# 教师排课系统需求分析文档

**第 七 组**

目录

[一、引言 4](#_Toc13182)

[1.1编写目的和范围 4](#_Toc8096)

[1.2术语表 4](#_Toc8375)

[1.3参考资料 5](#_Toc31661)

[1.4使用的文字处理和绘图工具 5](#_Toc1031)

[二、设计概述 6](#_Toc17270)

[2.1任务和目标 6](#_Toc20256)

[2.1.1需求概述 6](#_Toc5107)

[第一次需求概述 6](#_Toc31996)

[第二次需求概述 6](#_Toc3718)

[第三次需求概述 6](#_Toc23141)

[2.1.2运行环境概述 7](#_Toc516)

[2.1.3 要求与限制 7](#_Toc864)

[2.1.3.1硬件 7](#_Toc2263)

[2.1.3.2 时间 7](#_Toc17728)

[2.1.3.3 成本预算 7](#_Toc11596)

[2.1.4 详细设计方法与工具 7](#_Toc21054)

[2.1.5 人员分工与安排 8](#_Toc27990)

[三、需求分析 8](#_Toc23225)

[3.1可行性分析 8](#_Toc319)

[3.1.1操作可行性 8](#_Toc399)

[3.1.2 技术可行性 8](#_Toc9512)

[3.1.3经济可行性 9](#_Toc3121)

[3.2详细接口需求分析 9](#_Toc25416)

[四、总体方案确认 9](#_Toc29789)

[4.1系统总体结构确认 9](#_Toc3391)

[4.2 系统详细界面划分 10](#_Toc28837)

[4.2.1 应用系统与支撑系统的详细界面划分 10](#_Toc17666)

[4.2.2 系统内部详细界面划分 10](#_Toc29225)

[五、系统详细设计 10](#_Toc14891)

[5.1 功能结构图 10](#_Toc18737)

[5.2 系统结构设计及子系统划分 12](#_Toc9899)

[数据流图 12](#_Toc28555)

[系统功能模块详细设计 13](#_Toc11827)

[登录界面结构设计: 13](#_Toc11987)

[教师填写信息： 14](#_Toc7638)

[排课： 14](#_Toc18968)

[调课申请： 15](#_Toc13733)

[留言功能： 15](#_Toc19826)

[六、开发和测试生产环境说明 16](#_Toc13407)

[6.1 开发环境 16](#_Toc31624)

[6.2 测试及产品环境 16](#_Toc23914)

[软件测试 16](#_Toc20511)

[七、模块设计 17](#_Toc13220)

[7.1 E-R图 17](#_Toc2073)

[7.2 用例图 18](#_Toc14301)

[7.3功能设计说明 19](#_Toc25242)

[7.3.1 模块1：登录 19](#_Toc12274)

[7.3.1.1 子模块 20](#_Toc11310)

[7.3.1.1.1设计图 20](#_Toc25566)

[7.3.1.1.2 功能描述 20](#_Toc27672)

[7.3.1.1.3 输入数据 20](#_Toc26686)

[7.3.1.1.4 输出数据 20](#_Toc3388)

[教师部分 21](#_Toc2593)

[7.3.2 模块2：教师提交部分 21](#_Toc26435)

[7.3.2.1 子模块 21](#_Toc19990)

[7.3.2.1.1设计图 22](#_Toc29158)

[7.3.2.1.2 功能描述 22](#_Toc17612)

[7.3.2.1.3 输入数据 23](#_Toc31619)

[7.3.3 模块3：教师申请调课 23](#_Toc18903)

[7.3.3.1 子模块 23](#_Toc17767)

[7.3.3.1.1设计图 24](#_Toc9538)

[7.3.3.1.2 功能描述 24](#_Toc15228)

[7.3.3.1.3 输入数据 25](#_Toc3871)

[7.3.3.1.4 输出数据 25](#_Toc27185)

[管理员部分 25](#_Toc19513)

[7.3.4 模块4：管理员排课部分 25](#_Toc11480)

[7.3.4.1 子模块 25](#_Toc5509)

[7.3.4.1.2 功能描述 27](#_Toc8147)

[7.3.4.1.3 输入数据 28](#_Toc18451)

[7.3.4.1.3输出数据 28](#_Toc32070)

[7.3.4.1.4核心代码 28](#_Toc2440)

[7.3.5模块5：管理员联系部分 32](#_Toc32)

[7.3.5.1 子模块 32](#_Toc11627)

[7.3.5.1.2 功能描述 33](#_Toc2242)

[7.3.5.1.3 输入数据 33](#_Toc18886)

[7.3.6 模块6：时间设置 33](#_Toc22414)

[7.3.6.1 子模块 33](#_Toc7662)

[7.3.6.1.2 功能描述 33](#_Toc5400)

[7.3.6.1.3 输入数据 34](#_Toc17519)

[八、接口设计 34](#_Toc23772)

[8.1内部接口（暂无） 34](#_Toc12652)

[8.2 外部接口（暂无） 34](#_Toc26275)

[8.2.1 接口说明 34](#_Toc18680)

[8.2.2 调用方式 34](#_Toc3064)

[九、数据库设计 34](#_Toc29998)

[9.1 课程表（course） 35](#_Toc4699)

[9.2 课程表（KB1） 36](#_Toc8724)

[9.3 课程表（KB2） 36](#_Toc10271)

[9.4 工学院表 37](#_Toc30091)

[9.5 教务表 38](#_Toc12559)

[9.6 留言表（LIUYAN） 38](#_Toc3040)

[9.7联系表（connection） 40](#_Toc10566)

[9.8 回复表 40](#_Toc1462)

[9.9 规定时间（Time） 40](#_Toc22510)

[十、系统安全保密设计 41](#_Toc23133)

[10.1说明 41](#_Toc6329)

[10.2设计 41](#_Toc14810)

[10.2.1 身份验证部分 41](#_Toc31308)

[十一、系统出错处理 41](#_Toc25636)

[十二、设计和开发规范 41](#_Toc8450)

[12.1 数据库设计规范 41](#_Toc18281)

[12.2 .NET编码规范 41](#_Toc6307)

[12.3 代码目录结构 43](#_Toc1365)

[十三、 附录 43](#_Toc21772)

## 一、引言

## 1.1编写目的和范围

使开发人员在完成概要设计说明书的基础上,完成概要设计的各项功能规定，为程序员写出实际的程序代码提供依据。它是软件设计阶段所有任务和所有相关人员所需的参考资料。预期的读者为系统的开发者、使用者和指导老师。规范化软件的生产，给软件的设计、编码、测试、维护等提供依据。

## 1.2术语表

|  |  |  |
| --- | --- | --- |
| 序号 | 术语或缩略语 | 说明性质 |
| 1 | KB1 | 数据库-课表 |
| 2 | pass | 登录密码 |
| 3 | xueyuan | 学院 |
| 4 | number1 | 排序号 |
| 5 | id | 用户名 |
| 6 | yishiduan | 第一时间段开始时间 |
| 7 | yishijian | 第一时间段结束时间 |

## 1.3参考资料

<https://blog.csdn.net/cyg2001/article/details/52164207>

https://wenku.baidu.com/view/f2344a0e763231126fdb1105.html

## 1.4使用的文字处理和绘图工具

文字处理软件：word

绘图工具：visio

## 二、设计概述

## 2.1任务和目标

## 2.1.1需求概述

## 第一次需求概述

编写排课系统，并为工学院五位老师进行排课，每位老师负责教授的课程由教务给出。老师只需提交三个志愿时间段，然后系统就会根据课程优先级（管理员设置）和志愿时间进行智能排课，并最终生成课表。

## 第二次需求概述

编写排课系统，并为工学院五位老师进行排课，每位老师负责教授的课程由教务给出。老师只需提交三个志愿时间段，然后系统就会根据课程优先级（管理员设置）和志愿时间进行智能排课，并最终生成课表，并且可按单双周查看。

## 第三次需求概述

编写排课系统，并为工学院五位老师进行排课，每位老师负责教授的课程由教务给出。老师只需提交三个志愿时间段，然后系统就会根据课程优先级和志愿时间进行智能排课，并最终生成单双周课表。

并设置四个合课班级（两个班一起上课）一共10节课，如存在公选课3门，分别为高数、马原、毛概（每门课上2次，每节课2个班上），剩下四门是每个班单独上，可以显示各个班的课表。

## 2.1.2运行环境概述

.NET平台的核心是NET Framework，它为.NET平台下应用程序的运行提供基本框架。

## 2.1.3 要求与限制

## 2.1.3.1硬件

一台能连网配置良好的电脑，一位熟悉电脑操作的人员。

## 2.1.3.2 时间

工作人员须在2018.10.1-2018.12.18期间完成对系统的设计和测试。

## 2.1.3.3 成本预算

预算金额10000元

## 2.1.4 详细设计方法与工具

本系统采用C#语言，使用Visual Stdio.NET为主要开发工具。

## 2.1.5 人员分工与安排

文档编写：陈雪宇，褚峻仪，张煜茜；

代码、算法：羊亦可，谢子澎；

界面设计：翟伟；

后台数据库链接、代码搜索：范新宇。

## 三、需求分析

## 3.1可行性分析

可行性分析，系统可行性研究是系统发展过程中的一项重要的工作，其目的就是订出一套选择标准和一套选择程序及组成一个有效能的决策组织，如此，使用者便能依据正确的顺序解决相关的问题。下面将从操作可行性、技术可行性、经济可行性三个方面对排课系统进行分析。

## 3.1.1操作可行性

对于新系统的实施运作，由于组织人员都相当配合，所以运作方面没有太大的困难。而由于作业流程和程序设计都很详细规划，所以方案的运作都能按照作业程序来进行。而硬件方面，由于现有的PC和打印机极为普遍，所以没有太大的问题。

