**（分布式）学生信息管理系统**

### 1. 项目技术介绍

#### 1.1：基于soa的架构

SOA：Service Oriented Architecture面向服务的架构。也就是把工程拆分成服务层、表现层两个工程。服务层中包含业务逻辑，只需要对外提供服务即可。表现层只需要处理和页面的交互，业务逻辑都是调用服务层的服务来实现。

|  |
| --- |
|  |

#### 1.2：后端技术

|  |  |  |
| --- | --- | --- |
| **技术** | **名称** | **版本** |
| Spring Framework | 容器 | 4.2.4.RELEASE |
| SpringMVC | MVC框架 | 4.2.4.RELEASE |
| Mybatis | ORM框架 | 3.2.8 |
| MyBatis Generator | MyBatis逆向工程 | 3.2.8 |
| PageHelper | MyBatis分页插件 | 3.4.2 |
| Dubbo | 分布式服务框架进行项目间通信 | 2.5.3 |
| Zookeeper | 分布式协调服务 | 3.4.7 |
| Redis | 分布式缓存数据库 | 3.0.0 |
| Maven | 项目构建管理 | 3.3.9 |

#### 1.3：前端技术

|  |  |
| --- | --- |
| **技术** | **名称** |
| jQuery | javascript框架 |
| Bootstrap | 前端响应式框架 |
| Echarts | 图表统计 |
| wow.js && animate.css | 动画效果 |

#### 1.4：开发环境

|  |  |
| --- | --- |
| 操作系统 | Windows7 - 64bit |
| IDE | Eclipse Mars |
| 应用服务器 | Tomcat7 |
| JDK | jdk1.7.0\_80 |
| 反向代理服务器 | Nginx |

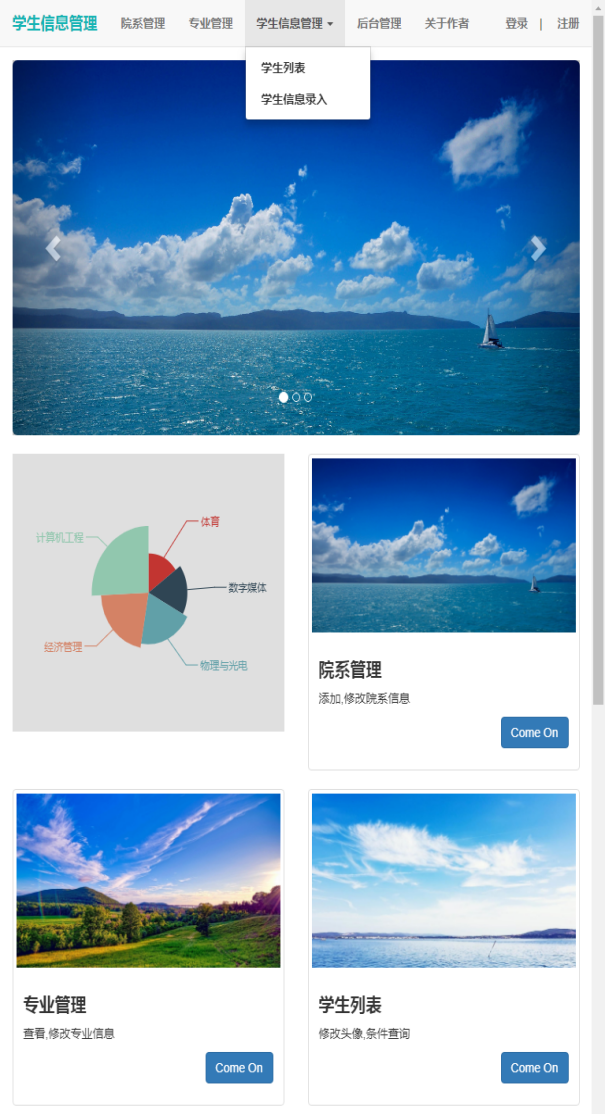
#### 1.5：生产环境

|  |  |
| --- | --- |
| 操作系统 | CentOS 7.3 |
| 应用服务器 | Tomcat7 |
| 反向代理服务器 | Nginx |
| JDK | jdk1.7.0\_80 |

