**北京科技大学 计算机与通信工程学院**

**软件工程课程设计**

**系统总体&详细设计报告**

**报告名称**：**基于B/S结构的会议室教室预约管理系统**

**学生姓名** ： **马嘉威**

**专 业**：  **计算机科学与技术**

**班 级**：  **计算机1401**

**学 号**： **41422001**

**任课教师**： **殷绪成**

目录

[第一部分 总体设计 1](#_Toc482709239)

[1引言 1](#_Toc482709240)

[1.1编写目的 1](#_Toc482709241)

[1.2背景 1](#_Toc482709242)

[1.3参考资料 1](#_Toc482709243)

[2总体设计 2](#_Toc482709244)

[2.1需求规定 2](#_Toc482709245)

[2.2运行环境 2](#_Toc482709246)

[2.3基本设计概念和处理过程 2](#_Toc482709247)

[2.3.1整个系统的结构图 3](#_Toc482709248)

[2.3.2各部分的结构图 3](#_Toc482709249)

[2.4结构 5](#_Toc482709250)

[2.4.1服务器端结构 5](#_Toc482709251)

[2.4.2客户端结构 6](#_Toc482709252)

[2.5功能器需求和程序的关系 6](#_Toc482709253)

[2.6人工处理过程 7](#_Toc482709254)

[3接口设计 7](#_Toc482709255)

[3.1用户接口 7](#_Toc482709256)

[3.2外部接口 8](#_Toc482709257)

[3.3内部接口 8](#_Toc482709258)

[4运行设计 9](#_Toc482709259)

[4.1运行模块组合 9](#_Toc482709260)

[4.1.1登录 10](#_Toc482709261)

[4.1.2客户信息修改模块 11](#_Toc482709262)

[4.1.3教室查询模块 12](#_Toc482709263)

[4.1.4教室预约模块 13](#_Toc482709264)

[4.1.5发布公告模块 14](#_Toc482709265)

[4.1.6教室评论模块 15](#_Toc482709266)

[4.1.7教室预约审批模块 16](#_Toc482709267)

[4.1.8密码丢失找回模块 17](#_Toc482709268)

[4.2运行控制 17](#_Toc482709269)

[4.2.1登录 18](#_Toc482709270)

[4.2.2密码找回/密码修改 18](#_Toc482709271)

[4.2.3个人信息修改 18](#_Toc482709272)

[4.2.4查询教室 18](#_Toc482709273)

[4.2.5预约教室 19](#_Toc482709274)

[4.2.6评论 19](#_Toc482709275)

[4.2.7公告 19](#_Toc482709276)

[4.2.8预约教室审批 19](#_Toc482709277)

[4.3运行时间 19](#_Toc482709278)

[5系统数据结构设计 20](#_Toc482709279)

[5.1逻辑结构设计要点 20](#_Toc482709280)

[5.1.1数据库各表属性如下 20](#_Toc482709281)

[5.1.2系统基本E-R图 21](#_Toc482709282)

[5.2物理结构设计要点 21](#_Toc482709283)

[5.3数据结构与程序的关系 24](#_Toc482709284)

[6系统出错处理设计 25](#_Toc482709285)

[6.1出错信息 25](#_Toc482709286)

[6.2补救措施 25](#_Toc482709287)

[第二部分     详细设计 26](#_Toc482709288)

[1引言 26](#_Toc482709289)

[1.1编写目的 26](#_Toc482709290)

[1.2项目背景 26](#_Toc482709291)

[1.3定义 26](#_Toc482709292)

[1.4参考资料 26](#_Toc482709293)

[2设计概述 27](#_Toc482709294)

[2.1运行环境概述 27](#_Toc482709295)

[2.1.1服务器环境 27](#_Toc482709296)

[2.1.2客户端环境 27](#_Toc482709297)

[2.2开发工具概述 28](#_Toc482709298)

[2.3基本设计原理概述 28](#_Toc482709299)

[2.4基本开发技术概述 28](#_Toc482709300)

[3模块详细设计 29](#_Toc482709301)

[3.1登录模块 29](#_Toc482709302)

[1功能描述 29](#_Toc482709303)

[2性能 29](#_Toc482709304)

[3输入项 30](#_Toc482709305)

[4输出项 30](#_Toc482709306)

[5设计方法（算法） 30](#_Toc482709307)

[6流程逻辑 32](#_Toc482709308)

[7相关数据库描述 32](#_Toc482709309)

[8存储分配 32](#_Toc482709310)

[9测试计划 32](#_Toc482709311)

[10尚未解决问题 33](#_Toc482709312)

[3.2客户信息修改模块 33](#_Toc482709313)

[1功能描述 33](#_Toc482709314)

[2性能 33](#_Toc482709315)

[3输入项 33](#_Toc482709316)

[4输出项 34](#_Toc482709317)

[5设计方法（算法）&流程逻辑 35](#_Toc482709318)

[6相关数据库描述 35](#_Toc482709319)

[7存储分配 35](#_Toc482709320)

[8测试计划 35](#_Toc482709321)

[9尚未解决问题 36](#_Toc482709322)

[3.3教室查询模块 36](#_Toc482709323)

[1功能描述 36](#_Toc482709324)

[2性能 36](#_Toc482709325)

[3输入项 36](#_Toc482709326)

[4输出项 37](#_Toc482709327)

[5设计方法（算法） 37](#_Toc482709328)

[6流程逻辑 38](#_Toc482709329)

[7相关数据库描述 38](#_Toc482709330)

[8存储分配 38](#_Toc482709331)

[9测试计划 38](#_Toc482709332)

[10尚未解决问题 39](#_Toc482709333)

[3.4教室预约模块 39](#_Toc482709334)

[3.4.1功能 39](#_Toc482709335)

[3.4.2性能 39](#_Toc482709336)

[3.4.3输入项 39](#_Toc482709337)

[3.4.4输出项 40](#_Toc482709338)

[3.4.5设计方法（算法） 40](#_Toc482709339)

[3.4.6流程逻辑 43](#_Toc482709340)

[3.4.7相应数据库描述 43](#_Toc482709341)

[3.4.8存储分配 43](#_Toc482709342)

[3.4.9测试计划 44](#_Toc482709343)

[3.4.10尚未解决问题 44](#_Toc482709344)

[3.5发布公告模块 44](#_Toc482709345)

[1功能描述 44](#_Toc482709346)

[2性能 44](#_Toc482709347)

[3输入项 44](#_Toc482709348)

[4输出项 45](#_Toc482709349)

[5设计方法（算法） 45](#_Toc482709350)

[6流程逻辑 46](#_Toc482709351)

[7相关数据库描述 47](#_Toc482709352)

[8存储分配 47](#_Toc482709353)

[9测试计划 47](#_Toc482709354)

[10尚未解决问题 47](#_Toc482709355)

[3.6教室评论模块 47](#_Toc482709356)

[3.6.1功能描述 47](#_Toc482709357)

[3.6.2性能 48](#_Toc482709358)

[3.7.3输入项 48](#_Toc482709359)

[3.6.4输出项 48](#_Toc482709360)

[3.6.5设计方法（算法） 48](#_Toc482709361)

[3.6.6流程逻辑 51](#_Toc482709362)

[3.6.7相关数据库描述 51](#_Toc482709363)

[3.6.8存储分配 52](#_Toc482709364)

[3.6.9测试计划 52](#_Toc482709365)

[3.6.10尚未解决问题 52](#_Toc482709366)

[3.7教室预约审批模块 52](#_Toc482709367)

[3.7.1功能描述 52](#_Toc482709368)

[3.7.2性能 52](#_Toc482709369)

[3.7.3输入项 52](#_Toc482709370)

[3.7.4输出项 53](#_Toc482709371)

[3.7.5设计方法（算法） 53](#_Toc482709372)

[3.7.6流程逻辑 55](#_Toc482709373)

[3.7.7相关数据库描述 55](#_Toc482709374)

[3.7.8存储分配 56](#_Toc482709375)

[3.7.9测试计划 56](#_Toc482709376)

[3.7.10尚未解决问题 56](#_Toc482709377)

[3.8密码丢失找回模块 56](#_Toc482709378)

[3.8.1功能描述 56](#_Toc482709379)

[3.8.2性能 57](#_Toc482709380)

[3.8.3输入项 57](#_Toc482709381)

[3.8.4输出项 57](#_Toc482709382)

[3.8.5设计方法（算法） 58](#_Toc482709383)

[3.8.6流程逻辑 59](#_Toc482709384)

[3.8.7相关数据库描述 60](#_Toc482709385)

[3.8.8存储分配 60](#_Toc482709386)

[3.8.9测试计划 60](#_Toc482709387)

[3.8.10尚未解决问题 61](#_Toc482709388)

[4用户接口设计 61](#_Toc482709389)

[4.1用户界面设计 61](#_Toc482709390)

[1、页面统计： 61](#_Toc482709391)

[2、页面详细设计： 61](#_Toc482709392)

[4.2用户体验设计 62](#_Toc482709393)

# 第一部分 总体设计

# 1引言

## 1.1编写目的

a,该总体设计说明书的目的在于根据基于B/S结构的会议室教室预约管理系统需求分析说明书提出该系统的概要设计，即系统的处理流程，结构，接口设计，运行设计，系统数据结构设计以及系统出错处理设计等。

b,希望起到一个承上启下的作用，对需求分析做出大概实现方案，以及对下一步详细设计部分的实现打下基础。

c,通过对实际软件系统的总体设计，把握软件系统的整体架构，达到对书本知识的运用，体会面向数据流的设计方法，理解总体设计的设计原理：模块化，抽象，逐步求精，信息隐藏和局部化，模块对立，即高内聚低耦合。

## 1.2背景

a，软件系统名称：基于B/S结构的会议室教室预约管理系统

b，该项目选题于《软件工程课程设计》，我们组选的是第一个课题“基于B/S结构的会议室教室预约管理系统”

c，此次项目是以计算机与通信工程学院的会议室教室预约为对象，分析、设计并实现一个预约管理的网络电子化系统，方便教师、学生进行会议室、教室的借用。

d，在进行总体设计的说明书的编写前，软件课程设计小组已经完成该项目的需求分析，对该软件系统的功能需求和非功能需求有了详细说明

## 1.3参考资料

1.软件工程导论（第六版）张海藩 编著 /2010-07-01 /清华大学出版社

2.《软件工程总体设计示例模板》百度百科/文库/知道

3.已提交的《基于B/S结构的会议室教室预约管理系统需求分析报告》

# 2总体设计

## 2.1需求规定

北京科技大学计算机与通信工程学院为方便教师和学生预约机电信息楼的教室，同时方便管理员管理教室，需开发一个基于B/S架构的会议室教室预约管理系统。该系统将为用户提供教室信息（课程、容量、是否为空等），管理员将通过账号以及密码登录，对教室信息、预约申请信息、用户信息，留言以及公告信息进行管理；用户分为教师，学生，管理员，这三种用户都可以分别对相应权限的会议室、教室占用状态浏览和查询（包括，时间、缘由、借用人等信息），同时可以对会议室、教室进行预约和取消预约，并且可以修改个人信息。预定成功的用户会收到系统发送的短信并凭借短信去收发室领取该教室的钥匙。长期预定，以及对长期预定过程中可能存在冲突的解决办法，比如审批和抽签等。

a.登录：本系统的登录主要包括的是系统管理员的登录，学生的登录和教师的登录，这三种角色只要输入各自的用户名，密码和验证码选择登录角色就可以登录本系统

b.系统管理员：系统管理员的功能主要包括教室管理，基础信息管理和账户管理。账户管理包括后台用户管理，修改密码，账户信息和编辑信息四个主要的功能。基础信息管理主要包括学生管理，教师管理。教室管理模块主要包括教室管理，教室预约审批，教学楼管理和教室类型管理。