## 3.1.2 技术可行性

在软件技术方面，我们采用SQL Server 2008作为后台数据库，使用C#开发基于Visual Studio 2010的排课系统，因为这种技术已实用化，设计的页面精美，功能强大。

## 3.1.3经济可行性

经济可行性是对将要开发的系统的成本效益进行的量度，可以说是可行性分析中最重要的一个，以低成本追求高效率是我们的目标。排课系统是通过计算机来自动地解决各教师合理有效利用时间的问题，避免教师在上课时间上的冲突，使排课时间分配均匀，这就大大的节省了时间，提高了工作效率，避免了人力、物力、财力的浪费，因此本系统可行。

## 3.2详细接口需求分析

分为两个界面，各自接口如下：

教师界面：教学安排表、排课信息录入、课表显示、导出打印（单双周、班级）、联系管理员、教师信息（联系方式）、管理员回复

管理员：课程信息录入、一键排课、联系教师、课表显示（单双周、班级）、导出课表（单双周、班级）、同意或驳回调课申请

## 四、总体方案确认

着重解决系统总体结构确认及界面划分问题。

## 4.1系统总体结构确认

对系统组成、逻辑结构及层次进行确认，对应用系统、支撑系统及各自实现的功能进行确认，细化集成设计及系统工作流程，特别要注意因软件的引进造成的系统本身结构和公司其他系统的结构变化。