### 2.设计要求

#### 2.1 全局使用响应式布局，移动端自适应

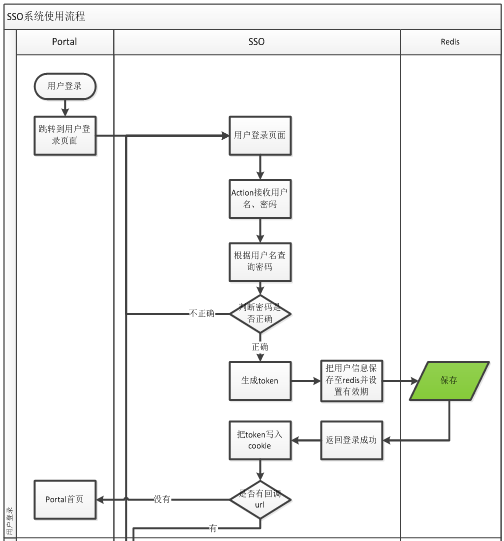
**PC端：**  **移动端：**

#### 2.2 用户登录、注册页面

用户的登录、注册功能。用户只有登录后，才能进入后台管理。登录注册功能添加**验证码，点击更换验证码，**以及注册的时候，**Ajax**验证用户名是否存在，并判断用户输入信息的合法性，并有相应提示。用户注册的默认**权限**为：游客，其它权限(超级管理员，管理员)。

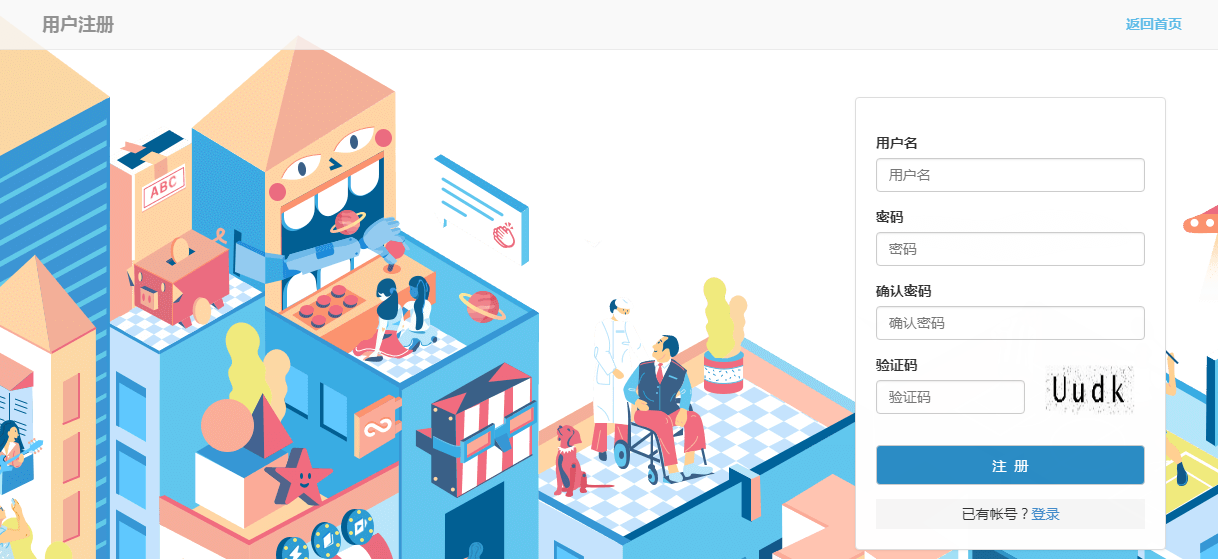
##### 2.2.1登录流程图：



用户输入账号密码后，后台接收数据，查询数据库，如果查询不打，则返回错误提示。如果从数据库查询的到，先使用UUID生成字符串，把uuid作为token，存到redis作为Key，User对象转化为JSON字符串作为Value，最后将token到客户端cookie中。

