

# [教学服务系统]

之信息管理子系统

## 需求说明书

[V1.0]

组长：毛宇晗（yhmao@zju.edu.cn）

组员：钱昕，包君泽，章栩倩，李了

日期：2014-6

# 目录

1.引言 .....	6
1.1 编写目的.....	6
1.2 背景 .....	6
1.3 定义 .....	7
1.4 参考资料.....	8
1.5 项目概述.....	9
2.总体设计.....	10
2.1 需求规定.....	10
2.1.1 系统功能.....	10
2.1.2 系统性能.....	11
2.1.3 输入输出要求.....	11
2.1.4 数据管理能力要求.....	11
2.2 运行环境.....	11
2.2.1 运行服务器.....	11
2.2.2 前端框架.....	14
2.2.3 数据库.....	16
2.2.4 支持软件.....	16
2.3 基本设计概念和处理流程.....	17
2.4 结构 .....	19
2.4.1 按功能划分（横向划分） .....	19
2.4.2 按处理流程划分.....	24
2.5 内部接口.....	24
2.6 人工处理过程.....	25
2.7 尚未解决的问题.....	25
3.详细设计.....	26
3.1 登录模块设计说明.....	26
3.1.1 模块描述.....	26
3.1.2 功能.....	26
3.1.3 性能.....	26
3.1.4 输入项.....	26
3.1.5 输出项.....	27
3.1.6 设计方法.....	27
3.1.7 流程逻辑.....	28
3.1.8 测试方法.....	28
3.2 密码找回模块设计说明.....	29
3.2.1 模块描述.....	29
3.2.2 功能.....	29
3.2.3 性能.....	29
3.2.4 输入项.....	29
3.2.5 输出项.....	30
3.2.6 设计方法.....	30
3.2.7 流程逻辑.....	31

3.2.8 测试方法.....	31
3.3 修改密码模块设计说明.....	32
3.3.1 模块描述.....	32
3.3.2 功能.....	32
3.3.3 性能.....	32
3.3.4 输入项.....	32
3.3.5 输出项.....	33
3.3.6 设计方法.....	33
3.3.7 流程逻辑.....	34
3.3.8 测试计划.....	34
3.4 修改用户信息模块设计说明.....	35
3.4.1 模块描述.....	35
3.4.2 功能.....	35
3.4.3 性能.....	36
3.4.4 输入项.....	36
3.4.5 输出项.....	36
3.4.6 设计方法.....	37
3.4.7 流程逻辑.....	38
3.4.8 测试方法.....	38
3.5 查看用户信息模块设计说明.....	39
3.5.1 模块描述.....	39
3.5.2 功能.....	39
3.5.3 性能.....	39
3.5.4 输入项.....	40
3.5.5 输出项.....	40
3.5.6 设计方法.....	40
3.4.7 流程逻辑.....	41
3.5.8 测试计划.....	42
3.6 用户添加/删除模块设计说明.....	42
3.6.1 模块描述.....	42
3.6.2 功能.....	42
3.6.3 性能.....	43
3.6.4 输入项.....	43
3.6.5 输出项.....	44
3.6.6 设计方法.....	44
3.6.7 流程逻辑.....	46
3.6.8 测试方法.....	46
3.7 个人信息查看模块设计说明.....	47
3.7.1 模块描述.....	47
3.7.2 功能.....	47
3.7.3 性能.....	47
3.7.4 输入项.....	48
3.7.5 输出项.....	48
3.7.6 设计方法.....	48

3.7.7 流程逻辑.....	49
3.7.8 测试计划.....	49
3.8 课程信息查询模块设计说明.....	50
3.8.1 模块描述.....	50
3.8.2 功能.....	50
3.8.3 性能.....	50
3.8.4 输入项.....	50
3.8.5 输出项.....	51
3.8.6 设计方法.....	51
3.8.7 流程逻辑.....	52
3.8.8 测试计划.....	52
3.9 课程添加/删除模块设计说明 .....	53
3.9.1 模块描述.....	53
3.9.2 功能.....	53
3.9.3 性能.....	53
3.9.4 输入项.....	53
3.9.5 输出项.....	54
3.9.6 设计方法.....	54
3.9.7 流程逻辑.....	56
3.9.8 测试计划.....	56
3.10 课程信息修改模块设计说明.....	57
3.10.1 模块描述.....	57
3.10.2 功能.....	57
3.10.3 性能.....	57
3.10.4 输入项.....	57
3.10.5 输出项.....	58
3.10.6 设计方法.....	58
3.10.7 流程逻辑.....	59
3.10.8 测试计划.....	59
3.11 系统安全与运行日志模块设计说明.....	60
3.11.1 模块描述.....	60
3.11.2 功能.....	60
3.11.3 性能.....	61
3.11.4 输入项.....	61
3.11.5 输出项.....	61
3.11.6 设计方法.....	61
3.11.7 流程逻辑.....	62
3.11.8 测试计划.....	63
3.12 查看系统日志模块设计说明.....	63
3.12.1 模块描述.....	63
3.12.2 功能.....	64
3.12.3 性能.....	64
3.12.4 输入项.....	64
3.12.5 输出项.....	64

3.12.6 设计方法.....	64
3.12.7 流程逻辑.....	65
3.12.8 测试计划.....	66
3.13 查看数据库表格模块设计说明.....	66
3.13.1 模块描述.....	66
3.13.2 功能.....	66
3.13.3 性能.....	66
3.13.4 输入项.....	67
3.13.5 输出项.....	67
3.13.6 设计方法.....	67
3.13.7 流程逻辑.....	68
3.13.8 测试计划.....	68
3.14 备份数据模块设计说明.....	69
3.14.1 模块描述.....	69
3.14.2 功能.....	69
3.14.3 性能.....	69
3.14.4 输入项.....	69
3.14.5 输出项.....	70
3.14.6 设计方法.....	70
3.14.7 流程逻辑.....	70
3.14.8 测试计划.....	71
3.15 一键销毁系统模块设计说明.....	71
3.15.1 模块描述.....	71
3.15.2 功能.....	71
3.15.3 性能.....	71
3.15.4 输入项.....	72
3.15.5 输出项.....	72
3.15.6 设计方法.....	72
3.15.7 流程逻辑.....	73
3.15.8 测试计划.....	73
4.接口设计.....	74
4.1 用户接口.....	74
4.1.1 欢迎界面.....	74
4.1.2 登陆界面.....	74
4.1.3 修改密码界面.....	75
4.1.4 查看用户信息界面.....	75
4.1.5 修改用户信息界面.....	76
4.1.6 检索用户界面.....	76
4.1.7 添加用户界面.....	77
4.1.8 检索课程界面.....	77
4.1.9 查看课程基本信息.....	78
4.1.10 修改课程基本信息界面.....	78
4.1.11 添加课程界面.....	79
4.2 外部接口.....	79

4.3 内部接口.....	79
5.系统数据结构设.....	80
5.1 概念结构设计.....	80
5.2 逻辑结构设计.....	80
5.3 物理结构设计.....	81
6.运行设计 .....	82
6.1 运行模块组合 .....	82
6.2 运行控制.....	82
6.3 运行时间.....	83
7.系统出错设计.....	83
7.1 出错信息.....	83
7.2 补救措施.....	84
7.3 系统维护设计.....	84

# 1.引言

## 1.1 编写目的

为了规范教学服务系统子系统——信息管理系统的编码过程，编写本需求说明书以供软件开发者在实现过程中进行参考。本需求说明书描述了本系统的各项功能和性能需求，明确阐述了各功能的实现过程、实用范围和客户的使用场景，并为其提供了一个遵循的标准。另外，本需求说明书也为程序员和客户提供了一个沟通的桥梁，使得开发者和客户对需求有明确的了解，消除歧义，也是为了使软件开发过程更顺利。

本说明书的预期读者包括：

- 客户
- 项目经理
- 本系统及其他子系统的软件开发人员
- 软件质量分析员
- 软件测试人员

## 1.2 背景

**软件系统名称：**

教学服务系统

**任务提出者：**

浙江大学 2013-2014 夏学期软件工程基础课程任课老师——王新宇

**开发者：**

浙江大学 2013-2014 夏学期软件工程基础课程部分学生组成的项目组第一小组

**用户：**

教务处管理人员、系统管理员、任课老师、学生

**实现该软件的网络范围：**

由多台 PC 组成的局域网

**本机该子系统与其他子系统的关系：**

本子系统——信息管理系统只是教学服务系统中的一部分，与项目组中其他小组的系统可以整合成教学服务系统

## 1.3 定义

### 用户权限：

用户所能进行的操作，例如查询、修改成绩和个人信息。

### 用例：

每个用例提供了一个或多个场景，该场景说明了系统是如何和最终用户或其它系统互动，即谁可以用系统做什么，从而获得一个明确的业务目标。

### IPO 图：

输入加工输出图的简称，用来描述每个模块的输入、输出和数据加工过程。

### 数据流图：

能全面地描述信息系统逻辑模型，用少数几种符号综合地反映出信息在系统中的流动、处理和存储情况。

### 状态图：

描述一个实体基于事件反应的动态行为，显示了该实体如何根据当前所处的状态对不同的事件做出反应。

### CRC 卡：

标准索引卡集合的集合，包括类名、类的职责、类的协作关系三部分，每一张卡片表示一个类。

### 数据词典：

定义数据流图中的各个成分的具体含义，对每一个数据流、文件、加工给出详细定义。

### 系统集成：

根据用户的需求，通过优化和连接技术将分离的子系统整合成一个有效的整体，使性能达到最优。



## 1.4 参考资料

- 《软件工程——实践者的研究方法》  
作者： Roger S.Pressman(美)  
译者： 郑人杰 马素霞 白晓颖  
出版社： 机械工业出版社  
出版年月： 2007 年 1 月
- 《软件工程导论》  
作者： 张海藩  
出版社： 清华大学出版社  
出版年月： 2008-2
- 《需求分析与系统设计》  
作者： 麦斯阿塞克  
出版社： 机械工业出版社  
出版年： 2009-9
- 软件需求说明书国家标准规范( GB856T——88)
- 网页设计心理学  
作者： Susan M. Weinschenk  
出版社： 人民邮电出版社  
出版年月： 2013-1
- Sina App Engine  
<http://sae.sina.com.cn/>
- Bootstrap 中文网  
<http://www.bootcss.com/>
- w3school 在线教程  
<http://www.w3school.com.cn/>
- 股票交易系统概要设计说明书 v4.0  
作者： 浙江大学软件工程基础课程 2012 级王新宇班第一大组第一小组

## 1.5 项目概述

一个完整的教学服务系统，应实现以下功能模块：

### 1. 基础信息管理子系统

- 1) 用户基本信息管理
- 2) 课程基本信息管理
- 3) 用户权限管理：不同类型用户拥有不同的权限，系统管理员拥有最高权限
- 4) 系统安全管理：保证整个系统的安全性

### 2. 自动排课子系统

- 1) 教学资源管理：增加或更新每个教室的基本信息
- 2) 系统根据必要信息自动排课
- 3) 手动调课功能：更加特殊要求在自动化排课基础上进行手动调整，并监测冲突
- 4) 课表查询和打印
- 5) 性能指标与相关约束：需要充分利用教学资源来满足老师和学生

### 3. 选课子系统

- 1) 专业培养方案制定：学生在制定自己的培养方案后方能选课
- 2) 搜索和浏览课程信息
- 3) 学生可以根据培养方案和课程容量进行选课
- 4) 选课结果浏览
- 5) 充分考虑全部学生人数，控制同时在线选课人数，同时管理人员可以设定第一轮选课和补选的时间
- 6) 管理员可以根据特殊要求人工选课
- 7) 性能指标与相关约束：至少支持 200 人同时选课，登录后长时间无操作自动下线

### 4. 课程资源共享子系统

- 1) 教师和学生可以上传和下载与课程有关的资料
- 2) 课程作业的布置和提交
- 3) 系统管理员可以对大量教学资源进行管理，例：高频下载资料可以置于醒目处
- 4) 提供教学资源的检索功能

### 5. 论坛交流子系统

- 1) 教师可以发布课程相关信息公告

- 2) 教师和学生可以发表带相关附件的帖子
  - 3) 用户可以回复帖子，并且用户间可以在线交流
  - 4) 论坛管理员可以对热门帖子和热门回复进行统计分析
  - 5) 支持对已发布的帖子进行检索
6. 在线测试子系统
- 1) 教师可以对题库进行管理
  - 2) 教师可以手动从题库中生成试卷，也支持系统更加选定范围自动生成试卷
  - 3) 学生可以在线完成测试，即时出分
  - 4) 教师可以使用统计分析功能进行成绩分析
  - 5) 学生可查看历史成绩
  - 6) 性能指标与相关约束：支持至少 100 个学生同时在线测试
7. 成绩管理子系统
- 1) 教师可以输入并修改学生成绩，但在正式提交后，教师需要提交修改理由才能修改成绩
  - 2) 教师提交成绩后，学生可以查看自己的成绩和相关统计分析
  - 3) 成绩分析：教师以图形化的方式获得某门课程的成绩分布、平均分等信息，学生可以获得包括 GPA，平均分和总学分等在内的统计分析

其中，我们组负责的是第一个子系统——基础信息管理子系统，要实现用户和课程基本信息的管理，以及负责整个系统的权限管理和安全防护。因为这个系统与其他子系统之间交互较多，在前期需要进行大量的交流沟通，确定基本的构架方案，在界面设计上也要力求风格一致。

## 2. 总体设计

### 2.1 需求规定

#### 2.1.1 系统功能

用户界面，包括所有用户登录界面，找回密码界面，修改密码界面，显示/更改个人资

料界面，上传头像界面，课程查找界面和显示课程基本信息界面；管理员查找/添加/删除课程界面，编辑课程信息界面，查找/添加/删除用户信息界面，查看用户日志界面，更改用户权限界面，以及超级管理员添加管理员界面。

### 2.1.2 系统性能

本系统的性能将由网络数据传输延迟、服务器响应时间内、数据库响应时间以及并发访问该系统的用户数量决定。服务器采用新浪 SAE 云解决方案以确保稳定性。客户端方面将实现 AJAX 技术，减轻服务器压力，增加带宽利用率，并未用户提供良好的界面体验。

### 2.1.3 输入输出要求

网页客户端前端采用 bootstrap 开发框架，力求为用户提供一个美观，友善的操作界面。用户通过点击按钮或者提交表单的方式向服务器下达指令与提供数据。

服务器后台处理后将结果显示到用户的网页界面上。

### 2.1.4 数据管理能力要求

安全：服务器将予以数据库最高等级的保护，防止黑客从后台下载数据库，防止通过网页 SQL 注入的方式从数据库中获取信息或者破坏数据库。

性能：对于频繁的访问数据库的操作，后台建立持久的数据库连接，以避免重复连接数据库耗费资源，并将常用数据保存在 cookies 中。

## 2.2 运行环境

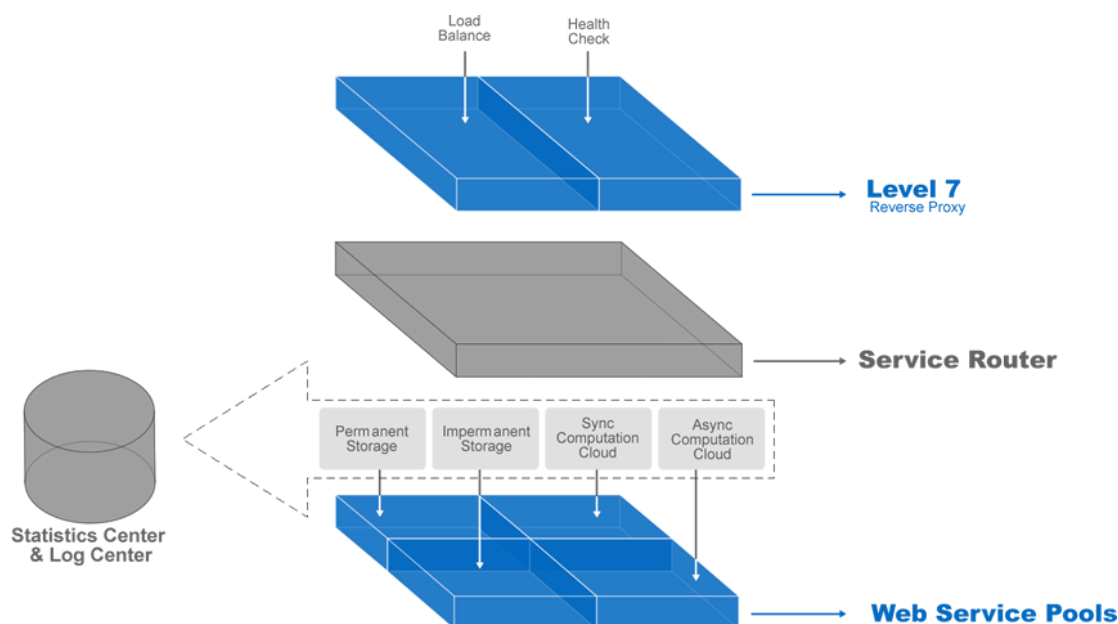
### 2.2.1 运行服务器

本系统使用 Sina App Engine 作为云端服务器

- SAE 整体架构

SAE 从架构上采用分层设计，从上往下分别为反向代理层、路由逻辑层、Web 计算服务

池。而从 Web 计算服务层延伸出 SAE 附属的分布式计算型服务和分布式存储型服务，具体又分成同步计算型服务、异步计算型服务、持久化存储服务、非持久化存储服务。各种服务统一向日志和统计中心汇报，参考下图：



SAE 整体架构图

**7 层反向代理层：**HTTP 反向代理，在最外层，负责响应用户的 HTTP 请求，分析请求，并转发到后端的 Web 服务池上，并提供负载均衡、健康检查等功能。

**服务路由层：**逻辑层，负责根据请求的唯一标识，快速的映射（ $O(1)$ 时间复杂度）到相应的 Web 服务池，并映射到相应的硬件路径。如果发现映射关系不存在或者错误，则给出相应的错误提示。该层对用户隐藏了很多具体地址信息，使开发者无需关心服务的内部实际分配情况。

**Web 服务池：**由一些不同特性的 Web 服务池组成。每个 Web 服务池实际是由一组 Apache Server 组成的，这些池按照不同的 SLA 提供不同级别的服务。每个 Web 服务进程实际处理用户的 HTTP 请求，进程运行在 HTTP 服务沙盒内，同时还内嵌同样运行在 SAE 沙盒内的解析引擎。用户的代码最终通过接口调用各种服务。

**日志和统计中心：**负责对用户所使用的所有服务的配额进行统计和资源计费，这里的配额有两种，一种是分钟配额，用来保证整个平台的稳定；一种是天配额，用户可以给自己的设定每天资源消耗的最高上限。日志中心负责将用户所有服务的日志汇总并备份，并提供检索查询服务。

**各种分布式服务：**SAE 提供覆盖 Web 应用开发主要方面的多种服务，用户可以通过 StdLib

（可以理解为 SAE PHP 版的 STL）很方便的调用它们。同时因为 Web 服务的多样性，SAE 的标准服务不可能满足所有场景的需求，所以 SAE 通过服务总线来对接第三方服务（如分词、全文检索等），SAE 也欢迎第三方服务商选择 SAE 来为开发者提供服务。

