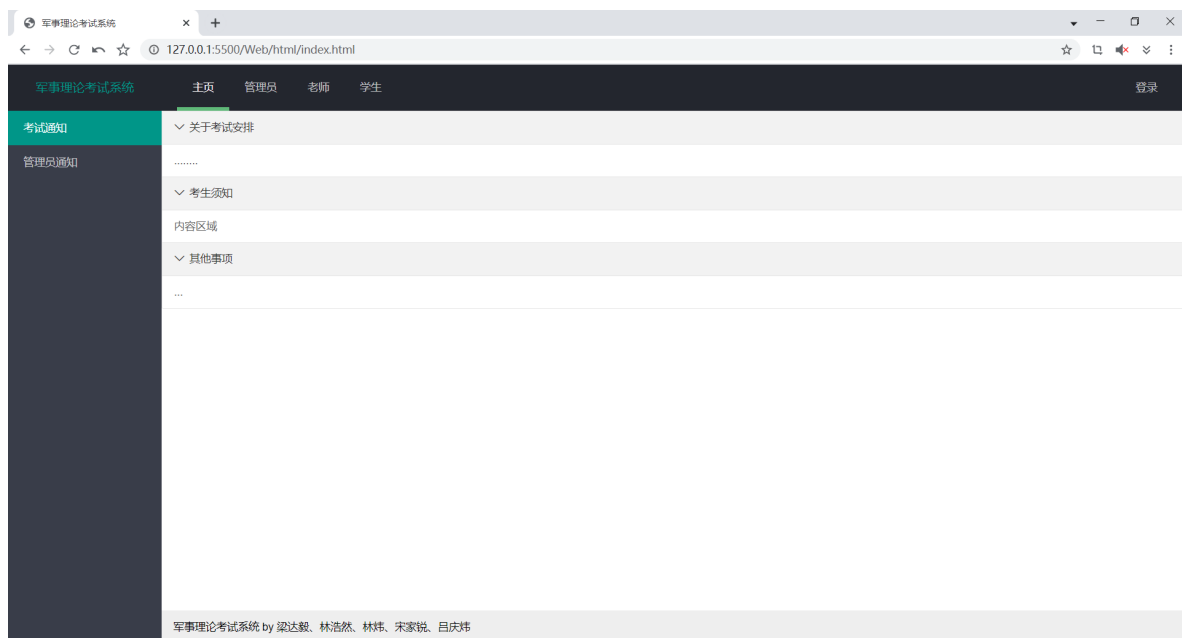
- 返回当前客户端的IP和端口号

# 5 用户使用说明书

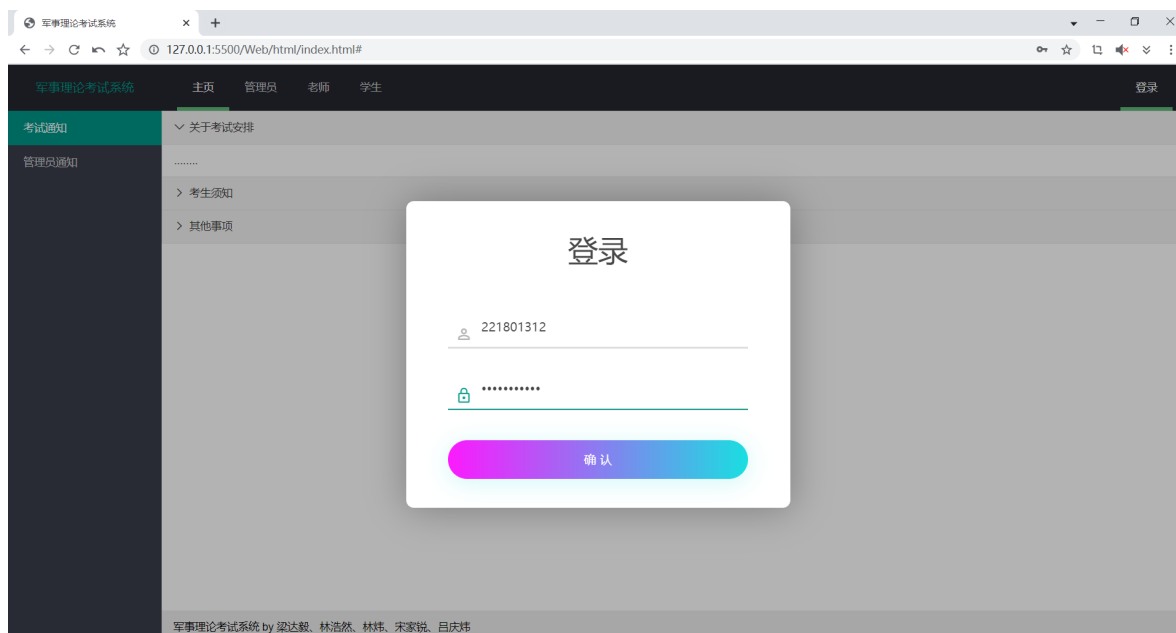
## 5.1 用户登录模块

用户登录模块可以实现对用户的合法性进行检查、通过用户登录账号对用户权限的判断，并通过不同权限的用户限制其能访问的页面。

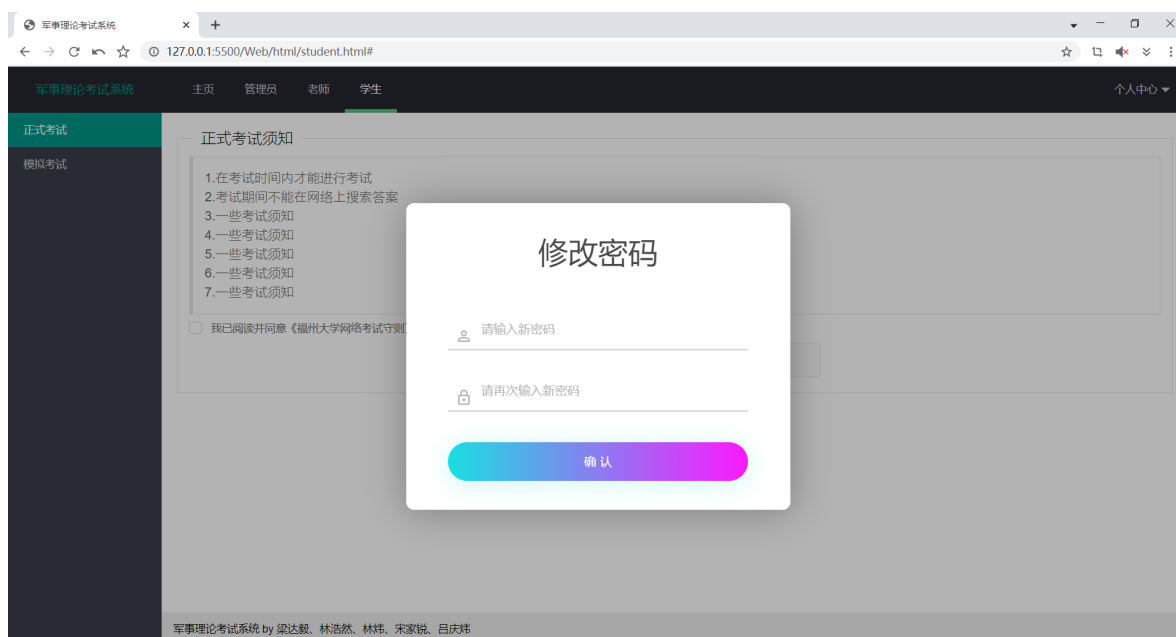
用户首次访问时主页如下，主页包括考试通知和管理员通知两个栏目，考试通知用于展示关于考试安排、考生须知、其他实现等，管理员通知用于展示管理员通知和系统使用手册。