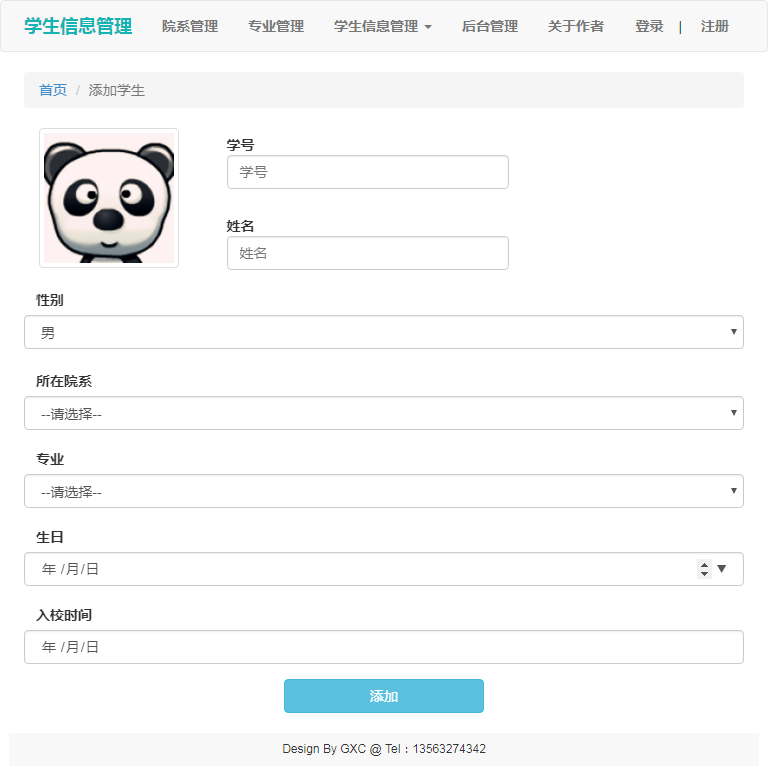
##### 2.2.2 登录效果图



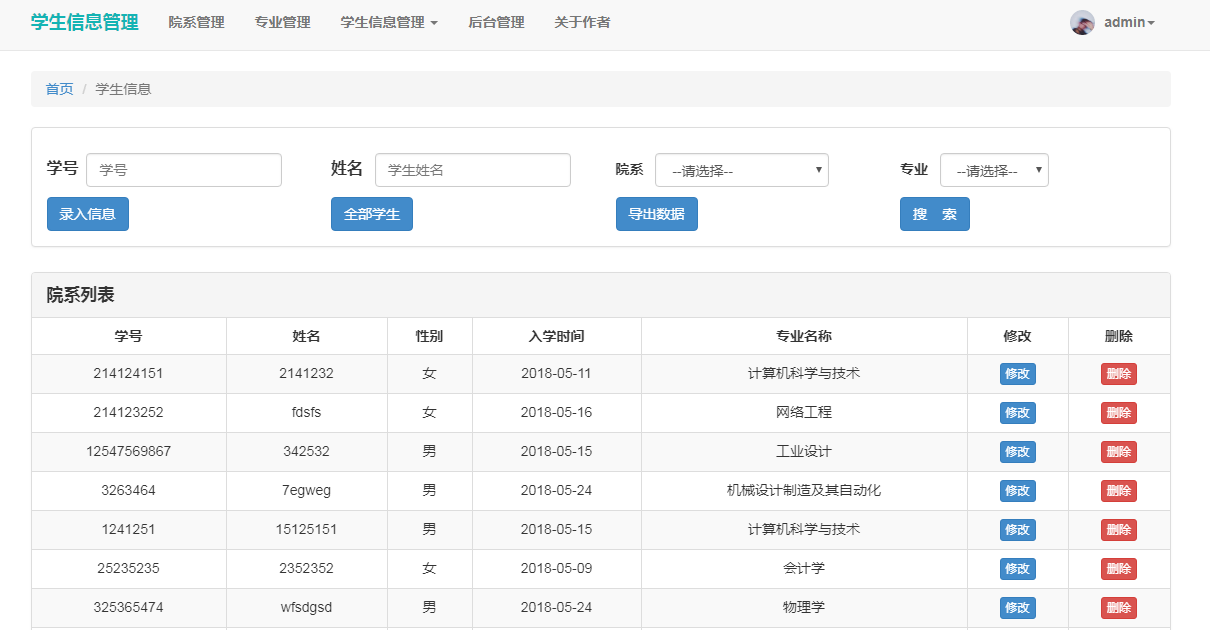


#### 2.3 前台管理页面

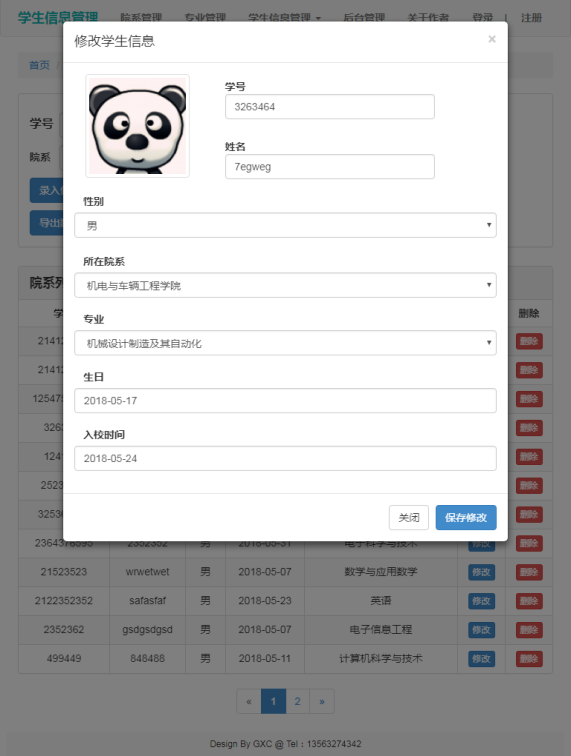
（1）录入学生基本信息功能的界面。用户可以通过菜单选项让程序呈现“录入学生基本信息”功能的界面，通过该界面可以录入学生的学号、姓名、性别、出生日期、专业、院系和照片(**点击头像上传照片**)等信息。