## 4.2 系统详细界面划分

## 4.2.1 应用系统与支撑系统的详细界面划分

应用系统与支撑系统之间的界面包括系统主服务器与其他服务器的服务范围及访问方式，网络及数据库对应用系统的支撑方式，全局数据的管理与存取方式等。

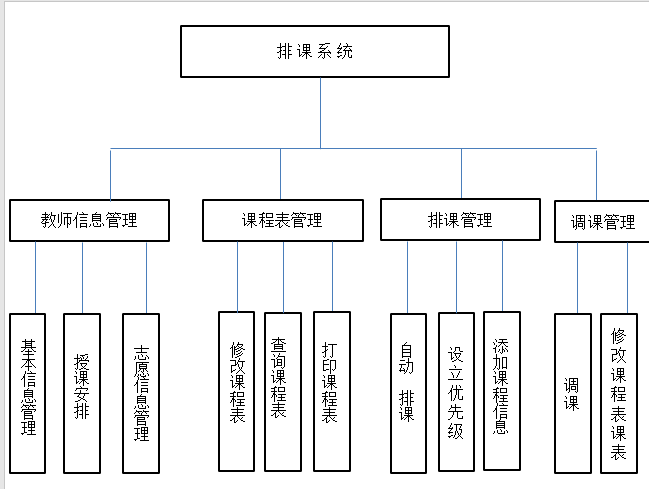
## 4.2.2 系统内部详细界面划分

系统各功能之间的界面包括覆盖范围，模块间功能调用涉及到的系统模块及方法，全局数据格式，系统性能要求等。

## 五、系统详细设计

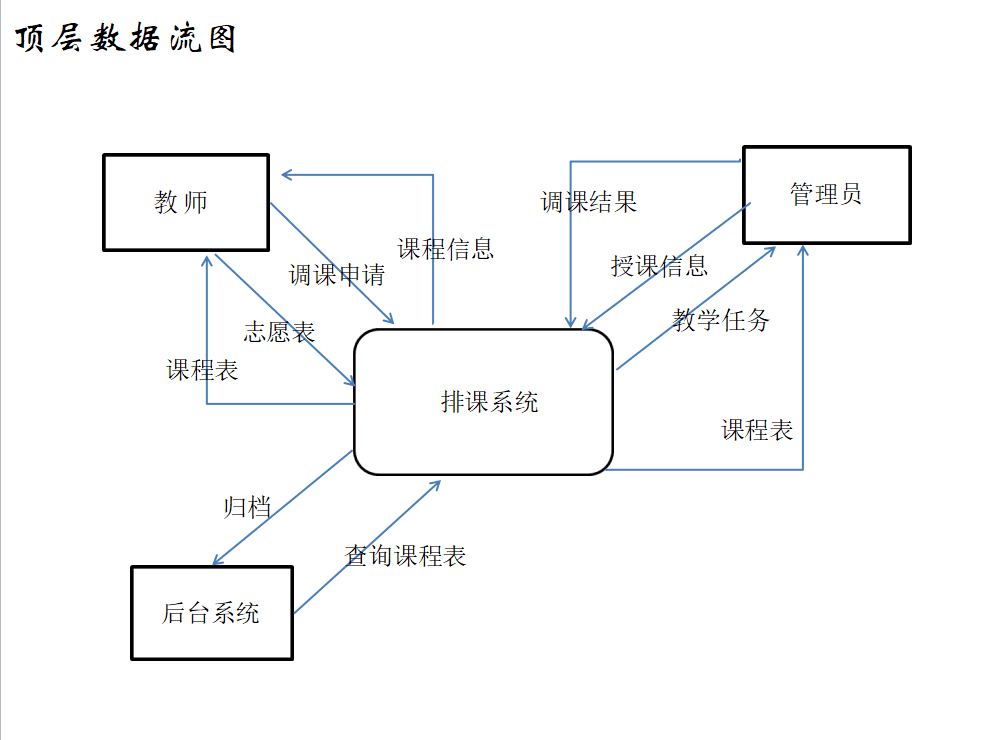
## 5.1 功能结构图

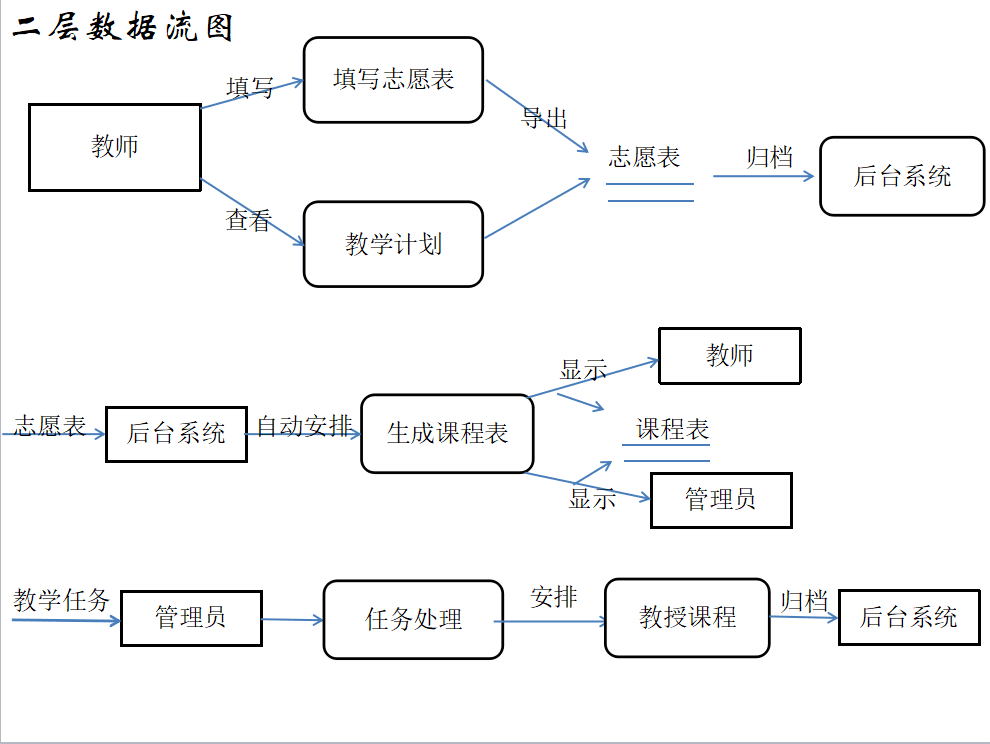
排课系统由四个部分：教师信息管理、课程表管理、排课管理、调课管理组成。



## 5.2 系统结构设计及子系统划分

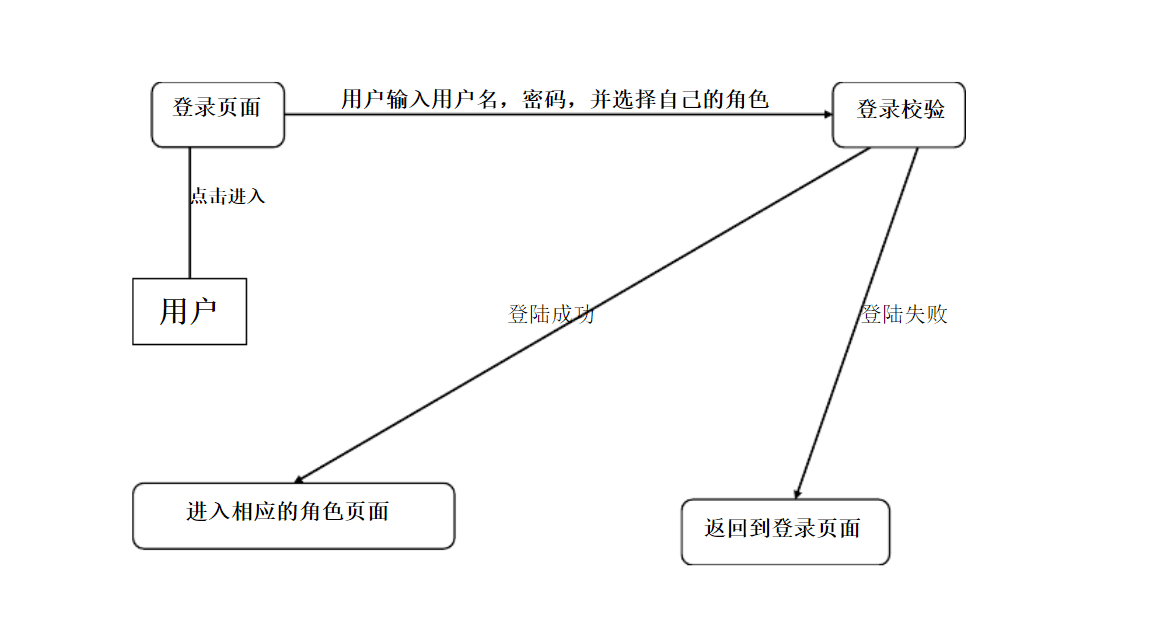
## 数据流图



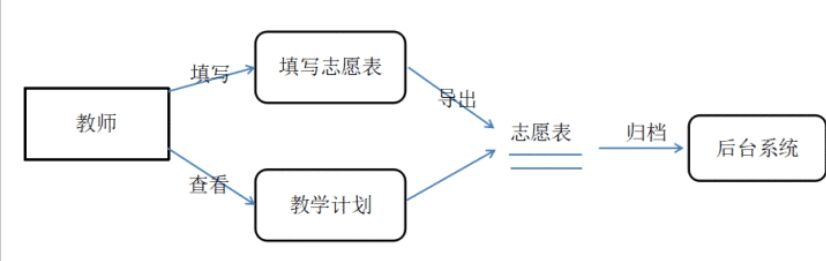


## 系统功能模块详细设计

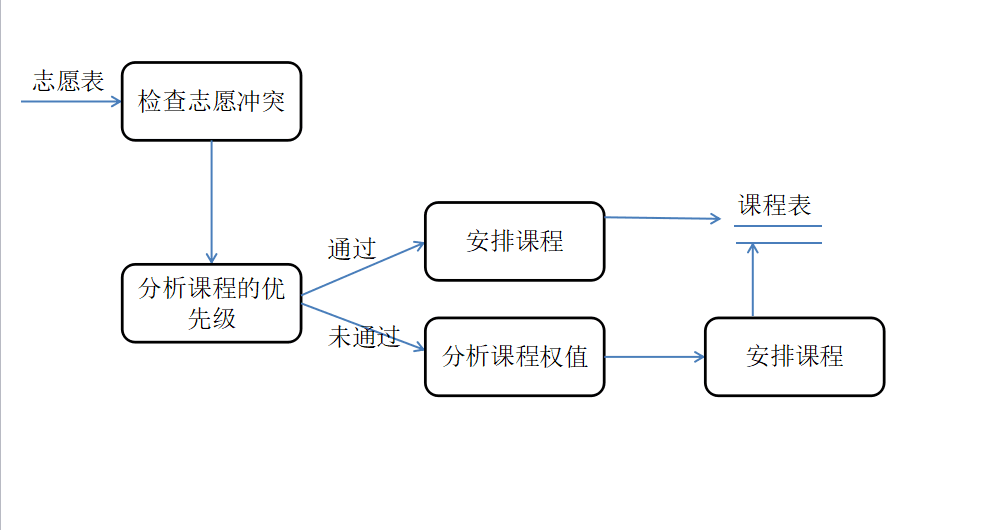
## 登录界面结构设计:



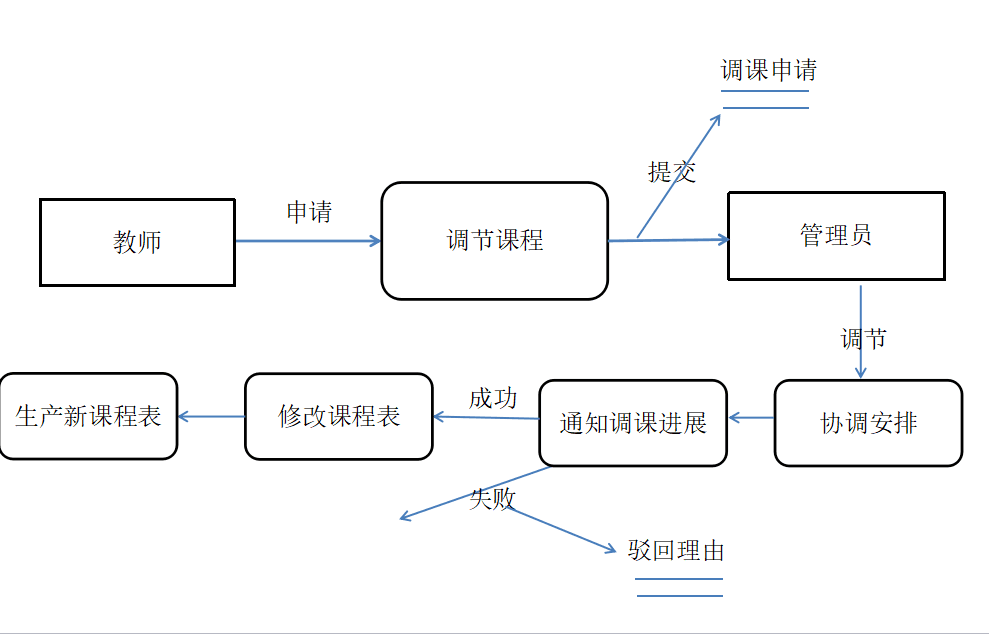
## 教师填写信息：



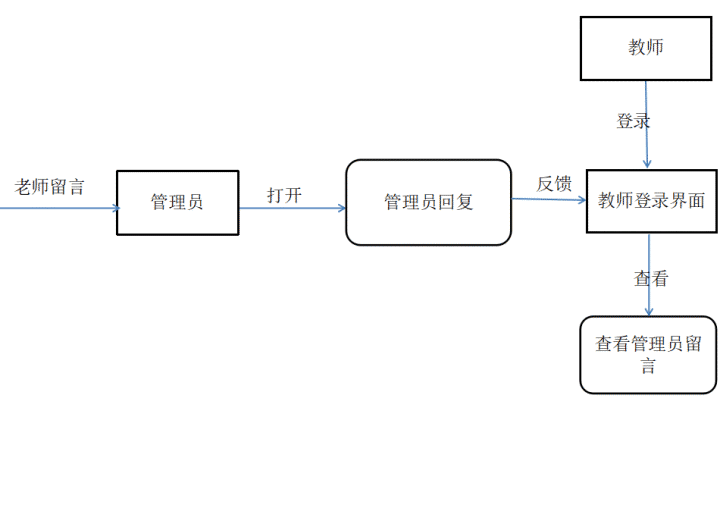
## 排课：



## 调课申请：



## 留言功能：



## 六、开发和测试生产环境说明

## 6.1 开发环境

开发语言选C#.NET,前端开发工具Microsoft Visual Studio2010，后台数据库采用SQL Server2008来实现数据的存取访问。

## 6.2 测试及产品环境

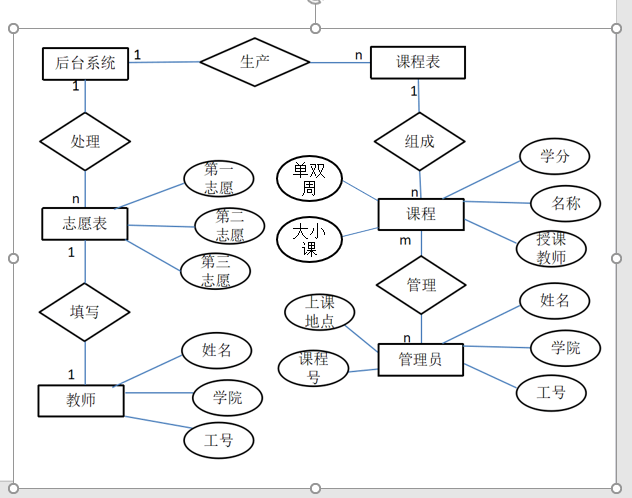
## 软件测试

测试计划——多台机器

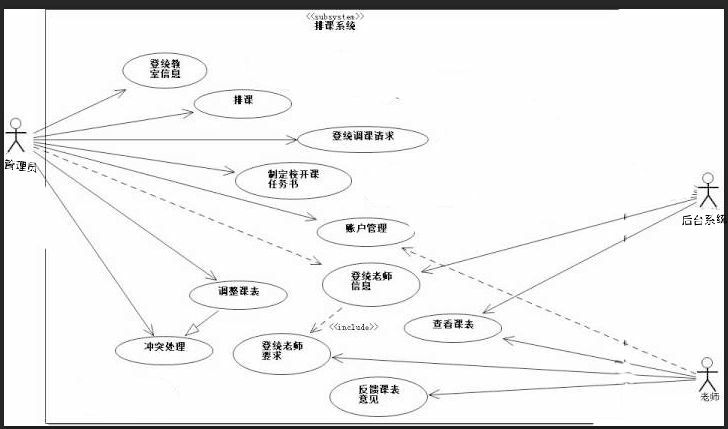
在真正的使用时，存在不同老师在多台电脑上使用该排课系统进行操作的情况，所以多台电脑更能真实反映系统是否支持多人同时操作，检测对多用户的实时交互的能力，也可以更好的反映出系统存在什么问题。而且，在现在计算机被普遍使用的情况下，找到多台电脑可以轻易实现，并不会消耗多余的成本。虽然演示时需要耗费一些人力，但这一缺点在总体看来可以忽略不计。