## ● 扩展性

扩展性是分布式系统的两个主要目的之一，SAE 作为公有云计算，同样把服务的扩展性作为架构设计的重要指标，要求在用户增长、压力提升的情况下，可以实现自动的服务扩展，同样的当压力降低时，可以将服务收缩，以节约资源，整个过程无需人工参与。SAE 人工只需做好容量规划和管理。目前国外的公有云计算架构的扩展性主要有两个思路：

静态扩展，用户和资源有强绑定关系。最典型的例子为亚马逊的 EC2 和 Ruby 云计算平台 Heroku，用户申请的资源和用户有严格的一对一关系，换句话说，A 用户申请的虚拟机在 A 退还资源前，B 用户不能使用，哪怕 A 用户的虚拟机处于闲置状态。

动态扩展，用户和资源没有强绑定关系。最典型的例子为 Google App Engine，用户申请的资源和用户没有严格的一对一关系，换句话说，处理 A 用户请求的进程在处理完之后，可以马上处理 B 用户的请求。

两种扩展性各有利弊，静态扩展的长处是为平台提供了良好的隔离性，资源可以固定的映射在某个用户下，但缺点是资源利用率不高；动态扩展的长处是资源利用率高，这样整个云计算平台的成本会很低，但缺点是对隔离性有更高的要求，因为资源可以在很短的时间被多个用户使用。相比较，在安全性上，动态扩展要比静态扩展的技术门槛更高。

在 SAE 平台上，我们采用以动态扩展为主，静态扩展为辅的兼而有之的设计。在 Web 计算池层，是典型的动态扩展，没有一个用户独占 Web 服务进程，而是所有用户以共享的方式使用 Web 服务进程，通过 Cache，热的用户自然在缓存层占据更多的位置。而在 SAE 的某些服务中，扩展性又是以静态扩展的方式展现，如 RDC（Relational DB Cluster）分布式数据库集群，当用户申请了 MySQL 服务，我们就会在 RDC 后端根据 SLA 的级别创建一主多从的 DB 给用户，在用户显式的删除该 DB 前，该 DB 都不会被别人使用。当然，通过 RDC，任何一个用户也无需知道后端 DB 的实际地址，只需访问 RDC 统一的 host 和 port 即可。

## ● 高可靠性

HA 是分布式系统的另一个主要目的，SAE 同样以提供服务的高可靠性为架构设计的重要指标。HA 的实现途径主要有两个，一个是硬件保证，一个是架构的冗余设计。

在 SAE 平台上，所有服务器都是新浪标准采购的硬件设备，运行在国内最好机房内，并进行多机房容灾，网络资源方面则享用门户网站所使用的带宽环境。另外，所有的硬件设备都有专门的运维部门负责，故障的响应速度和新浪内部服务一样。

在架构设计上，SAE 通过对所有服务都进行冗余设计来提供服务的高可靠性。这里的服务可以分成计算型和数据型两种类别讨论：

针对计算型程序，冗余设计就是程序在多节点运行。但这样会带来一致性问题，最主要的困扰就是选举问题，如何在多个节点中选出一个主节点来执行。比如 SAE 上的分布式定时服务 Cron，采用多点部署方式，多个计算节点相互隔离，通过时钟同步服务同时触发用户设定的定时任务，但要求只能有一个节点负责执行。为了解决这个问题，SAE 设计出了一套分布式锁算法来提供选举服务。该算法可以在牺牲某些特定条件下的一致性来提供比 Paxos 算法更高的可靠性(3 台机器在最高任意 2 台机器发生故障的情况下整个选举过程仍然正常，而 Paxos 算法最多容忍 1 台)。目前，该算法正在申请专利，并广泛应用在 SAE 内部。

针对数据型服务，SAE 主要是通过复制来保证服务的高可靠性。SAE 上的数据存储服务普遍采用被动复制和主动复制两种方式。如 SAE 上 MySQL 之间的主从 Binlog 同步就是典型的被动复制，TaskQueue、DeferredJob 等服务也采用被动复制的方式，用户的任务描述会写到主内存级队列中，主队列利用后台线程将写操作同步到从队列上，一旦主队列发生故障，从队列会快速的切换为主队列。另外 SAE 上也有部分服务采用主动复制（双写复制）的方式来保证 HA，比如 Cron，当用户通过 App 的工程配置文件 appconfig.yaml 设定定时任务时，任务信息会以双写的方式写到多个持久化 DB 中，以供后续的到时触发。

另外，SAE 在整体架构设计时，充分考虑服务之间的“优雅降级”，尽量降低服务之间的耦合度，我们要求任何一个服务都不要假设其他服务是可靠的。目前在 SAE 平台上的所有服务均不存在单点设计，服务的平均 HA 在 99.95%，即年平均服务不可用时间在 4 到 5 个小时之间。

## 2.2.2 前端框架

网页客户端前端采用 bootstrap 开发框架，力求为用户提供一个美观，友善的操作界面。



- **Bootstrap 前端开发框架**

Bootstrap 是 Twitter 推出的一个开源的用于前端开发的工具包。它由 Twitter 的设计师 Mark Otto 和 Jacob Thornton 合作开发,是一个 CSS/HTML 框架。Bootstrap 提供了优雅的 HTML 和 CSS 规范,它即是由动态 CSS 语言 Less 写成。Bootstrap 一经推出后颇受欢迎,一直是 GitHub 上的热门开源项目,包括 NASA 的 MSNBC (微软全国广播公司) 的 Breaking News 都使用了该项目。

- **特点**

Bootstrap 是基于 jQuery 框架开发的,它在 jQuery 框架的基础上进行了更为个性化和人性化的完善,形成一套自己独有的网站风格,并兼容大部分 jQuery 插件。

- **组件**

Bootstrap 中包含了丰富的 Web 组件,根据这些组件,可以快速的搭建一个漂亮、功能完备的网站。其中包括以下组件:

下拉菜单、按钮组、按钮下拉菜单、导航、导航条、面包屑、分页、排版、缩略图、警告对话框、进度条、媒体对象等

- **Javascript 插件**

Bootstrap 自带了 13 个 jQuery 插件,这些插件为 Bootstrap 中的组件赋予了“生命”。其中包括: 模式对话框、标签页、滚动条、弹出框等。

- **定制自己的框架代码**

可以对 Bootstrap 中所有的 CSS 变量进行修改,依据自己的需求裁剪代码。



## ● Bootstrap Less

Less 是一个 CSS 预处理器，让 CSS 具有动态性。另一方面，Bootstrap 是一个快速开发 Web App 和站点的工具包。这样，您可以在 CSS 中使用 Bootstrap 的 Less 变量、mixins（混合）和 nesting（嵌套）。

### 2.2.3 数据库

MySQL 是 SAE 为用户提供的分布式 MySQL 数据库集群服务，可以支持百万级的数据库访问。SAE 的 MySQL 数据库集群服务有以下特点：

- 通过其强隔绝性为开发者提供了更高的安全性,保障开发者的数据安全。
- 当某个数据库发生故障和延迟时,系统能够自动切换,保证服务稳定可靠。
- 能通过对 SQL 的智能预处理降低用户触发分钟配额的可能。
- 整体数据库集群性能提升。

SAE 平台为支持几乎所有 MySQL 的特性。公共版支持 MyISAM 引擎,企业版支持 InnoDB,详见企业版介绍。需要注意的是 SAE 的 MySQL 管理页面显式开启才能使用。

目前 MySQL 服务对用户开放的操作：

- select, insert, update, delete
- create table
- alter table
- drop table
- index

可以通过在 MySQL 的管理页面中集成的 PhpMyAdmin 里创建数据库和数据表。

### 2.2.4 支持软件

服务器端：

操作系统：SAE 分布式 Web 服务开发、运行平台

PHP：PHP5.0

数据库：Mysql5.6

Web 服务器：Apache

Mysql 管理软件：PHPMyadmin 或其它管理软件

开发工具：Sublime3、Github

办公软件：Microsoft Office 2013 和 Microsoft Office 2011 for Mac 系列产品

测试工具:测试人员自行选择,推荐使用 LoadRunner

建模工具:根据项目情况,选择合适方便的建模工具

客户端:

桌面端浏览器：IE8-IE11，Safari，Chrome 等多浏览器支持

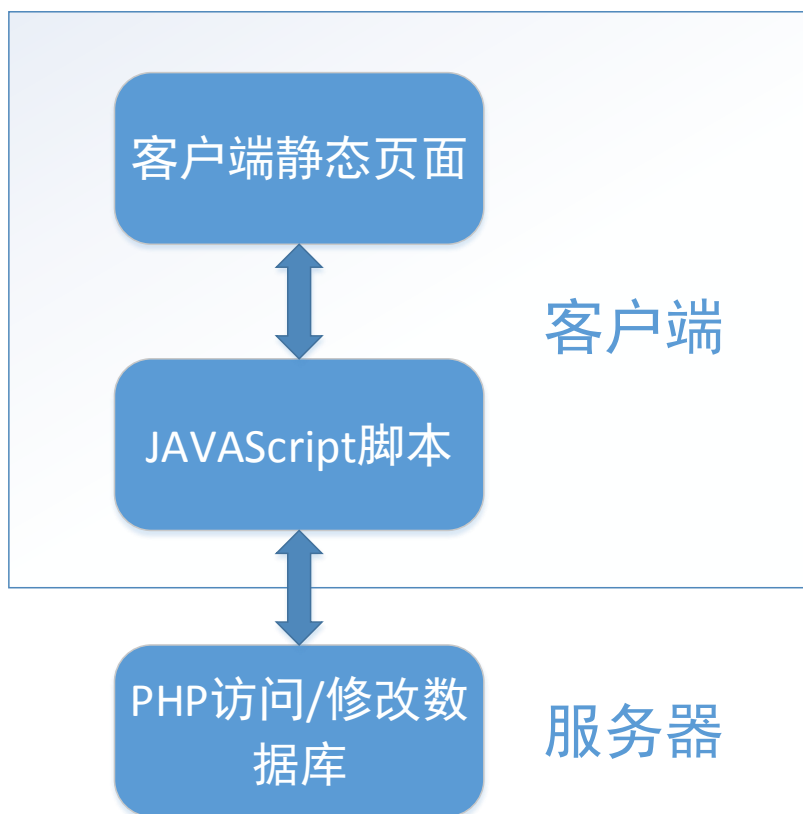
手机端浏览器：Safari，Chrome，UC 等多浏览器支持

## 2.3 基本设计概念和处理流程

服务器：以 SAE 的 Apache 为服务器，服务器端采用 PHP 语言编写，数据库采用 MySQL。

客户端：完美兼容桌面端和手机端 Chrome、Safari、IE8 以上浏览器。采用 Bootstrap 前端框架，界面简洁美观，兼容性强。以及 Ajax 技术减轻服务器压力，增加服务器带宽利用率。

处理流程图如下：



**客户端静态页面：**通过 **Bootstrap** 前端框架，设计一个简洁，美观的界面。

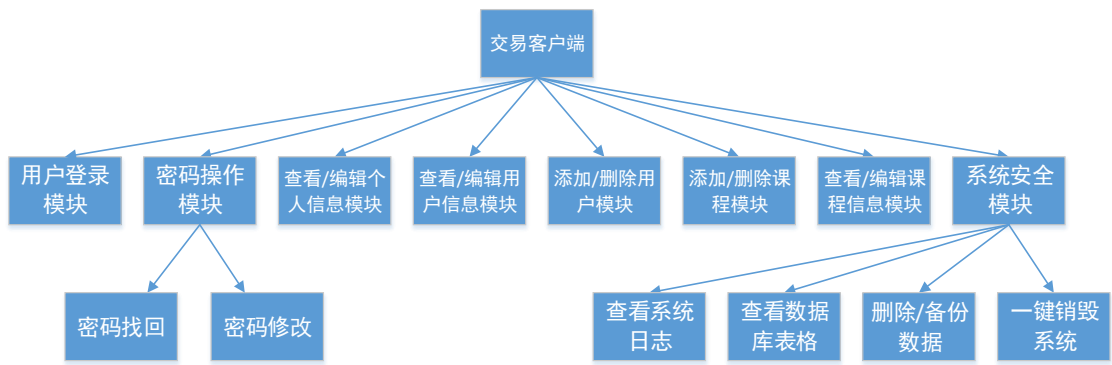
**JavaScript 脚本：**客户端静态页面中，各种文本框与按钮的操作均能触发脚本函数，脚本通过创建 **XMLHttpRequest** 对象，与服务器进行异步交互。并且，对象接受服务器反馈信息后，能通过脚本函数对客户端静态页面实现无刷新的更新信息。

**PHP 访问/修改数据库：**此模块通过 **url** 参数方式从客户端得到数据与命令，并且对其进行安全检测，而后按照要求访问/修改数据库，并且返回操作结果或是查询结果。

2.4 结构

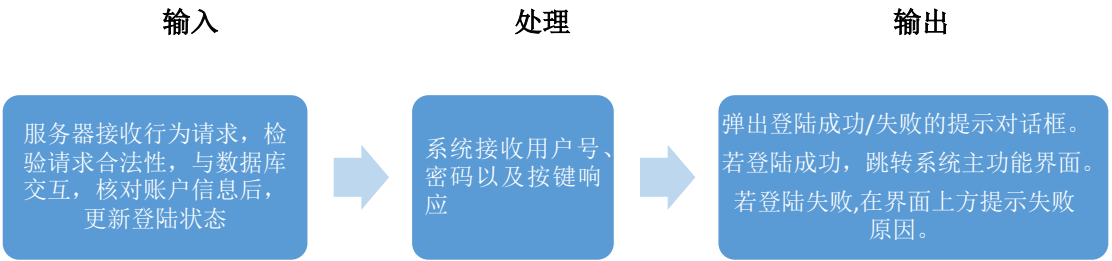
2.4.1 按功能划分（横向划分）

客户端 HIPO 图

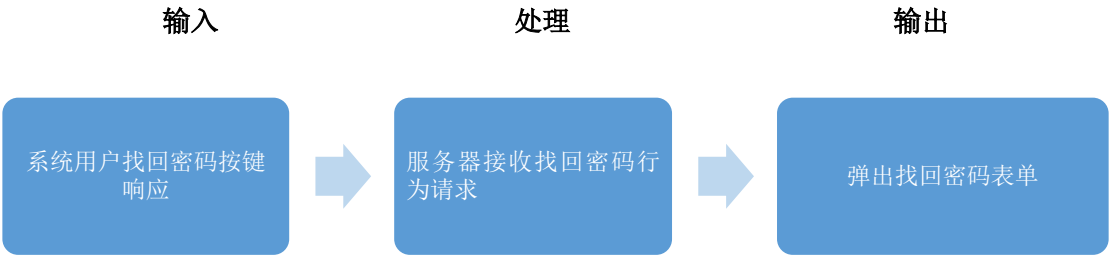


客户端 IPO 图

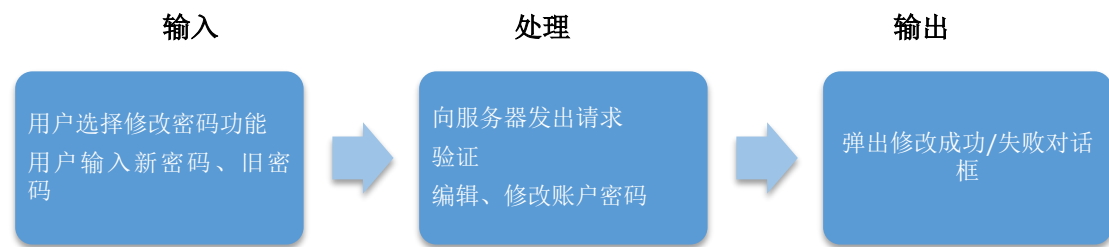
登陆模块:



密码找回模块:



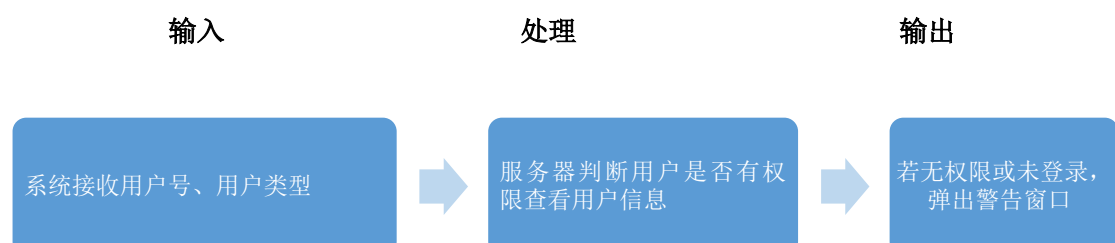
### 修改密码模块:

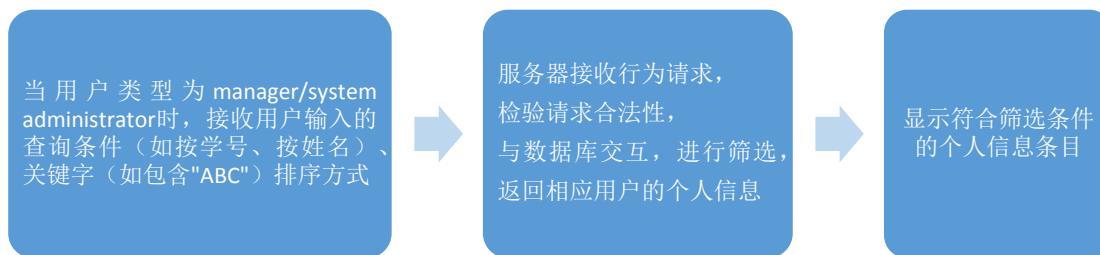


### 修改用户信息模块:

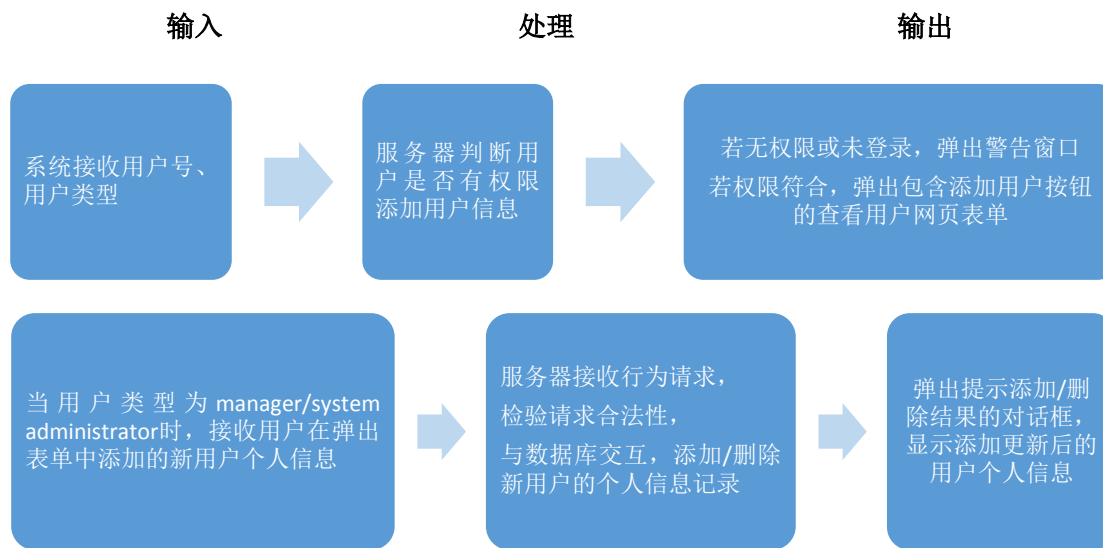


### 查看用户信息模块:

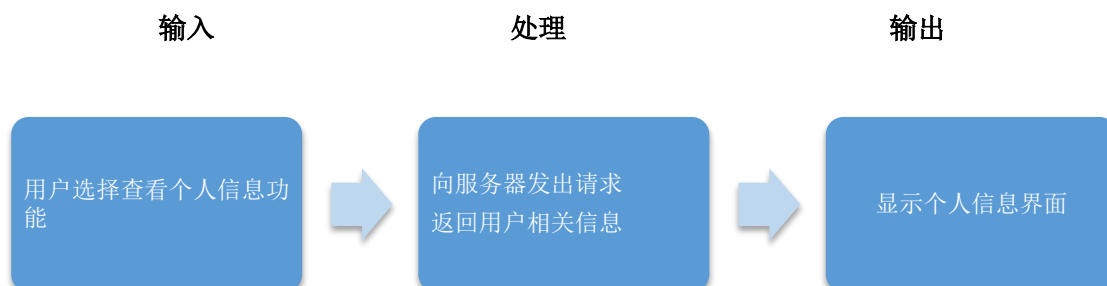




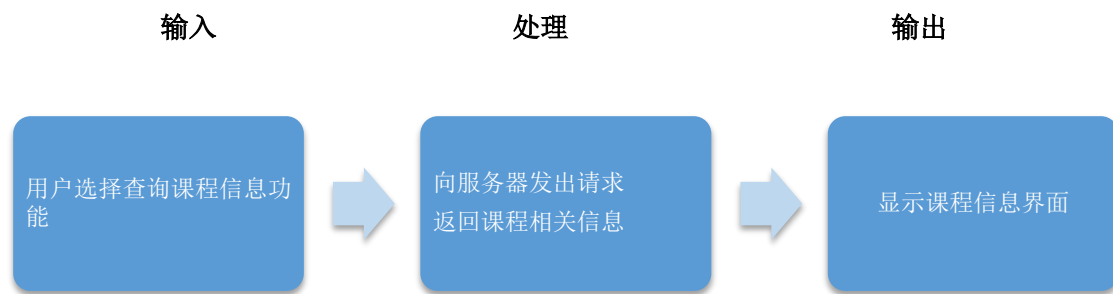
#### 用户添加/删除模块：



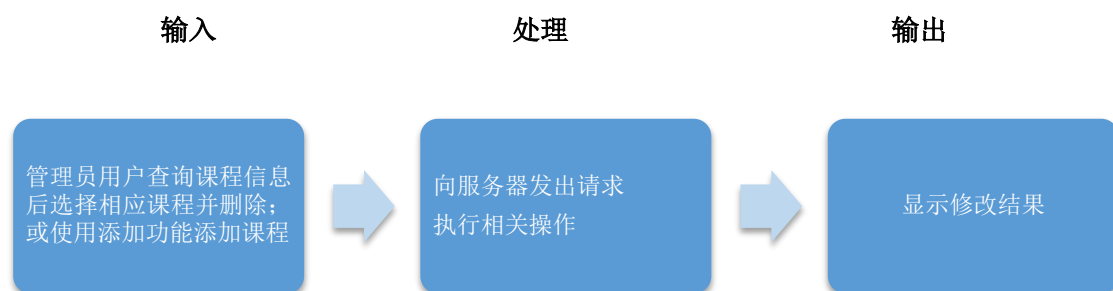
#### 个人信息查看模块：



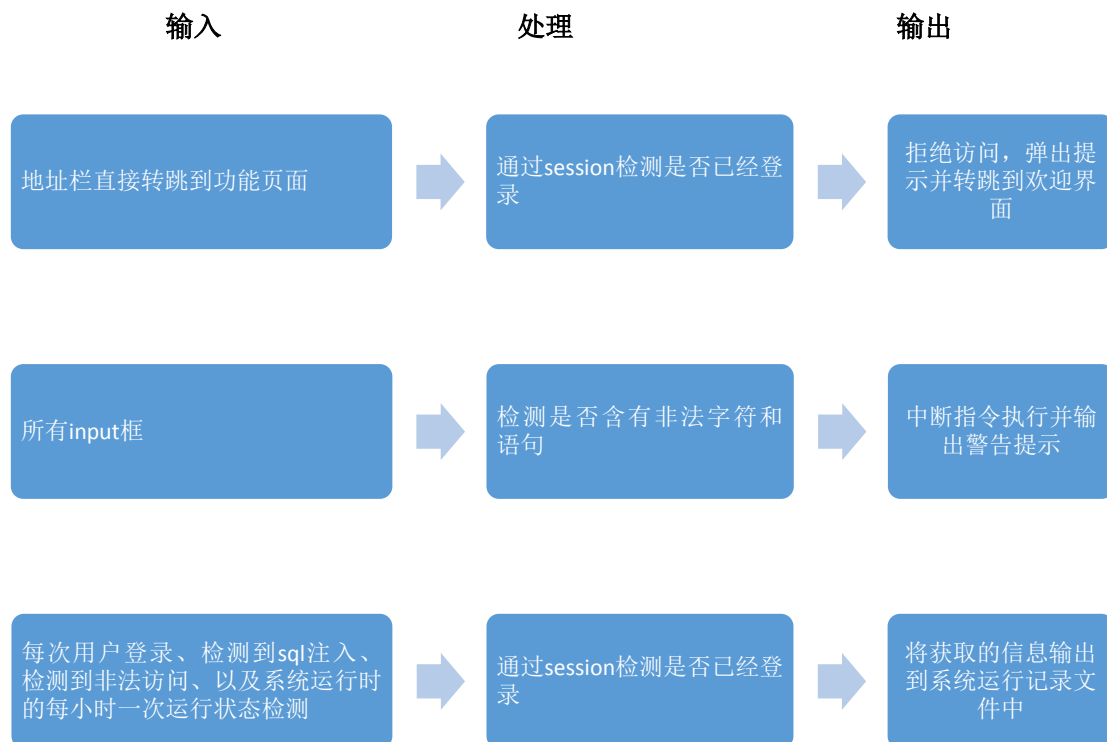
### 课程信息查询模块:



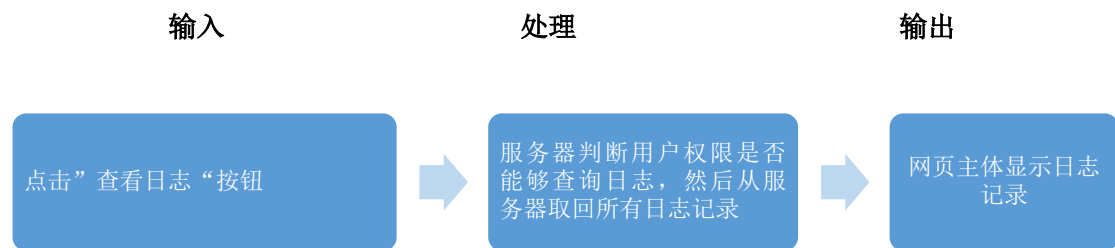
### 课程添加/删除模块:



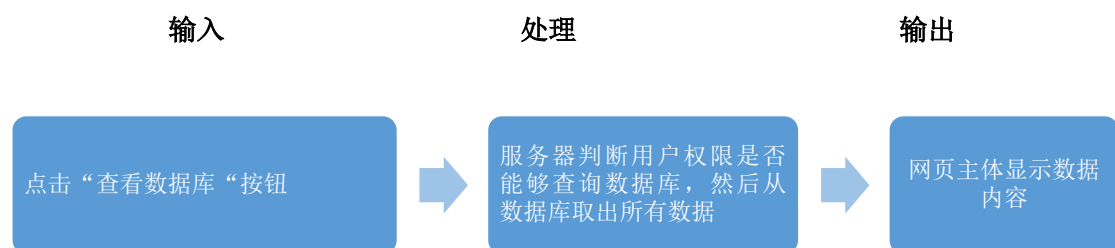
### 系统安全与运行日志模块:



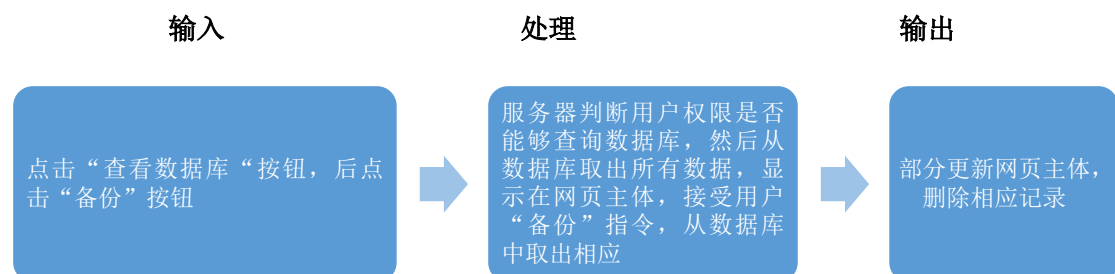
#### 查看系统日志模块:



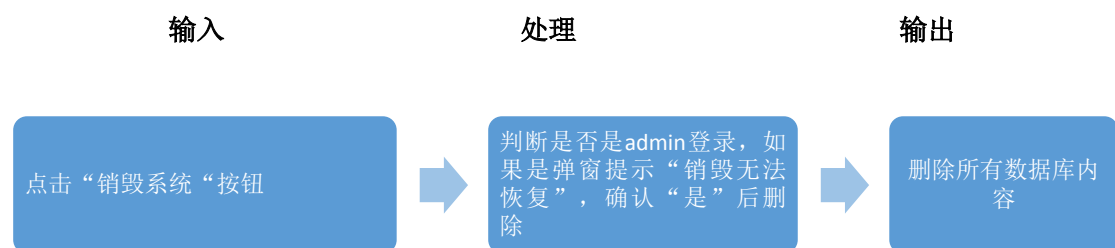
#### 查看数据库表格模块:



#### 备份数据模块:

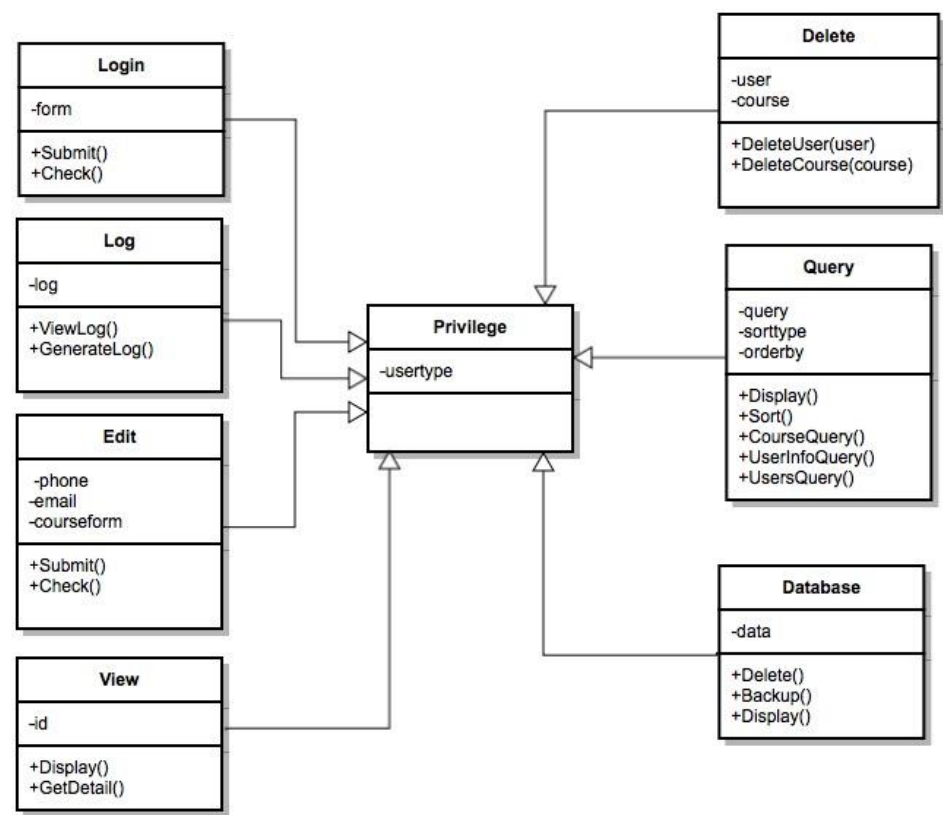


#### 一键销毁系统模块:





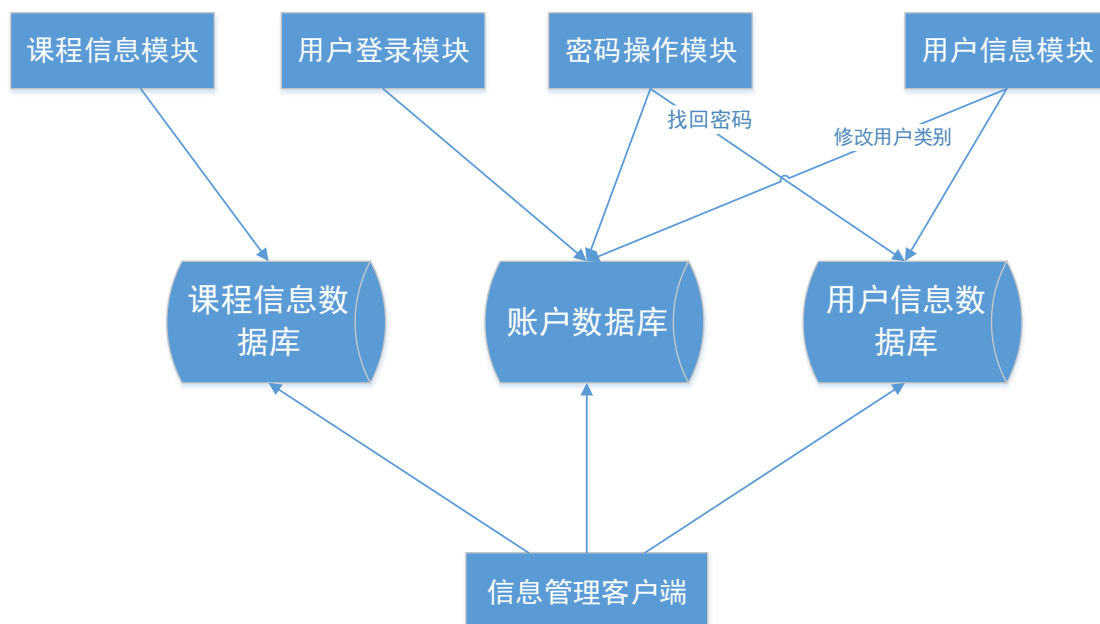
2.4.2 按处理流程划分



2.5 内部接口

本系统内不同模块通过共享数据库完成通信。

	用户信息模块	课程信息模块	密码操作模块
信息管理客户端	共享用户信息数据库 (user_info), 账户数据库 (accounts)	共享课程信息数据库 (course_info), 账户数据库 (accounts)	共享账户数据库 (accounts), 用户信息数据库 (user_info)



图表-各模块共享数据库表

## 2.6 人工处理过程

系统出现无法修复的错误，需要通过系统管理员查看系统日志，查看数据库，判断错误并进行修正。

## 2.7 尚未解决的问题

1. 为了系统与数据的安全，系统管理员有备份数据库的功能。但目前还不知道在 **PHP** 中以怎样的方式实现.sql 后缀文件的导出。
2. 为管理员设置了一键销毁系统的操作，但目前还不知道在 **PHP** 中如何以具体的代码实现这样的功能。
3. 现在设计的找回密码是通过邮箱和基本信息找回，但是这样的方式安全性有待提高。目前还不知道要怎样实现更安全的密码找回功能。

# 3.详细设计

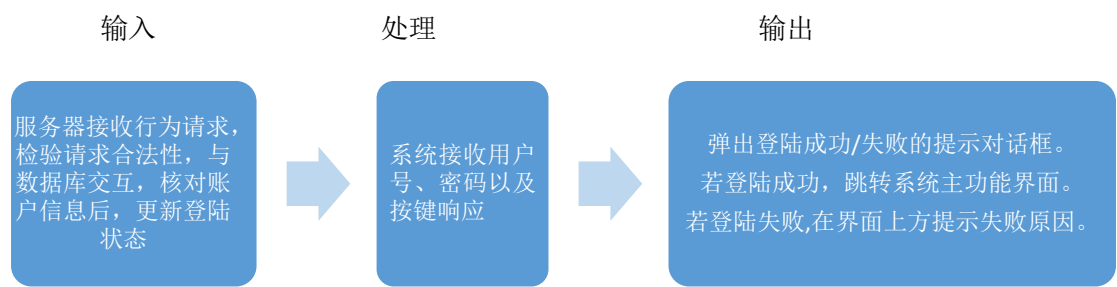
## 3.1 登录模块设计说明

### 3.1.1 模块描述

本模块以网页表单的形式显示，是用户使用客户端时继欢迎界面后的第一个输入界面，在本界面身份认证成功后，用户才能获得授权使用本系统的后续功能。目的是对用户的身份进行认证。

### 3.1.2 功能

IPO 图如下，



图表-登录模块 IPO 图

### 3.1.3 性能

根据用户输入的 ID、密码进行判断，如果输入的内容匹配则显示登陆成功，否则进行相应提示。

### 3.1.4 输入项

名称	标识	HTML 标签/数据类型	输入方式
用户号	user_id	<session>	/
用户密码	user_password	<session>	/
提交按钮	submit_bottom	<bottom>	鼠标单击