c.教师：教师功能模块主要包括教室的预约和个人账户的管理。

d.学生：学生功能模块主要包括教室的预约和个人账户的管理。

e.安全性：明确区分系统中不同用户权限；用户登陆密码不可见、不可复制；系统数据的机密性，完整性，可管理性，独立性，可备份和恢复性

f.审计性：在“增、删、改、查”的“业务SQL”里嵌入“日记记录SQL”外

g.易用性：使用个性化的人机交互界面，为用户设计友好的界面接口

## 2.2运行环境

a.硬件环境：

CPU：至少是双核处理器

内存：至少是1G内存

硬盘：至少是10G以上

显示器：VGA以及更高

b.软件环境：

操作系统：Windows XP及以上操作系统，Linux，OS等等

Web服务器：Apache2.0以上版本

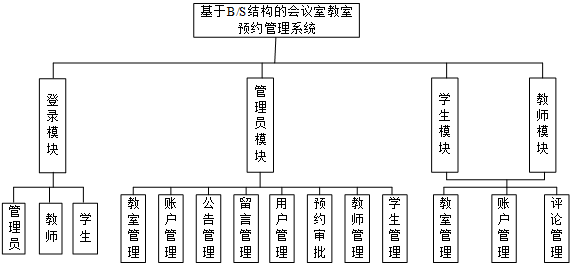
数据库：MYSQL5.0及以上数据库

服务器脚本：PHP5.2以上版本

## 2.3基本设计概念和处理过程

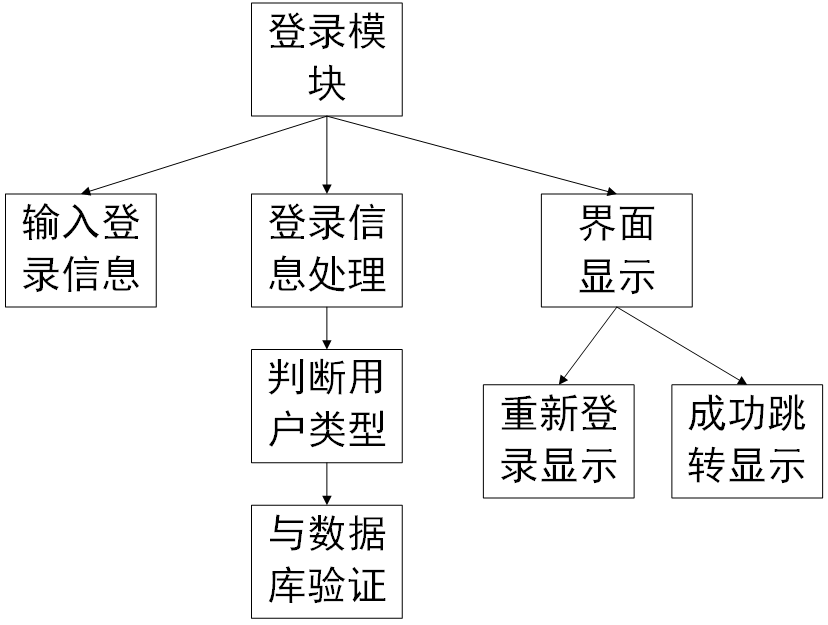
基于数据流的结构化设计方法

### 2.3.1整个系统的结构图

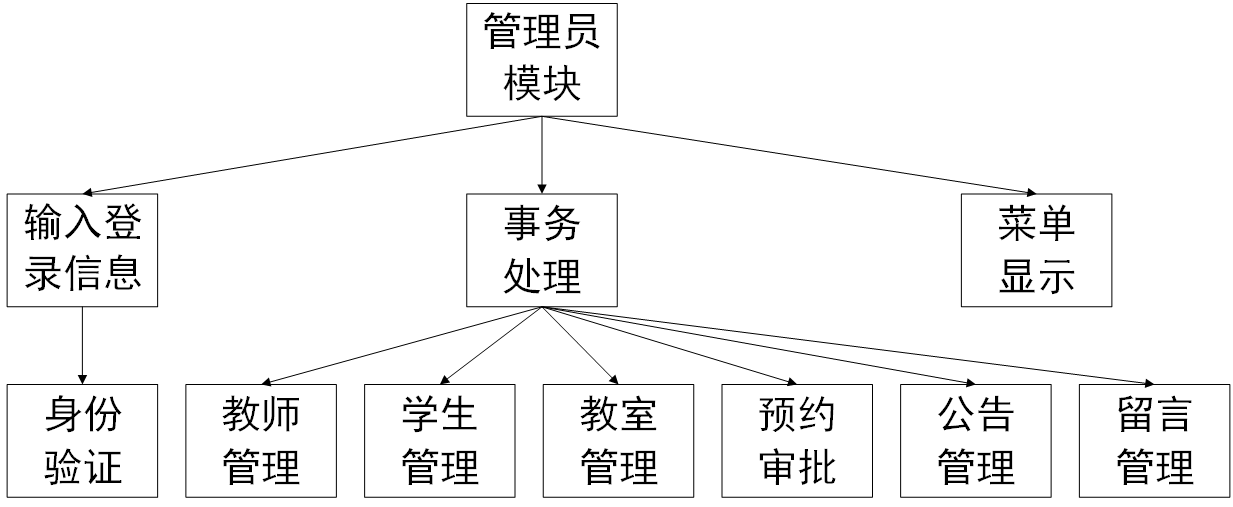


### 2.3.2各部分的结构图

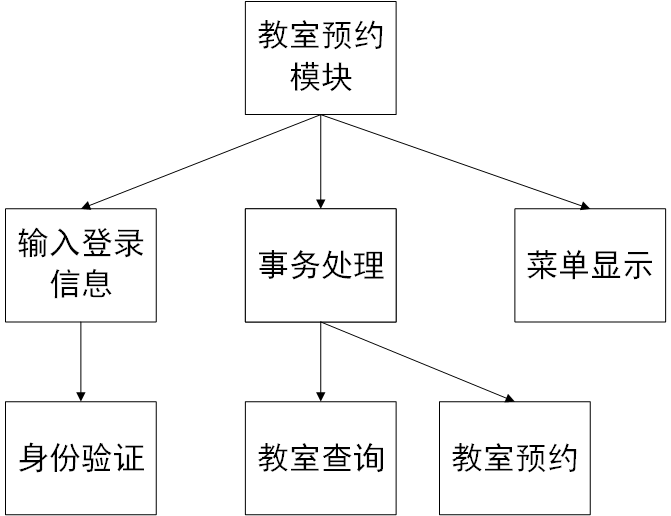
①登录模块：



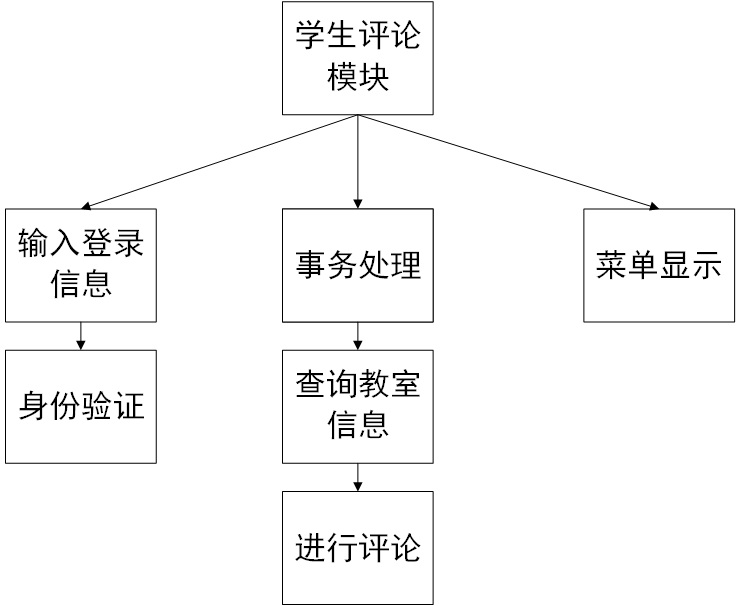
②管理员模块：



③教室预约模块：

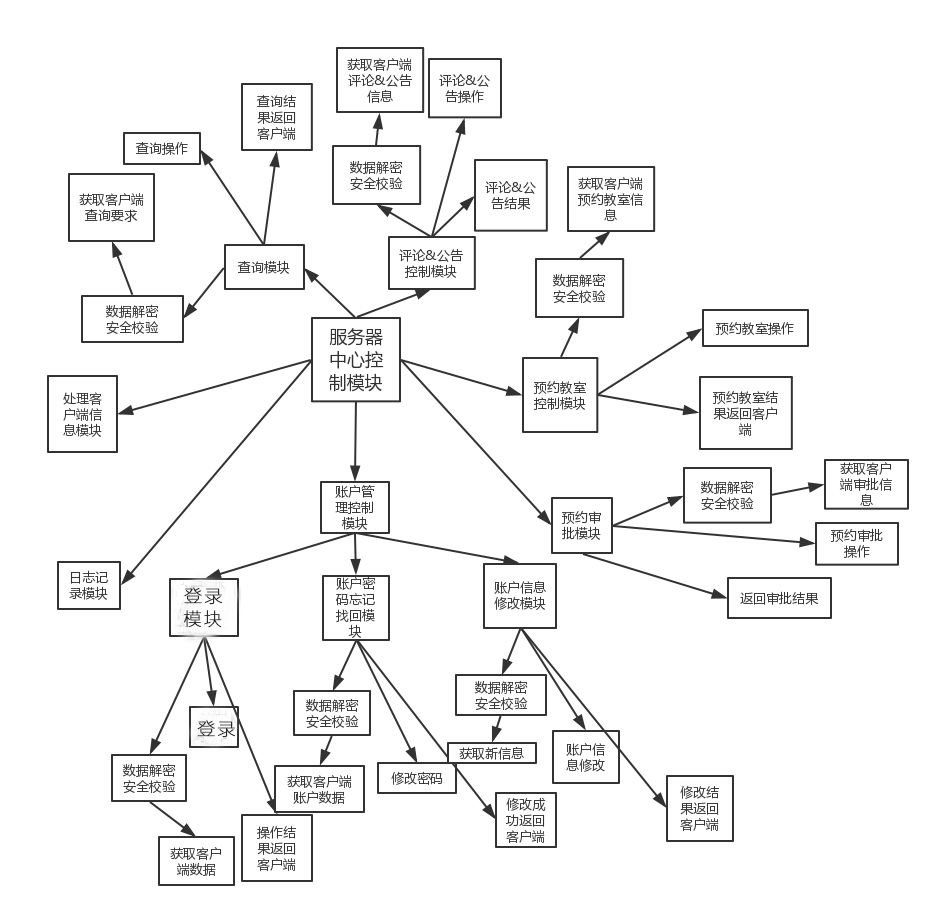


④评论模块：

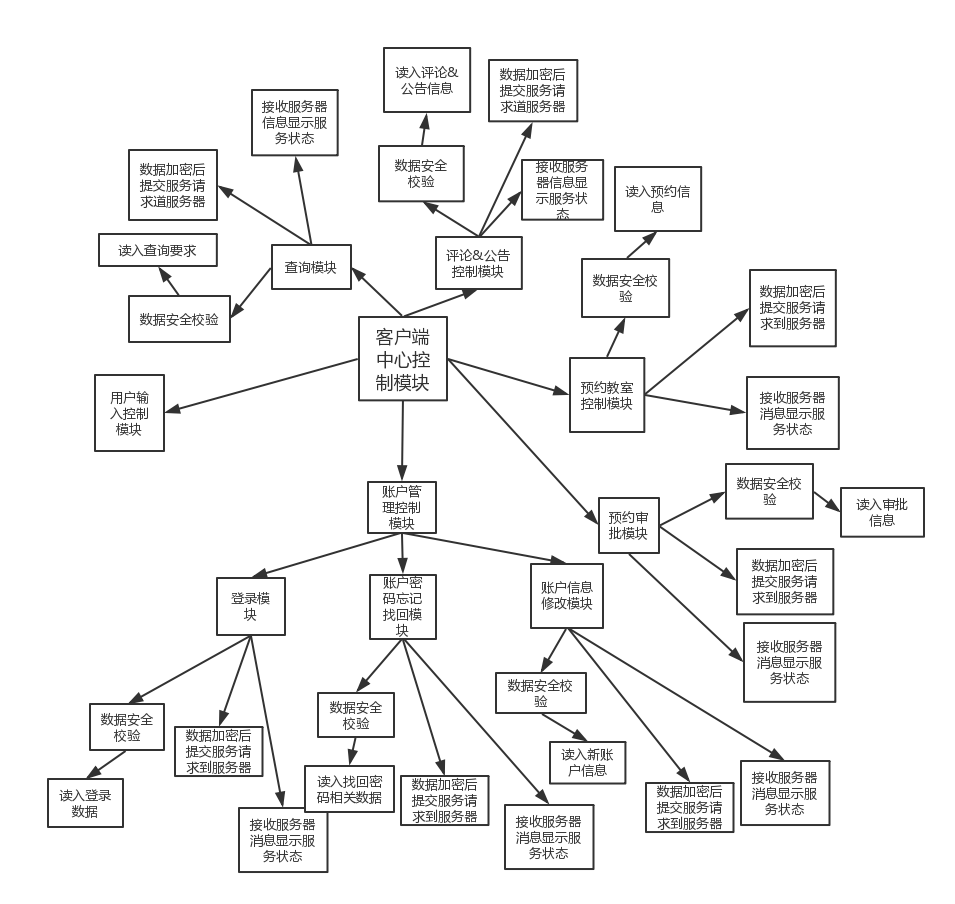


## 2.4结构

### 2.4.1服务器端结构



### 2.4.2客户端结构



## 2.5功能器需求和程序的关系

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 用户输入控制模块 | 登录模块 | 账户密码忘记找回模块 | 账户信息修改模块 | 教室预约模块 | 预约审批模块 | 查询模块 | 评论模块 | 公告模块 | 日志记录模块 | 显示服务状态 |
| 账户登录 | √ | √ |  |  |  |  |  |  |  | √ | √ |
| 忘记密码找回 | √ | √ | √ | √ |  |  |  |  |  | √ | √ |
| 账户信息修改 | √ |  |  | √ |  |  |  |  |  | √ | √ |
| 查询教室 | √ |  |  |  |  |  | √ |  |  | √ | √ |
| 预约教室 | √ |  |  |  | √ |  |  |  |  | √ | √ |
| 审批预约教室 | √ |  |  |  |  | √ |  |  |  | √ | √ |
| 评论教室 | √ |  |  |  |  |  |  | √ |  | √ | √ |
| 发布公告 | √ |  |  |  |  |  |  |  | √ | √ | √ |

## 2.6人工处理过程

①预约教室审批：不同用户拥有不同的预约教室权限，用户主要分为学生，老师和管理员，管理员拥有最高权限，可以对所有教室进行管理，对预约教室进行审批，有长期预定教室的权限（一个学期以上），对于学生和老师进行短期教室的预约需要提交“预约表”，在管理员进行人工审核之后，系统发送短信通知用户。

②教室钥匙管理：用户预约教室成功之后，需要在预约时间之前到楼管处签到，凭借短信通知获得教室钥匙，管理员标记教室为“正在使用”，否则用户不按规定时间签到则为违规一次，发布公告通知。

# 3接口设计

## 3.1用户接口

采用图形用户界面：

1、登录界面：输入账号，密码，选择登录角色，点击登录进行登录

2、忘记密码找回：在登录界面点击“忘记密码”进入忘记密码找回界面，输入账号，新密码，再次输入新密码，手机号，获取验证码，验证码正确，提示“密码修改成功”，返回登录界面

3、账户信息修改：个人中心->信息修改界面，输入账户信息，按确认后提交内容，返回“修改成功”或“失败”提示

4、查询教室界面：直接点击查询，列出所有可预约及不可预约教室列表；选择查询标准，再点击查询，列出所有符合查询要求的教室；再搜索框输入关键字查询，列出所有与关键字相关的查询结果

5、预约教室：进入某一可预约教室界面，点击“预约”，弹出预约申请表，按要求填写预约申请表，点击提交，完成教室预约，返回“等待管理员审批界面”

6、审批预约教室界面：管理员登录进入教室管理->预约管理，出现教室预约申请列表，点击某一申请表查看，点击“允许预约”标记该教室状态为“已预约”，并且系统发送短信到该预约用户通知预约成功，返回预约申请列表；或者点击“禁止预约”，系统发送短信到该预约用户通知预约失败

7、评论教室界面：用户登录成功并且进入某一教室信息界面，在最下方有评论区，用户点击“评论”，弹出评论文本卡框，填写完成后点击评论，进入“评论成功”或“失败”界面

8、发布公告界面：管理员登录成功后进入公告管理界面，点击“新建公告”，弹出公告文本框，输入公告，点击发布，弹出“发布成功”或者“失败”，返回公告列表界面

## 3.2外部接口

服务器端采用PHP编写程序，通过mysql管理工具访问教室预约系统数据库

## 3.3内部接口

该系统是基于B/S架构的，所以客户端界面是web浏览器，客户通过在浏览器界面输入各项数据，进行不同的操作请求进入相应的操作模块，每个操作模块用户按要求输入相应的数据，经过合格性校验后，作为一个表单提交给服务器端；服务器端接受到数据后进行表单的解析，将不同的数据分配给相应的处理模块处理，并将护理结果以表单的形式返回给客户端浏览器；服务器端与数据库相连，通过数据库语言访问数据库实现数据库操作，并将操作结果返回客户端。

操作模块如下：