## 七、模块设计

## 7.1 E-R图

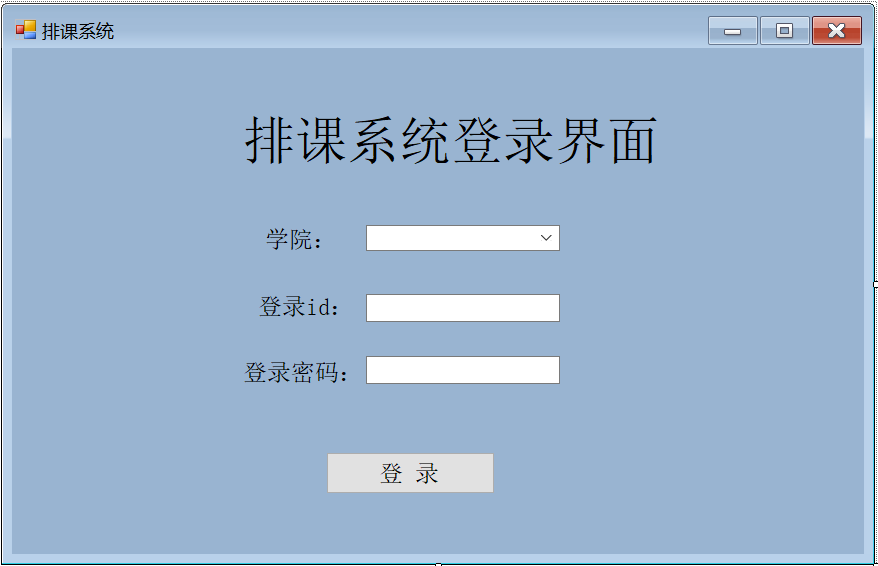


## 7.2 用例图



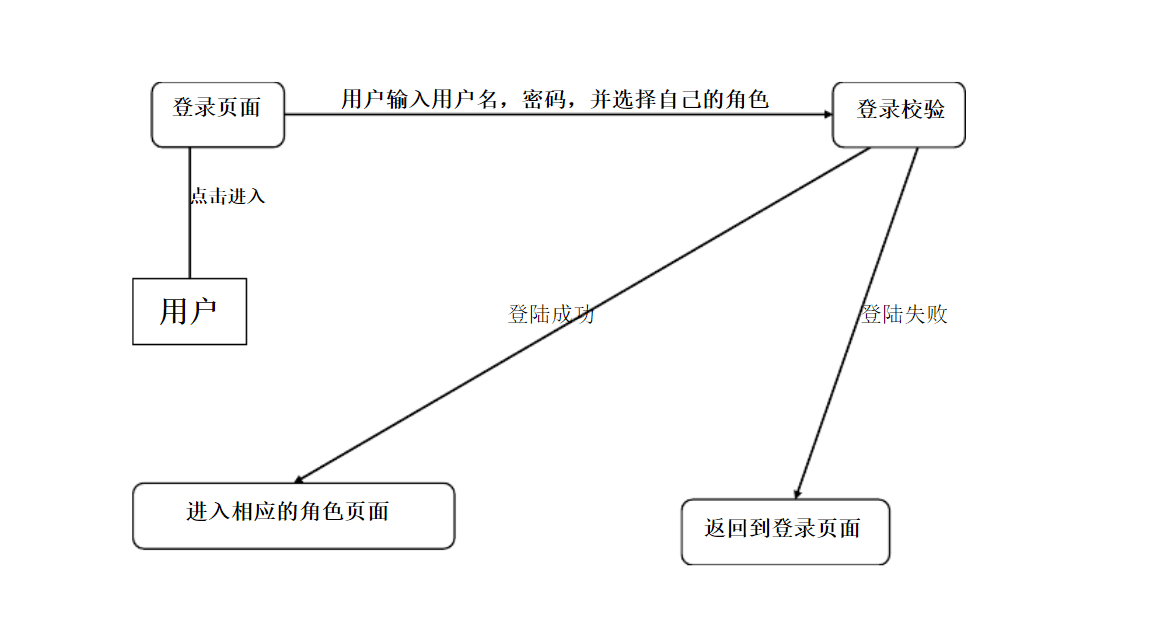
## 7.3功能设计说明

## 7.3.1 模块1：登录



## 7.3.1.1 子模块

## 7.3.1.1.1设计图



## 7.3.1.1.2 功能描述

登录分为管理员登录和教师登录，各自权限不同，老师进入系统查看所授课程（单双周、大小课）及提交志愿时间，管理员进行排课管理和回复申请。

## 7.3.1.1.3 输入数据

教师或者管理员的用户名和密码

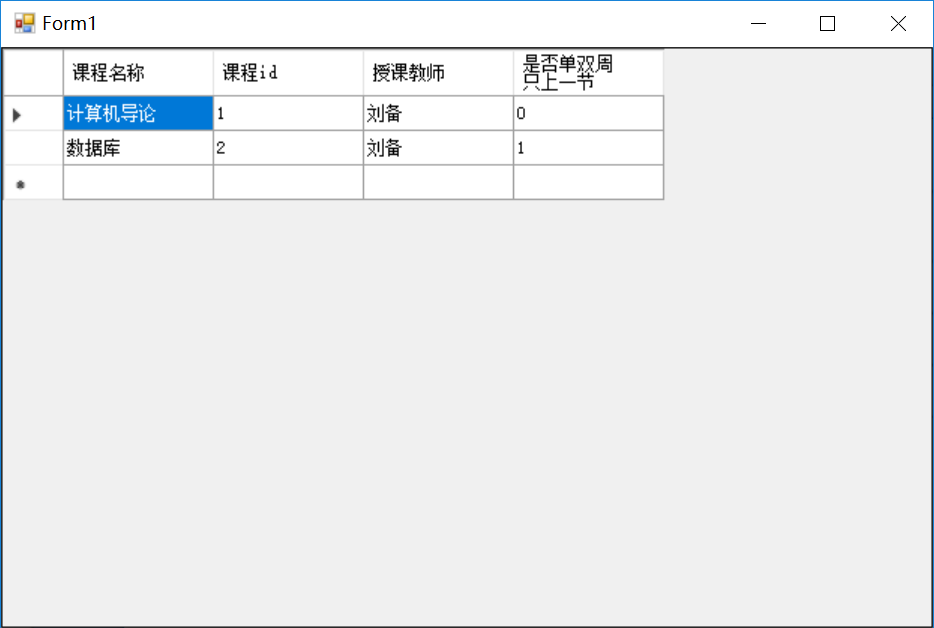
## 7.3.1.1.4 输出数据

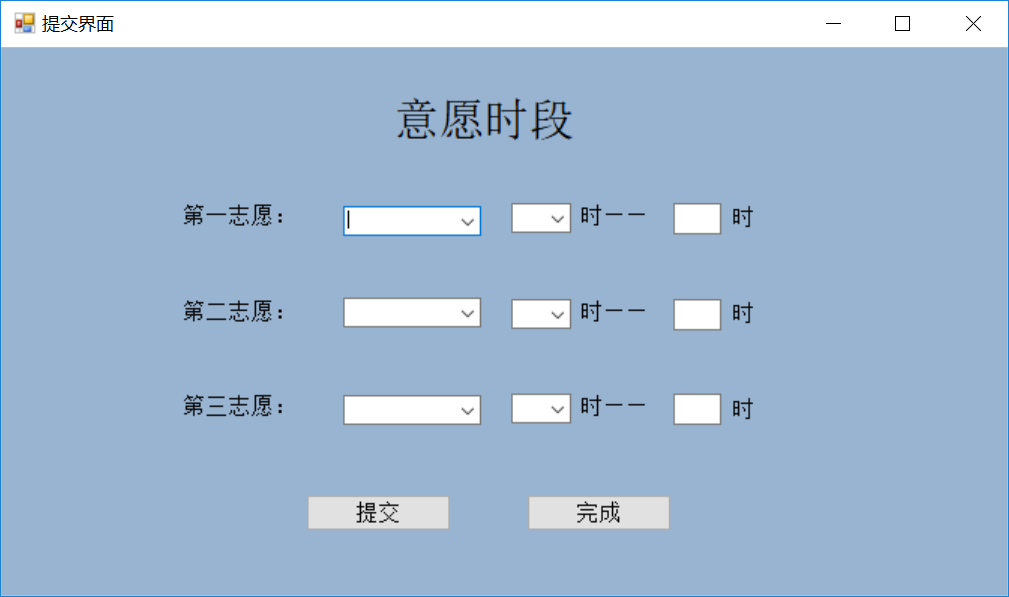
当出现非法操作时，提示出错。如“学院不能为空”“未查询到该账号”“账户密码不对”“请重新输入”