图表-登陆模块输入项表

### 3.1.5 输出项

名称	标识	类型	输出方式
提示信息（修改失败）	message	char(100)	警告框文字
主功能界面	main_page	page	新网页

图表-登陆模块输出项表

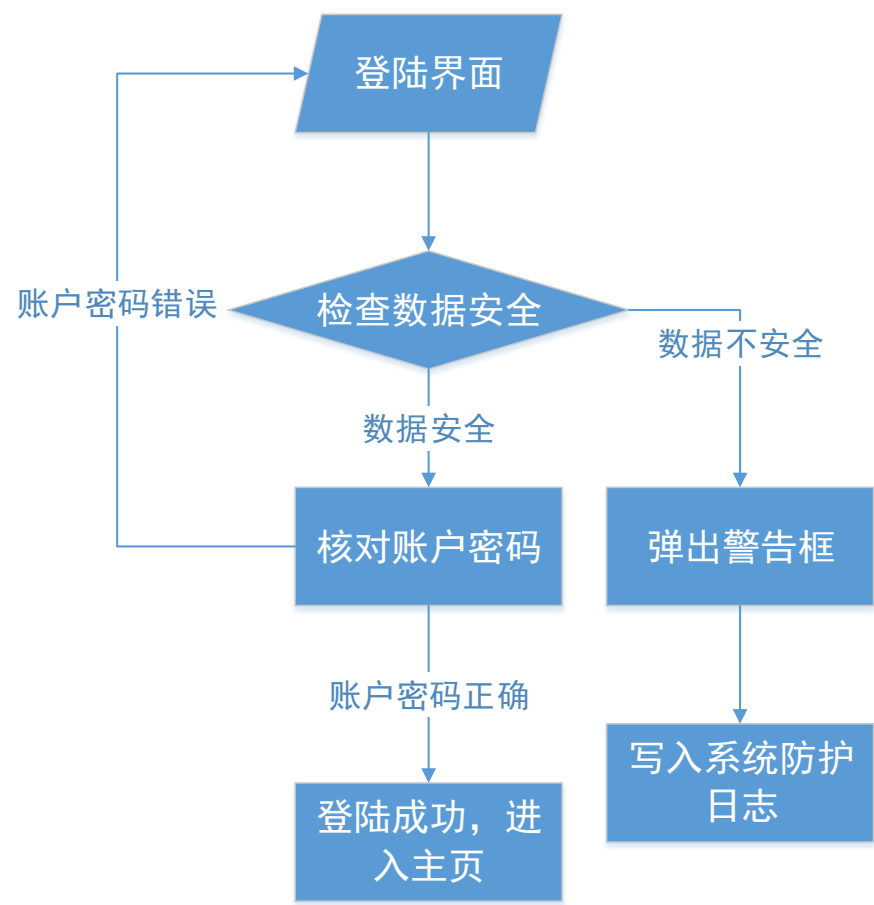
### 3.1.6 设计方法

当客户端点击登录时，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
If (! CheckSafty( id , password ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';
    SystemLogWrite( );
}
else//输入符合规范，无安全问题
{
    ConnectDB( );//连接数据库
    If (DBLoginQuery( id , password ) )
    {
        //若账号密码存在
        Setcookie;
        $home_url = 'logged.php';
        header('Location: '.$home_url);
    }
    else//若查到的记录不对，则设置错误信息
    {
        $error_msg = 'Sorry, you must enter a valid username and password to log in.';
    }
}
?>
```

而后客户端的函数被调用，将数据写入客户端。

3.1.7 流程逻辑



图表-登陆模块流程逻辑图

3.1.8 测试方法

测试要求通过远程计算机，创建相应用户账户信息数据库，并且添加测试数据。

输入数据	预期结果
表单输入正确的用户号、密码，有登陆按钮行为	跳转该用户对应权限的主功能界面
表单输入的用户号、密码其一有误	界面上方警告框提示出错信息
表单数据中包含 sql 语句	界面上方警告框提示出错信息

图表-登陆模块测试计划表

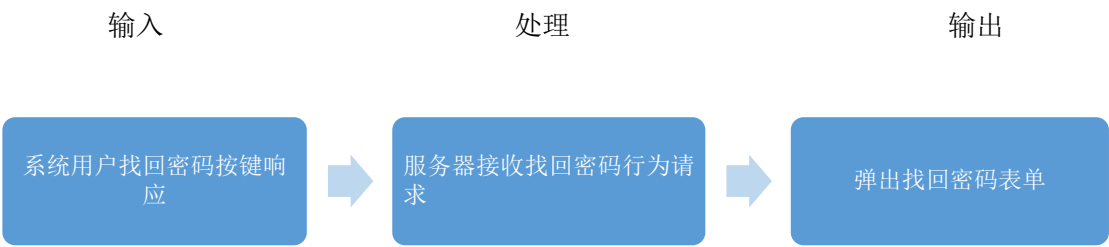
## 3.2 密码找回模块设计说明

### 3.2.1 模块描述

此模块以网页表单的形式显示，整合于登陆模块界面中，是为了使暂时回忆不出密码的用户有途径找回密码。

### 3.2.2 功能

IPO 图如下，



图表-密码找回模块 IPO 图

### 3.2.3 性能

用户可以通过回答验证问题来找回密码，若回答成功则返回密码。

### 3.2.4 输入项

名称	标识	HTML 标签/数据类型	输入方式
找回密码按钮	quiry_bottom	<bottom>	鼠标单击
验证问题答案	answer	char(45)	表格文字

图表-密码找回模块输入项表

### 3.2.5 输出项

名称	标识	类型	输出方式
找回密码逐步提示界面	quiry_page	form	网页表单
找回密码验证问题	message_1	char(45)	表单文字
成功找回密码提示信息	message_1	char(45)	表单文字
找回密码失败提示信息	message_2	char(45)	表单文字
原密码（若成功）	password	char(45)	表单文字

图表-密码找回模块输出项表

### 3.2.6 设计方法

当点击找回密码模块时，跳转到相关界面。输入邮箱和安全提问的答案后点击提交，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php

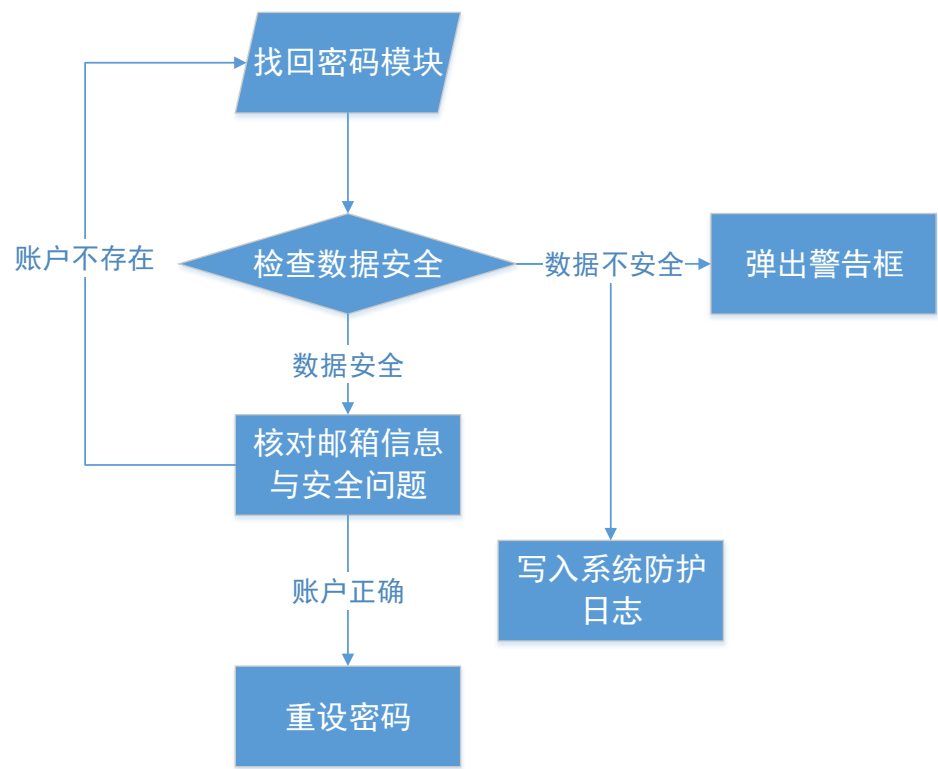
If (! CheckSafty( email , answer )) //若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';
    SystemLogWrite( );
}

else//输入符合规范，无安全问题
{
    ConnectDB( );//连接数据库
    If (ChecMatch( email , answer ) )
    {
        //若邮件和答案匹配
        EditPsd( email , newpsd );//修改密码
    }

    else//若查到的记录不对，则设置错误信息
    {
        $error_msg = 'Sorry, your email address does not exist or your answer is wrong.';
    }
}

?>
```

3.2.7 流程逻辑



图表-找回密码模块测试流程逻辑图

3.2.8 测试方法

与登录模块集成测试。

测试要求通过远程计算机，创建相应用户信息数据库，并且添加测试数据。

输入数据	预期结果
用户的找回密码行为请求	弹出找回密码逐步提示表单（上有验证问题）
用户的验证问题答案	系统判断用户输出答案正确/错误，显示相应提示信息。若答案正确，界面显示原有密码。

图表-找回密码模块测试计划表



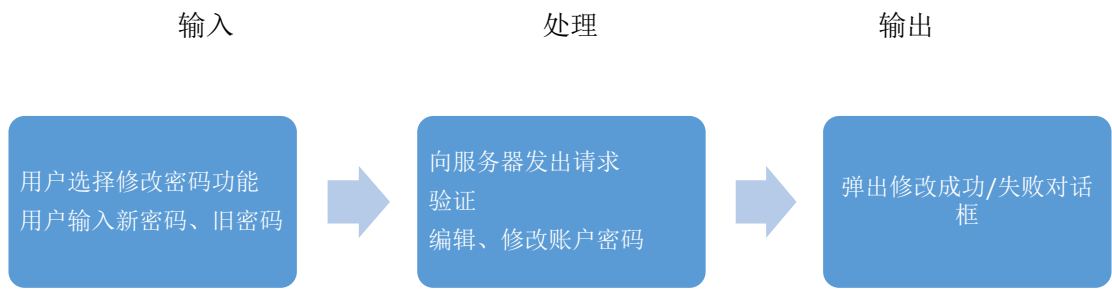
### 3.3 修改密码模块设计说明

#### 3.3.1 模块描述

此模块是用户身份认证后可以使用的模块，用户在提交正确的旧密码之后可以输入新密码给服务器进行修改。

#### 3.3.2 功能

IPO 图如下，



图表-修改密码模块 IPO 图

#### 3.3.3 性能

用户提交正确的旧密码后即可输入新密码进行修改。

#### 3.3.4 输入项

名称	标识	HTML 标签	输入方式
旧密码	old_password	string	文本框输入
新密码	new_passworld	string	文本框输入
修改	modify_button	<button>	鼠标点击

图表-修改密码模块输入项图

### 3.3.5 输出项

名称	标识	类型	输出方式
修改结果	modify_result	enum{修改成功, 修改失败}	对话框

图表-修改密码模块输出项图

### 3.3.6 设计方法

当点击修改密码模块时,跳转到相关界面。输入旧密码以及两遍新密码点击提交,触发脚本,将信息提交给服务器,服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录,则继续以下语句;未登录则跳转至登陆界面

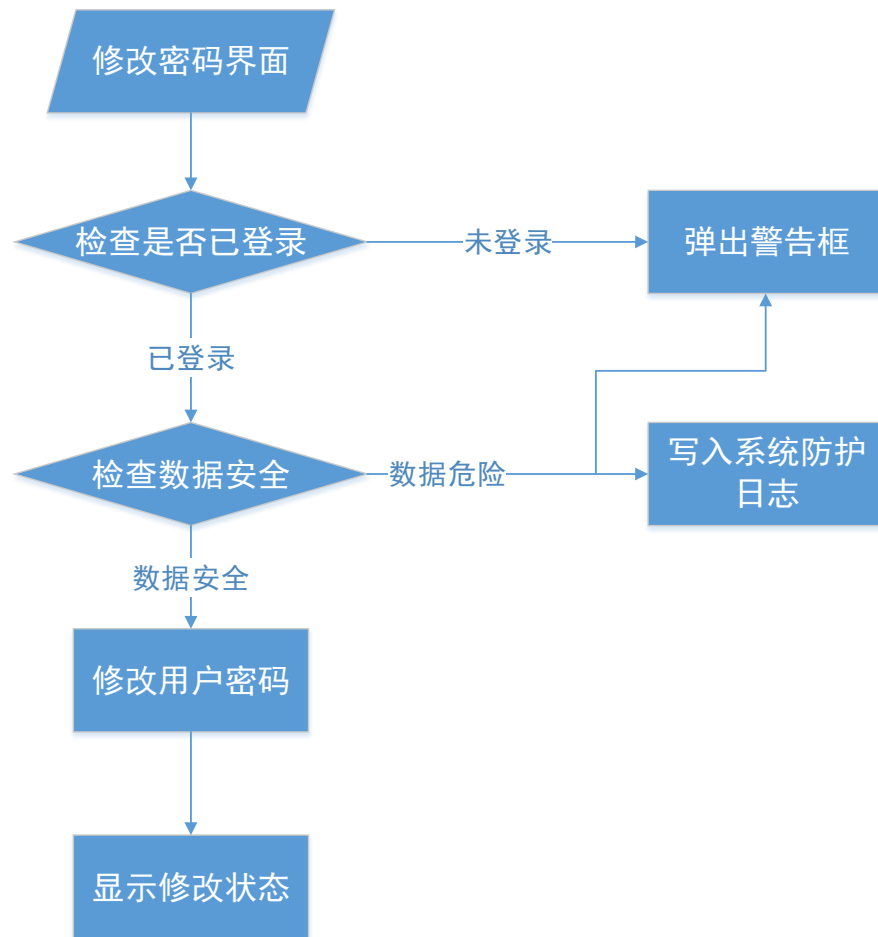
if(! CheckSame (new_psd1 , new_psd2 ))//若两次输入新密码不相同
    $error_msg = 'Please enter the same password.';
else//两次输入密码相同
{
    if (! CheckSafty( old_psd , new_psd1 , new_psd2 )) //若输入不符合规范,存在安全问题
    {
        $error_msg = ' Your behavior is malicious.';
        SystemLogWrite( );
    }
    else//输入符合规范,无安全问题
    {
        ConnectDB( );//连接数据库

        EditPsd($_SESSION["user_id"],newpsd );//修改密码
    }
    else//若查到的记录不对,则设置错误信息

        $error_msg = 'Sorry, your email address does not exist or your answer is wrong.';
}
```

```
}  
?>
```

### 3.3.7 流程逻辑



图表-修改密码模块流程逻辑图

### 3.3.8 测试计划

输入数据	预期结果
正确的旧密码，两次相同的新密码	弹窗提示修改成功
正确的旧密码，两次不同的新密码	弹窗提示新密码不匹配
错误的旧密码	弹窗提示旧密码错误
用户尚未登录	拒绝执行操作，返回登录界面
数据中包含 sql 语句或特殊字符	拒绝执行该指令，并弹出警告

图表-修改密码模块测试计划表

## 3.4 修改用户信息模块设计说明

### 3.4.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 manager/system administrator 用户可使用的功能模块，整合于查询用户信息界面中，存在目的是让 manager/system administrator 得以修改 student/teacher(/manager)用户的个人信息。

### 3.4.2 功能

IPO 图如下，



图表-修改用户信息模块 IPO 图

### 3.4.3 性能

满足权限要求的用户(manager/system administrator)只要选择在每条用户信息记录右侧的修改用户信息功能按钮, 按键点击, 数据库中就可弹出默认值为用户原信息的表单。系统可接收用户在表单中的修改内容, 更新数据库, 并在原页面下方显示修改更新后的结果。

### 3.4.4 输入项

(斜体为显示用户信息模块对修改用户功能结果的呈现方式)

名称	标识	HTML 标签/数据类型	输入方式
用户号	user_id	<session>	/
用户类型	user_type	<session>	/
修改按钮	alter_bottom	<bottom>	鼠标单击
<i>用户号</i>	<i>user_id</i>	<i>char(20)</i>	<i>表格文字</i>
<i>用户姓名</i>	<i>user_name</i>	<i>varchar(45)</i>	<i>表格文字</i>
<i>用户性别</i>	<i>gender</i>	<i>char(1)</i>	<i>表格文字</i>
<i>出生日期</i>	<i>birth</i>	<i>date</i>	<i>表格文字</i>
<i>用户年龄</i>	<i>age</i>	<i>tinyint(4)</i>	<i>表格文字</i>
<i>入学日期</i>	<i>enroll_time</i>	<i>char(4)</i>	<i>表格文字</i>
<i>所属院系</i>	<i>department</i>	<i>varchar(45)</i>	<i>表格文字</i>
<i>联系号码</i>	<i>phone</i>	<i>char(15)</i>	<i>表格文字</i>
<i>联系邮箱</i>	<i>email</i>	<i>varchar(45)</i>	<i>表格文字</i>

图表-修改用户信息模块输入项表

### 3.4.5 输出项

(斜体为显示用户信息模块对修改用户功能结果的呈现方式)

名称	标识	类型	输出方式
提示信息 (修改成功/失败)	message	char(45)	对话框文字
<i>用户号</i>	<i>user_id</i>	<i>char(20)</i>	<i>表格文字</i>

用户姓名	<i>user_name</i>	<i>varchar(45)</i>	表格文字
用户性别	<i>gender</i>	<i>char(1)</i>	表格文字
出生日期	<i>birth</i>	<i>date</i>	表格文字
用户年龄	<i>age</i>	<i>tinyint(4)</i>	表格文字
入学日期	<i>enroll_time</i>	<i>char(4)</i>	表格文字
所属院系	<i>department</i>	<i>varchar(45)</i>	表格文字
联系号码	<i>phone</i>	<i>char(15)</i>	表格文字
联系邮箱	<i>email</i>	<i>varchar(45)</i>	表格文字

图表-修改用户信息模块输出项表

### 3.4.6 设计方法

当点击编辑用户信息时，跳转到相应界面，输入用户 id 并点击提交后，查看到个人信息，点击编辑相关信息后，相关信息变为可编辑的文本框，点击确认按钮触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限

if (! CheckSafty( user_info ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';

    SystemLogWrite( );
}

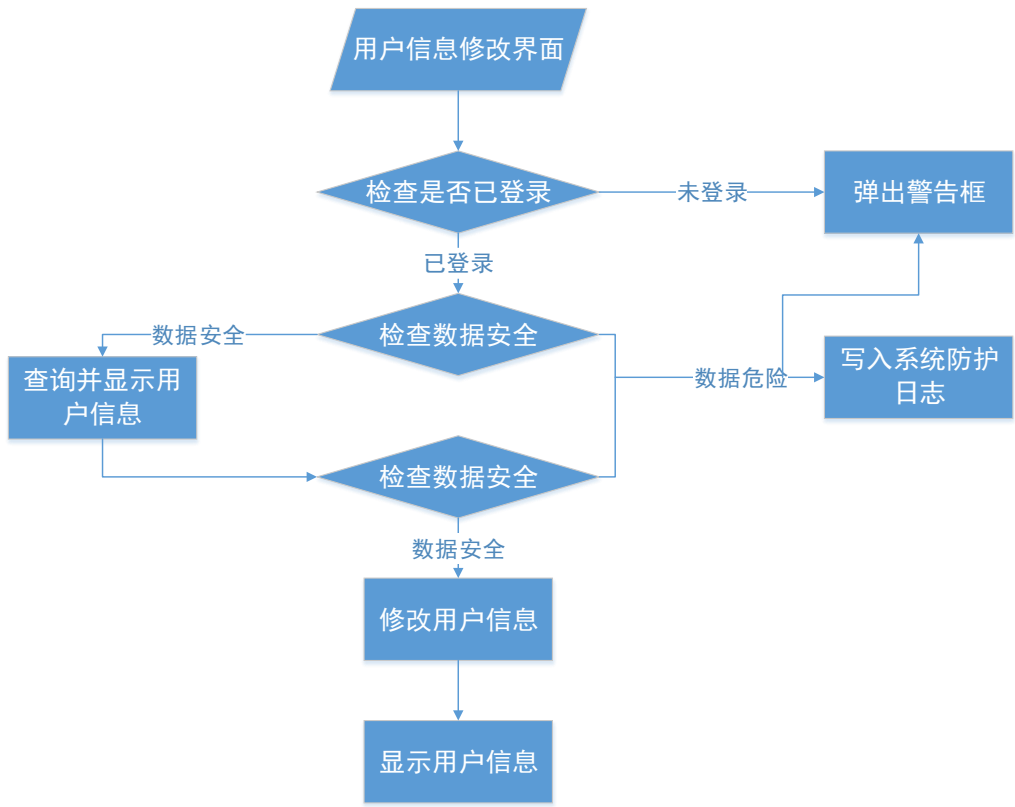
else//输入符合规范，无安全问题
{
    ConnectDB( );//连接数据库

    ModInfoQuery( info_type , user_info )//编辑个人信息
}

?>
```