# 4运行设计

## 4.1运行模块组合

各个不同的运行控制所经历的内部模块及支持软件如下粗红线所示：

### 4.1.1登录



### 4.1.2客户信息修改模块



### 4.1.3教室查询模块



### 4.1.4教室预约模块



### 4.1.5发布公告模块



### 4.1.6教室评论模块



### 4.1.7教室预约审批模块



### 4.1.8密码丢失找回模块



## 4.2运行控制

运行控制按照各模块间逻辑结构和函数调用关系来实现。在各个模块中，客户端收集用户输入信息，判断输入信息所属模块，以表单形式反馈到服务器端，服务器端对数据进行解析，再匹配到相应的控制模块。最后实现正确的运行路径。

### 4.2.1登录

1、登录时用户需提供账号，密码，验证码和用户身份。用户的账号及密码会被存储于数据库中，以用作登录校验，密码修改，教室预约等。

2、账户每次登录系统，系统的日志管理系统都会记录登录时信息，包括用户账号，登录时间，登录地点等。

3、用户首次登陆时有初始密码为“000000”或者身份证后六位，用户可在首次登录后修改密码

4、用户的账号为标示用户的唯一id，用户的部分个人信息学在首次登录之前由管理员录入系统，如账号（即学号），电话号码，身份证号，初始密码，用户类型等

### 4.2.2密码找回/密码修改

1、用户通过email 找回密码，管理员在初始化用户信息时必须填写用户正确的邮箱，否则将不能使用系统所提供的密码重设功能。

2、必须由该用户通过邮箱确认。

3、在登录界面提供一个链接，或者在登录失败后提供一个链接。链接名定义为：“忘了密码，需要找回吗？”

4、当用户点击找回密码时，弹出找回密码表单。用户输入账号（如果是登录失败后，我们可以用session填充表单内容），新密码，确认新密码，验证码等信息。户点击提交后，开始密码重设功能流程。

5、密码重设时，用户绑定的邮箱会收到一个链接，点击该链接跳转到密码修改成功界面

### 4.2.3个人信息修改

1、用户登录成功，点击个人中心，打开信息修改界面

2、弹出信息修改表单，输入账户信息，按确认后提交内容

3、弹出“修改成功”或“失败”提示，并返回个人信息界面

### 4.2.4查询教室

1、用户登录成功后，点击教室查询，打开教室查询界面

2、教室查询界面初始化为所有可见教室，用户不输入任何信息直接点击查询或者刷新页面，教室列表依然默认显示所有可见教室

3、可见教室列表根据用户权限和浏览器界面大小显示

4、用户可通过“关键字查询”或者“模式匹配”两种方式找到相关教室

5、“关键字查询”：用户在搜索框输入部分教室名称，数据库自动匹配相关教室显示在搜索框列表

6、“模式匹配”：用户通过选择教室容量，位置，楼层等等教室相关属性，然后点击搜索显示相关教室到教室列表

### 4.2.5预约教室

1、用户登录成功后点击预约教室，打开教室预约功能

2、教室预约功能首先跳转到教室查询界面

3、用户通过查询到符合期望教室，打开教室信息界面

4、用户在教室信息界面点击“预约”，跳转到“教室预约单填写”界面

5、用户填写“教室预约单”，点击“提交”

6、弹出“预约成功，等待管理员审批”，并跳转回教室信息界面

### 4.2.6评论

1、用户登录成功后点击评论教室，进入教室评论功能

2、点击评论教室后跳转到教室查询界面，查询相关教室，进入教室信息界面

3、在教室信息最下方点击评论，弹出评论文本框

4、输入评论内容，点击提交

5、返回教室信息界面，最下方评论已更新

### 4.2.7公告

1、管理员登录成功后点击公告管理，进入公告列表

2、在公告列表界面可对公告进行增删改查

3、点击某一条公告，可对其进行修改和删除功能

4、点击增加公告，弹出公告内容文本框，编辑好后，点击提交，返回公告列表界面，界面自动刷新显示

5、一般用户（学生和老师）在登录之后的首页只能看到最近的几条公告，公告按时间自动下沉

### 4.2.8预约教室审批

1、管理员登录成功后点击教室管理，进入预约管理界面

2、该界面显示所有预约单列表

3、点击某一申请表查看，点击“允许预约”标记该教室状态为“已预约”，并且系统发送信息到该预约用户的邮箱通知预约成功，返回预约申请列表

4、或者点击“禁止预约”，系统发送信息到该预约用户邮箱通知预约失败

## 4.3运行时间

1、B/S工作模式结构下，用户工作界面是通过WWW浏览器来实现，极少部分事务逻辑在前端（Browser）实现，但是主要事务逻辑在服务器端（Server）实现，形成所谓三层3-tier结构。

2、客户端网页程序没有实现逻辑功能，系统主要消耗时间为打开浏览器的时间。

3、客户端和服务器通信占用网络传输时间，网络传输时间主要由网络传输速度决定，在网速较高的环境下不是主要考虑因素。

4、服务器端对数据库操作占用服务器的cpu时间，数据库操作时间取决于服务器性能，同时访问量的多少，所使用数据库功能以及数据库查询语言的逻辑结构，在大规模访问量或者实现较难查询功能时，改时间可能成为系统性能的瓶颈。

5、基于此次实现的系统，具有较少访问量，实现查询逻辑简单，所以该系统的运行时间

# 5系统数据结构设计

## 5.1逻辑结构设计要点

### 5.1.1数据库各表属性如下

学生：姓名，学号，班级，年级，邮箱，电话号码，头像，密码

老师：姓名，教职工号，职称，所属院系，邮箱，电话号码，头像，密码

管理员：姓名，账号，电话号码，邮箱，头像，密码

教室：类型，教室编号，座位容量，地点，预约状态，图片

预约申请记录表：预约号，教室编号，教室类型，申请时间段，申请人账号，申请原因，审核人账号，审核结果

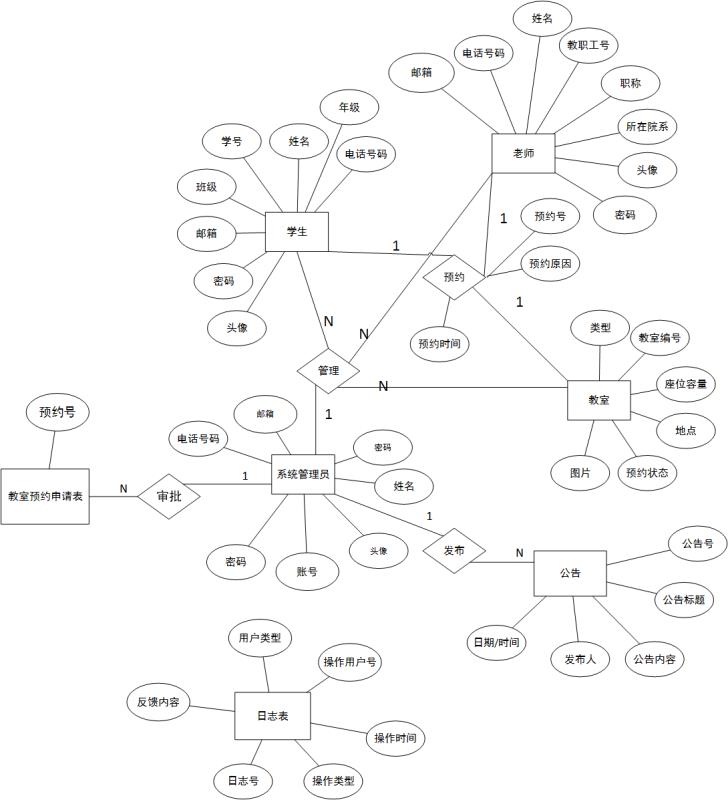
公告表：公告号，公告标题，公告内容，日期/时间，发布人

评论表：评论号，教室编号，评论内容，评论人

日志表：日志号，操作类型，操作时间，操作用户号，用户类型，反馈内容

(有下划线的属性为主键primary key)