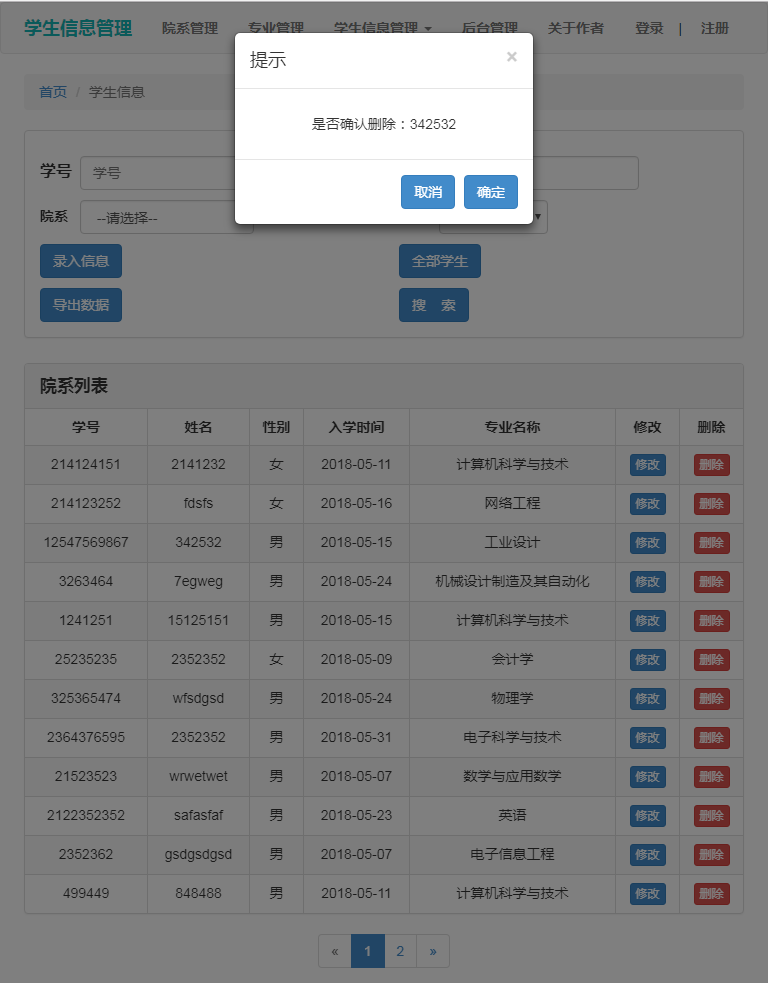
（2）多条件查询学生列表，可以根据学号、姓名**模糊查询**学生信息，也可以按照专业，院系查询学生信息，以及学生列表的**分页**，如果未填写查询条件，则查询全部学生信息。还可以点击**导出数据**，自动生成条件查询后学生列表的**Excel表格**。



（3）修改学生基本信息功能的界面。用户可以通过点击学生列表的“修改”按钮让页面呈现“修改学生信息”功能的模态框，通过该界面可以对已录入的学生信息进行修改点击图片，可以修改图片。**注：只有管理员身份才可修改学生信息**



（4）删除学生基本信息。用户可以通过点击学生列表的“删除”按钮，弹出模态框提醒，点击“确定”删除学生信息(及其服务器端上传的头像信息)。**注：只有管理员身份才可删除学生信息**