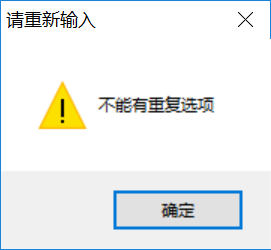
## 教师部分

## 7.3.2 模块2：教师提交部分

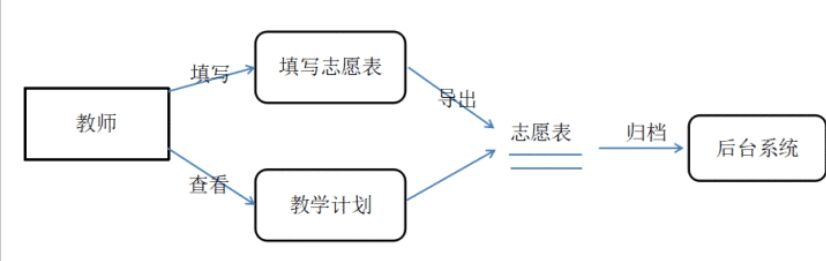
## 7.3.2.1 子模块







## 7.3.2.1.1设计图



## 7.3.2.1.2 功能描述

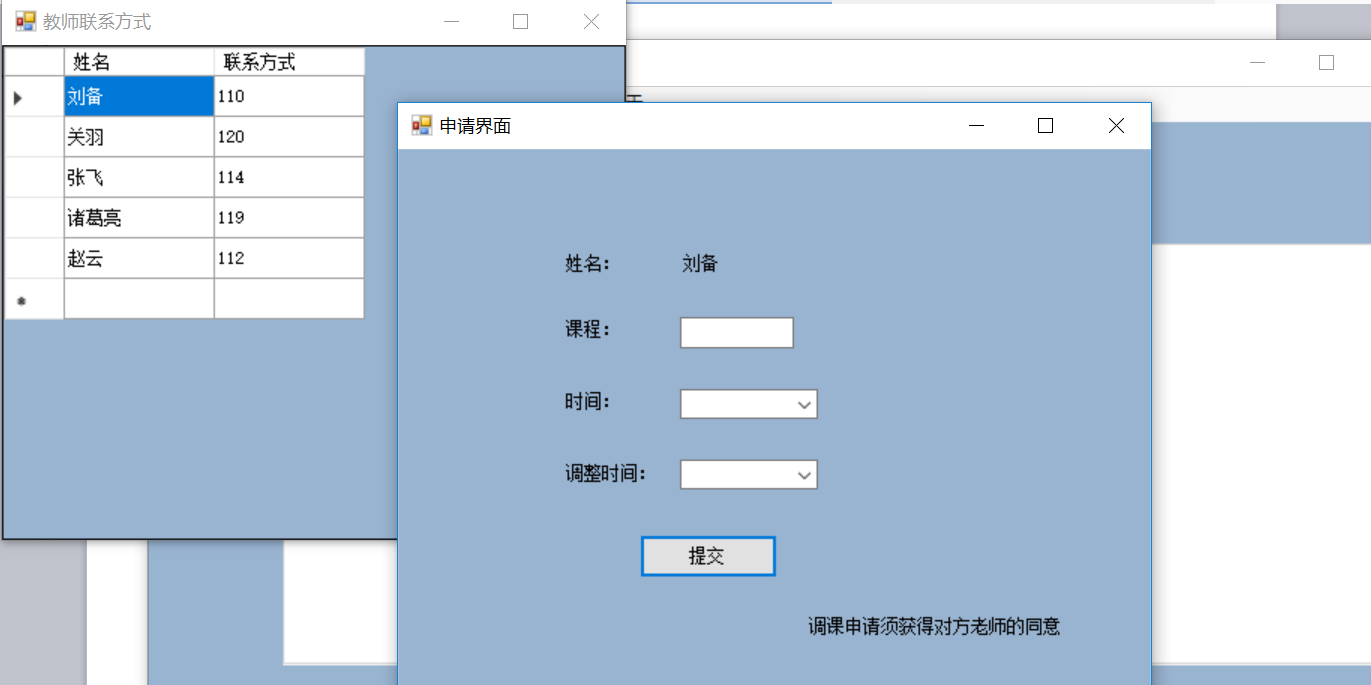
教师查看课程安排表（显示是否需要单双周上课及大小课），教师填写志愿时间（不可重复）生成志愿表至后台系统，由后台系统自动安排生成课程表。