### 5.1.2系统基本E-R图

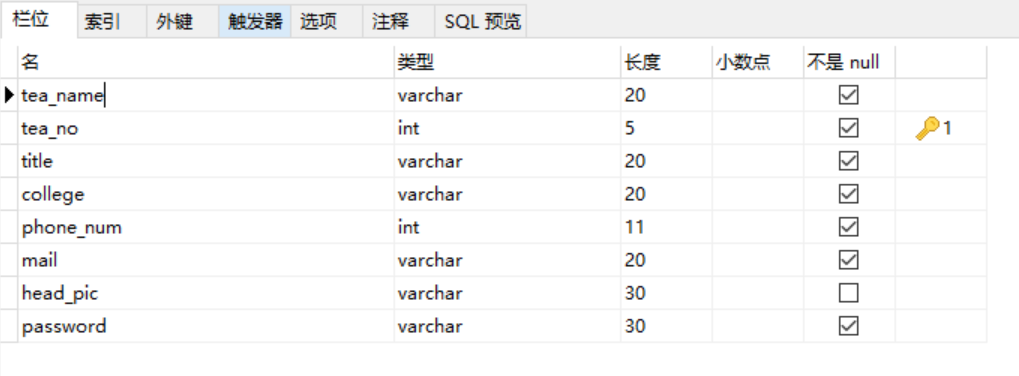


## 5.2物理结构设计要点

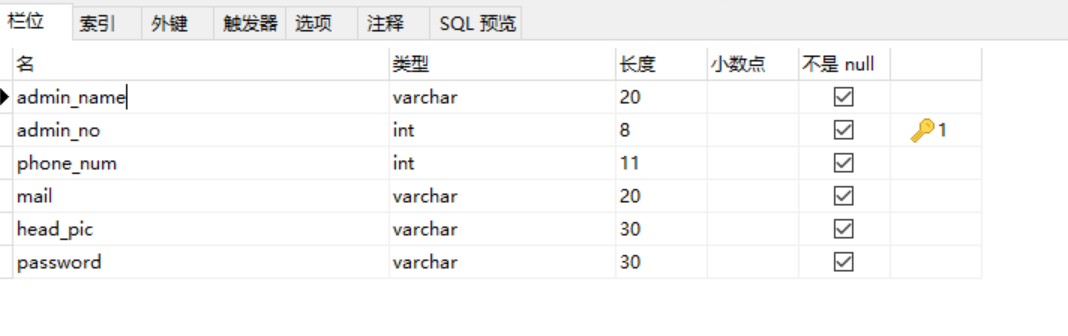
学生表（student）：



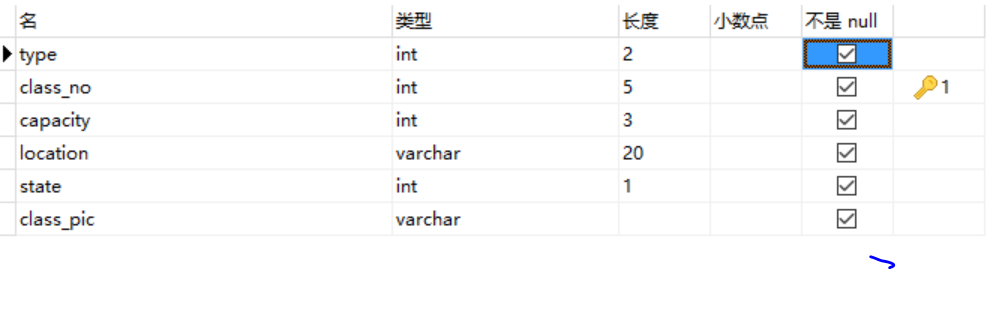
教师表（teacher）：



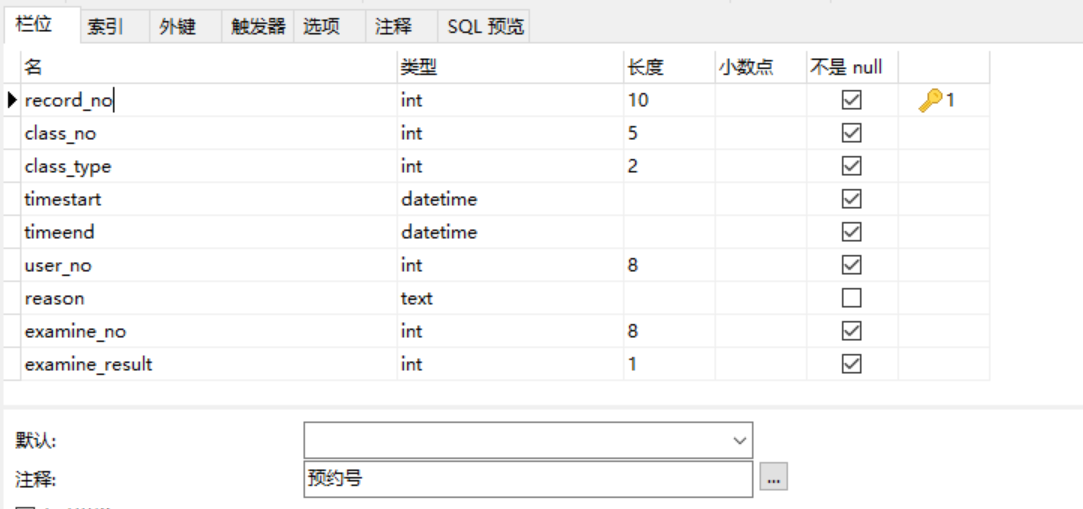
管理员表（admin）：



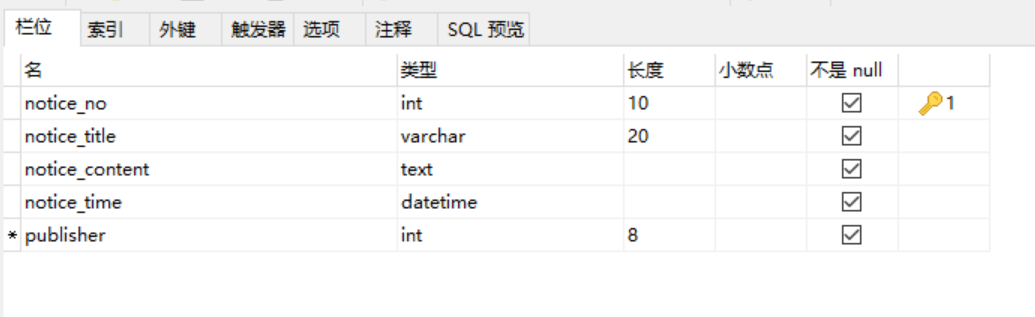
教室表（classroom）：



预约申请记录表（record\_table）:



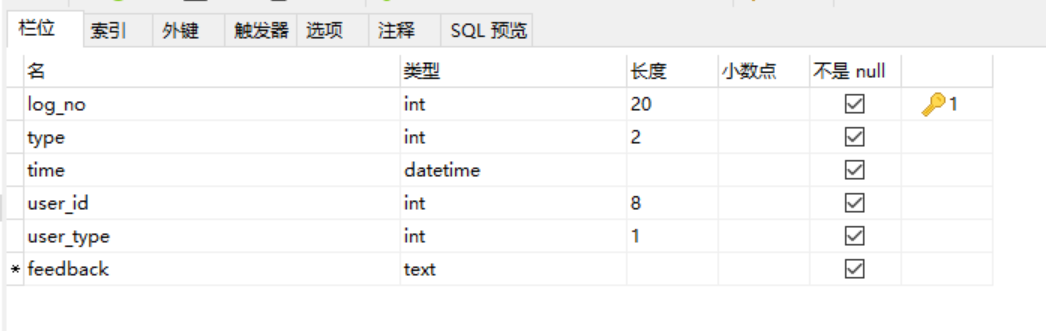
公告表(notice\_table)：



评论表(comment\_table)：



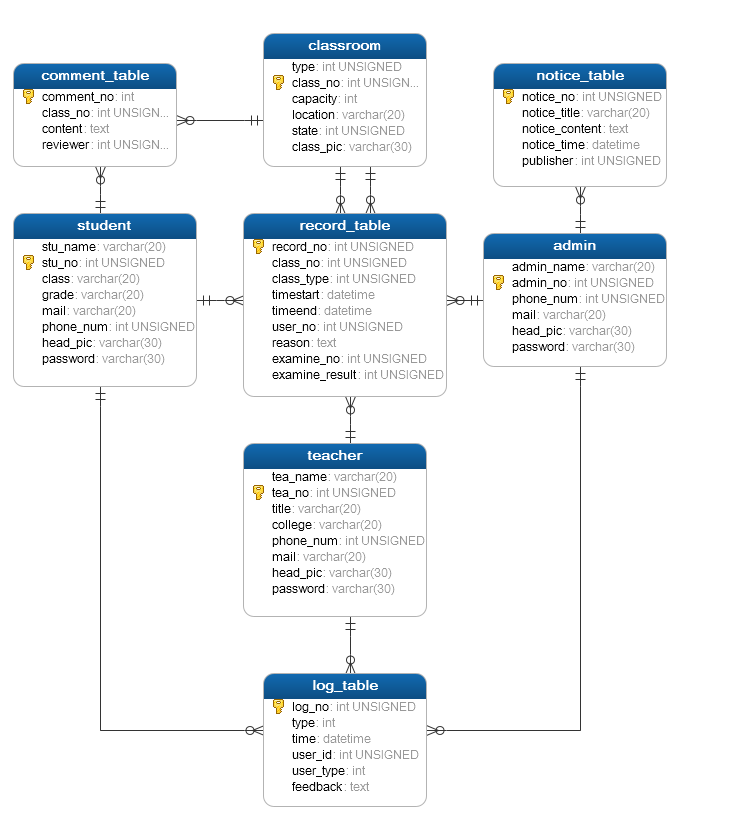
日志表(log\_table):



## 5.3数据结构与程序的关系

本数据库具体通过navicat for mysql来建立，其物理结构如下：

数据库总体结构:



# 6系统出错处理设计

## 6.1出错信息

|  |  |  |
| --- | --- | --- |
| 系统输出信息形式 | 含义 | 处理方法 |
| 服务器访问繁忙 | 由于网络速较差忙，或者服务器访问量较大，数据库软件繁忙，链接数据库配置不正确等一些因素引起的数据库连接不上 | 等待连接，提示返回前一界面 |
| 账号已在另一台主机登录 | 同一账号不能在两台主机上同时登录 | 对该账号进行登陆锁定，退出登录后释放锁定，可以在其他主机登录 |
| 重复插入提示 | 由于网速较差等原因，导致客户重复执行多次相同数据库操作 | 数据库查重，提示用户重复操作 |
| 重新输入密码或者修改密码 | 用户密码输入错误 | 提示用户重新输入密码或者显示修改密码链接，用户修改密码 |

## 6.2补救措施

故障出现后的补救措施:

1、用户造成的数据库冗余或更新失败，由管理员等超级用户修复数据库

2、由于服务器访问量过大造成服务器崩溃，管理员升级服务器或者限制服务器访问人数

# 第二部分     详细设计

# 1引言

## 1.1编写目的

第一部分总体设计部分，已解决了该系统在需求分析部分的程序模块设计问题。包括将该系统划分成若干个模块、模块结构设计、各个模块之间的接口定义、模块之间传信息传递模式，以及数据结构等。