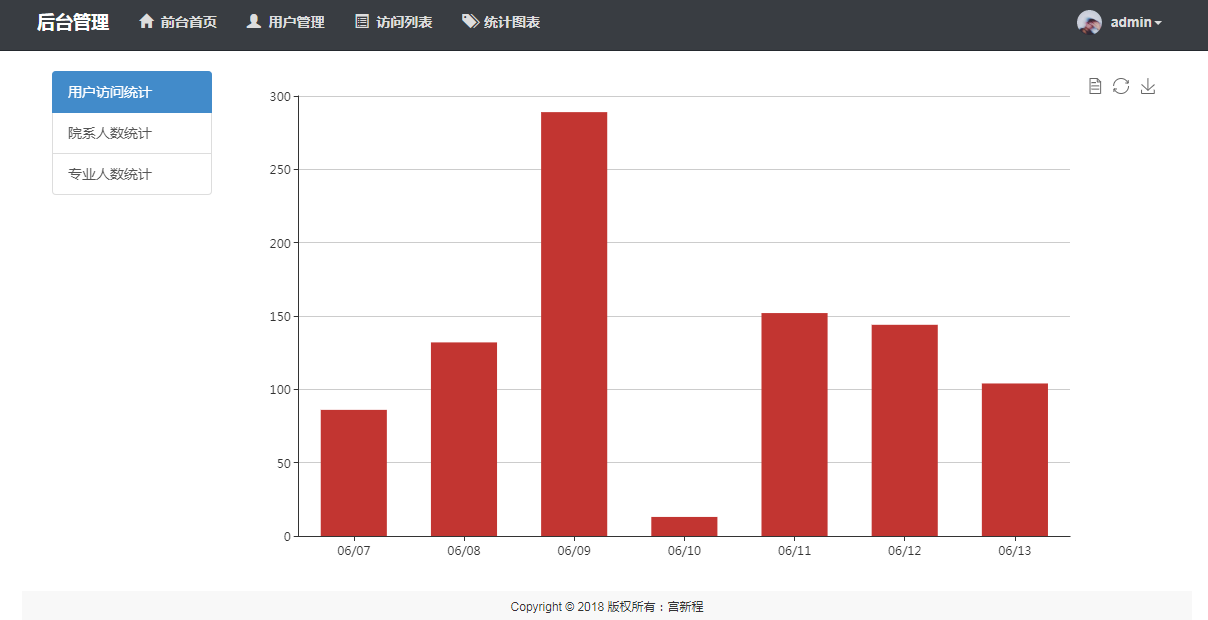


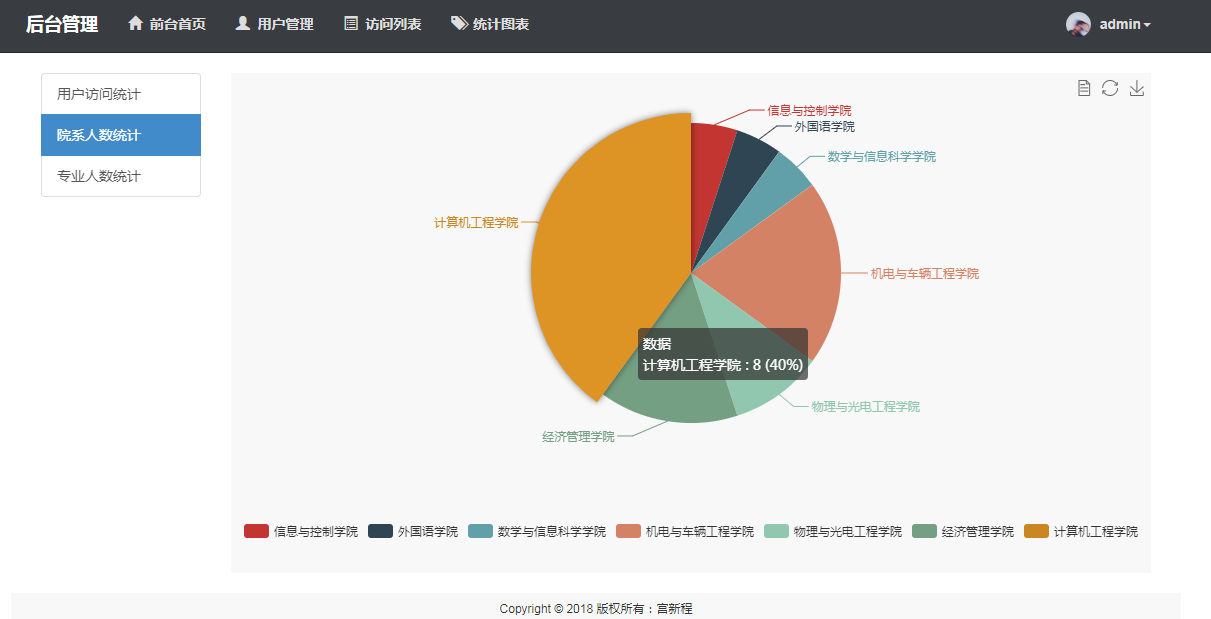
（5）院系、专业管理同学生管理(增删改查)，此处省略。

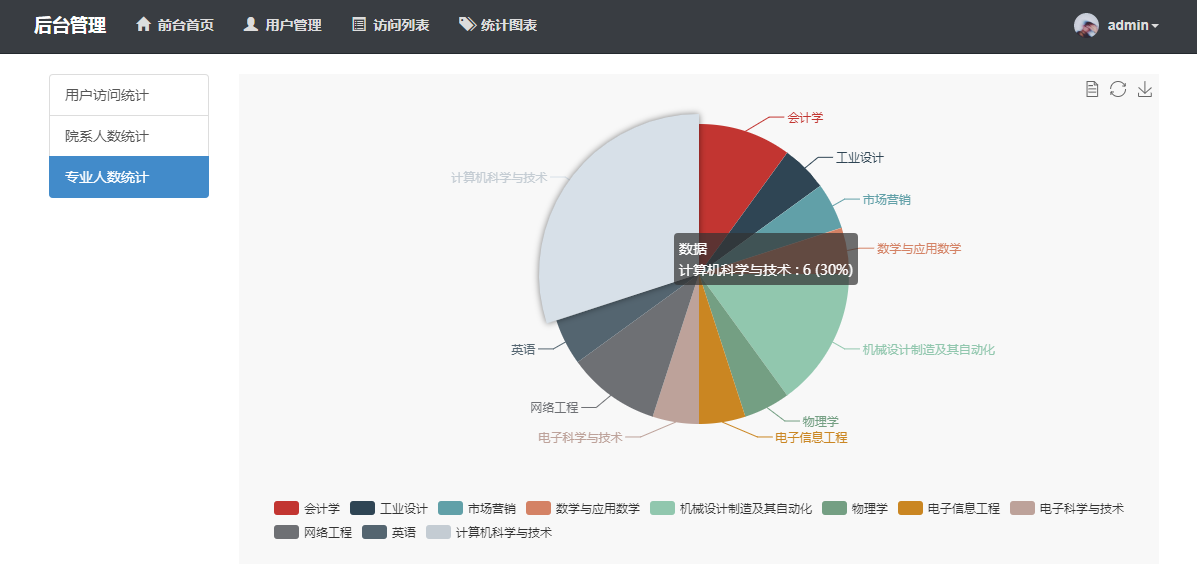
#### 2.4 后台管理页面

##### 2.4.1 统计图表

用户访问统计（统计每天网站的访问量，并以图表形式展现）、院系人数统计、专业人数统计，统计图表使用 Echarts框架。

