

本科毕业论文（设计）

概要设计说明书

|  |  |
| --- | --- |
| **学 生 姓 名** | **龙雅容** |
| **学号** | **20151002004** |
| **专业** | **软件工程** |
| **年级班级** | **2015级2班** |
| **指导教师** | **朱毅（教授）** |
| **所在学院** | **软件工程学院** |
| **提交日期** | **2019年04月19日** |

2019 年 4 月

成都信息工程大学 软件工程学院

目录

[1 引言 - 1 -](#_Toc2639)

[1.1 编写目的 - 1 -](#_Toc1612)

[1.2 背景 - 1 -](#_Toc27798)

[1.3 术语。 - 1 -](#_Toc21676)

[1.4 参考资料 - 1 -](#_Toc3709)

[2 总体设计 - 1 -](#_Toc511)

[2.1 系统体系结构 - 1 -](#_Toc7949)

[2.2 系统功能结构 - 2 -](#_Toc4565)

[2.3 运行环境 - 3 -](#_Toc2451)

[2.3.1 硬件环境 - 3 -](#_Toc26691)

[2.3.2 软件环境 - 3 -](#_Toc6123)

[2.4 系统的关键技术 - 3 -](#_Toc3455)

[3 功能模块设计说明 - 3 -](#_Toc10250)

[3.1 功能模块列表 - 3 -](#_Toc3011)

[3.2 初级预算 - 4 -](#_Toc5885)

[3.2.1 模块编号和功能描述 - 4 -](#_Toc8402)

[3.2.2 操作者 - 5 -](#_Toc21685)

[3.2.3 与本模块相关的码表和表 - 5 -](#_Toc2323)

[3.2.4 界面设计与说明 - 5 -](#_Toc28503)

[3.2.5 输入信息 - 5 -](#_Toc6179)

[3.2.6 输出信息 - 6 -](#_Toc6634)

[3.2.7 算法 - 8 -](#_Toc13124)

[3.2.8 处理流程 - 8 -](#_Toc4504)

[3.2.9 类设计 - 9 -](#_Toc9250)

[3.3 最终预算 - 11 -](#_Toc7650)

[3.3.1 模块编号和功能描述 - 11 -](#_Toc11958)

[3.3.2 操作者 - 11 -](#_Toc2927)

[3.3.3 与本模块相关的码表和表 - 11 -](#_Toc27477)

[3.3.4 界面设计与说明 - 12 -](#_Toc15970)

[3.3.5 输入信息 - 12 -](#_Toc11822)

[3.3.6 输出信息 - 12 -](#_Toc8286)

[3.3.7 算法 - 14 -](#_Toc20421)

[3.3.8 处理流程 - 15 -](#_Toc3995)

[3.3.9 类设计 - 15 -](#_Toc1576)

[3.4 实际账单 - 17 -](#_Toc24510)

[3.4.1 模块编号和功能描述 - 17 -](#_Toc15607)

[3.4.2 操作者 - 17 -](#_Toc16976)

[3.4.3 与本模块相关的码表和表 - 17 -](#_Toc2671)

[3.4.4 界面设计与说明 - 18 -](#_Toc23556)

[3.4.5 输入信息 - 18 -](#_Toc4637)

[3.4.6 输出信息 - 19 -](#_Toc13622)

[3.4.7 算法 - 20 -](#_Toc27114)

[3.4.8 处理流程 - 21 -](#_Toc26434)

[3.4.9 类设计 - 21 -](#_Toc31886)

[3.5 用户管理 - 23 -](#_Toc13728)

[3.5.1 模块编号和功能描述 - 23 -](#_Toc31929)

[3.5.2 操作者 - 23 -](#_Toc30961)

[3.5.3 与本模块相关的码表和表 - 23 -](#_Toc22132)

[3.5.4 界面设计与说明 - 24 -](#_Toc13147)

[3.5.5 输入信息 - 24 -](#_Toc14465)

[3.5.6 输出信息 - 24 -](#_Toc8429)

[3.5.7 算法 - 24 -](#_Toc24004)

[3.5.8 处理流程 - 25 -](#_Toc27607)

[3.5.9 类设计 - 25 -](#_Toc16131)

[3.6 其它信息管理 - 26 -](#_Toc11201)

[3.6.1 模块编号和功能描述 - 26 -](#_Toc22388)

[3.6.2 操作者 - 27 -](#_Toc20616)

[3.6.3 与本模块相关的码表和表 - 27 -](#_Toc15092)

[3.6.4 界面设计与说明 - 27 -](#_Toc23589)

[3.6.5 输入信息 - 27 -](#_Toc29416)

[3.6.6 输出信息 - 28 -](#_Toc11404)

[3.6.7 算法 - 28 -](#_Toc32520)

[3.6.8 处理流程 - 29 -](#_Toc24005)