按照软件工程瀑布模型，接下来应该进行软件工程生命周期的详细设计部分，详细设计报告中将对如何具体实现该系统进行阐述。

在本阶段中，过程设计和人机界面设计是主要任务。确定应该如何具体地实现所要求的系统，设计出程序的“蓝图”，从而在编码阶段可以根据这个蓝图写出实际的程序代码。

主要任务有：参考在需求分析报告中所描述的数据、功能、运行、非功能需求，并根据总体设计部分所确定的结构设计，接口设计和数据设计，逐个实现软件系统模块的程序。

## 1.2项目背景

a，软件系统名称：基于B/S结构的会议室教室预约管理系统

b，该项目选题于《软件工程课程设计》第一个课题“基于B/S结构的会议室教室预约管理系统”

c，此次项目是以计算机与通信工程学院的会议室教室预约为对象，分析、设计并实现一个预约管理的网络电子化系统，方便教师、学生进行会议室、教室的借用。

d，在进行详细设计的说明书的编写前，软件课程设计小组已经完成该项目的需求分析和总体设计部分，对该软件系统的需求和结构有了详细说明

e，本部分将进行详细的物理设计、算法设计、人机界面设计，程序数据结构设计，数据库设计等。

## 1.3定义

1.Github：一个面向开源及私有软件项目的托管平台。

2. API：（Application Programming Interface,应用程序编程接口）是一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力，而又无需访问源码，或理解内部工作机制的细节。

## 1.4参考资料

1.软件工程导论（第六版）张海藩 编著 /2010-07-01 /清华大学出版社

2.《软件工程详细设计模板》百度百科/文库/知道

3.已提交的《基于B/S结构的会议室教室预约管理系统需求分析报告》

4. 《基于B/S结构的会议室教室预约管理系统总体&详细设计报告》第一部分 总体设计部分

5. CSDN (Chinese Software Developer Network) 中国最大的IT社区和服务平台

6. 《PHP和MySQL Web开发(原书第4版)》机械工业出版社

# 2设计概述

## 2.1运行环境概述

### 2.1.1服务器环境

（1）采用Linux系统的优秀发行版Centos系统作为服务器系统。 Linux是免费使用和自由传播的类Unix操作系统，基于POSIX和UNIX的多用户、多任务、支持多线程和多CPU。它能运行主要的UNIX工具软件、应用程序和网络协议。其中centos是使用最广的发行版，版本优秀、纯开源免费、社区活跃、使用者多。服务器采用阿里云的云服务器，简单高效、处理能力可弹性伸缩的计算服务，帮助我们快速构建更稳定、安全的应用，提升运维效率，降低 IT 成本，使我们更专注于核心业务创新。

（2）采用LAMP组合架构的技术，即是： Linux + Apache + Mysql / MariaDB + Perl/PHP/Python是一组常用来搭建动态网站或者服务器的开源软件，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，是比较强大的Web应用程序平台。

（3）选用Apache作为Web服务器软件。Apache是世界使用排名第一的Web服务器软件。它可以运行在几乎所有广泛使用的计算机平台上，由于其跨平台和安全性被广泛使用，是最流行的Web服务器端软件之一。它快速、可靠并且可通过简单的API扩充，将Perl/Python/PHP等解释器编译到服务器中。并且其开源的特性，使得开发人员可以定制个性化的服务，强大的开源社区有许多潜在的可能问题都可以解决。

（4）使用MySQL关系型数据库管理系统。在WEB应用方面MySQL是最好的关系数据库管理系统应用软件之一。 MySQL是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。同样其开源的特性为我们提供了大量便于部署以及问题解决的资料。

（5）脚本语言使用PHP。PHP是通用开源脚本语言，语法吸收了C语言、Java和Perl的特点，拥有自己独特的优势。优雅简捷的语法，与Mysql和apache的完美结合，成为了快速搭建高效稳定服务器的不二之选。

（6）LAMP是一种互联网领域广泛搭配使用的技术架构，它们的配合是非常适合互联网类产品的，如网站，网页，或者后台系统等。

### 2.1.2客户端环境

（1）客户端操作系统可以是Windows XP及以上操作系统，Linux，OS等等

（2）浏览器可以是Internet Explorer 简称ie，google chrome浏览器，firefox浏览器，safari浏览器，opera浏览器等

## 2.2开发工具概述

（1）Dreamweaver网页制作软件，该软件已成为专业级网页制作程序，支持HTML、CSS、PHP、JSP以及ASP等众多脚本语言的语法着色显示，同时提供了模板套用功能，支持一键式生成网页框架功能。是初学者或专业级网站开发人员必备之选择工具。

（2）PS（Photoshop）图像处理软件：用于对网页图片进行润色或特殊效果处理，是一款网页制作必备之软件。

（3）Zend Studio是Zend Technologies公司开发的PHP语言集成开发环境(IDE)。除了有强大的PHP开发支持外也支持HTML、js、CSS，但只对PHP语言提供调试支持

（4）Sublime Text 是一个代码编辑器，也是HTML和散文先进的文本编辑器。

（5）Navicat for MySQL是一套管理和开发MySQL或MariaDB的理想解决方案，支持单一程序，可同时连接到MySQL和MariaDB。这个功能齐备的前端软件为数据库管理、开发和维护提供了直观而强大的图形界面，给MySQL或MariaDB新手以及专业人士提供了一组全面的工具。

## 2.3基本设计原理概述

B/S架构软件系统基本设计原理：

首先在PHP程序中获取数据库中存储的数据，再通过加载模板引擎将 数据分配出去，然后将模板文件再通过模板引擎加载并处理后输出。所以PHP程序只是创建动态数据，加载模板引擎并将动态数据分配给模板，完成了PHP程序的工作。而模板的设计也只需要页面设计人 员独立完成，使用HTML、CSS及JavaScript等前台页面设计语言编写。另外，在模板文件中还需要使 用模板引擎可以解析的标记，将PHP中分配过来的动态数据在模板中引用。

## 2.4基本开发技术概述

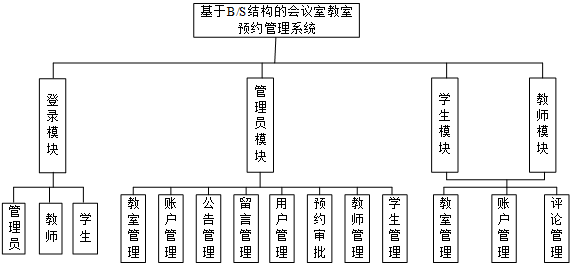
1、B/S(Browser/Server)结构即浏览器和服务器结构。它是随着Internet技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户工作界面是通过WWW浏览器来实现，极少部分事务逻辑在前端(Browser)实现，但是主要事务逻辑在服务器端(Server) 实现，形成所谓三层3-tier结构。这样就大大简化了客户端电脑载荷，减轻了系统维护与升级的成本和工作量，降低了用户的总体成本。以目前的技术看，局域网建立B/S结构的网络应用，并通过 Internet/Intranet模式下数据库应用，相对易于把握、成本也是较低的。

2、PHP是一种HTML内嵌式的语言，PHP与微软的ASP颇有几分相似，都是一种在服务器端执行的嵌入HTML文档的脚本语言，语言的风格类似于C语言，现在被很多网站的编程人员广泛运用。 PHP独特的语法混合了C、Java、Perl以及PHP自创新的语法，它可以比CGI或者Perl更快速的执行动态网页。PHP是将程序嵌入到 HTML文档中去执行，执行效率比完全生成HTML标记的CGI要高许多;与同样是嵌入HTML文档的脚本语言Javascript相比，PHP 在服务器端执行，充分利用了服务器的性能；PHP执行引擎还会将用户经常访问的PHP程序驻留在内存中，其他用户再一次访问这个程序时就不需要重新编译程序了，只要直接执行内存中的代码就可以了，这也是PHP高效率的体现之一。

# 3模块详细设计

基于B/S结构的教室管理预约系统的总体设计部分，已经将系统划分为子模块，下面对各个子模块进行功能、性能、算法、输入、输出等的详细设计。

下面是模块之间的总体结构图：



## 3.1登录模块

### 1功能描述

1、登录时用户需提供账号，密码，验证码和用户身份。用户的账号及密码会被存储于数据库中，以用作登录校验，密码修改，教室预约等。

2、账户每次登录系统，系统的日志管理系统都会记录登录时信息，包括用户账号，登录时间，登录地点等。

3、用户首次登陆时有初始密码为“000000”或者身份证后六位，用户可在首次登录后修改密码

4、用户的账号为标示用户的唯一id，用户的部分个人信息学在首次登录之前由管理员录入系统，如账号（即学号），电话号码，身份证号，初始密码，用户类型等

5、用户点击登录按钮，返回登录失败或者登录成功进入下一个页面

### 2性能

登录过程总需要进行密码的验证功能，需要用户提交密码，系统对数据库保存密码字符串进行解密对比，不同的用户账号密码加密结果不同，具有随机性。

### 3输入项

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入媒体 | 输入数据来源 | 输入方式 | 安全保密性 | 数量 | 频度 |
| 账号 | 主键 | int | 8 | 键盘，鼠标 | 用户 | 手动 |  | 待定 | |
| 密码 |  | Varchar |  | 键盘，鼠标 | 用户 | 加密算法 |
| 账号类型 |  | Int | 2 | 鼠标 | 用户 |  |

### 4输出项

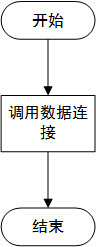
登陆成功：跳转用户首页（根据用户类型加载首页）

输出：账号，用户头像

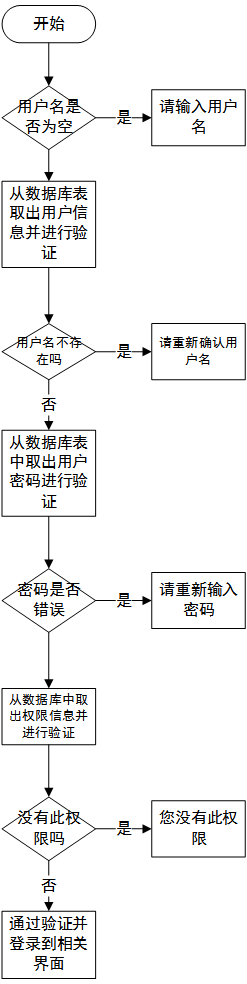
登陆失败：提示账号或密码错误，提示找回密码或者返回登陆界面

### 5设计方法（算法）

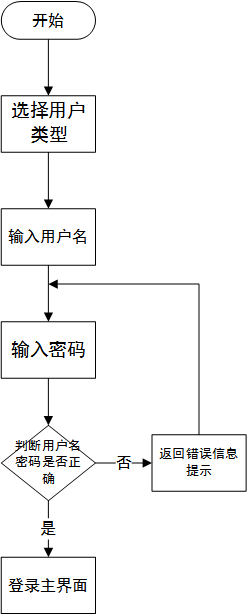
1. 登录窗口事件



1. 登录单击事件



### 6流程逻辑



### 7相关数据库描述

该过程涉及到teacher表中的tea\_no,password字段,student表中的stu\_no,password字段， admin表中的admin\_no,password字段。

（数据表详细字段设计参考总体设计部分）

### 8存储分配

服务器自动分配内存，将数据存入数据库

### 9测试计划

黑盒测试：

设计测试用例：

|  |  |  |
| --- | --- | --- |
| 输入1 | 正确 | 不正确 |
| 账号 | √ |  |
| 密码 | √ |  |
| 用户类型 | √ |  |
| 输入2 | 正确 | 不正确 |
| 账号 | √ |  |
| 密码 |  | √ |
| 用户类型 | √ |  |
| 输入1 | 正确 | 不正确 |
| 账号 |  | √ |
| 密码 |  | √ |
| 用户类型 |  | √ |

### 10尚未解决问题

密码的具体加密问题有待商榷

## 3.2客户信息修改模块

### 1功能描述

1、用户登录成功，点击个人中心，打开信息修改界面

2、弹出信息修改表单，输入账户信息，按确认后提交内容

3、弹出“修改成功”或“失败”提示，并返回个人信息界面

### 2性能

对于不同的用户有不同的修改信息权限，管理员作为超级用户对教师和学生的信息有修改权限

### 3输入项

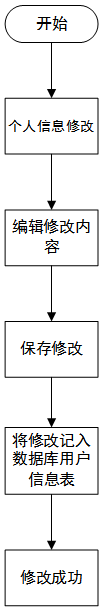
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入媒体 | 输入数据来源 | 输入方式 | 安全保密性 | 数量 | 频度 |
| 账号 | 主键 | int | 8 | 系统 | 数据库 | 自动 | 无 | 待定 | |
| 用户姓名 |  | Varchar | 20 | 键盘，鼠标 | 用户 | 手动 |
| 电话号码 |  | Int | 2 | 键盘，鼠标 | 用户 |
| 邮箱 |  | Varchar | 30 | 键盘，鼠标 | 用户 |
| 头像 |  | Varchar | 30 | 键盘，鼠标 | 用户 |