而后客户端的函数被调用，将数据写入客户端。

3.4.7 流程逻辑



图表-修改用户信息模块流程逻辑图

3.4.8 测试方法

与查询用户信息模块集成测试。

测试要求通过远程计算机，创建相应用户信息数据库，并且添加测试数据。

输入数据	预期结果
session 用户类型为 manager/system administrator	在查询用户信息结果的每条记录右侧显示修改用户个人信息功能按钮
session 用户类型为 student/teacher	提示“无权限修改用户个人信息”
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面
鼠标对修改用户个人信息按钮的点击	弹出表单，默认值为该用户原来的个人信息
表单数据中包含 sql 语句或不合法输入	拒绝执行该指令并弹出提示警告

表单输入了合法的用户个人信息	在数据库中修改该用户个人信息，并至查询用户信息部分显示
----------------	-----------------------------

图表-修改用户信息模块测试计划表

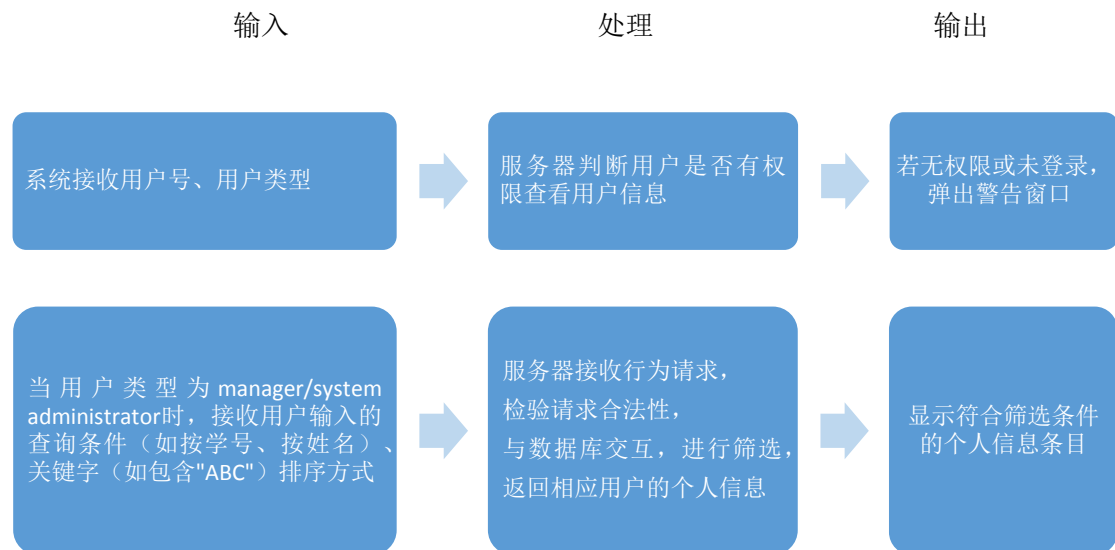
## 3.5 查看用户信息模块设计说明

### 3.5.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 `manager/system administrator` 用户可使用的功能模块，通过网页表单传送数据，存在目的是查询系统内各用户信息。

### 3.5.2 功能

IPO 图如下，



图表-查看用户信息模块 IPO 图

### 3.5.3 性能

只要选择查询条件，输入查询关键词和排序方式，按键点击，原页面下方就可按照选择的排列顺序显示符合筛选条件的个人信息条目。



### 3.5.4 输入项

名称	标识	HTML 标签	输入方式
用户号	user_id	<session>	/
用户类型	user_type	<session>	/
查询条件	condition	<select>	在下拉菜单中选择
查询关键词	keyword	<input>	在文本框输入
排序方式	order	<select>	在下拉菜单中选择

图表-查看用户信息模块输入项表

### 3.5.5 输出项

名称	标识	类型	输出方式
用户号	user_id	char(20)	表格文字
用户姓名	user_name	varchar(45)	表格文字
用户性别	gender	char(1)	表格文字
出生日期	birth	date	表格文字
用户年龄	age	tinyint(4)	表格文字
入学日期	enroll_time	char(4)	表格文字
所属院系	department	varchar(45)	表格文字
联系号码	phone	char(15)	表格文字
联系邮箱	email	varchar(45)	表格文字

图表-查看用户信息模块输出项表

### 3.5.6 设计方法

当点击查看用户信息时，跳转到相应界面，输入用户 id 并点击提交后，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
```

```
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限

ConnectDB( );

$User_info = SelectPersonInfo( user_id );

//在数据库中查询用户相关信息，将查询结果返回一个包含个人详细信息对象

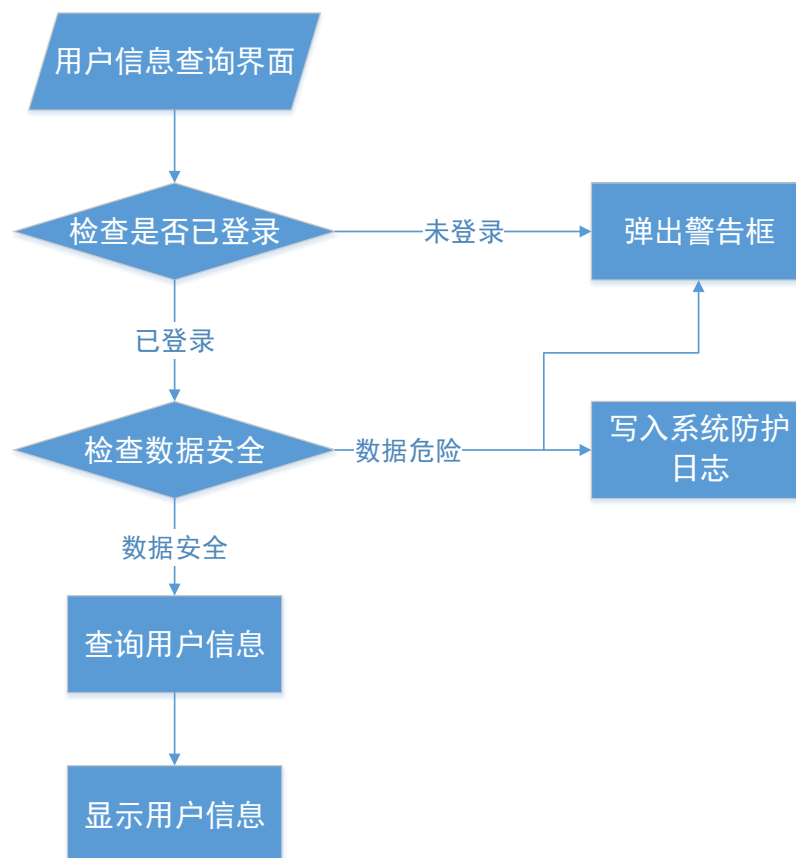
echo $User_info ;

//返回结果

?>
```

而后客户端的函数被调用，将数据写入客户端。

### 3.4.7 流程逻辑



图表-查看用户信息模块流程逻辑图

### 3.5.8 测试计划

与用户登录模块集成测试。

测试要求通过远程计算机，创建相应用户信息数据库，并且添加测试数据。

输入数据	预期结果
数据库中存在的用户号	返回相关用户信息并按所选排序方式返回
数据库中不存在的用户号	提示“未搜索到相关用户”
数据库中存在的用户姓名	返回相关用户信息并按所选排序方式返回
数据库中不存在的用户姓名	提示“未搜索到相关用户”
数据库中存在的用户入学日期	返回相关用户信息并按所选排序方式返回
数据库中不存在的用户入学日期	提示“未搜索到相关用户”
数据库中存在的用户所属院系	返回相关用户信息并按所选排序方式返回
数据库中不存在的用户所属院系	提示“未搜索到相关用户”
数据中包含 sql 语句或特殊字符	拒绝执行该指令并弹出警告
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

图表-查看用户信息模块测试计划表

## 3.6 用户添加/删除模块设计说明

### 3.6.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 manager/system administrator 用户可使用的功能模块，整合于查询用户信息界面中，存在目的是让 manager/system administrator 得以添加或删除 student/teacher(/manager)用户。

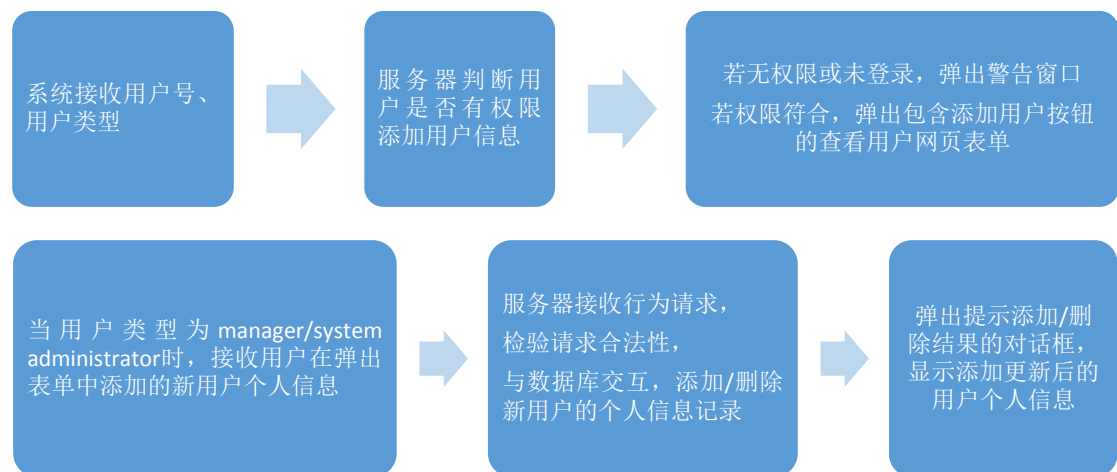
### 3.6.2 功能

IPO 图如下，

输入

处理

输出



图表-用户添加/删除模块 IPO 图

### 3.6.3 性能

满足权限要求的用户(manager/system administrator)只要选择添加用户功能，在弹出表单中输入新用户个人信息，按键点击，数据库中就可添加一条新用户的个人信息记录，并在原页面下方增加显示；同理，当选择删除功能时，在数据库中删除对应的用户信息。

### 3.6.4 输入项

名称	标识	HTML 标签/数据类型	输入方式
用户号	user_id	<session>	/
用户类型	user_type	<session>	/
用户号	user_id	char(20)	表格文字
用户姓名	user_name	varchar(45)	表格文字
用户性别	gender	char(1)	表格文字
出生日期	birth	date	表格文字
用户年龄	age	tinyint(4)	表格文字
入学日期	enroll_time	char(4)	表格文字
所属院系	department	varchar(45)	表格文字
联系号码	phone	char(15)	表格文字

联系邮箱	email	varchar(45)	表格文字
------	-------	-------------	------

图表-用户添加/删除模块输入项表

### 3.6.5 输出项

(斜体为显示用户信息模块对添加用户功能结果的呈现方式)

名称	标识	类型	输出方式
添加结果 (成功/失败)	adduser_result	Enum{添加成功, 添加失败}	对话框文字
删除信息 (成功/失败)	deleteuser_result	Enum{删除成功, 删除失败}	对话框文字
<i>用户号</i>	<i>user_id</i>	<i>char(20)</i>	<i>表格文字</i>
<i>用户姓名</i>	<i>user_name</i>	<i>varchar(45)</i>	<i>表格文字</i>
<i>用户性别</i>	<i>gender</i>	<i>char(1)</i>	<i>表格文字</i>
<i>出生日期</i>	<i>birth</i>	<i>date</i>	<i>表格文字</i>
<i>用户年龄</i>	<i>age</i>	<i>tinyint(4)</i>	<i>表格文字</i>
<i>入学日期</i>	<i>enroll_time</i>	<i>char(4)</i>	<i>表格文字</i>
<i>所属院系</i>	<i>department</i>	<i>varchar(45)</i>	<i>表格文字</i>
<i>联系号码</i>	<i>phone</i>	<i>char(15)</i>	<i>表格文字</i>
<i>联系邮箱</i>	<i>email</i>	<i>varchar(45)</i>	<i>表格文字</i>

图表-用户添加/删除模块输出项表

### 3.6.6 设计方法

添加用户

当点击添加用户时, 跳转到相应界面, 输入添加用户所需的用户信息并点击提交后, 触发脚本, 将信息提交给服务器, 服务器执行以下 PHP 代码。

```
<?php
```

```

include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限

if (! CheckSafty( user_id,user_name, and so on ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';

    SystemLogWrite( );
}

else//输入符合规范，无安全问题
{

    ConnectDB( );//连接数据库

    AddUser( user_id , user_name ,and so on )//添加用户
}

?>

```

## 删除用户

当点击删除用户时，跳转到相应界面，输入相关查询信息并点击提交后，返回用户列表，点击列表后的删除按钮触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```

<?php

include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限

if (! CheckSafty( user_info ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';

    SystemLogWrite( );
}

else//输入符合规范，无安全问题
{

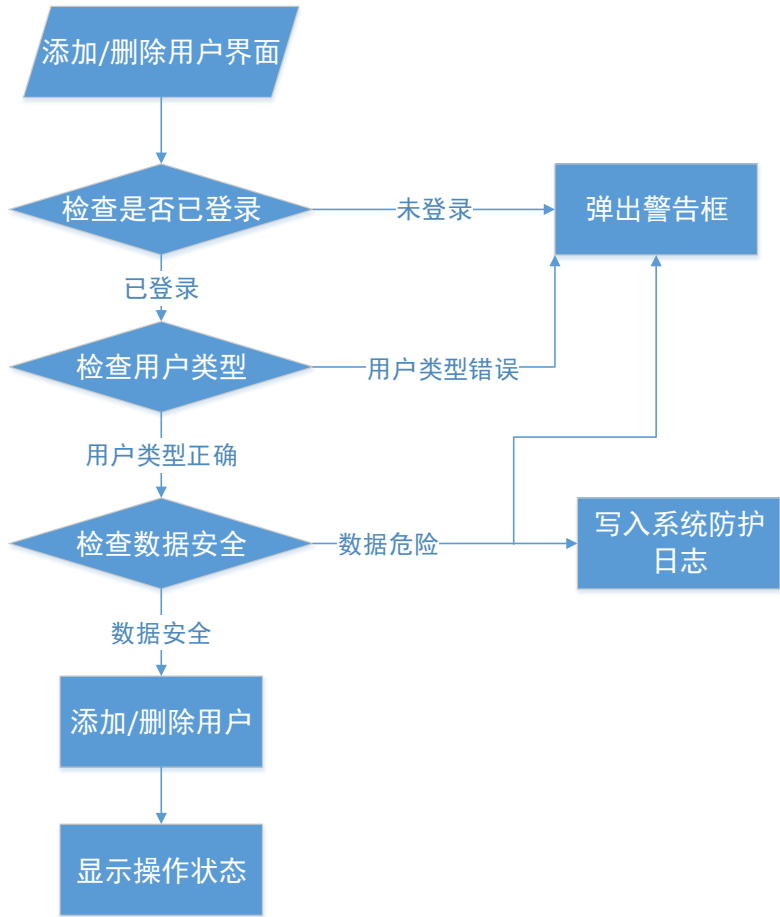
    ConnectDB( );//连接数据库

    DeleteUser( user_id)//删除用户
}

```

?>

3.6.7 流程逻辑



图表-用户添加/删除模块流程逻辑图

3.6.8 测试方法

与查询用户信息模块集成测试。

测试要求通过远程计算机，创建相应用户信息数据库，并且添加测试数据。

输入数据	预期结果
session 用户类型为 manager/system administrator	弹出添加新用户个人信息表单
session 用户类型为 student/teacher	提示“无权限添加用户”
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

数据中包含 sql 语句或不合法输入	拒绝执行该指令并弹出提示警告
合法的用户个人信息	在数据库中添加新用户个人信息，并至查询用户信息部分显示

图表-添加用户模块测试计划表

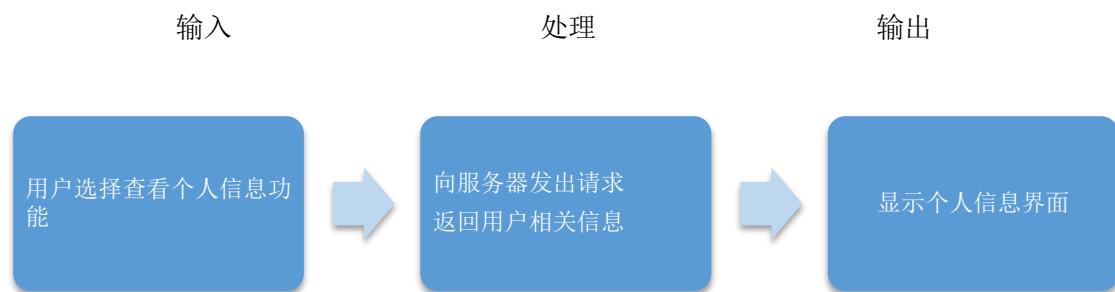
## 3.7 个人信息查看模块设计说明

### 3.7.1 模块描述

本模块是所有用户登陆后可以使用的功能模块，通过网页表单传送数据，实现目的是查看个人资料信息。

### 3.7.2 功能

IPO 图如下，



图表-课程信息修改模块 IPO 图

### 3.7.3 性能

登陆后进入个人界面，点击“查看”按钮即可查看相关信息。



### 3.7.4 输入项

名称	标识	html 标签	输入方式
查看	check	<button>	鼠标点击

图表-个人信息查看模块输入项表

### 3.7.5 输出项

名称	标识	类型	输出方式
用户号	user_id	char(20)	表格文字
用户姓名	user_name	varchar(45)	表格文字
用户性别	gender	char(1)	表格文字
出生日期	birth	date	表格文字
用户年龄	age	tinyint(4)	表格文字
入学日期	enroll_time	char(4)	表格文字
所属院系	department	varchar(45)	表格文字
联系号码	phone	char(15)	表格文字
联系邮箱	email	varchar(45)	表格文字

图表-用户添加/删除模块输出项表

### 3.7.6 设计方法

当点击查看个人信息时，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

ConnectDB( );

$self_info = SelectPersonInfo($_SESSION["user_id"]);

//在数据库中查询个人相关信息，将查询结果返回一个包含个人详细信息对象
```

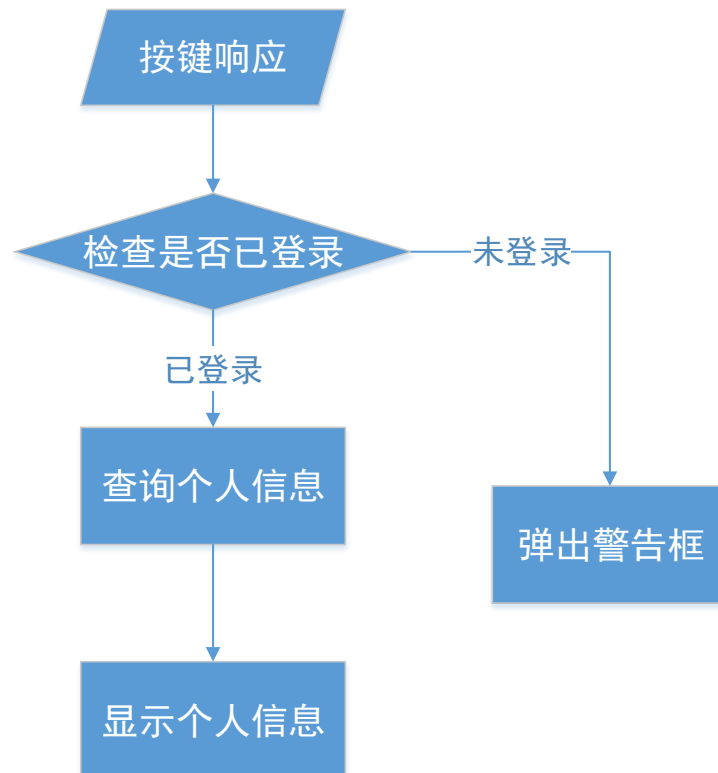
```
echo $Self_info ;
```