[3.6.9 类设计 - 29 -](#_Toc429)

[4 内部接口设计 - 30 -](#_Toc19854)

[4.1 接口1 - 31 -](#_Toc6993)

[4.2 接口2 - 33 -](#_Toc21828)

[4.3 接口3 - 34 -](#_Toc22377)

[4.4 接口4 - 35 -](#_Toc13290)

[4.5 接口5 - 36 -](#_Toc23751)

[4.6 接口5 - 37 -](#_Toc12673)

# 引言

## 编写目的

本文档是用于基于网络爬虫的ACM预算系统开发的概要设计，主要包含了功能模块和系统框架的设定。期望的读者为后续开发人员和测试及维护人员。

## 背景

ACM团队的账目繁琐，如涉及到住宿、交通、报名等费用，并且在实际发费中，总会遇到意外。因此在预算是需要考虑全面，让预算后的金额弹度够大。这导致预算相当麻烦。不仅需要上网查询相关费用，而且还要进行费用的选择，耗时耗力。同时ACM团队日益壮大，人数的增加，更加加重了预算和账目整理的难度。

为了让ACM团队投入更多的精力在日常训练和比赛中，而开发ACM预算系统。因此ACM预算系统针对的用户为ACM团队，让ACM团队轻松预算以及账目清晰明了

## 术语。

ACM：国际大学生程序设计大赛

ACM团队：参加ACM的所有成员，包括教练和参赛队员

初级预算：从目的地、时间等相关信息，获得合适的车次，住宿等，自动形成预算表

最终预算：在初始预算的基础上，确定相关信息，重新生成预算表。

## 参考资料

本系统的《需求规格说明书》

# 总体设计

## 系统体系结构

|  |
| --- |
|  |
| 图2‑1 系统体系结构图 |

## 系统功能结构

|  |
| --- |
| 系统功能图 |

图2‑2 系统功能结构图

## 运行环境

本系统是基于python中Django架构的ACM预算系统，通过浏览器显示前端并需要联网。

### 硬件环境

1. 描述本软件运行对服务器、客户端的硬件要求：

服务器：Windows10,64位操作系统，普通处理器2.40GHZ，RAM4GB；

客户端：Windows10,64位操作系统，普通处理器2.40GHZ，RAM4GB；

1. 描述本软件运行所使用的外围设备，主要包括：

输入/输出设备：普通键盘/屏幕

### 软件环境

1. 操作系统；Windows10
2. 数据库系统；mysql
3. 开发平台及工具；pycharm
4. 通信协议；WebSocket protocol、TCP/IP

## 系统的关键技术

Python爬虫，Django框架，CSS样式与jquery

# 功能模块设计说明

## 功能模块列表

表3‑1 功能模块列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块**  **编号** | **模块**  **名称** | **对应需求**  **功能编号** | **所对应**  **需求功能** | **实现**  **优先级** |
| YS01 | 初始预算管理 | YS\_cjys01 | 初始预算创建 | 高 |
| YS01 | 初始预算管理 | YS\_cjys02 | 初始预算修改 | 高 |
| YS01 | 初始预算管理 | YS\_cjys03 | 初始预算查询 | 高 |
| YS01 | 初始预算管理 | YS\_cjys04 | 初始预算删除 | 高 |
| YS02 | 最终预算管理 | YS\_zzys01 | 最终预算创建 | 高 |
| YS02 | 最终预算管理 | YS\_zzys02 | 最终预算修改 | 高 |
| YS02 | 最终预算管理 | YS\_zzys03 | 最终预算查询 | 高 |
| YS02 | 最终预算管理 | YS\_zzys04 | 最终预算删除 | 高 |
| YS03 | 实际账单管理 | YS\_zm01 | 实际账单创建 | 高 |
| YS03 | 实际账单管理 | YS\_zm02 | 实际账单修改 | 高 |
| YS03 | 实际账单管理 | YS\_zm03 | 实际账单查询 | 高 |
| YS03 | 实际账单管理 | YS\_zm04 | 实际账单删除 | 高 |
| YS03 | 实际账单管理 | YS\_zm05 | 实际账单总结 | 高 |
| YS04 | 用户信息管理 | YS\_adm01 | 用户注册 | 高 |
| YS04 | 用户信息管理 | YS\_adm02 | 用户登录 | 高 |
| YS04 | 用户信息管理 | YS\_adm03 | 用户注销 | 高 |
| YS04 | 用户信息管理 | YS\_adm04 | 用户信息查询 | 高 |
| YS04 | 用户信息管理 | YS\_adm05 | 用户信息修改 | 高 |
| YS05 | 其它信息管理 | YS\_qt01 | 车次信息管理 | 高 |
| YS05 | 其它信息管理 | YS\_qt02 | 酒店信息管理 | 高 |
| YS05 | 其它信息管理 | YS\_qt01 | 航班信息管理 | 高 |