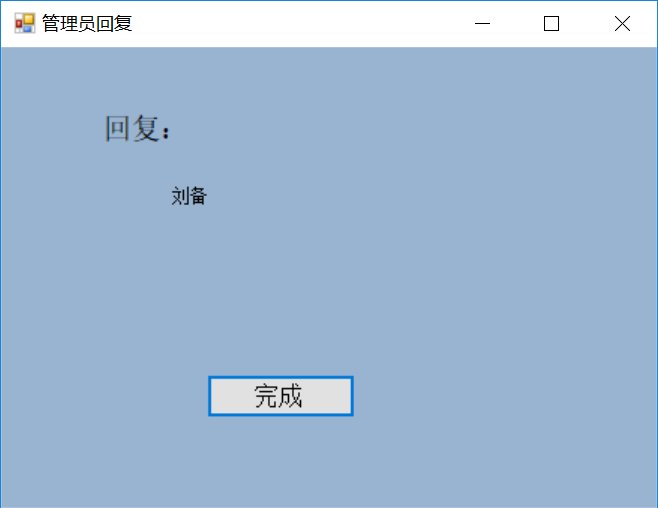
## 7.3.2.1.3 输入数据

志愿（包括第一志愿、第二志愿、第三志愿）时间

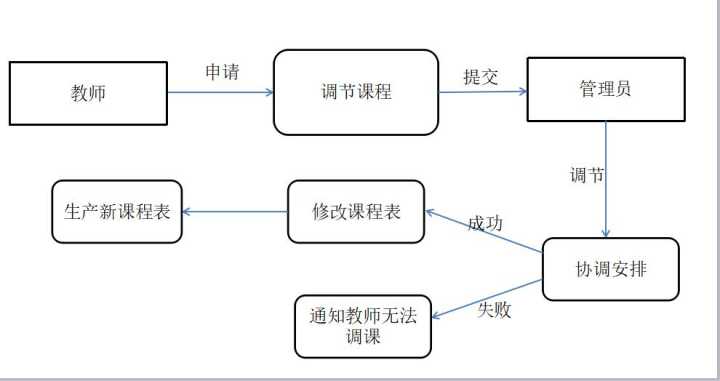
## 7.3.3 模块3：教师申请调课

## 7.3.3.1 子模块





## 7.3.3.1.1设计图



## 7.3.3.1.2 功能描述

教师提交要调课的课程及时间，管理员收到协调安排，并修改课程表，若无法调课则驳回。

## 7.3.3.1.3 输入数据

需要调课的课程及时间

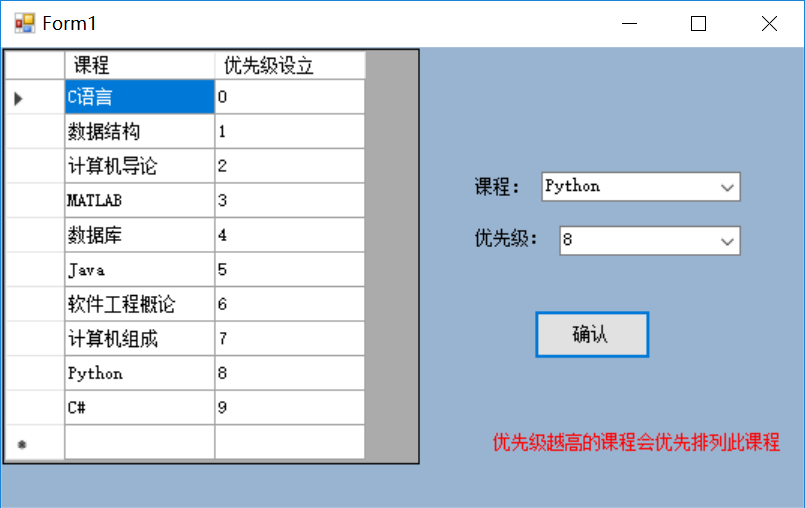
## 7.3.3.1.4 输出数据

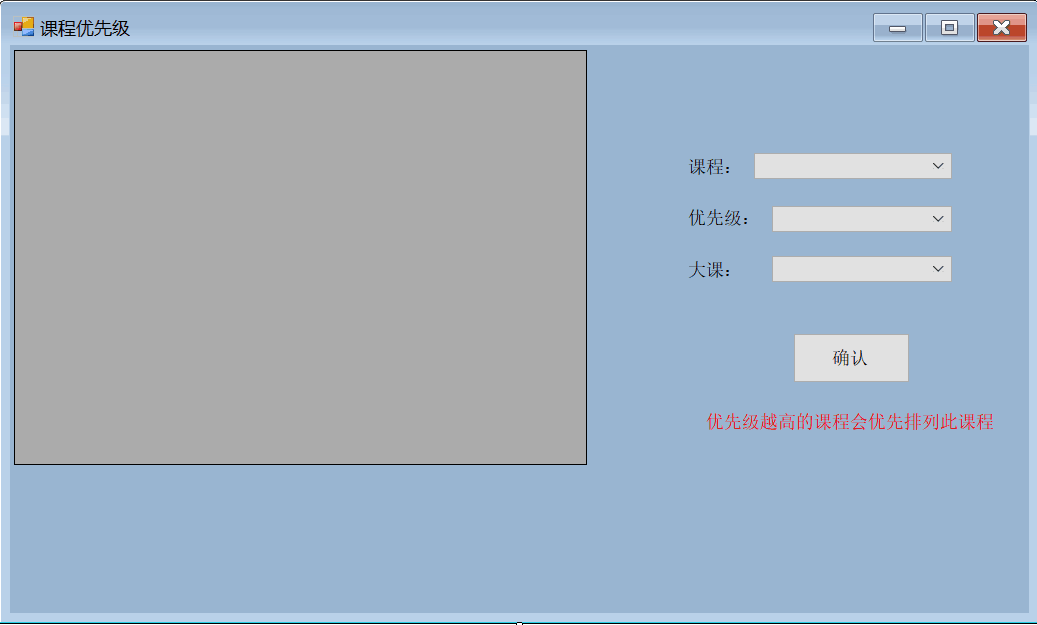
管理员回复

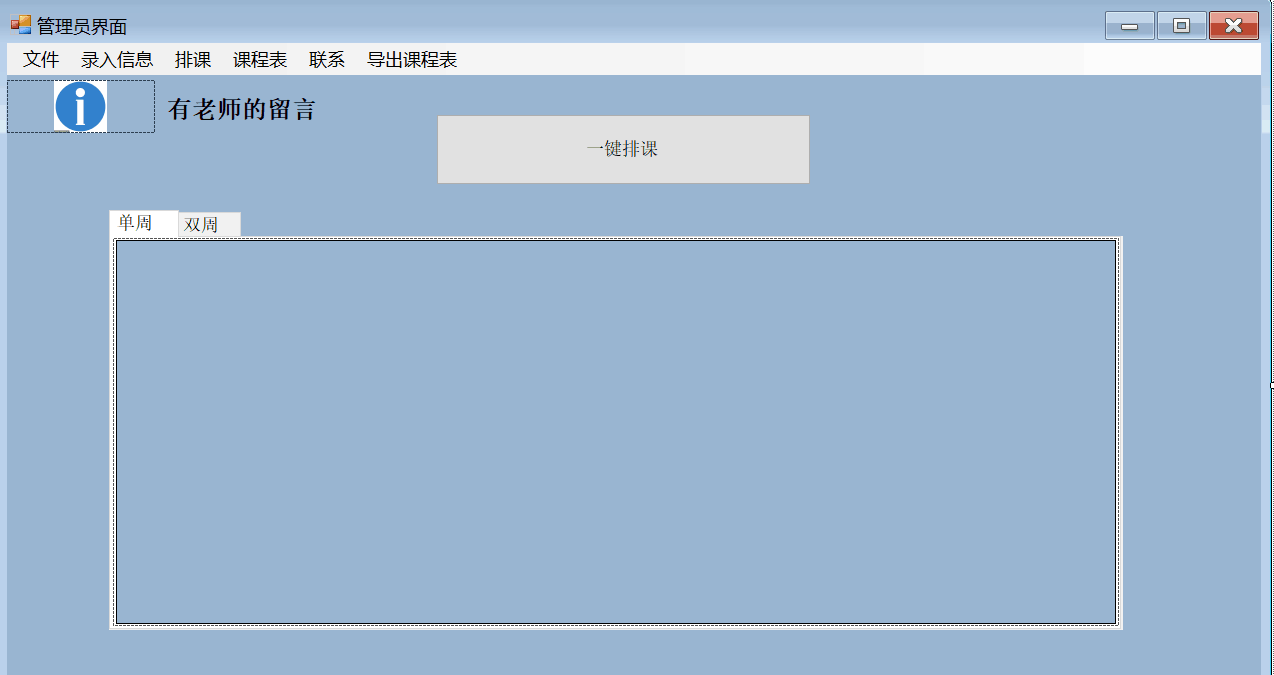
## 管理员部分

## 7.3.4 模块4：管理员排课部分

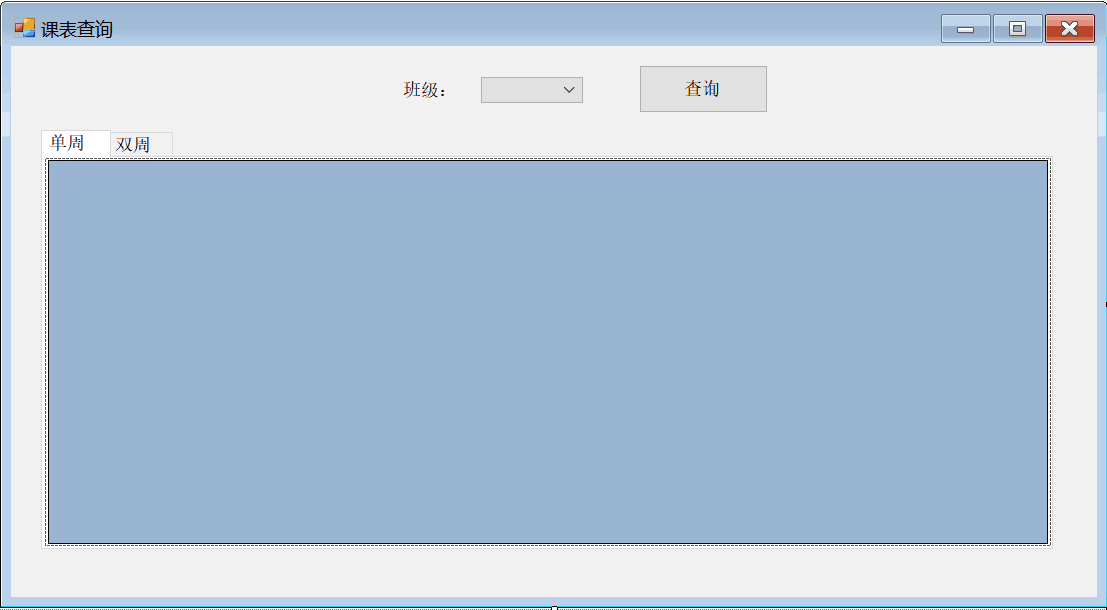
## 7.3.4.1 子模块











## 7.3.4.1.2 功能描述

管理员根据课程的重要程度进行课程优先级安排以及是否合课，点击排课按钮，系统根据提交时间和课程优先级进行排课，若存在教师没有提交志愿时间则将随机安排上课时间，最终生成单双周课表，也可以按照班级查询课表。