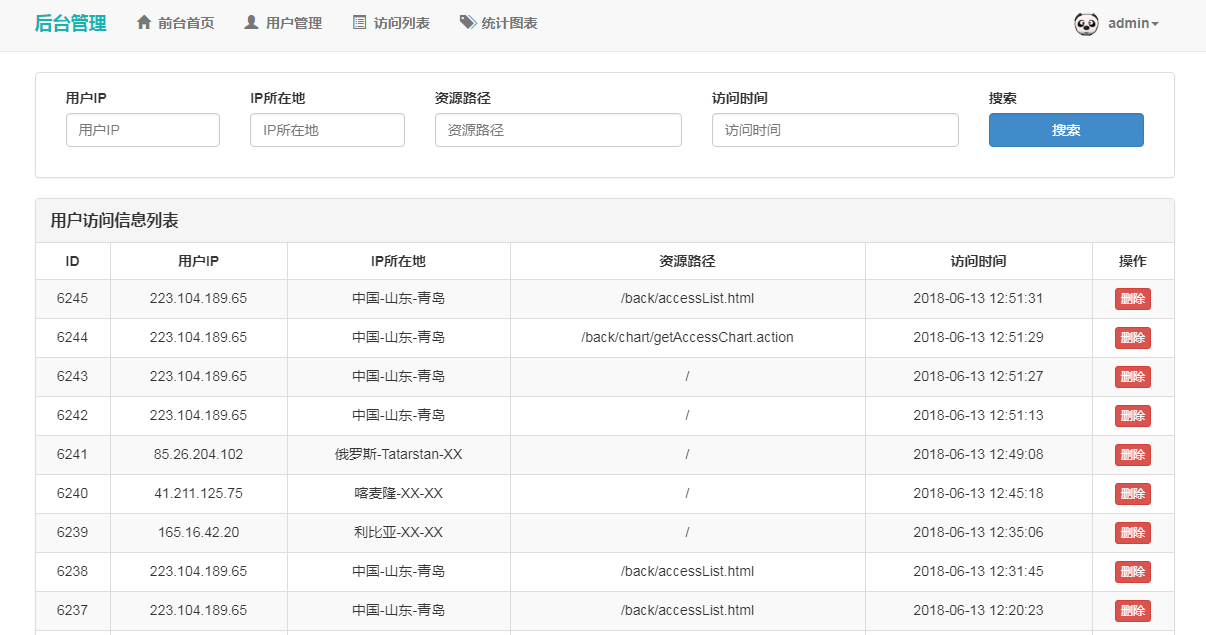






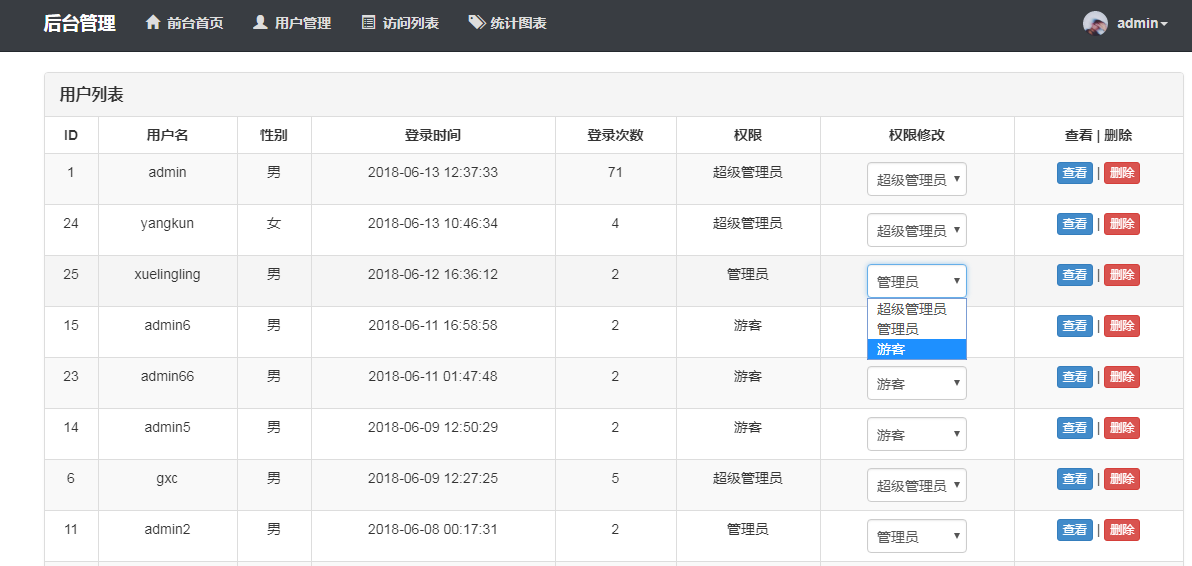
##### 2.4.2 客户端访问列表

当有用户访问网站时，可以记录访问者的ip地址，并根据ip地址获取ip所在地，并记录用户的访问时间 和 访问路径等，并可根据条件查询访问列表信息。