用户可以通过右上角登录按钮进行登录，当用户未登录时点击管理员、老师或学生按钮会强制弹出登录界面。



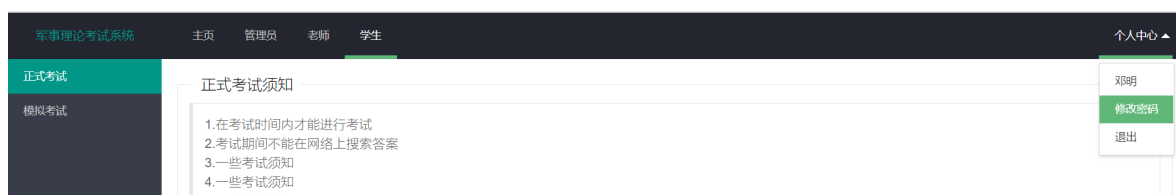
若用户输入的用户名和密码正确，将会跳转到登录用户对应权限的界面。当用户登录密码为初始密码时，会弹出修改密码窗口强制用户修改密码。若不正确则会弹出用户名或密码错误。



## 5.2 学生用户

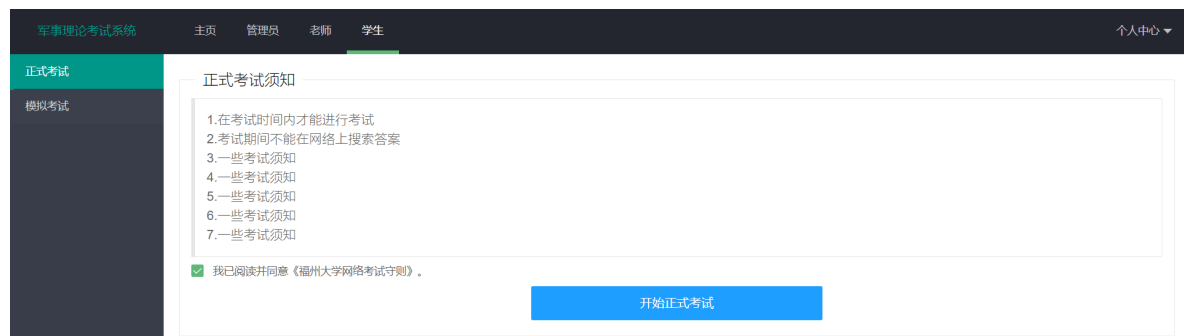
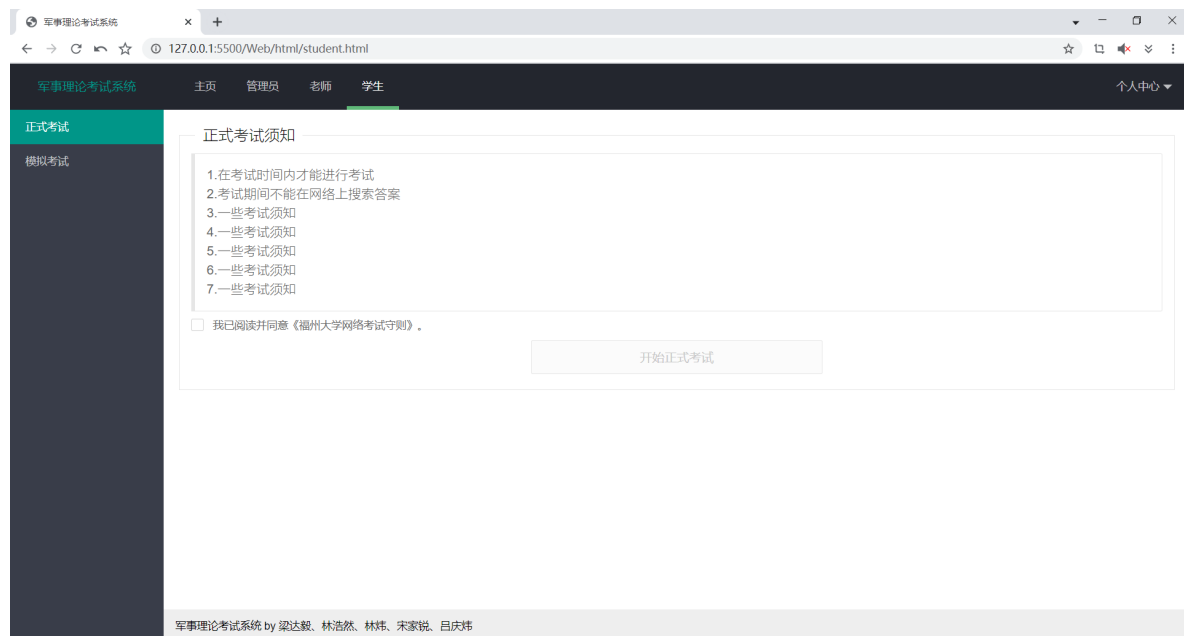
### 5.2.1 学生用户个人中心