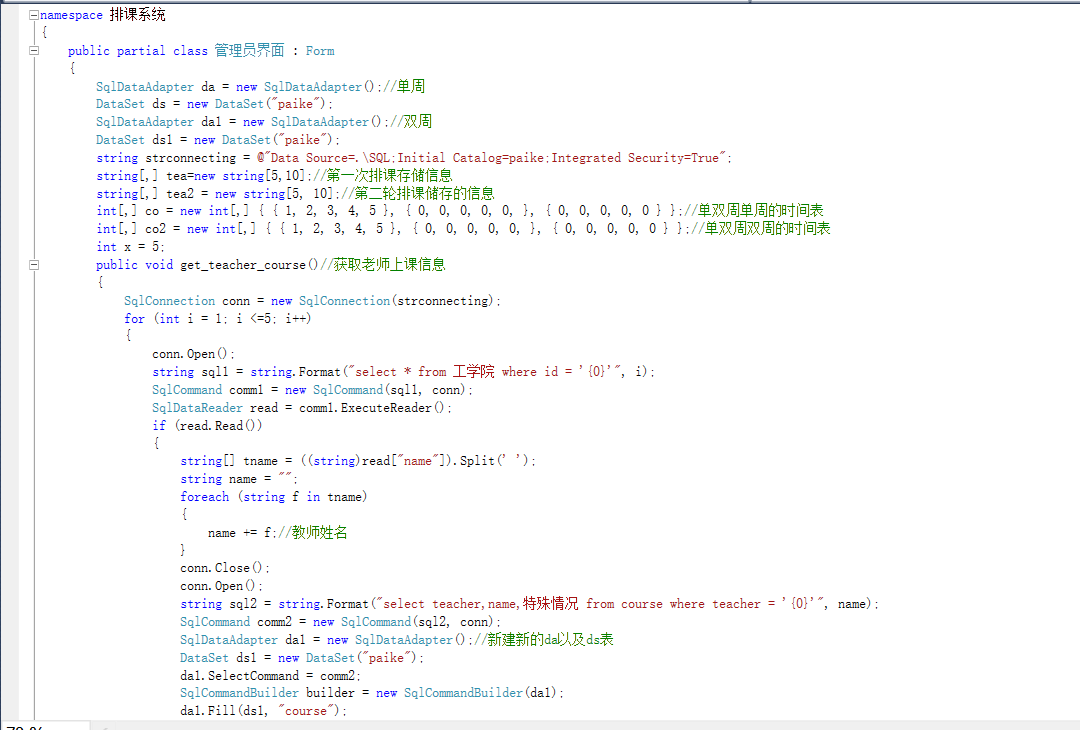
## 7.3.4.1.3 输入数据

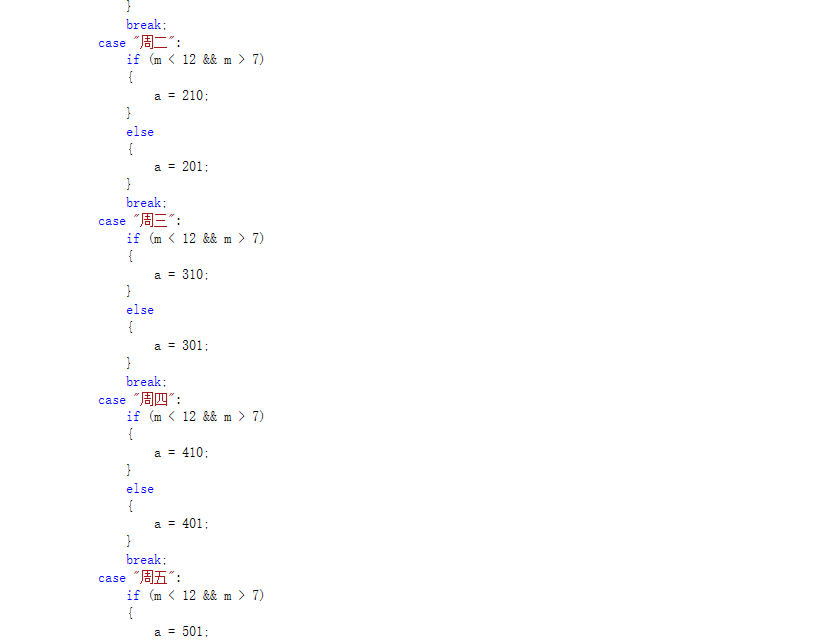
课程优先级

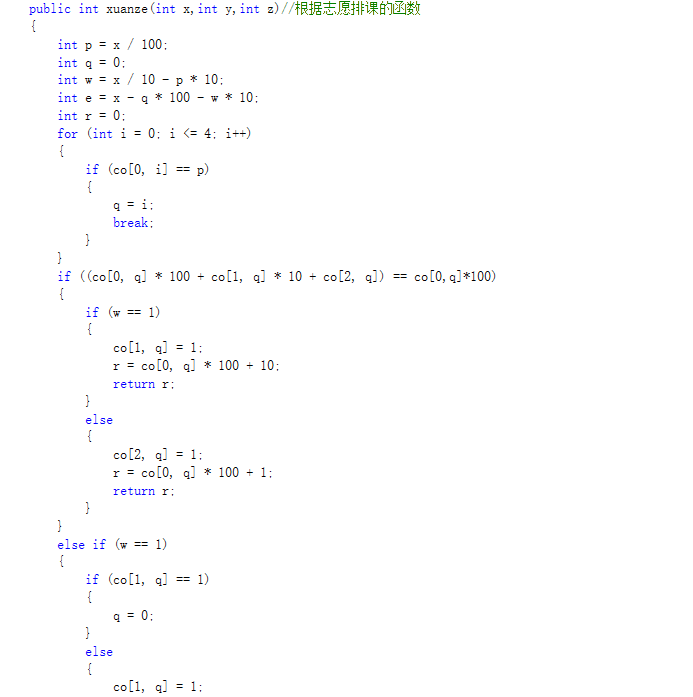
## 7.3.4.1.3输出数据

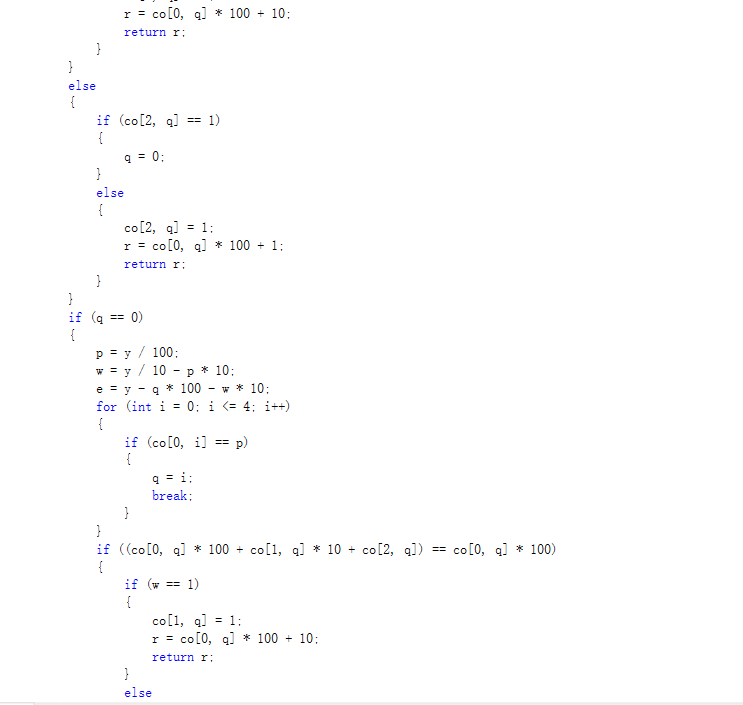
单双周课表，班级课表

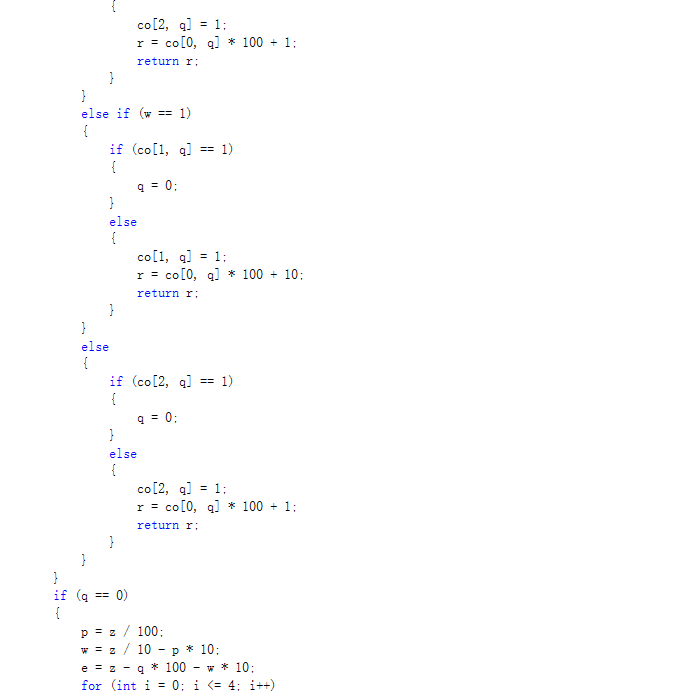
## 7.3.4.1.4核心代码





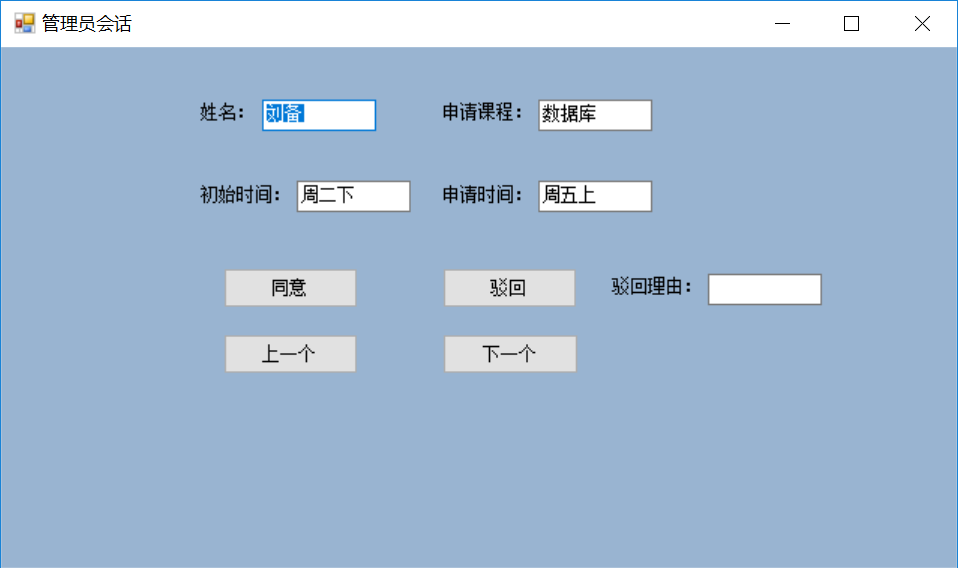






## 7.3.5模块5：管理员联系部分

## 7.3.5.1 子模块



## 7.3.5.1.2 功能描述

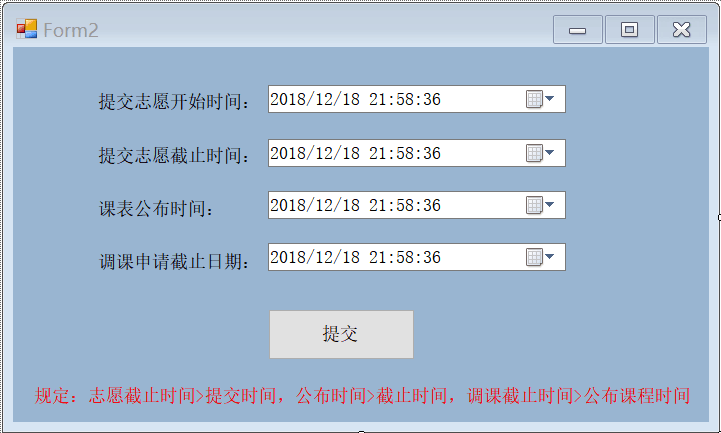
若有教师提交调课申请，管理员将根据课程实际情况，对教师的申请同意或者驳回。

## 7.3.5.1.3 输入数据

同意或驳回理由

## 7.3.6 模块6：时间设置

## 7.3.6.1 子模块



## 7.3.6.1.2 功能描述

管理员设定志愿开始、截止时间，课表公布时间，调课申请截止时间。提交志愿、调课需在规定时间内进行。若不在规定时间内，则无法操作。

## 7.3.6.1.3 输入数据

志愿开始、截止时间，课表公布时间，调课申请截止时间

## 八、接口设计

## 8.1内部接口（暂无）

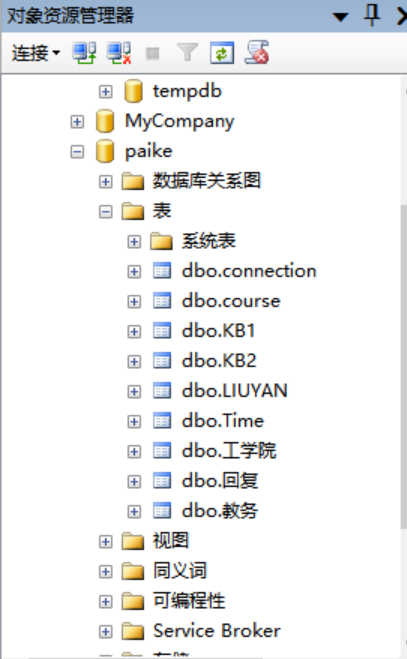
## 8.2 外部接口（暂无）

## 8.2.1 接口说明

## 8.2.2 调用方式

## 九、数据库设计

本系统采用SQL Server 2008作为前台数据库，数据库名称为paike，其中包含9张数据库表。



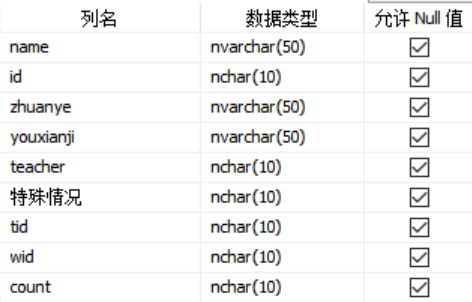
## 9.1 课程表（course）

字段名称解释：

name：姓名

id：用户名

zhuanye：专业



## 9.2 课程表（KB1）

字段名称解释：

JIE ：节数

MON: 星期一

TUE：星期二

WEN：星期三

THU：星期四

FRI:星期五



## 9.3 课程表（KB2）

字段名称解释：

JIE ：节数

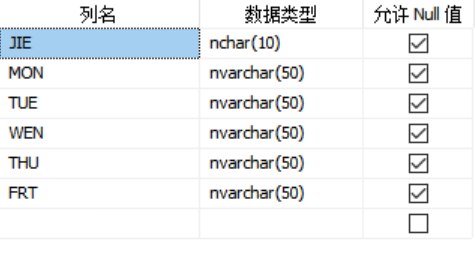
MON: 星期一

TUE：星期二

WEN：星期三

THU：星期四

FRI:星期五



## 9.4 工学院表

字段名称解释：

id：用户名（教师）

pass：密码

name：姓名

number：数量

number1:编号

telenum:电话



## 9.5 教务表

字段名称解释：

id：用户名（教务）

pass：密码



## 9.6 留言表（LIUYAN）

字段名称解释：

liuyan：留言

id：用户名

zhou：周

yishiduan：第一时段的开始时间

yishijian：第一时段的结束时间

ershiduan：第二时段的开始时间

ershijian：第二时段的结束时间

sanshiduan：第三时段的开始时间

sanshijian：第三时段的结束时间

erzhou：二周

sanzhou：三周

yizhou：一周

xueyuan：学院

name：姓名



## 9.7联系表（connection）

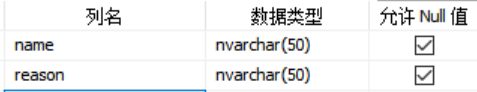


## 9.8 回复表

字段名称解释：

name：姓名

reason：回复结果



## 9.9 规定时间（Time）

time1：提交志愿开始时间

Time2：提交志愿截止时间

Time3：课表公布时间

Time4：调课申请截止时间



## 十、系统安全保密设计

## 10.1说明

为了保证系统整体安全而设计的安全措施。

## 10.2设计

## 10.2.1 身份验证部分

例如：对信任的用户,颁发身份验证码,通过该标识进行身份识别

## 十一、系统出错处理

若学院为空则提示“学院不能为空”，然后用户重新输入。

若账号不存在则提示“未查询到该账号”，然后用户重新输入。

若账号密码错误无法登陆则提示“账户密码不对”，然后用户重新输入。

## 十二、设计和开发规范

## 12.1 数据库设计规范

逐步消除数据依赖中不适合的部分，使模式中的各关系模式达到一定程度的“分离”，即“一事一地”的模块设计原则。

## 12.2 .NET编码规范

**函数命名**

函数命名规则：动词+名词（或东西），每个字母第一个字母必须大写。单词之间不加“\_”.

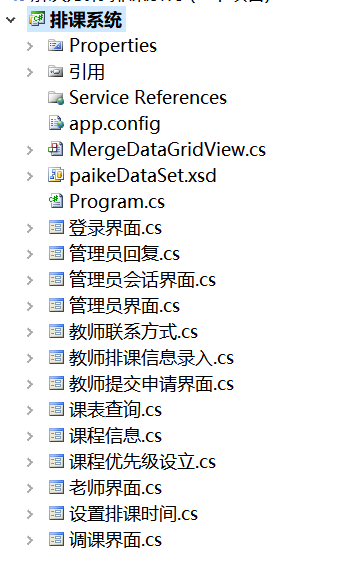
**表字段命名**

在命名表时，用单数形式表明名称。

在命名表的列时，不要重复表的名称。

不要在列的名称中包含数据类型。如果后来有必要更改数据类型，这将减少工作量。

## 12.3 代码目录结构



## 十三、 附录

**会议记录**  
褚峻仪  
1.先到先得：  
优点 可以根据时间优势选到想要的时间  
缺点 老师过于被动 用户体验差 选上心仪的时间概率低 对相对年老不精通电脑的老师不友好  
一键排课：实现不了  
提交需求： 老师提交第一第二志愿 更大可能地满足老师的需求 减少用户需要花费的时间  
缺点 会产生冲突的课程 需要调节  
2 网页  成本相对较少 容易打印 但需要每次登陆  
手机app 开发成本相对较高   