### 4输出项

修改成功：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入媒体 | 输入数据来源 | 输入方式 | 安全保密性 | 数量 | 频度 |
| 账号 | 主键 | int | 8 |  | 数据库  数据库 | 自动 | 无 | 待定 | |
| 用户姓名 |  | Varchar | 20 |
| 电话号码 |  | Int | 2 |
| 邮箱 |  | Varchar | 30 |
| 头像 |  | Varchar | 30 |

### 5设计方法（算法）&流程逻辑



### 6相关数据库描述

该模块使用到admin表，teacher表，student表。

（具体表字段设计见总体设计数据结构设计部分）

### 7存储分配

服务器自动分配内存，数据库存储数据

### 8测试计划

使用黑盒测试：

1、输入正确的信息，提交数据库，查看数据库信息是否更新

2、输入不完整的信息，提交，查看是否提交成功，并查看数据表内容是否修改

3、输入非正确格式的信息，提交，查看是否提交成功，并查看数据表内容是否修改

### 9尚未解决问题

密码修改是否包含在信息修改中仍在商榷

## 3.3教室查询模块

### 1功能描述

1、用户登录成功后，点击教室查询，打开教室查询界面

2、教室查询界面初始化为所有可见教室，用户不输入任何信息直接点击查询或者刷新页面，教室列表依然默认显示所有可见教室

3、可见教室列表根据用户权限和浏览器界面大小显示

4、用户可通过“关键字查询”或者“模式匹配”两种方式找到相关教室

5、“关键字查询”：用户在搜索框输入部分教室名称，数据库自动匹配相关教室显示在搜索框列表

6、“模式匹配”：用户通过选择教室容量，位置，楼层等等教室相关属性，然后点击搜索显示相关教室到教室列表

### 2性能

1、“关键字查询”功能涉及到正则表达式的匹配问题，正则表达式的编写的影响性能的关键

2、“模式匹配”功能涉及到数据库的查询方式，数据库查询语句的设计，数据库结构的设计，数据库视图的设计是影响性能的关键

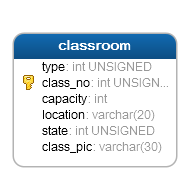
### 3输入项

关键字查询：

输入与教室有关的汉字

模式匹配：

选择classroom表中的相关字段，根据字段值进行匹配



### 4输出项

1、关键字查询：

搜索框出现相关教室称，点击某一条目进入相关教室详细信息页面

2、直接点击搜索，列出符合模式匹配相关教室列表

### 5设计方法（算法）

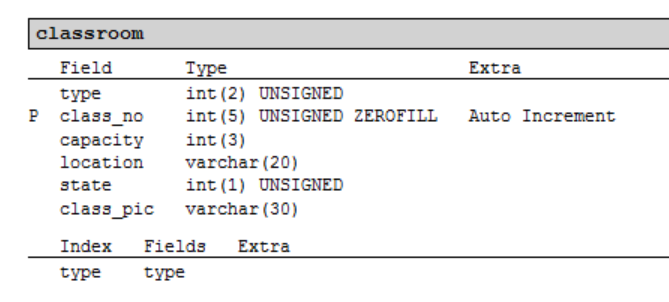


### 6流程逻辑



### 7相关数据库描述

改模块主要涉及classroom表的操作



### 8存储分配

根据服务器自动分配，根据数据库排序规则存储

### 9测试计划

1、白盒测试：对搜索框自动匹配函数进行路径测试，输入不同参数，测试匹配的正则表达式

2、黑盒测试：在搜索框输入相关和无关文字查看显示结果

在匹配模式选择不同的规则查看教室显示列表

### 10尚未解决问题

关键字查询使用到正则表达式匹配功能，对于这部分知识还有所欠缺，需要通过相关文档或者api实现

## 3.4教室预约模块

### 3.4.1功能

1、用户（学生或者老师）通过预约模块对教室进行预约操作

2、用户登录成功后点击预约教室，打开教室预约功能，教室预约功能首先跳转到教室查询界面

3、用户通过查询到符合期望教室，打开教室信息界面

4、用户在教室信息界面点击“预约”，跳转到“教室预约单填写”界面

5、用户填写“教室预约单”，点击“提交”

7、“教室预约单”以表单的形式提交到数据库

6、弹出“预约成功，等待管理员审批”，并跳转回教室信息界面

### 3.4.2性能

1、在安全性需求上，不同用户拥有不同的访问数据库权限，因此在教室预约界面显示不同的数据库列表，可以预约不同的教室

2、在性能要求方面，要求用户可以精确的查找到相关的教室，并完成教室预约表单的提交

3、在人机交互方面，要求系统反馈良好的交互信息，包括出错处理和响应时间反馈等

### 3.4.3输入项

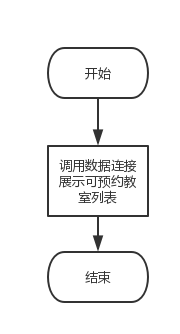
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 输入媒体 | 输入数据来源 | 安全保密性 | 数量 | 频度 |
| 预约号 | 主键 | int | 10 | 自动生成 |  | 数据表 | 无 | 待定 | |
| 预约教室号 | 外键 | Int | 5 | 手动输入 | 键盘、鼠标 | 用户 | 无 |
| 教室类型 | 外键 | Int | 2 | 自动匹配 |  | 数据表 | 无 |
| 预约时间 |  | Datetime |  | 系统生成 |  | 系统 | 无 |
| 预约时间段 |  | Datetime |  | 用户选择 | 鼠标 | 用户 | 无 |
| 用户账号 | 外键 | int | 8 | 系统匹配 |  | 数据表 | 无 |
| 预约原因 |  | text |  | 手动输入 | 键盘、鼠标 | 用户 | 无 |

### 3.4.4输出项

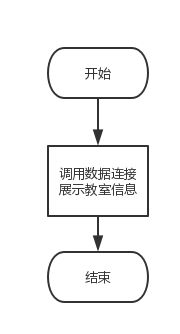
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 输入媒体 | 输入数据来源 | 安全保密性 | 数量 | 频度 |
| 预约号 | 主键 | int | 10 | 自动生成 |  | 数据表 | 无 | 待定 | |
| 预约教室号 | 外键 | Int | 5 | 自动生成 |  |
| 教室类型 | 外键 | Int | 2 | 自动生成 |  |
| 预约时间 |  | Datetime |  | 自动生成 |  |
| 预约时间段 |  | Datetime |  | 自动生成 |  |
| 用户账号 | 外键 | int | 8 | 自动生成 |  |
| 预约原因 |  | text |  | 自动生成 |  |

### 3.4.5设计方法（算法）

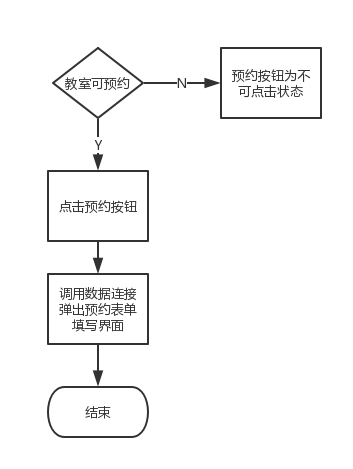
1、打开教室预览窗口界面（单击“预约教室”按钮）



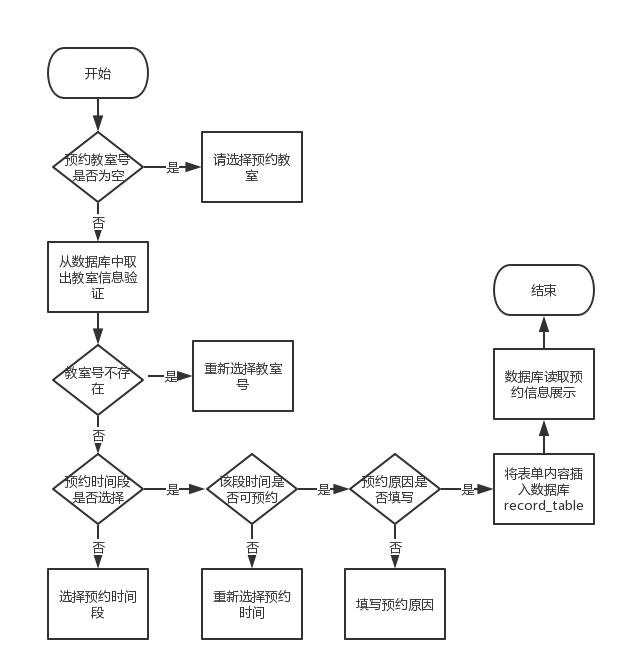
2、打开教室信息界面（单击某一教室条目）



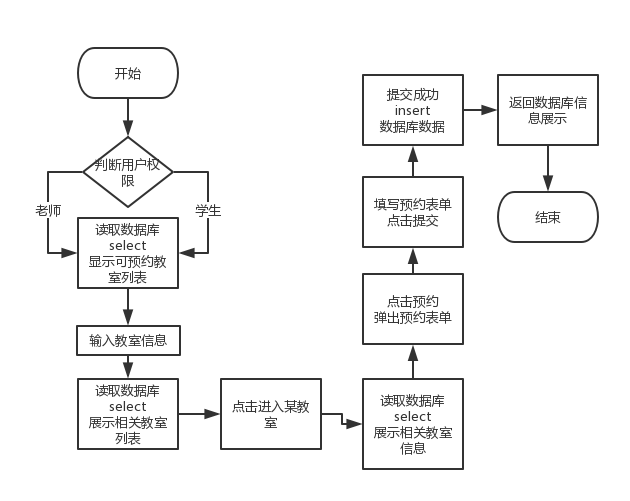
3、点击“预约”，跳转



4、点击“提交”，跳转

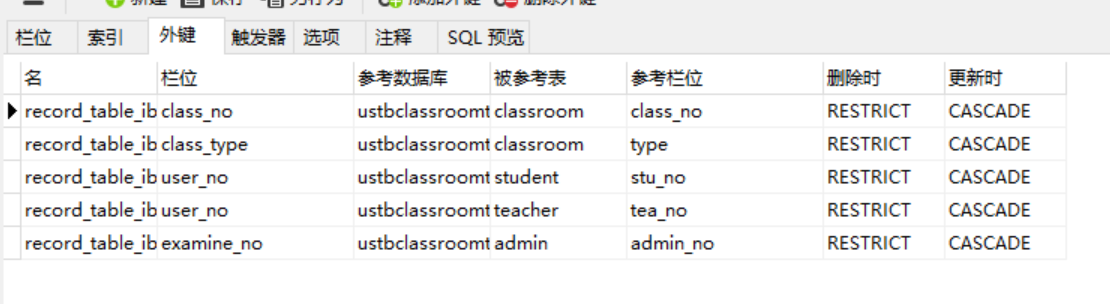


### 3.4.6流程逻辑



### 3.4.7相应数据库描述

该过程涉及到record\_table表,classroom表，student表，teacher表，admin表，record\_table表中的class\_no,class\_type字段作为外键参考classroom表中的class\_no,class\_type字段，record\_table表中的user\_no参考student表，teacher表中的stu\_no,tea\_no,admin\_no字段。



### 3.4.8存储分配

本模块自动分配内存，最后存储在数据库中

### 3.4.9测试计划

首先对本模块的各个子模块函数进行白盒测试，然后进行子模块函数组合进行单元测试，单元测试主要用到黑盒测试技术。输入正确的条件，输出预期结果，输入错误的条件，输出预期结果，并做出错处理。