本系统允许用户修改密码、和登出。修改密码按钮将会弹出修改密码窗口，登出按钮将会直接登出。



### 5.2.2 学生主页界面

学生主页允许学生参加正式考试和模拟考试。正式考试和模拟考试界面均有考试须知，并要求同学勾选《福州大学网络考试守则》才允许进入考试。



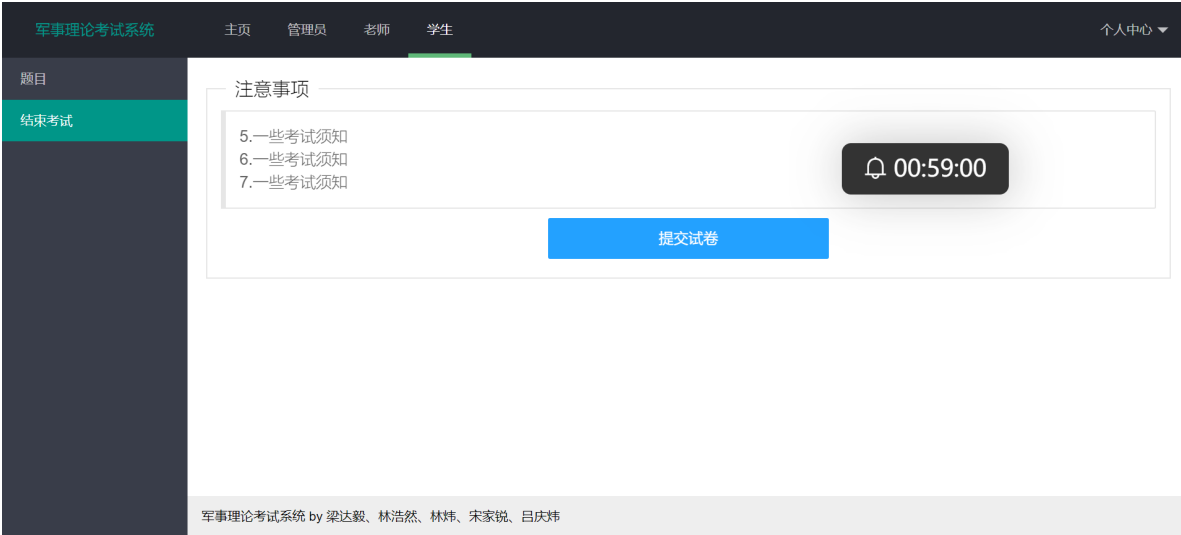
### 5.2.3 学生正式考试界面

正式考试界面支持显示考试剩余时间，学生可以通过右上角结束考试按钮提前结束考试。



### 5.2.4 学生模拟考试界面

模拟考试界面与正式考试界面大体相同，在学生提交试卷之后，可以看到模拟考试的答题情况。



## 5.3 教师用户

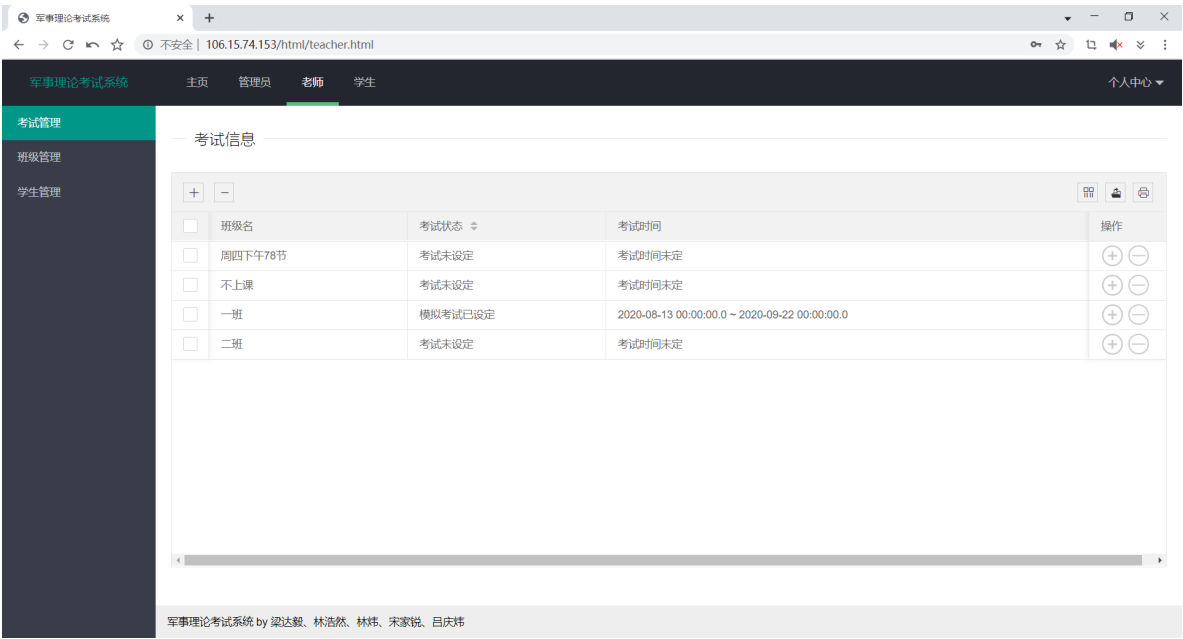
### 5.3.1 教师用户个人中心

本系统允许用户修改密码、和登出。修改密码按钮将会弹出修改密码窗口，登出按钮将会直接登出。

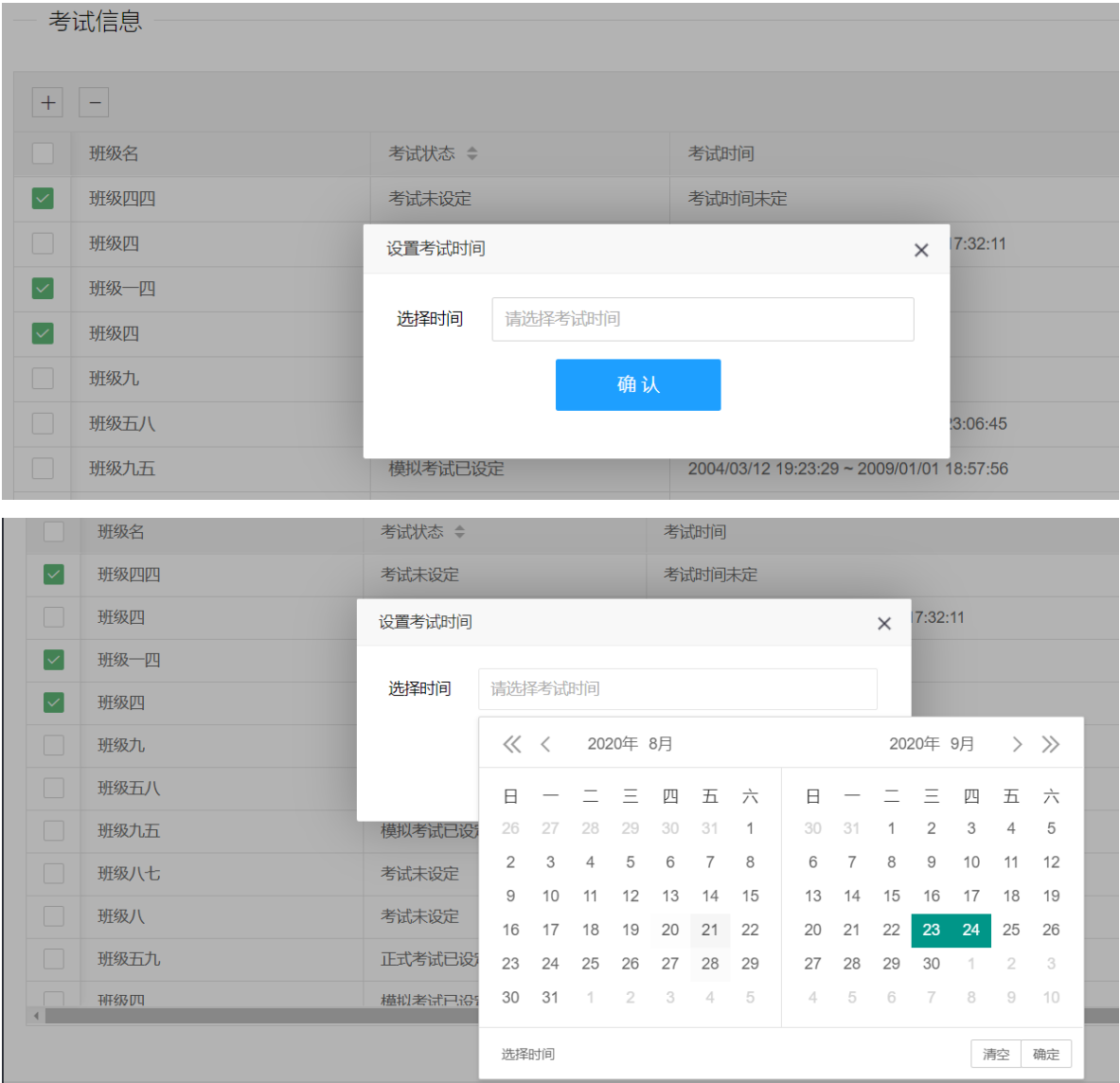


### 5.3.2 考试管理

考试管理界面如下。



教师可以通过“考试信息”的“+”选择班级的考试时间，若当前班级已处于考试状态，将不允许教师添加考试状态。教师可以通过多选班级批量设定考试时间。



设置考试时间

选择时间

请选择考试时间

开始时间

结束时间

时

分

秒

08

00

00

09

01

01

10

02

02

11

03

03

12

04

04

13

05

05

时

分

秒

09

00

00

10

01

01

11

02

02

12

03

03

13

04

04

14

05

05

返回日期

清空

确定

通过“考试信息”的“-”可以删除班级的模拟考试状态，将不允许教师删除正式考试的状态。

| +                                   | - | 班级名  | 考试状态    | 考试时间                                      | 操作                        |
|-------------------------------------|---|------|---------|-------------------------------------------|---------------------------|
| <input type="checkbox"/>            |   | 班级四四 | 考试未设定   | 考试时间未定                                    | <div>+</div> <div>-</div> |
| <input type="checkbox"/>            |   | 班级四  | 正式考试已设定 | 1987/10/18 05:36:15 ~ 1972/09/12 17:32:11 | <div>+</div> <div>-</div> |
| <input type="checkbox"/>            |   | 班级一四 | 考试未设定   |                                           | <div>+</div> <div>-</div> |
| <input type="checkbox"/>            |   | 班级四  | 考试未设定   |                                           | <div>+</div> <div>-</div> |
| <input type="checkbox"/>            |   | 班级九  | 考试未设定   |                                           | <div>+</div> <div>-</div> |
| <input checked="" type="checkbox"/> |   | 班级五八 | 模拟考试已设定 | 2:00 ~ 1986/07/10 23:06:45                | <div>+</div> <div>-</div> |
| <input checked="" type="checkbox"/> |   | 班级九五 | 模拟考试已设定 | 2004/03/12 19:23:29 ~ 2009/01/01 18:57:56 | <div>+</div> <div>-</div> |
| <input type="checkbox"/>            |   | 班级八七 | 考试未设定   | 考试时间未定                                    | <div>+</div> <div>-</div> |