```
//返回结果
```

```
?>
```

而后客户端的函数被调用，将数据写入客户端。

### 3.7.7 流程逻辑



### 3.7.8 测试计划

与用户登录模块集成测试。

测试要求通过远程计算机，创建相应课程信息数据库，并且添加测试数据。

输入数据	预期结果
点击“查看”按钮	显示相应用户信息
数据中包含 sql 语句或特殊字符	拒绝执行该指令并弹出警告
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

图表-系统安全与运行日志模块测试计划表

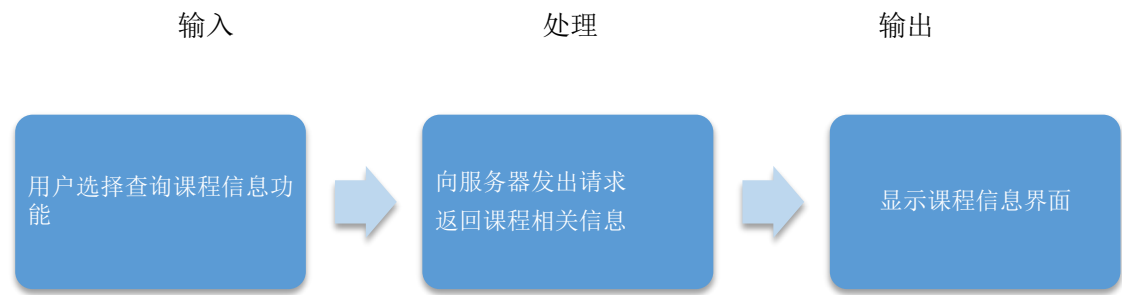
### 3.8 课程信息查询模块设计说明

#### 3.8.1 模块描述

本模块是所有用户登陆后可以使用的功能模块，通过网页表单传送数据，实现目的是显示相应的课程信息。

#### 3.8.2 功能

IPO 图如下，



图表-课程信息查询模块 IPO 图

#### 3.8.3 性能

在课程查看界面输入课程的课程号或名称等信息，点击按钮即可查看相关信息。

#### 3.8.4 输入项

名称	标识	html 标签	输入方式
课程号	cid	<input>	在文本框输入
课程名称	cname	<input>	在文本框输入

图表-课程信息查询模块输入项表

### 3.8.5 输出项

名称	标识	类型	输出方式
课程号	cid	char(20)	表格文字
课程名称	cname	varchar(60)	表格文字
课程学分	credit	float(2,1)	表格文字
开课院系	cdepartment	varchar(40)	表格文字
课程介绍	course_intro	blob	表格文字

图表-用户添加/删除模块输出项表

### 3.8.6 设计方法

当点击查看课程信息时，跳转到相应界面，输入课程 id 并点击提交后，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面

ConnectDB( );

$Class_info = SelectClassInfo( user_id );

//在数据库中查询用户相关信息，将查询结果返回一个包含课程详细信息对象

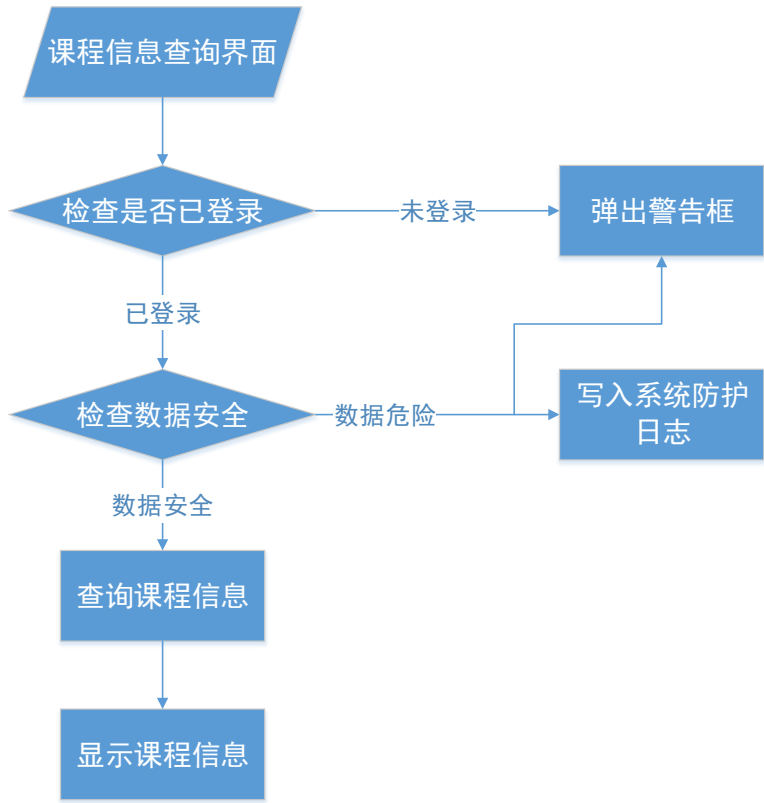
echo $Class_info ;

//返回结果

?>
```

而后客户端的函数被调用，将数据写入客户端。

3.8.7 流程逻辑



3.8.8 测试计划

与用户登录模块集成测试。

测试要求通过远程计算机，创建相应课程信息数据库，并且添加测试数据。

输入数据	预期结果
数据库中存在的课程号	显示相应课程信息
数据库中存在的课程名称	显示相应课程信息
数据库中不存在的课程号	弹出对话框提示课程不存在
数据库中不存在的课程名称	弹出对话框提示课程不存在
数据中包含 sql 语句或特殊字符	拒绝执行该指令并弹出警告
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

图表-系统安全与运行日志模块测试计划表

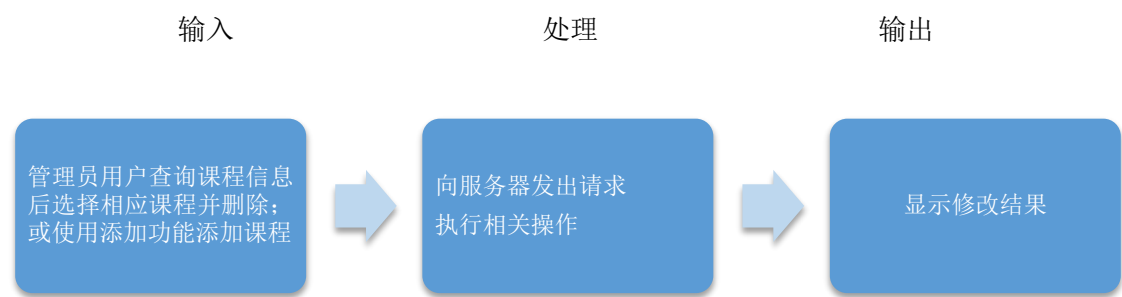
### 3.9 课程添加/删除模块设计说明

#### 3.9.1 模块描述

本模块是用户以 `manager` 类型登陆之后可以使用的功能模块，可以在课程界面实现相应课程的添加和删除。

#### 3.9.2 功能

IPO 图如下，



图表-课程添加/删除模块 IPO 图

#### 3.9.3 性能

在列表选定课程，点击“删除”按钮即可实现课程的删除；点击“添加”按钮后可添加新的课程。

#### 3.9.4 输入项

（斜体字为课程查询模块的输入）

名称	标识	html 标签	输入方式
<i>课程号</i>	cid	<input>	在文本框输入
<i>课程名称</i>	cname	<input>	在文本框输入
鼠标点击	click	<button>	鼠标点击

图表-课程添加/删除模块输入项表

### 3.9.5 输出项

名称	标识	类型	输出方式
删除结果	deletecourse_result	Enum{删除成功，删除失败}	对话框
添加结果	addcourse_result	Enum{添加成功，添加失败}	对话框

图表-课程添加/删除模块输入项表

### 3.9.6 设计方法

添加课程：

当点击添加课程时，跳转到相应界面，输入添加课程所需的课程信息并点击提交后，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面
CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限
If (! CheckSafty( class_id,class_name, and so on ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';
    SystemLogWrite( );
}
else//输入符合规范，无安全问题
{
    ConnectDB( )//连接数据库
    AddUser( class_id , class_name ,and so on )//添加课程
}
?>
```

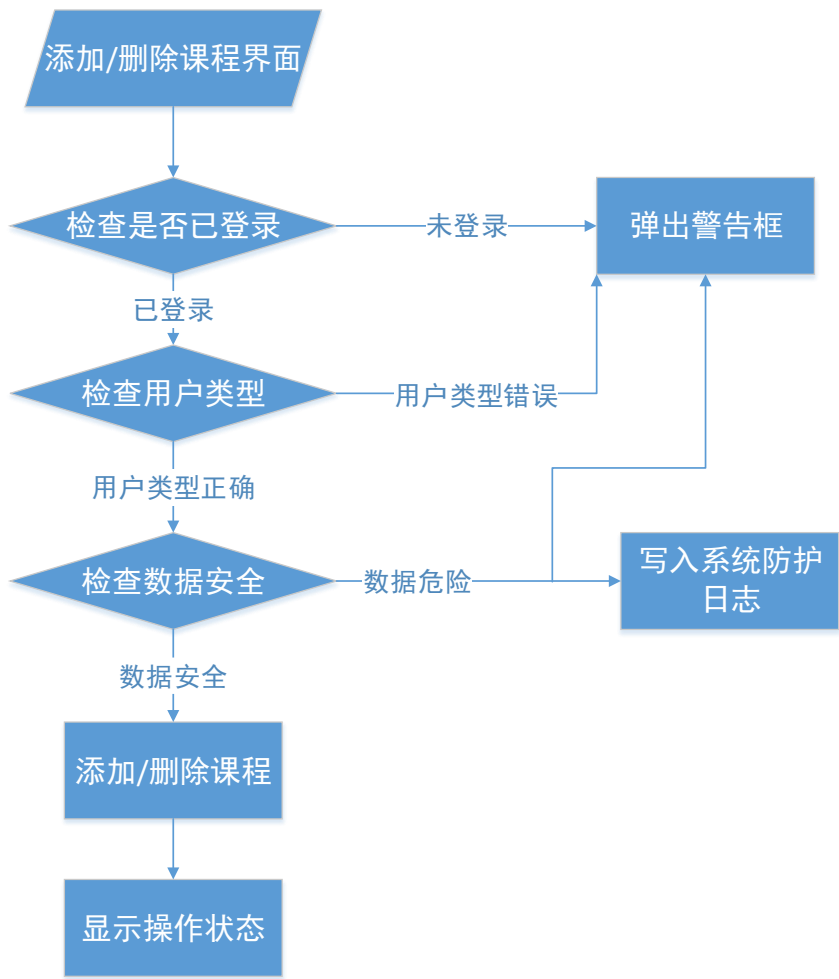
## 删除课程

当点击删除课程时，跳转到相应界面，输入相关查询信息并点击提交后，返回课程列表，点击列表后的删除按钮触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';
//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面
CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限
If (! CheckSafty( class_info ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';
    SystemLogWrite( );
}
else//输入符合规范，无安全问题
{
    ConnectDB( )//连接数据库
    DeleteClass( class_id)//删除用户
}
?>
```



3.9.7 流程逻辑



3.9.8 测试计划

与用户登录模块、课程查询模块集成测试。

测试要求通过远程计算机，创建相应课程信息数据库，并且添加测试数据。

输入数据	预期结果
选定课程后点击“删除”按钮	对话框提示“删除成功”
点击“添加”按钮	对话框提示“添加成功”
添加的课程数据中包含 sql 语句或特殊字符	拒绝执行该指令并弹出警告
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

图表-课程添加/删除模块测试计划表

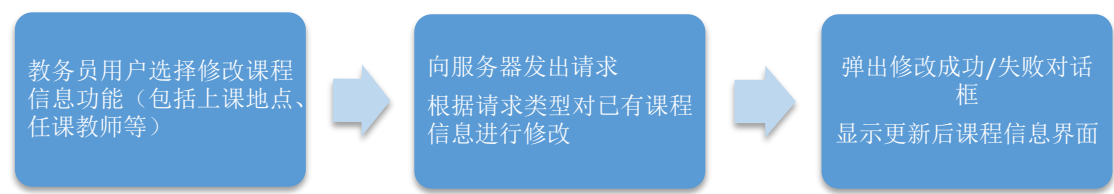
### 3.10 课程信息修改模块设计说明

#### 3.10.1 模块描述

本模块是用户以 manager 身份登录后可以使用的功能模块，通过网页表单传送数据，实现目的是修改相应的课程信息。

#### 3.10.2 功能

IPO 图如下，



图表-课程信息修改模块 IPO 图

#### 3.10.3 性能

在课程列表内选定需要修改的课程，在相应内容的文本框内输入更新后的信息，点击确认修改即可。

#### 3.10.4 输入项

名称	标识	HTML 标签	输入方式
课程号	cid	<input>	在文本框输入
课程名称	cname	<input>	在文本框输入
课程学分	credit	<select>	在下拉菜单中选择
开课院系	cdepartment	<select>	在下拉菜单中选择
课程介绍	Course_intro	<input>	在文本框输入
修改	modify_button	<button>	鼠标点击

图表-课程信息修改模块输入项表

3.10.5 输出项

名称	标识	类型	输出方式
修改结果	modify_result	Enum{修改成功, 修改失败}	对话框

图表-用户添加/删除模块输入项表

3.10.6 设计方法

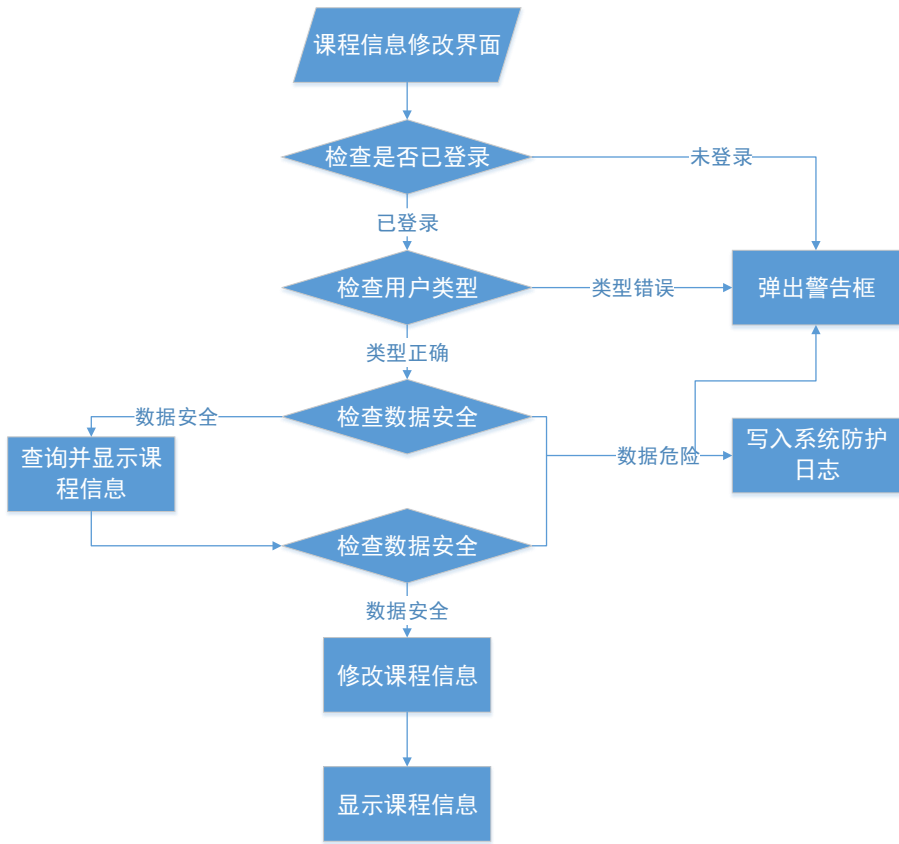
当点击编辑课程信息时，跳转到相应界面，输入课程 id 并点击提交后，查看到课程信息，点击编辑相关信息后，相关信息变为可编辑的文本框，点击确认按钮触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面
CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限
if (! CheckSafty( user_info ))//若输入不符合规范，存在安全问题
{
    $error_msg = ' Your behavior is malicious.';
    SystemLogWrite( );
}
else//输入符合规范，无安全问题
{
    ConnectDB( );//连接数据库
    ModInfoQuery( info_type , class_info )//编辑个人信息
}
?>
```

而后客户端的函数被调用，将数据写入客户端。

3.10.7 流程逻辑



3.10.8 测试计划

与用户登录模块集成测试。

测试要求通过远程计算机，创建相应课程信息数据库，并且添加测试数据。

输入数据	预期结果
符合要求的修改输入	对话框提示“修改成功”
数据库中不存在的用户号	提示 “未搜索到相关用户”
数据中包含 sql 语句或特殊字符	拒绝执行该指令并弹出警告
用户尚未登录，绕过登录窗口访问此模块	拒绝执行用户指令并返回登陆界面

图表-课程信息修改模块测试计划表

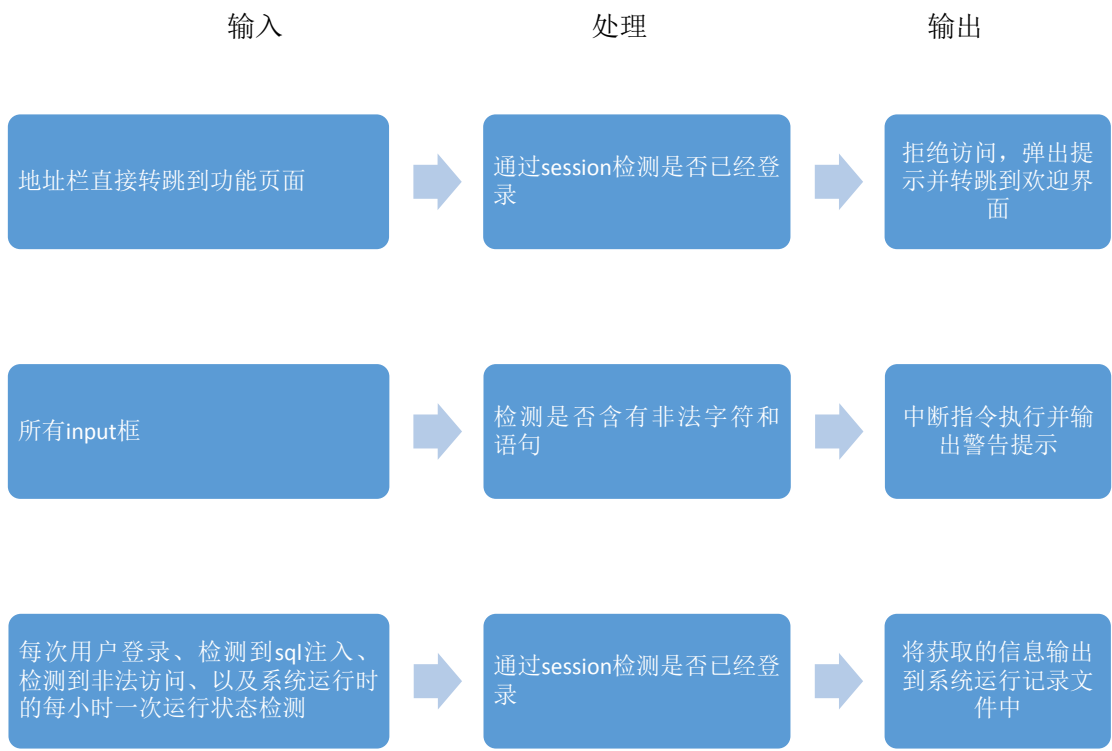
### 3.11 系统安全与运行日志模块设计说明

#### 3.11.1 模块描述

此模块穿插与各个基础功能模块之中，主要目的是防止非法访问、sql 注入以及系统运行日志的记录。防止非法访问是指在未登录的情况下直接在地址栏转跳到功能模块，此时系统会拒绝访问并输出提示。防止 sql 注入是指检测 input 输入中的特殊字符，当有非法字符或语句出现的时候拒绝执行并输出提示。系统运行日志的记录会在用户登录的时候、上述两种情况发生的时候将他们写入系统日志文件。同时系统运行日志每小时一次将系统运行情况写入系统日志，用以之后系统管理员分析系统运行状况。

#### 3.11.2 功能

IPO 图如下，



图表-系统安全与运行日志模块 IPO 图



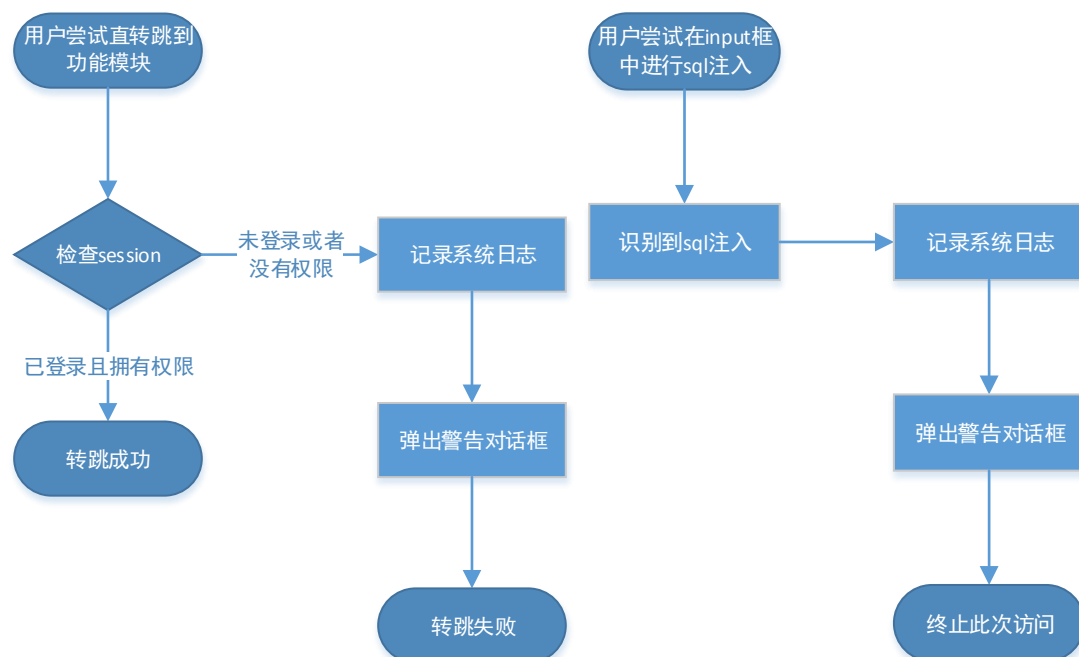
```