### 3.4.10尚未解决问题

1、详细设计部分以基本实现该模块在需求和总体设计部分的功能

2、依然可能出现功能不完善，性能不达标的可能

3、数据约束不够完善，数据表视图不够清晰

4、功能单一化，用户界面接口不活泼

## 3.5发布公告模块

### 1功能描述

1、管理员登录成功后点击公告管理，进入公告列表

2、在公告列表界面可对公告进行增删改查

3、点击某一条公告，可对其进行修改和删除功能

4、点击增加公告，弹出公告内容文本框，编辑好后，点击提交，返回公告列表界面，界面自动刷新显示

5、一般用户（学生和老师）在登录之后的首页只能看到最近的几条公告，公告按时间自动下沉

### 2性能

1、在安全性需求上，不同用户拥有不同的访问数据库权限，因此公告界面显示不同的内容

2、在性能要求方面，要求管理员可以精确的查看按照时间顺序排列的公告

3、在人机交互方面，要求系统反馈良好的交互信息，包括出错处理和响应时间反馈等

### 3输入项

输入公告标题，公告内容，发布公告时间，发布者等等

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 安全保密性 |
| 公告标题 |  | varchar | 20 | 管理员 | 无 |
| 公告内容 |  | text |  | 管理员 |
| 发布时间 |  | Datetime |  | 管理员 |
| 发布人 | 外键 | int | 8 | 自动生成 |

### 4输出项

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 输入媒体 | 输入数据来源 | 安全保密性 | 数量 | 频度 |
| 公告号 | 主键 | int | 10 | 自动生成 |  | 数据表 | 无 | 待定 | |
| 公告标题 |  | varchar | 20 | 自动生成 |  |
| 公告内容 |  | text |  | 自动生成 |  |
| 发布时间 |  | Datetime |  | 自动生成 |  |
| 发布人 | 外键 | int | 8 | 自动生成 |  |

### 5设计方法（算法）

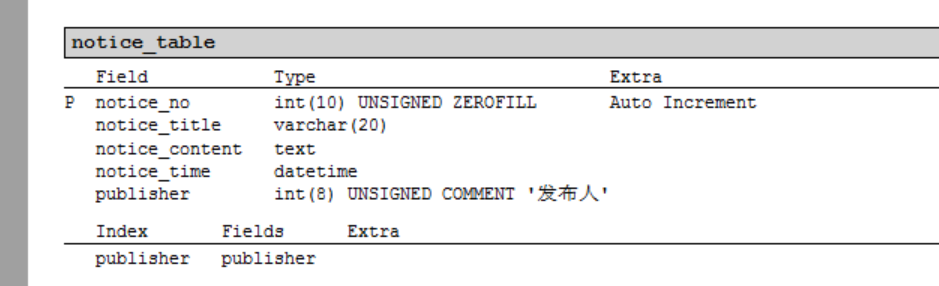


### 6流程逻辑



### 7相关数据库描述

该数据库涉及到的数据表为notice\_table



### 8存储分配

数据表内根据公告号自动递增的形式进行存储公告

### 9测试计划

黑盒测试：输入正确格式的公告，进行提交，检查数据库条目是否有变动，刷新页面显示是否有新公告

输入错误格式公告（没有标题，内容，发布者，发布时间等），点击查看是否可以提交，并查看返回信息

### 10尚未解决问题

没有设置公告的不同级别，根据公告的不同级别进行置顶

## 3.6教室评论模块

### 3.6.1功能描述

1、用户登录成功后点击评论教室，进入教室评论功能

2、点击评论教室后跳转到教室查询界面，查询相关教室，进入教室信息界面

3、在教室信息最下方点击评论，弹出评论文本框

4、输入评论内容，点击提交

5、返回教室信息界面，最下方评论已更新

### 3.6.2性能

评论提交后可及时刷新界面显示最新评论内容

### 3.7.3输入项

用户主要输入评论内容

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 安全保密性 |
| 评论编号 | 主键 | Int | 10 | 自动生成 | 无 |
| 评论内容 |  | text |  | 用户 |
| 评论时间 |  | Datetime |  | 系统生成 |
| 评论人 | 外键 | int | 8 | 自动生成 |

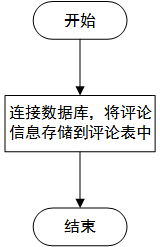
### 3.6.4输出项

评论成功后系统返回教室信息界面，返回评论内容

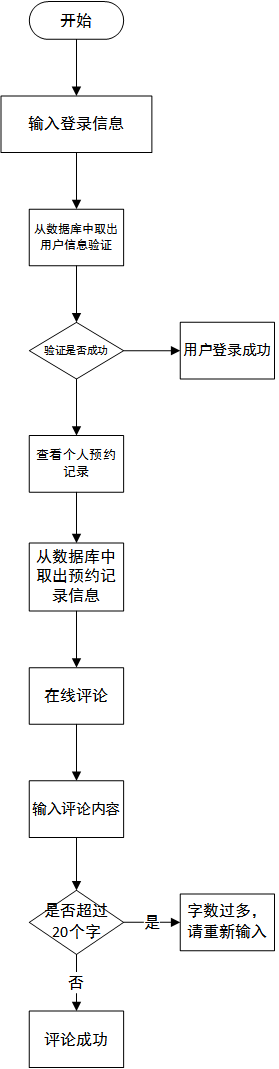
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 安全保密性 |
| 评论编号 | 主键 | Int | 10 | 自动生成 | 无 |
| 评论内容 |  | text |  | 自动生成 |
| 评论时间 |  | Datetime |  | 系统生成 |
| 评论人 | 外键 | int | 8 | 自动生成 |

### 3.6.5设计方法（算法）

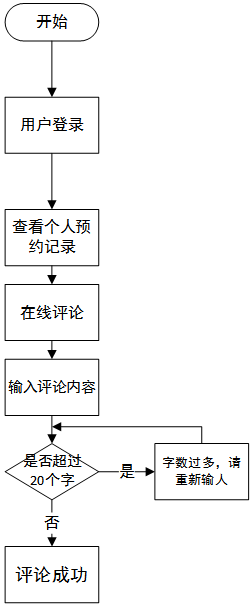
1. 教室评论窗体事件



2.评论事件

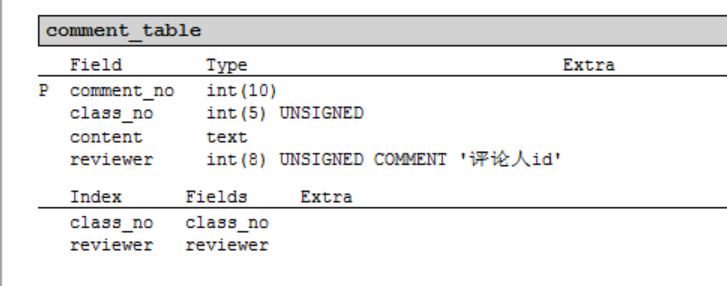


### 3.6.6流程逻辑



### 3.6.7相关数据库描述

该模块主要用到的数据表为comment\_table表



### 3.6.8存储分配

数据表内根据评论号自动递增的形式进行存储评论

### 3.6.9测试计划

黑盒测试：输入正确格式的评论，进行提交，检查数据库条目是否有变动，刷新页面显示是否有新公告

在评论界面点击返回或者取消评论，查看是否运行正常

### 3.6.10尚未解决问题

还无法实现评论的动态更新，需要刷新界面才能显示最新评论

## 3.7教室预约审批模块

### 3.7.1功能描述

1、管理员登录成功后点击教室管理，进入预约管理界面

2、该界面显示所有预约单列表

3、点击某一申请表查看，点击“允许预约”标记该教室在该实践段内的状态为“已预约”，并且系统发送信息到该预约用户的邮箱通知预约成功，返回预约申请列表

4、或者点击“禁止预约”，系统发送信息到该预约用户邮箱通知预约失败

### 3.7.2性能

1、系统允许多个管理员多地同时审核申请表

2、在进行预约审批时如果存在两个以上管理员同时操作同一张申请表，其中一个管理员先执行表单提交后，其他管理员表单提交失败或者显示该申请表已被审核

### 3.7.3输入项

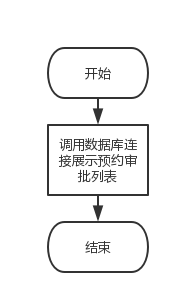
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 输入媒体 | 输入数据来源 | 安全保密性 | 数量 | 频度 |
| 预约号 | 主键 | int | 10 | 自动生成 |  | 数据表 | 无 | 待定 | |
| 审核人账号 | 外键 | int | 8 | 系统生成 |  | 数据表 | 无 |
| 审核结果 |  | Int | 1 | 管理员选择 | 鼠标 | 管理员 | 无 |

### 3.7.4输出项

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 输入媒体 | 输入数据来源 | 安全保密性 | 数量 | 频度 |
| 预约号 | 主键 | int | 10 | 自动生成 |  | 数据表 | 无 | 待定 | |
| 预约教室号 | 外键 | Int | 5 | 自动生成 |  | 数据表 | 无 |
| 教室类型 | 外键 | Int | 2 | 自动生成 |  | 数据表 | 无 |
| 预约时间 |  | Datetime |  | 自动生成 |  | 数据表 | 无 |
| 预约时间段 |  | Datetime |  | 自动生成 |  | 数据表 | 无 |
| 用户账号 | 外键 | int | 8 | 自动生成 |  | 数据表 | 无 |
| 预约原因 |  | text |  | 自动生成 |  | 数据表 | 无 |
| 审核人账号 | 外键 | int | 8 | 自动生成 |  | 数据表 | 无 |
| 审核结果 |  | Int | 1 | 自动身生成 |  | 数据表 | 无 |