班级表上方右侧工具栏可供教师筛选列、导出表到Excel、打印等操作。

考试信息

| +                        | - | 班级名 | 考试状态    | 考试时间                                      | 操作                        |
|--------------------------|---|-----|---------|-------------------------------------------|---------------------------|
| <input type="checkbox"/> |   | 班级二 | 考试未设定   | 考试时间未定                                    | <div>+</div> <div>-</div> |
| <input type="checkbox"/> |   | 班级十 | 模拟考试已设定 | 1977/09/06 20:18:49 ~ 1988/09/12 02:01:00 | <div>+</div> <div>-</div> |
| <input type="checkbox"/> |   | 班级七 | 模拟考试已设定 | 1999/03/27 03:49:18 ~ 1971/08/02 11:02:31 | <div>+</div> <div>-</div> |
| <input type="checkbox"/> |   | 班级一 | 模拟考试已设定 | 1984/08/24 20:13:38 ~ 1993/05/30 04:52:21 | <div>+</div> <div>-</div> |

在班级表处可查找班级名、考试状态、考试时间，其中班级名和考试状态允许排序以供快速查找。而右侧的“操作”工具栏可设置单个班级的考试状态或删除单个班级的考试状态。

| +                        | - | 班级名  | 考试状态  | 考试时间   | 操作                        |
|--------------------------|---|------|-------|--------|---------------------------|
| <input type="checkbox"/> |   | 班级七八 | 考试未设定 | 考试时间未定 | <div>+</div> <div>-</div> |
| <input type="checkbox"/> |   | 班级四  | 考试未设定 | 考试时间未定 | <div>+</div> <div>-</div> |
| <input type="checkbox"/> |   | 班级一  | 考试未设定 | 考试时间未定 | <div>+</div> <div>-</div> |

21

### 5.3.3 班级管理

班级管理页面如下，支持上传班级表、筛选列、导出到Excel、打印班级表、删除班级表等操作。

军事理论考试系统

127.0.0.1:5500/Web/html/teacher.html#

个人中心

考试管理

班级管理

学生管理

班级管理

班级信息

班级表

| 班级名  | 考试状态    | 考试时间                                      | 操作 |
|------|---------|-------------------------------------------|----|
| 班级八  | 正式考试已设定 | 1992/08/19 19:56:43 ~ 2006/12/22 05:06:47 |    |
| 班级五三 | 模拟考试已设定 | 2018/03/01 23:15:36 ~ 1998/01/16 03:06:58 |    |
| 班级七  | 模拟考试已设定 | 1984/12/22 01:52:09 ~ 1985/03/05 12:02:35 |    |
| 班级八七 | 考试未设定   | 考试时间未定                                    |    |
| 班级二  | 考试未设定   | 考试时间未定                                    |    |
| 班级九  | 模拟考试已设定 | 1997/12/28 15:41:04 ~ 1993/08/07 02:46:17 |    |
| 班级二  | 模拟考试已设定 | 2001/03/16 00:49:02 ~ 1990/12/06 00:20:26 |    |
| 班级四  | 模拟考试已设定 | 2010/04/20 22:32:43 ~ 1974/09/26 18:23:12 |    |
| 班级八三 | 正式考试已设定 | 2007/02/06 23:50:21 ~ 1985/01/26 22:26:40 |    |
| 班级五二 | 正式考试已设定 | 1981/05/12 19:20:56 ~ 1982/03/13 14:39:13 |    |
| 班级五  | 模拟考试已设定 | 2013/09/14 03:27:21 ~ 2017/04/25 12:18:23 |    |