移植到不同平台比较麻烦 维护多个版本的成本比较高 优点：用户体验佳 界面更灵活   
3 一台计算机：不能真实的模拟具体排课过程   
多台计算机：能简便的反应真实排课情况  
陈雪宇  
1.先到先得  
优点：时间决定一切，教师通过自己的时间优势，选择自己想选择的课程。  
缺点：按照先后顺序选课，不能在网上并发操作，选课设备不理想或选课当天有其它事情的老师可能会选不到自己心仪的时间。选课间接的变相成了抢课，增加教师心理负担。  
2.提交需求  
优点：老师可以在规定时间内，提交自己的愿望，管理人员尽量满足其愿望，使得系统显得更加人性化。设立第一，第二，第三志愿，通过一次次的协调，极少数人的三个志愿都无法满足。符合少数服从多数的潮流，三个志愿都无法满足的教师，可以在下个学期的排课中获得优先满足其志愿的权利。受设备，或其他偶然发生事件干扰性较小。  
缺点：耗时长，效率低，大大增加了管理人员的工作量，操作不易实现。  
3.网页端  
优点：可以支持很多设备，在排课时可以通过电脑来进行，排课结束后，若想要看自己选择的结果，排课结果等，也可以随时用手机查看，很方便。排课结果容易打印。  
缺点：  
4.客户端  
优点：不提议  
缺点：需要在手机或者电脑上下载APP，每次换一个设备，都有在相应的设备上下载，浪费内存空间，不方便。  
5.实机演示  
优点：不受设备，适合提交需求的情况。  
缺点：不能真实反映排课操作，无法测试该系统是否支持多人同时操作。  
6.多台机器同时操作  
优点：能真实反映排课操作，可简单测试该系统是否支持多人同时操作。适合先到先得的情况。  
缺点：需要多台设备和多个人操作，开销大。操作人数有限。  
谢子澎  
我觉得还是设立优先级像填志愿一样，一来保证了用户的选择权以保证良好的用户体验，二来更符合大学教师的实际情况，因为可以选择自己确切的空闲时间来授课(其实时间做研究什么的)  
第二个问题就是虽然网页端挺不错，但是从实际情况来看，我们对于前端的知识还是有一定缺乏，不利于实现，所以我觉得包括其他组都会选择客户端形式，其实客户端响应的时间比网页端快一点  
第三个问题我觉得还是五人五台电脑同时操作比较好，因为这样更符合现实，才能真实反映系统对多用户的实时交互，毕竟如果只能一个用户仅能在一个时间对系统进行操作，影响工作效率之余还造成用户的不良体验。  
 张煜茜  
先到先得相对于志愿筛选相对公平，但是容易造成令用户体验不佳。这就使得教师对排课系统的满意度急剧下降。教师在涉及自身时间安排的排课问题上看上去自由，实则没有任何的自由度,容易造成用户的不满。  
志愿筛选的优点是将排课活动转变成教务人员和教师共同参与的活动,大家通过共享的排课应用平台通过沟通和协商来共同完成排课过程。这样既满足了教师的排课需求,又能够有效地降低教务人员的排课负担,真正体现出人性化的特征。缺点是增加了后台管理人员的负担，志愿相同数量过多时也容易造成问题。  
一键排课优点是实现计算机的自动化处理,解决课表编排时教师、学生、教室、时间等相关因素的冲突问题。缺点是:教师、学生与教室、时间这些物理性因素同等对待,忽视了人的属性,使得现有排课系统缺乏人性化特征,教师和学生接受排课结果但并不满意。而且登录人数过多时容易造成网络阻塞，无法登陆的情况。  
基于平台的话优点是可以简化用户的操作难易程度，缺点是开发成本高，后期维护开销大，不同版本需要不同的安装需求。  
一台机器成本小，操作简便，但是不容易模拟出真实情况，多台机器同时可以较为接近真实情况，而且可以获取多人的使用体验感受但是人力资源耗费较大。  
羊亦可  
1.问题一我的最初是想支持一键排课的，因为一键排课完全可以以教务的初始要求来安排，并且我的那个界面的表格是可以设立优先级，不过可能很难实现，先到先得在算法上可能很容易实现，但是，感觉可能最后会形成类似于抢课的状态，就好比两个老师同时提交了他们的排课申请信息，算法会优先根据哪个老师来排课，而且从现实中考虑应该由教务来安排课程时间，我觉得老师应该不能安排排课时间，不仅有冲突，而且之后如果两个老师同时看中了一个时间段，一个老师已经选了，那么在私下交流调换，修改起来也很麻烦，老师输入课程信息，一键排课，同时优先级的设立，通过教务后台管理员来调配，这种应该是最合理的，老师尽量不要拥有排课的权力，主体用户应该是教务。  
2.问题二基于第一个问题，老师输入课程信息，教务的管理员来处理排课信息，我觉得客户端比较好实现，因为主客户端在教务管理员那里  
3.问题三支持五人同时吧，这样子老师有自己单独的客户端，提交信息到局域网数据库，然后由管理员的排课端统一排课，之后将课表发给老师，如果有老师觉得不妥，从根本上来讲，这是工作，我觉得的工作的优先级应该是最高的，虽然有点不太合乎情理，我觉得最后尽量修改吧  
翟伟  
1.先到先得吧 设立优先级大家都选最高级不是跟没选一个样吗 一件排课的话 一点选择都没有 先到先得只要你来得早 来的早的人还是能满足需求的  
2.网页 成本低 开发一个客户端还要下载成本还高  
3.既然先到先得 那肯定要多人测试了 至少5台计算机一起吧  
范新宇  
1.先到先得要好一点，因为排课系统想要让老师自己做选择，但是老师的选择有可能冲突，就要用先到先得的办法来解决冲突，缺点就是老师可能选不到自己想要的课。  
2.我觉得网页好一点，毕竟登录网站和下载一个客户端比还是要简单一点，有些老师可能不喜欢下载客户端，占内存还有点麻烦，网页的浏览起来比较方便。  
3.我觉得多台机器同时比较好，因为老师选课的时候就应该是不同老师在多台电脑上进行操作，这样演示的话比较符合实际，可以更好的反映出系统存在什么问题。  
  
第二次需求已完成（单双周）  
第三次需求部分未实现，已实现按班级查询课表，未实现合班上课。原因代码可拓展性较差，若想实现该功能，需耗费大量精力，经过反复考虑，最后决定放弃实现合班上课功能。