## 初级预算

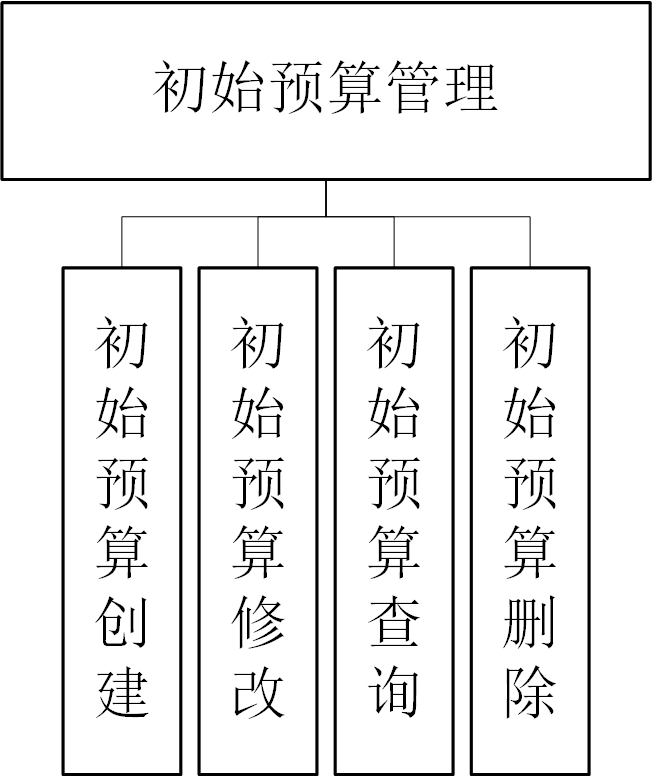


图3-2 初始预算模块机构图

### 模块编号和功能描述

YS\_cjys01：用户对初始预算进行创建

YS\_cjys02：用户对初始预算进行修改

YS\_cjys03：用户对初始预算进行查询

YS\_cjys04：用户对初始预算进行删除

### 操作者

教练和总教练

### 与本模块相关的码表和表

表3‑3 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| STra | 出发车次信息表 | √ |  | 提供数据 |
| ETra | 回程车次信息表 | √ |  | 提供数据 |
| SAir | 出发航班信息表 | √ |  | 提供数据 |
| EAir | 回程航班信息表 | √ |  | 提供数据 |
| CName | 竞赛账单信息表 | √ |  | 建立竞赛账单中的初始预算 |
| Acc | 酒店信息表 | √ |  | 提供数据 |
| Admi | 用户信息表 | √ |  | 建立者 |
| Eleb | 初级预算信息表 | √ |  | 数据计算与显示 |