注：该页面只能用户权限为**管理员**的用户访问，其他权限无法访问。只有管理员身份才可删除访问记录



##### 2.4.3用户列表

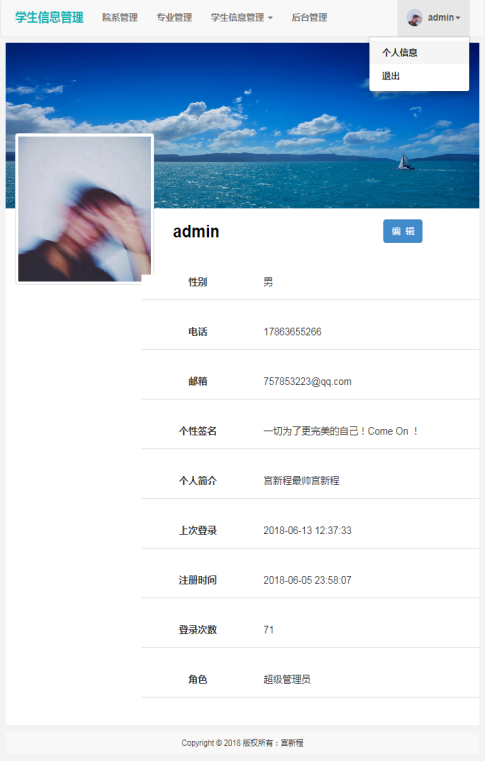
所有的注册用户的列表。可点击“查看按钮”查询用户信息详情，修改用户权限(只有超级管理员权限才可)，删除用户(只有超级管理员权限才可)，原本打算添加用户列表的多条件查询，因为时间问题未添加上，后续跟进。注：只有管理员权限才可打开该页面