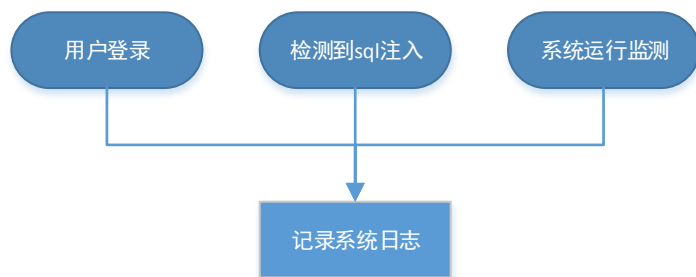
        return 1;
    }
    .....

    If (! CheckSafty( input ))//若输入不符合规范，存在安全问题
    {
        $error_msg = ' Your behavior is malicious.';
        SystemLogWrite( );
    }
    else
    .....
    ?>

```

### 3.11.7 流程逻辑





### 3.11.8 测试计划

与用户登录模块集成测试。

测试要求通过远程计算机，创建相应用户信息数据库，并且添加测试数据。

输入数据	预期结果
在登陆页面输入用户名密码	每次尝试登录都会被记录到系统日志当中
用户试图在未登录的情况下从地址栏转跳到功能模块或者在登录的情况下从地址栏转跳到不属于自己权限的功能模块	操作被中断并记录入系统日志，弹出警告对话框提示“非法访问”并转跳到欢迎页面或登陆页面。
在任一 input 框试图进行 sql 注入	被系统识别到 sql 注入，拒绝此次操作并记录到系统日志中，弹出警告对话框“非法输入”

图表-系统安全与运行日志模块测试计划表

## 3.12 查看系统日志模块设计说明

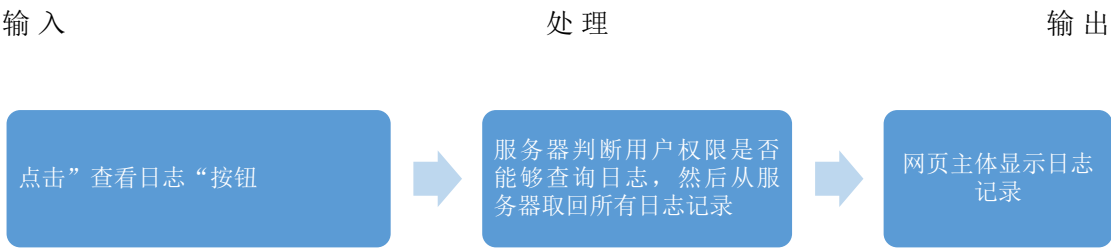
### 3.12.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 **System Administrator** 用户可使用的功能模块。在主页的导航栏上显示“查看日志”按钮，点击后网页主体显示所有的日志记录。



3.12.2 功能

IPO 图如下，



图表-查看系统日志模块 IPO 图

3.12.3 性能

只要点击按钮，通过用户验证，即可展示所有日志记录。

3.12.4 输入项

名称	标识	HTML 标签	输入方式
查看日志	log	<button>	点击按钮

图表-查看系统日志模块输入项表

3.12.5 输出项

名称	标识	类型	输出方式
日记记录	log_record	text	网页主体分页显示

图表-查看系统日志模块输出项表

3.12.6 设计方法

当点击查看系统日志时，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';

//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面
```

```
CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限

$file=fopen("systemlog.txt","r")or exit("Unable to open file!");//打开系统日志

while(!feof($file))

{

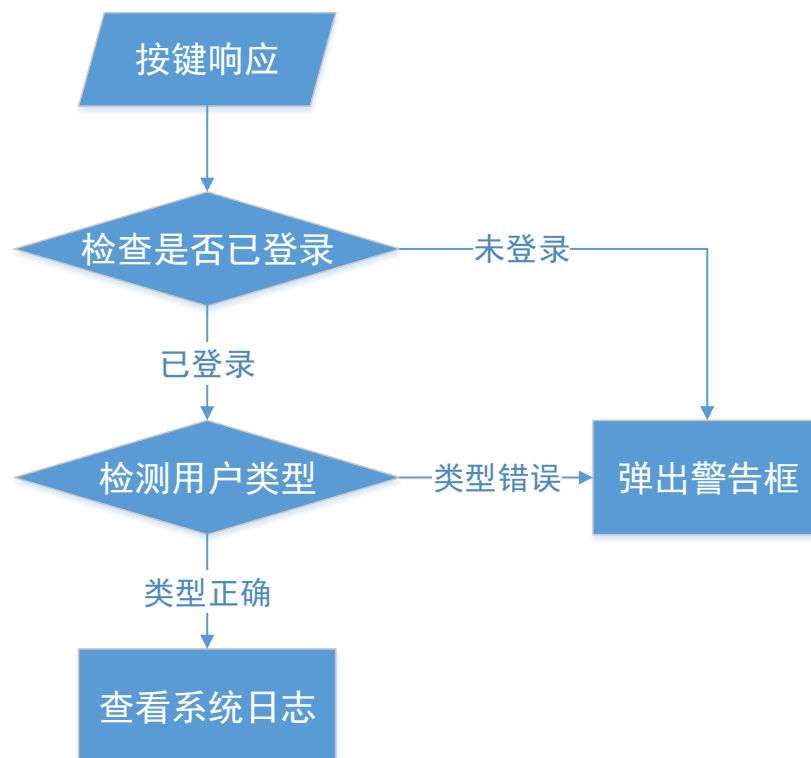
    echo fgets($file). "<br />";//逐行输出系统日志

}

fclose($file);

?>
```

### 3.12.7 流程逻辑



3.12.8 测试计划

与用户登录模块集成测试。

输入数据	预期结果
用非 admin 账户登录	看不到“查看日志”按钮
执行一次非法操作， 点击按钮	看到最新的操作记录

图表-查看系统日志模块测试计划表

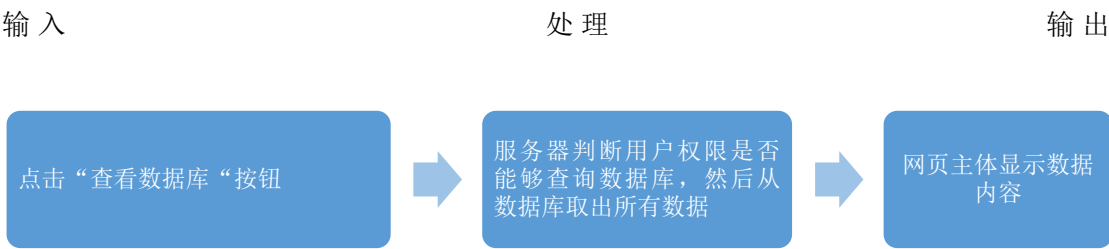
3.13 查看数据库表格模块设计说明

3.13.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 System Administrator 用户可使用的功能模块。在主页的导航栏上显示“查看数据库”按钮，点击后网页主体显示数据库里所有的表。

3.13.2 功能

IPO 图如下，



图表-查看数据库表格模块 IPO 图

3.13.3 性能

只要点击按钮，通过用户验证，即可展示所有数据库表格。

### 3.13.4 输入项

名称	标识	HTML 标签	输入方式
查看数据库	database	<button>	点击按钮

图表-查看数据库表格模块输入项表

### 3.13.5 输出项

名称	标识	类型	输出方式
数据库表格	database_record	表单内容	网页主体分页显示

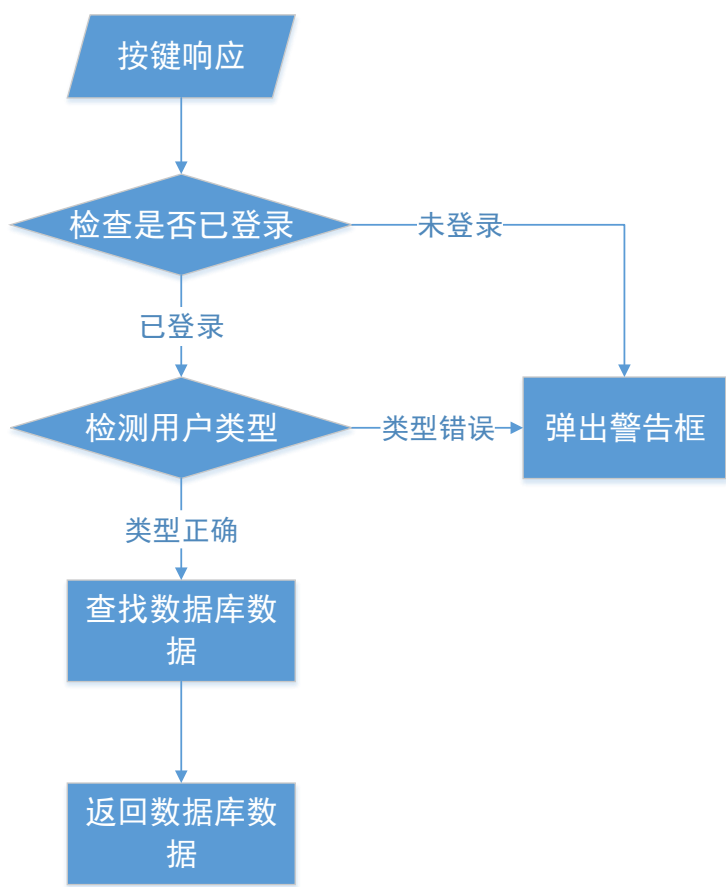
图表-查看数据库表格模块输出项表

### 3.13.6 设计方法

当点击查看数据库表格时，触发脚本，将信息提交给服务器，服务器执行以下 PHP 代码。

```
<?php
include 'check_access.php';
//确认是否登录。若登录，则继续以下语句；未登录则跳转至登陆界面
CheckUserType($_SESSION['usertype'])//确认用户是否具有相关权限
ConnectDB( );//连接数据库
$DB=SelectDatabase( );
echo $DB;
?>
```

3.13.7 流程逻辑



3.13.8 测试计划

与用户登录模块集成测试。

输入数据	预期结果
用非 admin 账户登录	看不到“查看数据库”按钮
添加一条新数据，点击按钮	看到最新的数据

图表-查看数据库模块测试计划表

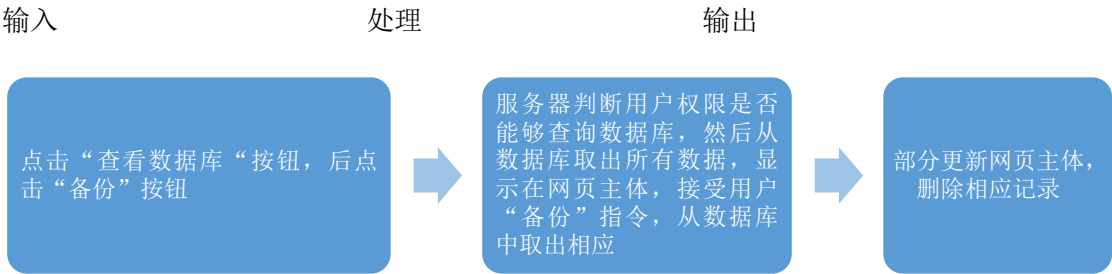
### 3.14 备份数据模块设计说明

#### 3.14.1 模块描述

此模块以网页表单的形式显示，是在登陆界面后 **System Administrator** 用户可使用的功能模块。在主页的导航栏上显示“查看数据库”按钮，点击后网页主体显示数据库里所有的表，在每个表格右上角显示“备份”按钮，可是实现备份功能。

#### 3.14.2 功能

IPO 图如下，



图表-备份数据模块 IPO 图

#### 3.14.3 性能

只要点击按钮，通过用户验证，即可展示所有数据库表格，并可以对其进行相应的操作。  
备份后可以在本地得到 SQL 文本文件。

#### 3.14.4 输入项

名称	标识	HTML 标签	输入方式
备份数据库	backup	<button>	点击按钮

图表-备份数据模块输入项表

### 3.14.5 输出项

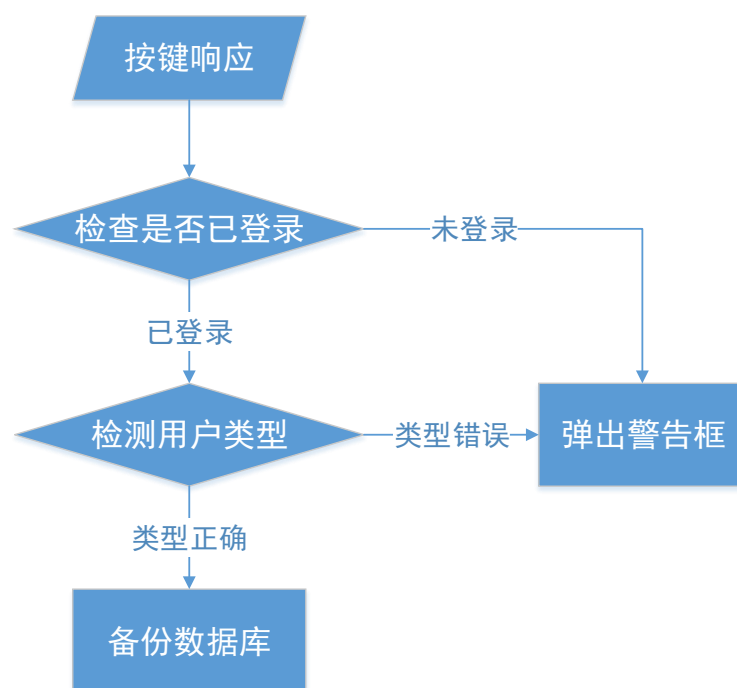
名称	标识	类型	输出方式
数据库表格	new_database	表单内容	网页主体分页显示
SQL 语句	backup_sql	SQL	一个包含 sql 语句的文本文件

图表-备份数据模块输出项表

### 3.14.6 设计方法

见 2.7 尚未解决的问题.