### 3.7.5设计方法（算法）

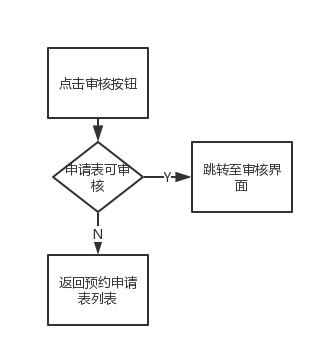
1、打开教室预览窗口界面（教室管理->预约审批）



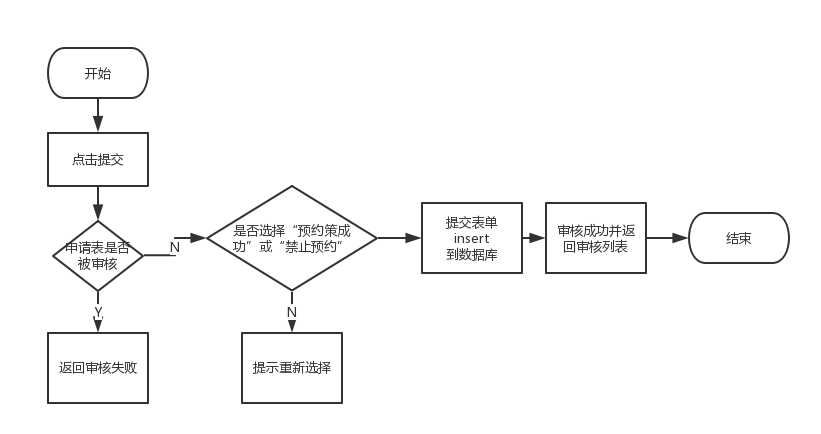
2、打开申请表界面（单击某一预约申请表条目）



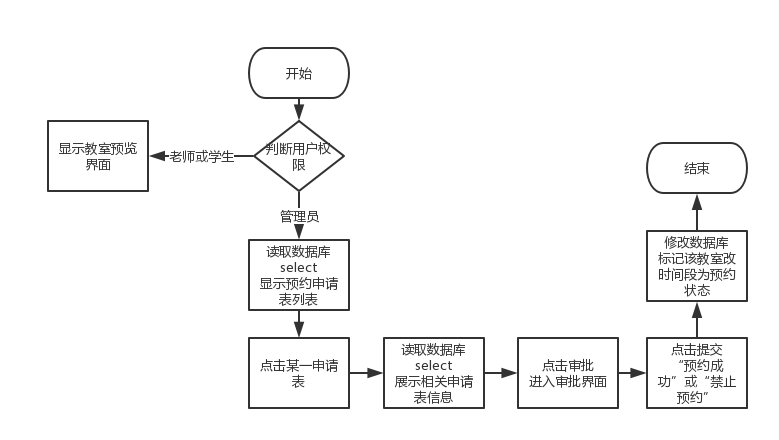
3、点击审核，跳转



4、点击“提交”，跳转

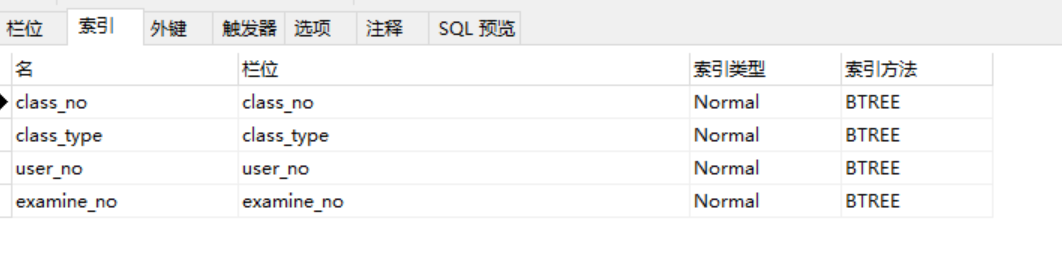


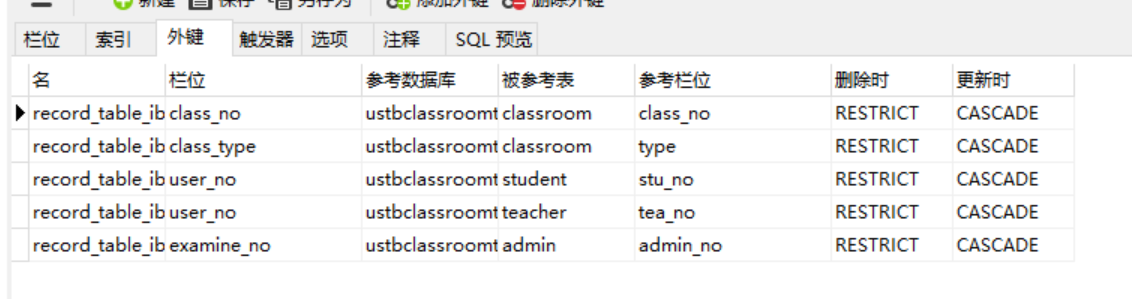
### 3.7.6流程逻辑



### 3.7.7相关数据库描述

该过程涉及到record\_table表,classroom表，student表，teacher表，admin表，record\_table表中的class\_no,class\_type字段作为外键参考classroom表中的class\_no,class\_type字段，record\_table表中的user\_no,examine\_no分别参考student表，teacher表，admin表中的stu\_no,tea\_no,admin\_no字段。





### 3.7.8存储分配

本模块自动分配内存，最后存储在数据库中

### 3.7.9测试计划

首先对本模块的各个子模块函数进行白盒测试，包然后进行子模块函数组合进行单元测试，包括打开申请表列表界面，打开申请表信息界面，提交返回等，单元测试主要用到黑盒测试技术。

### 3.7.10尚未解决问题

在管理员同时审批同一张申请表时的逻辑处理存在疑问，该处理方法有较多的解决方案，暂时没有确定

## 3.8密码丢失找回模块

### 3.8.1功能描述

1、用户在登录界面输入错误的账户名和密码，提示用户找回密码或者返回登陆界面。并在登录界面下面添加链接“找回密码”

2、用户通过email 找回密码，管理员在初始化用户信息时必须填写用户正确的邮箱，否则将不能使用系统所提供的密码重设功能。

3、找回密码时必须由该用户通过邮箱确认。

4、当用户点击找回密码时，弹出找回密码表单。用户输入账号（如果是登录失败后，我们可以用session填充表单内容），新密码，确认新密码，验证码等信息。户点击提交后，开始密码重设功能流程。

5、密码重设时，用户绑定的邮箱会收到一个链接，点击该链接跳转到密码修改成功界面

### 3.8.2性能

1、修改密码第一次输入与第二次输入密码不需刷新页面即可验证是否相同

2、点击修改密码后，服务器端立刻发送链接到邮箱，客户端立刻跳转到邮箱登录界面或者提醒用户查看邮箱

### 3.8.3输入项

找回密码需要输入用户账号，新密码，确认密码，验证码等

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 安全保密性 | 数量 | 频度 |
| 账号 | 主键 | int | 8 | 自动生成/用户输入 | 无 | 待定 | |
| 新密码 |  | Varchar | 30 | 用户输入 | 加密 |
| 确认密码 |  | Varchar | 30 | 用户输入 | 加密 |
| 验证码 |  | Varchar | 5 | 用户输入 | 无 |
| 邮箱 |  | Varchar | 30 | 数据库查找 | 无 |

### 3.8.4输出项

返回密码修改成功，并修改用户表密码为新密码

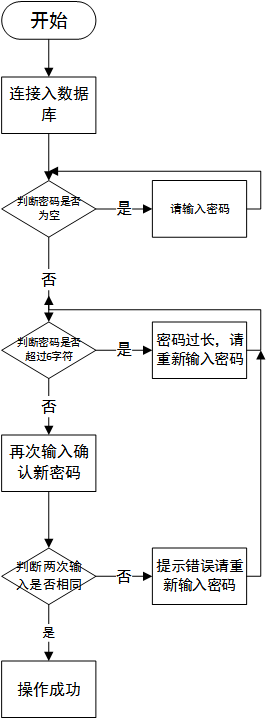
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 有效范围 | 输入方式 | 安全保密性 | 数量 | 频度 |
| 账号 | 主键 | int | 8 | 系统生成 | 无 | 待定 | |
| 新密码 |  | Varchar | 30 | 数据库 | 加密 |

### 3.8.5设计方法（算法）

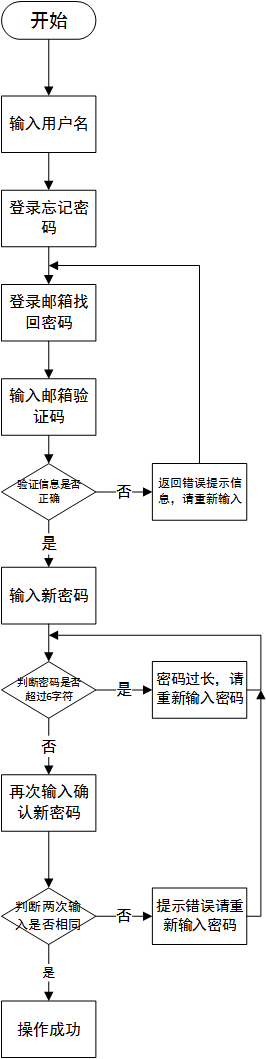
1. 忘记密码窗体事件



1. 按下确认按钮事件

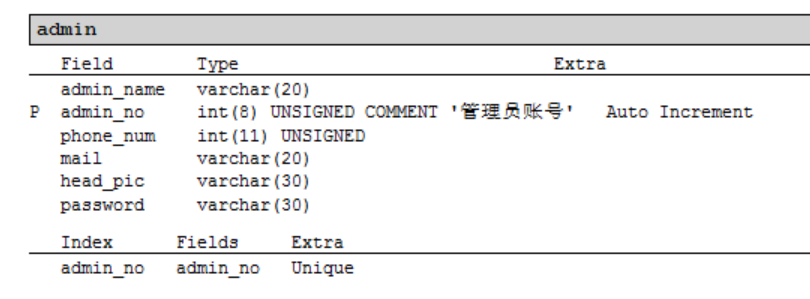


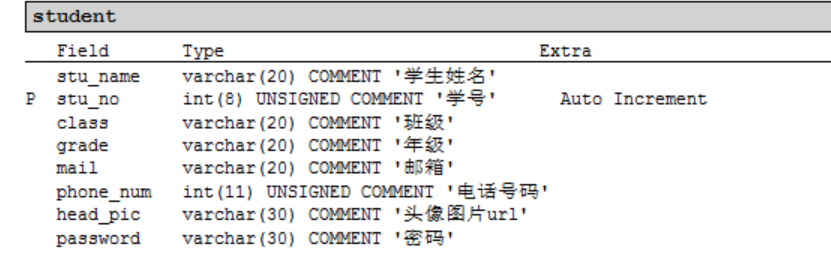
### 3.8.6流程逻辑

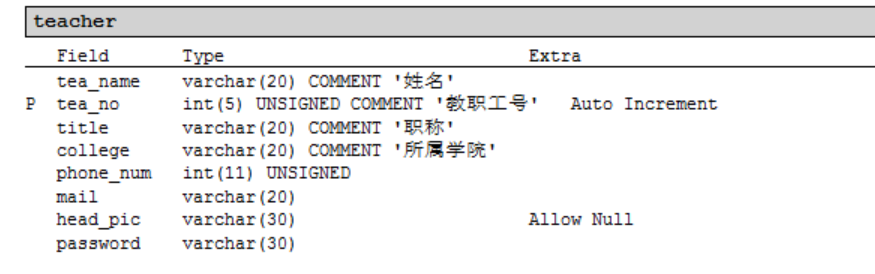


### 3.8.7相关数据库描述

该模块主要用到三张用户表teacher表，student表，admin表







### 3.8.8存储分配

系统根据用户账号查找相关数据表条目，发送验证链接到用户邮箱，修改成功后将用户新密码按照加密格式存入用户表覆盖原来密码

### 3.8.9测试计划

黑盒测试：

1、登录界面输入错误密码，查看是否提示“找回密码”

2、登录时输入密码错误，查看是否提示“重新输入密码”或者“找回密码”

3、点击找回密码跳转到密码修改界面，输入新密码和确认密码不一致，查看是否提示“两次输入密码不一致”

4、两次输入密码一致，点击提交，查看邮箱是否接收到验证链接

5、邮箱点击验证链接，查看密码是否修改成功，查看数据库个人信息密码条目是否更新

### 3.8.10尚未解决问题

用户应该可以使用多种方式进行密码找回，如密码问题，绑定手机短信，人脸识别等等，目前仅支持邮箱验证找回

# 4用户接口设计

## 4.1用户界面设计

### 1、页面统计：

一级页面1个，二级页面1个，三级页面3个，四级页面4个

### 2、页面详细设计：

\*登录页面（1）：

1)顶部：标题、名称。

2)左侧登录：工号/学号栏、密码栏、用户/管理员身份选择、登录按钮。

3)右侧背景图：logo

\*查询教室界面（2）：

1）顶部标题：标题、名称、背景、LOGO。

2）左侧导航：用户姓名、用户组、导航按钮、退出按钮。

3）右侧公告栏：显示最近时间的公告条目

4）中间顶部搜索框：键入相关文字提示相关教室

5）中间下部教室条目：显示当前部分可预约教室

\*教室信息界面（3）

1）顶部标题：标题、名称、背景、LOGO。

2）左侧导航：用户姓名、用户组、导航按钮、退出按钮。

3）右侧公告栏：显示最近时间的公告条目

4）中间教室详细信息：教室号，教室位置，教室容量，预约按钮等

5）中下方评论区

\*预约教室弹框（4）

1） 顶部标题：标题、名称。

2） 中间预约信息：教室号，教室位置，教室容量，预约时间段，预约人等等

\*个人信息界面（3）

1）顶部标题：标题、名称、背景、LOGO。

2）左侧导航：用户姓名、用户组、导航按钮、退出按钮。

3）右侧公告栏：显示最近时间的公告条目

4）中部个人信息：显示账号，头像，联系方式，修改信息按钮等

\*个人信息修改弹框（4）

1） 顶部标题：标题、名称。

2） 中间个人信息修改：账号，头像，联系方式，提交按钮等

\*评论弹框（4）

评论文本输入框：开始，插入，引用，布局，图片等工具菜单

\*公告管理界面（3）

1）顶部标题：标题、名称、背景、LOGO。

2）左侧导航：用户姓名、用户组、导航按钮、退出按钮。

3）中间公告列表：公告号，发布时间，发布人，标题等

4）下方公告页数

\*发布公告弹框（4）

1） 顶部标题：标题、发布人，发布时间。

2） 中间公告内容区：开始，插入，引用，布局，图片等工具菜单

## 4.2用户体验设计

1、在每个主要页面的下方公布开发者电话，邮箱等联系方式，方便用户进行体验反馈和问题反馈

2、制作反馈问卷，在页面下方设置发聩链接，用户可填写用户反馈表