军事理论考试系统 by 梁达毅、林浩然、林炜、宋家锐、吕庆伟

上传按钮仅支持上传.xls、.xlsx 两种文件。

上传班级学生文件

点击上传，或将文件拖拽到此处

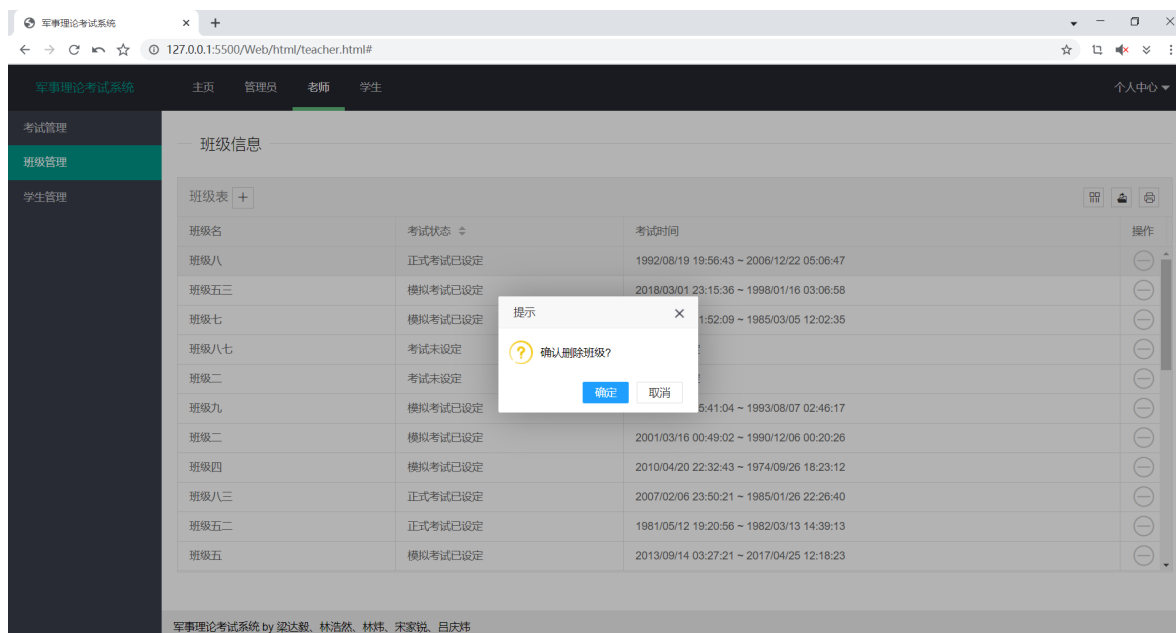
test.xlsx

请输入要上传的班级名

开始上传

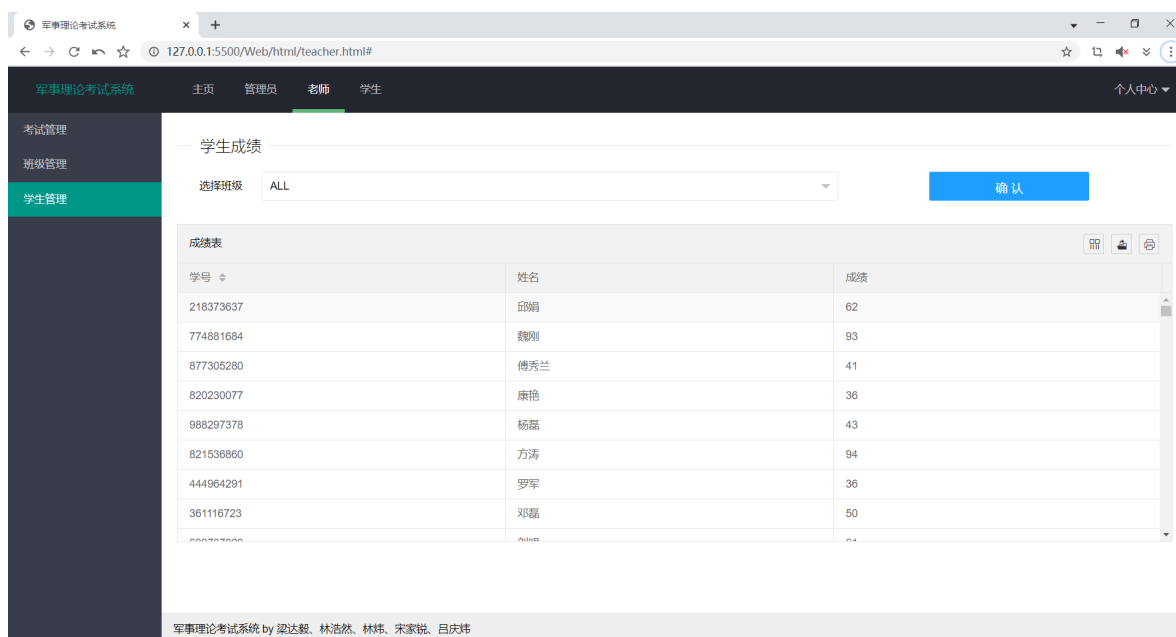
右侧操作按钮可让教师删除导入的班级表。



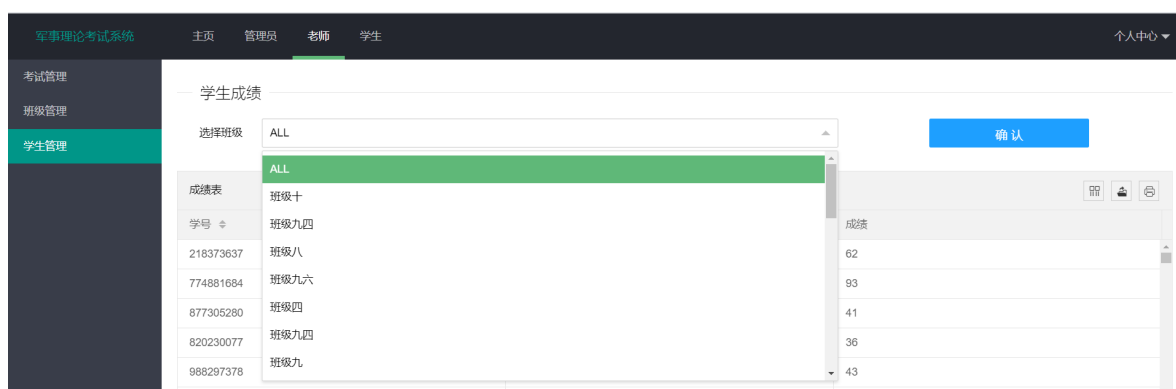


### 5.3.4 学生管理

学生管理界面如下，允许教师查看学生考试成绩、选择班级查看学生成绩、将学生成绩通过学号排序、通过姓名排序、通过成绩排序、筛选表格列信息、导出学生成绩到Excel、打印等操作。