#### 个人信息页面

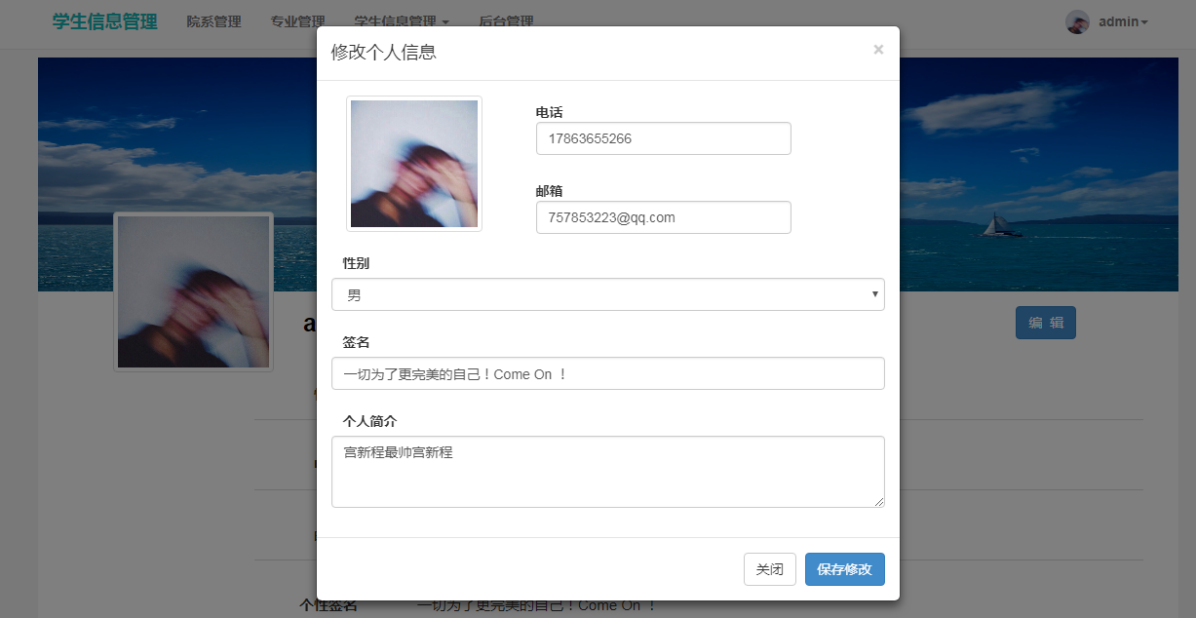
##### 2.5.1 查看个人信息

用户登录成功后，可以右上角点击“个人信息”，并跳转到个人信息页面。

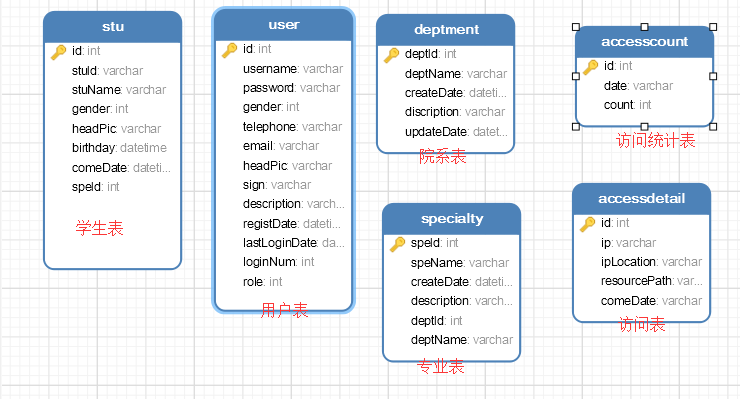


##### 修改个人信息

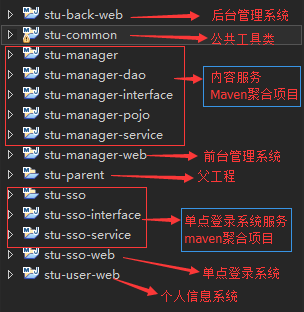
当点击个人信息的“编辑”按钮时，弹出修改用户信息的模态框，点击头像可修改用户头像。点击“保存”，修改用户信息。



### 3.数据库表设计



### 4.工程项目介绍



### 5.项目部署

首先启动Zookeeper、Redis、Nginx、MySQL等服务，使用maven构建工具的Tomcat热部署。命令：mvn tomcat7:run。

1. stu-manager（聚合项目）： [ 端口:8081 ]
2. stu-sso（聚合项目）：[端口:8084]
3. stu-manager-web：[端口:8082]
4. stu-sso-web：[端口:8085]
5. stu-back-web：[端口:8083]
6. stu-user-web：[端口:8086]