### 3.14.7 流程逻辑



### 3.14.8 测试计划

与用户登录、查看数据库模块集成测试。

输入数据	预期结果
用非 admin 账户登录	看不到“查看数据库”按钮
点击“备份”按钮	生成 SQL 文本文件，打开看可以看到生成表格的 SQL 语句

图表-备份数据库模块测试计划表

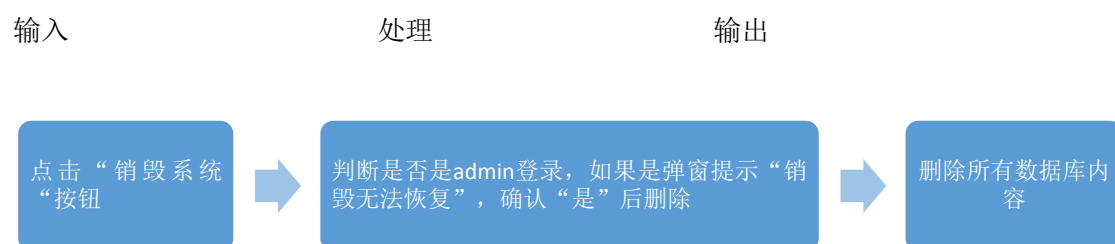
## 3.15 一键销毁系统模块设计说明

### 3.15.1 模块描述

此模块作为小组成员做大程完后的休闲模块，可以一键将所有数据库删除，导致系统无法正常使用。

### 3.15.2 功能

IPO 图如下，



图表-课程信息修改模块 IPO 图

### 3.15.3 性能

只要点击按钮，通过用户验证，并通过弹窗提示确认后即可删除数据库内容。



### 3.15.4 输入项

名称	标识	HTML 标签	输入方式
销毁系统	destroy	<button>	点击按钮
确认提示	flag	Js.alert	点击确认

图表-一键销毁系统模块输入项表

### 3.15.5 输出项

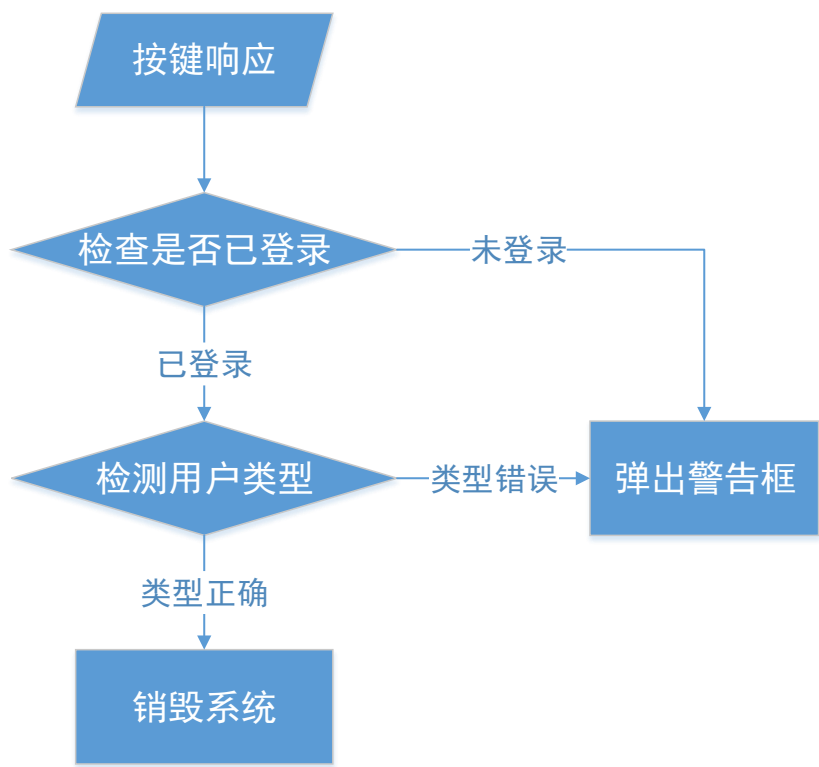
名称	标识	类型	输出方式
被销毁	destroyed	Js.alert	提示系统被销毁

图表-一键销毁系统模块输出项表

### 3.15.6 设计方法

见 2.7 尚未解决的问题。

3.15.7 流程逻辑



3.15.8 测试计划

与用户登录模块集成测试。

输入数据	预期结果
用非 admin 账户登录	看不到“销毁系统”按钮
点击按钮并确认后	弹窗提示系统被销毁，数据库被删除

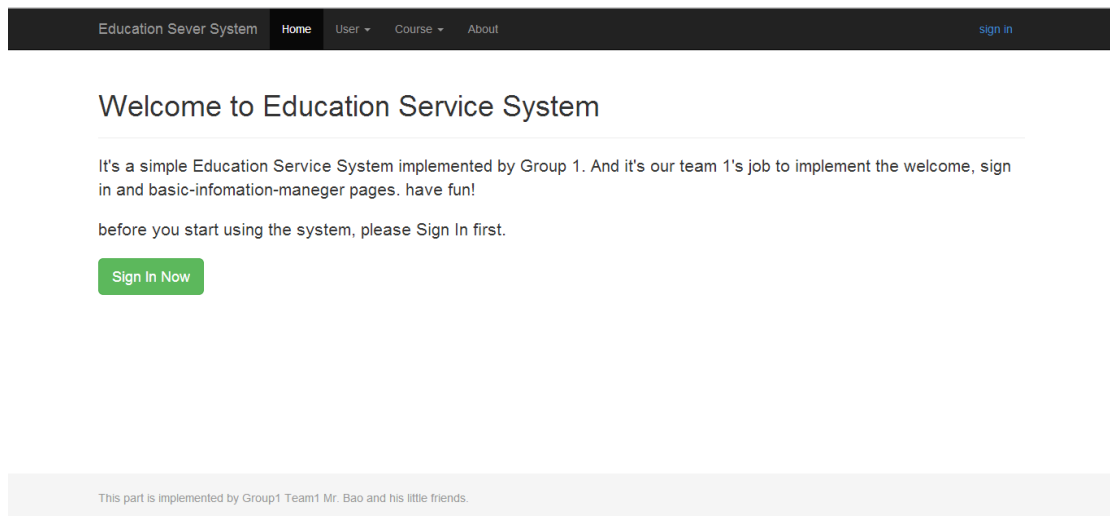
图表-一键销毁系统模块测试计划表

## 4.接口设计

### 4.1 用户接口

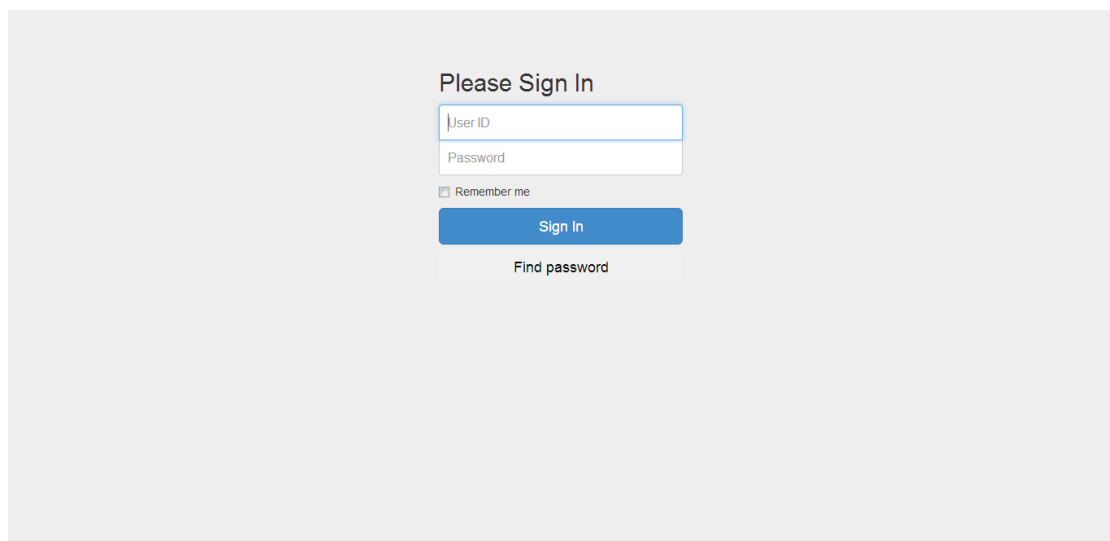
#### 4.1.1 欢迎界面

点击“sign in now”按钮登录。



#### 4.1.2 登陆界面

输入用户民、密码后点击“sign in”登录，忘记密码时可以用“find password”进行找回密码操作。



### 4.1.3 修改密码界面

用户可以在输入新密码之后点击按钮提交修改。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin sign out

## Change Password

Old password

Password

New password

Password

Input new password again

Password

Submit

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

### 4.1.4 查看用户信息界面

在查看用户基本信息。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin sign out

## User Infomation

Edit

User ID

admin

User Name

admin

Department

college of computer science

Gender

Male

Birth Day

1994-1-1

Enroll time

2012

Phone

180729128

Email

389313431@qq.com

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

### 4.1.5 修改用户信息界面

在查看用户界面按下修改按钮以后可以进行部分基本信息的修改。

Education Sever System

Home

User

Course

About

Hello,admin sign out

User Infomation

User ID

admin

User Name

admin

Department

college of computer science

Gender

Male

Birth Day

1994-1-1

Enroll time

2012

Phone

18072911111

Email

12345678@example.com

Submit

reset

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

### 4.1.6 检索用户界面

可在搜索后查看相关用户信息。

Education Sever System

Home

User

Course

About

sign in

User Information

Sort By

Order By

Keyword

Search

User ID	User Name	Department	Enroll Time	Phone	Action	
00001	Bao Junze	College of Computer Science	2012	123456	More	Delete
00002	Li ilao	College of Computer Science	2012	123456	More	Delete
00003	Mao Yuhan	College of Computer Science	2012	123456	More	Delete
00004	Zhang Xuqian	College of Computer Science	2012	123456	More	Delete
00005	Qian Xin	College of Computer Science	2012	123456	More	Delete

### 4.1.7 添加用户界面

可以在输入相关信息后添加新用户。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin sign out

Add User

User ID

User Name

Department

Gender

Birth Year

Birth Month

Birth Day

Submit

Reset

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

### 4.1.8 检索课程界面

可以在搜索课程关键字后查看相关课程信息。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin sign out

Course Information

Sort By ▾

Order By ▾

Lower Bound

Upper Bound

Keyword

Search

Course ID	Course Name	Department	Credit	Description	Action	
00001	Intro to Computer Science	College of Computer Science	2.0	Compulsory	More	Delete
00002	C Programming	College of Computer Science	2.0	Compulsory	More	Delete
00003	Data Structure Fundamentals	College of Computer Science	2.5	Mandatory	More	Delete
00004	Advanced Data Structures	College of Computer Science	1.5	Mandatory	More	Delete

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

## 4.1.9 查看课程基本信息

可以在搜索课程关键字后查看相关课程信息。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin [sign out](#)

Course Infomation

Edit

Course ID

00001

Course Name

Intro to Computer Science

Department

College of Computer Science

Credit

2.0

Description

Compulsory

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

## 4.1.10 修改课程基本信息界面

在查看课程信息页面按下 edit 后可以修改课程基本信息。

Education Sever System

Home

User ▾

Course ▾

About

Hello,admin [sign out](#)

Course Infomation

Course ID

Course Name

Department

Credit

Description

Submit

reset

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

### 4.1.11 添加课程界面

可在输入课程信息后点击按钮添加课程。

Education Sever System

HomeUserCourseAbout

Hello,admin sign out

Add Course

Course ID

Course Name

Department

Credit

Description

Submit

Reset

This part is implemented by Group1 Team1 Mr. Bao and his little friends.

## 4.2 外部接口

通过 php 内置的 `mysqli_connect()`函数与数据库 `mySQL` 建立连接。

## 4.3 内部接口

本系统与其他子系统通过共享数据以及 `SESSION` 调用完成通信

其余六个子系统通过检查`$_SESSION["user_id"]`来识别用户是否登录以及用户的 `ID` 是多少，通过检查`$_SESSION["usertype"]`来识别用户所属的组然后给予相应权限。

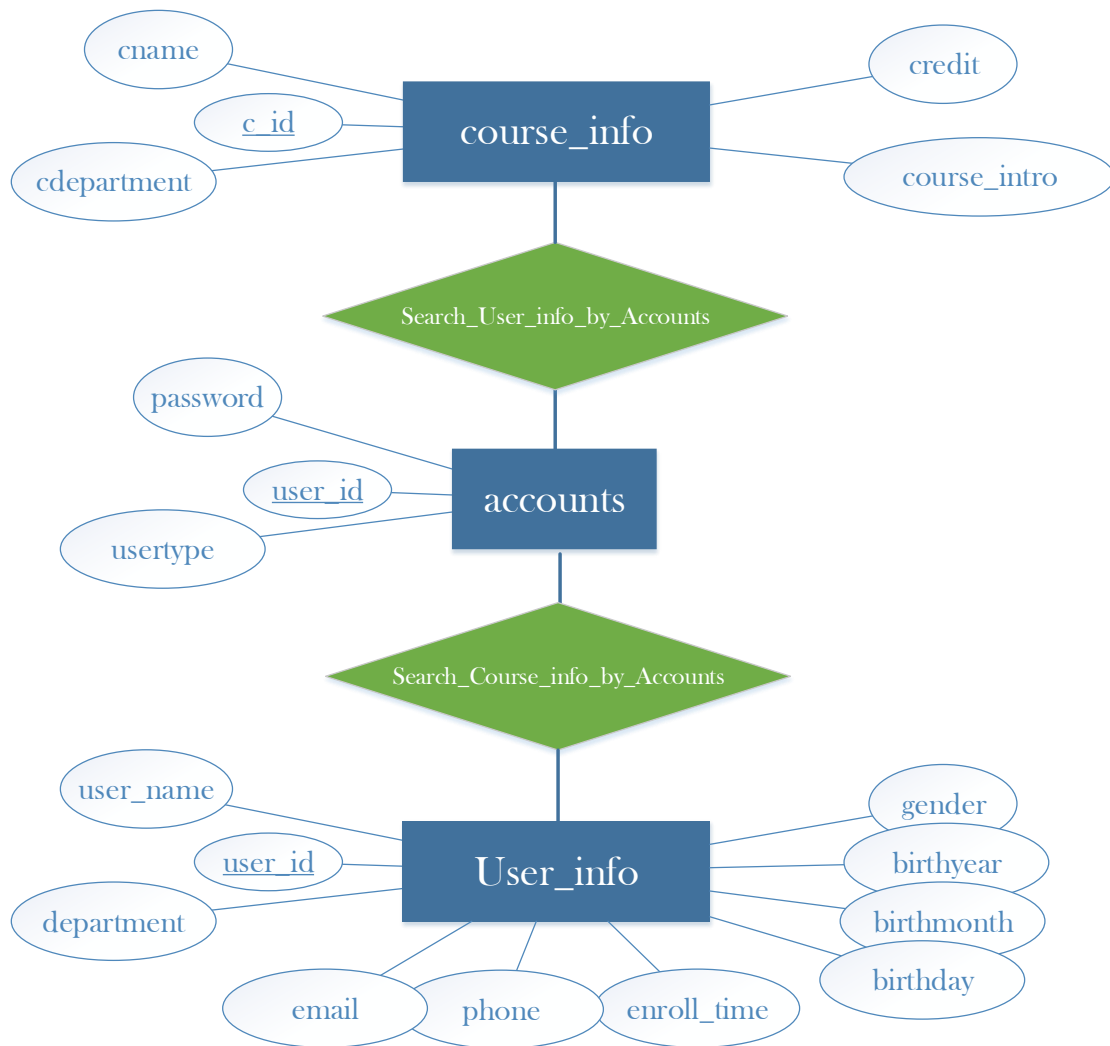
	自动排课子系统	选课子系统	课程资源共享子系统	论坛交流子系统	在线测试子系统	成绩管理子系统
基础信息管理子系统	用户登录信息和课程基本信息数据库	用户登录信息和课程基本信息数据库	用户登录信息和课程基本信息数据库	用户登录信息和用户基本信息数据库	用户登录信息和用户基本信息数据库	用户登录信息和用户基本信息数据库

图表-与其它子系统的数据库共享



## 5.系统数据结构设

### 5.1 概念结构设计



### 5.2 逻辑结构设计

账号信息管理: `accounts(user_id, password, usertype)`

用户基本信息: `user_info(user_id, username, department, gender, birthyear, birthmonth, birthday, enroll_time, phone, email)`

课程基本信息: `course_info(cid, cname, cdepartment, credit, course_intro)`

## 5.3 物理结构设计

### 1. 账号基本信息

字段	类型	是否为空	是否主键	备注
user_id	CHAR(20)	N	Y	账号 ID
password	CHAR(32)	N	N	密码
usertype	CHAR(7)	N	N	账号类型

### 2. 用户基本信息

字段	类型	是否为空	是否主键	备注
user_id	CHAR(20)	N	Y	账号 ID
username	VARCHAR(45)	N	N	用户姓名
department	VARCHAR(45)	N	N	所属部门
gender	CHAR(6)	N	N	性别
birthyear	INT(11)	N	N	出生年份
birthmonth	TINYINT(4)	N	N	出生月份
birthday	TINYINT(4)	N	N	出生日期
enroll_time	CHAR(4)	N	N	入学/入职年份
phone	CHAR(15)	Y	N	联系方式
email	VARCHAR(45)	Y	N	电子邮箱

### 3. 课程基本信息

字段	类型	是否为空	是否主键	备注
cid	CHAR(20)	N	Y	课程 ID
cname	VARCHAR(60)	N	N	课程名字
cdepartment	VARCHAR(40)	N	N	所属部门
credit	FLOAT(2,1)	N	N	学分
course_intro	BLOB	Y	N	课程介绍

## 6.运行设计

### 6.1 运行模块组合

本子系统按照功能划分模块，每个模块又按照流程划分为客户端界面，客户端脚本，服务器后台程序。出了主界面之外，功能模块之间相互不共享界面和客户端脚本，服务器后台程序只共享建立数据库连接的方法。

### 6.2 运行控制

**用户通过客户端界面进行操作：**

**用户登录模块：**用户通过在表单中输入正确的账户密码，从而登录系统，获得该用户类型应有的其他模块的操作权限。

**找回密码：**用户在表单中输入正确的用户邮件地址和个人安全问题答案，发送给服务器，即可在出现的页面中输入新的个人密码，发送给服务器。

用户登录成功后，通过主界面上的导航栏选择功能：

**个人信息查询/修改：**所有用户通过登录后获得授权。在此页面中可以查询到个人信息。部分信息后提供编辑按钮，点击后该条信息变为可编辑的文本框。用户在此表单中提交修改的个人信息，发送给服务器。

**用户信息查询/修改：**管理员及系统管理员登录后获得授权。在此表单中输入用户的 id，发送给服务器，即可返回用户的信息。所有信息后提供编辑按钮，点击后该条信息变为可编辑的文本框。管理员或系统管理员在此表单中提交修改的用户信息，发送给服务器。

**课程信息查询/修改：**所有用户通过登录后获得查询课程信息的授权。在此表单中输入课程的 id，发送给服务器，即可返回课程的信息。当用户类型为管理员或系统管理员时，所有信息后提供编辑按钮，点击后该条信息变为可编辑的文本框。在此表单中提交修改的课程信息，发送给服务器。

**修改密码：**所有用户通过登录后获得授权。在此表单中输入正确的旧密码和两次相同的新密码，发送给服务器。

## 6.3 运行时间

由于采用 Ajax 技术, 实现客户端静态更新信息, 静态分页等等, 因此, 服务器的带宽利用率大大提高, 唯一会影响服务器的运行时间的, 应该只是客户端频繁建立销毁数据库连接所带来的时间开销。

# 7.系统出错设计

## 7.1 出错信息

系统输出信息的形式	含义	处理办法
未登录时功能页面无法打开	没有登录时访问用户信息、课程信息等详情页面时无法转入相应页面	提示“非法连接”, 并提示用户登录
输入错误	在任何有输入框的地方输入非法的字符串, 例如 SQL 语句, 企图通过 SQL 注入来破坏数据库	对表单的输入进行过滤, 阻止非法输入
数据库无法连接	数据库配置不对	在 connectvar.php 中修改数据库配置
暂时无法连接网站服务器	由于短时间内出现大量攻击可能导致的服务器瘫痪	联系系统管理员进行紧急处理
账户密码错误	由于用户忘记密码或者被人恶意篡改密码造成用户无法登录	通过登录界面的“找回密码”功能对密码进行修改
数据库编码错误	由于用户端编码方式不一样, 在数据库读出写入的过程中也会涉及数据的编码问题	统一编码为 utf-8

## 7.2 补救措施

**确保数据完整性和一致性：**将数据库进行多处备份，并及时校对、更新；除线上保存之外，还可以进行线下存档。工程代码通过 `github` 进行完善的版本管理。

**恢复技术：**如果系统被攻击，数据库被修改，通过查看系统日志，获得更多信息并导入数据库备份文件进行恢复。

**紧急处理：**若网站被攻击瘫痪后，需要修改数据，联系系统管理员在后台进行数据更新。

## 7.3 系统维护设计

- 在可能出错的地方使用 `try-catch` 语句捕获异常，并输出相应的出错信息和可能的处理方法提示。
- 系统管理员定时查看系统日志，统计非法攻击来源和次数，并针对相应攻击加强安全防范措施。
- 系统管理员的登录不仅需要账户密码，还需要识别 IP，禁止非白名单 IP 的任何访问。
- 系统维护人员及时更新技术漏洞，增强代码的可靠性。