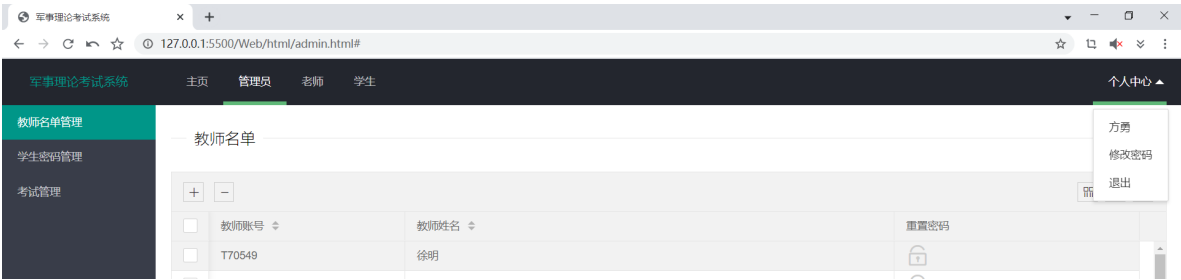
可选择要查看成绩的班级。



## 5.4 管理员用户

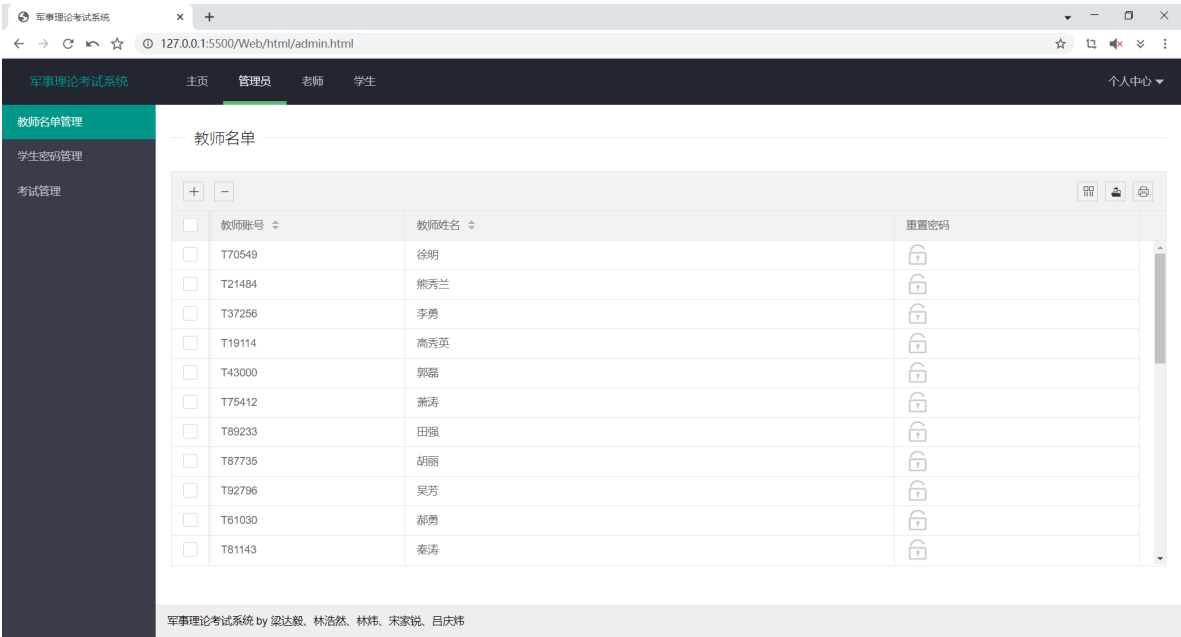
### 5.4.1 管理员用户个人中心

本系统允许用户修改密码、和登出。修改密码按钮将会弹出修改密码窗口，登出按钮将会直接登出。

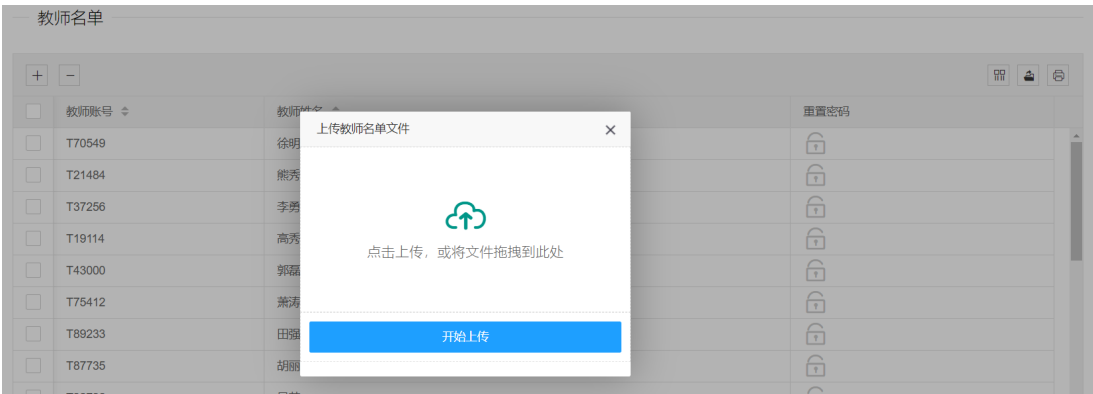


### 5.4.2 教师名单管理

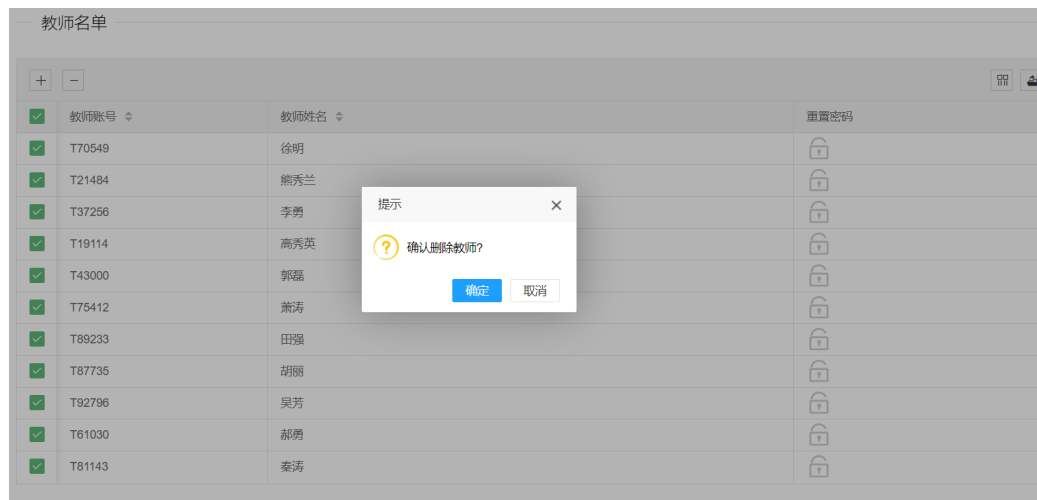
教师名单管理页面如下，支持管理员导入教师名单文件，批量删除教师、表格列筛选、导出为Excel、打印、表格可通过教师账号、教师姓名排序、重置教师密码等操作。



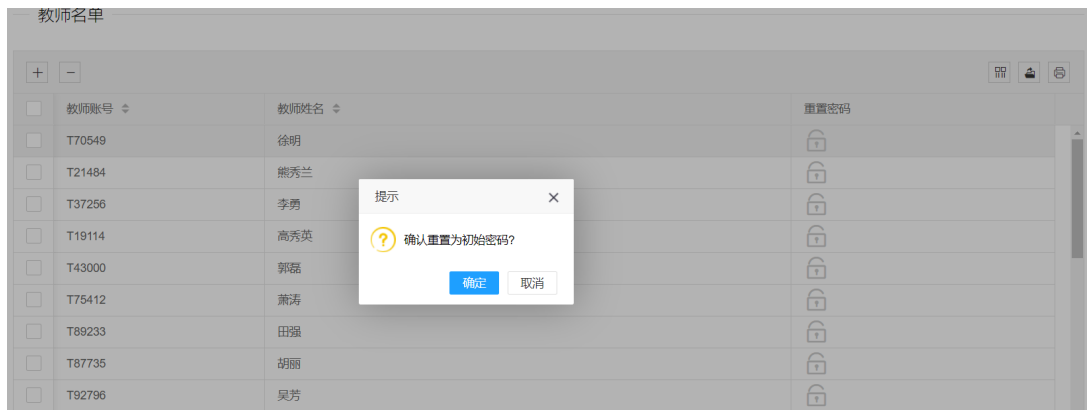
上传教师名单文件仅支持.xls或.xlsx文件。



支持批量删除教师。

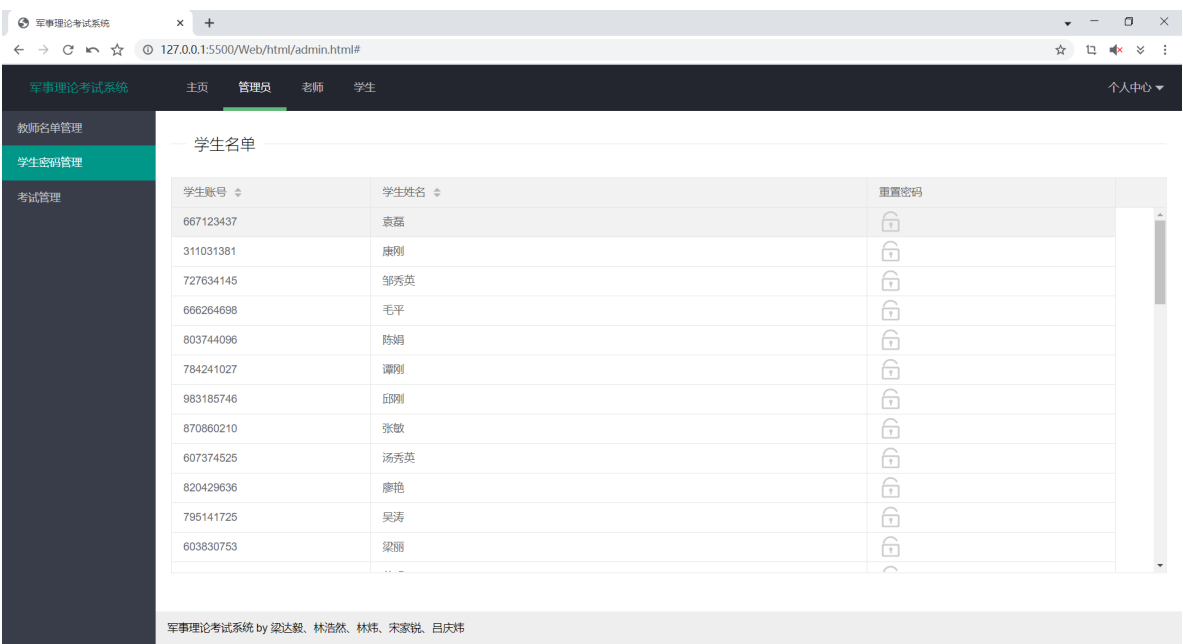


可以点击“重置密码”按钮重置当前教师密码为初始密码。



### 5.4.3 学生密码管理

学生密码管理的页面如下，支持管理员通过学生账号、学生姓名排序、重置学生密码为初始密码。



重置学生密码将会二次询问是否确定重置。

| 学生名单      |      |      |  |
|-----------|------|------|--|
| 学生账号      | 学生姓名 | 重置密码 |  |
| 667123437 | 袁磊   |      |  |
| 311031381 | 康刚   |      |  |
| 727634145 | 邹秀英  |      |  |
| 666264698 | 毛平   |      |  |
| 803744096 | 陈娟   |      |  |
| 784241027 | 谭刚   |      |  |
| 983185746 | 邱刚   |      |  |
| 870860210 | 张敏   |      |  |

提示

确认重置为初始密码?

确定取消

## 6 总结

在这次实践过程中，我们学到了非常多新的课堂之外的知识。比如github的使用，以及前后端分离开发的过程还有新的调试方法。

在项目初期，我们需要不断的制定模型，优化模型来使得这个数据库能够非常完善地实现这次的军事理论考试系统。首先就是建表，在建表的时候我们应当注意在不同的实体之间不要有过多的冗余属性，所以在这过程中，我们应当不断的优化这种关系，来使得之后系统对数据库操作的时候能够方便快捷且不易出错。在开发过程当中，我们采用了前后端分离式的开发，这就需要我们两边同时进行编程、调试，然而这又是一个完整的项目，就需要我们之间的代码是一个同步的状态，在使用github之前的我们，代码是非常分离的，都各自保存在本地的目录下，这样子非常不利于我们项目的开发，于是到中期，我们采用了github来进行代码管理与版本控制，这样就使得大家的代码都是一个同步的状态，彼此之间的进度也能够互相同步，这样的工具在这样的网课时代也更是我们开发过程中的好帮手。

在之后的程序调试，一些专门的工具例如postman、mockjs进行代码的调试，过程中也遇到了许多的困难，但是在大家的群策群力之下，这些bug也都被一一解决掉了。在最后的联调环节，虽然磕磕绊绊但也还算稳步进展，总而言之，这次的实践过程让我们第一次感受到了协同开发的历程，也更加了解了相关知识，是一次非常不错的历程。