### 界面设计与说明

网页端，水平表单，颜色对比明显

### 输入信息

比赛类型：

选择型。

地点：

字符串型。

最大值为15。

应具体到市。

格式为：xxx市。

出发/到达时间：

日期型。

应具体到年、月、日、时、分。

格式为：YYYY-MM-DD hh:mm

老师数量：

整型。

数字大于零。

队伍数量：

整型。

数字大于零。

女性数量：

整型。

数字大于等于零。

### 输出信息

比赛名称：

字符串型，最大值30，包含地点。

老师数量：

整型，不得小于零。

学生数量：

整型，不得小于3

学生数量 = 队伍数量 \* 3

队伍数量：

整型，不得小于零。

女性数量：

整型

初始出发/回程时间

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm

学生/老师外出天数

整型，大于零

学生/老师外出天数=学生/老师回程到达时间 - 学生/老师出发时间 + 1

学生/老师停留天数

整型，大于零

学生/老师停留天数=学生/老师回程出发时间 - 学生/老师到达时间 + 1

队伍/老师出发/回程车次：

字符串型，最大值15。

如果老师的交通工具为飞机，就不用输出老师出发/回程车次

队伍/老师出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

队伍/老师/来/回票价：

浮点型

队伍/老师/来/回交通费

浮点型

队伍/老师/来/回交通费=交通票价 \* 学生/老师数量

报名费

浮点型

补贴费

浮点型

酒店名称

字符串，最大值15

老师/学生房间数量

整型

房间数量需要根据老师到达和回程时间及女性数量进行判断

老师/学生酒店价格

浮点型

老师/学生酒店总费用

浮点型

老师/学生酒店总费用需根据老师/学生酒店数量及停留时间灵活判断

出发/回程航班：

字符串型，最大值15。

出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

老师的交通工具是火车，就不需要输出该数据

老师/队伍总预算

浮点型

老师/队伍总预算=交通费用+酒店总费用

初始总预算：

浮点型，队伍总预算+老师总预算。

### 算法

车次出发时间-车次回程到达时间=天数

几种可能的比较：老师坐飞机或坐地铁的不同时间及费用比较。

几种最值的比较，得出最合适的方案

### 处理流程

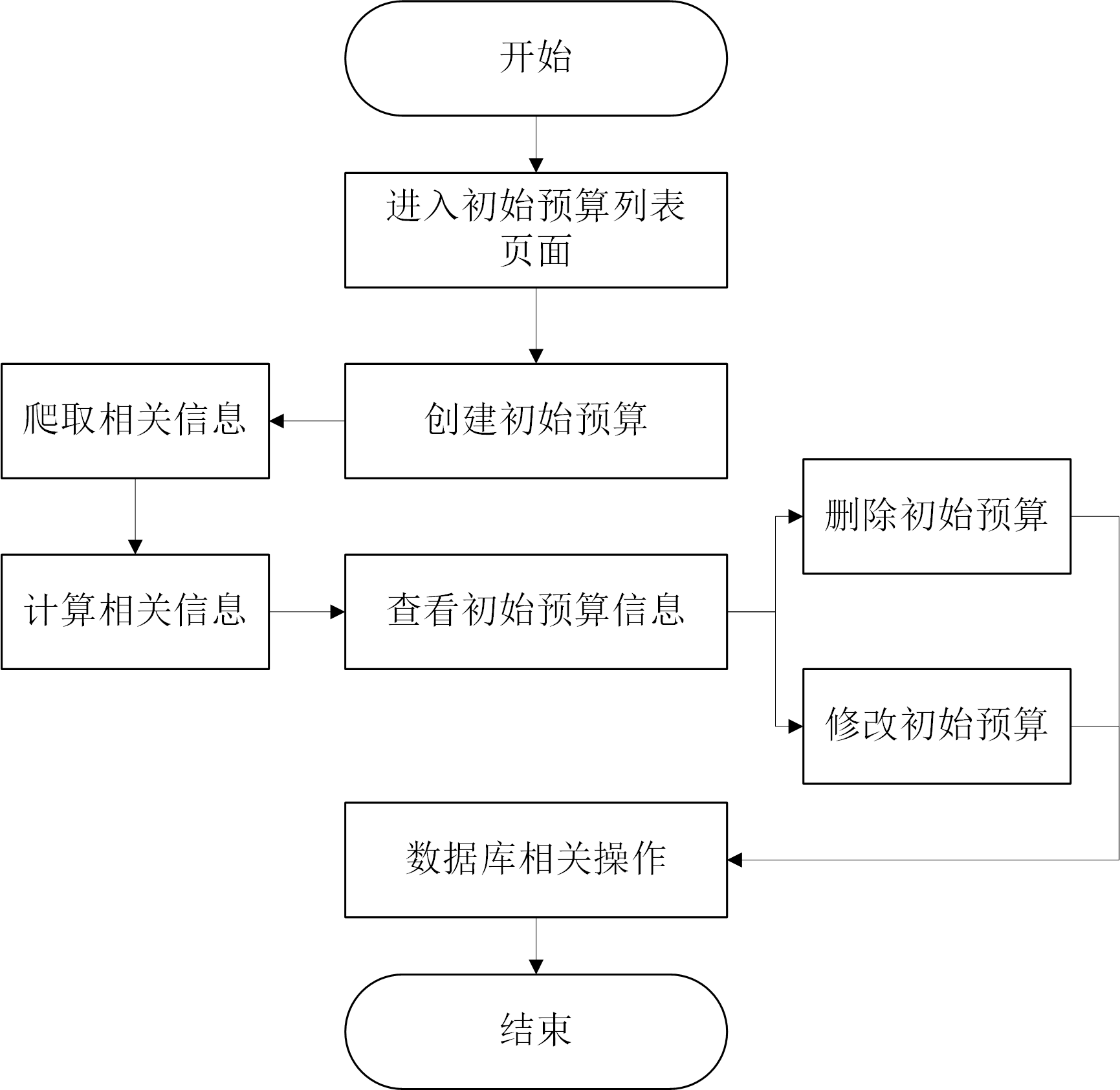


图3-4 初始预算管理流程图

### 类设计

#### 类图

|  |
| --- |
| **初始预算类图** |

图3‑5 初始预算类图

#### 类说明

1. STra类说明：
   1. 功能：出发车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
2. ETra类说明：
   1. 功能：回程车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
3. SAir类说明：
   1. 功能：出发航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
4. EAir类说明：
   1. 功能：回程航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
5. Acc类说明：
   1. 功能：酒店信息的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
6. Admi类说明：
   1. 功能：用户信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
7. Eleb类说明：
   1. 功能：初始预算信息的创建，删除，修改，查询多条或得到一条，其中STra、ETra、SAir、EAir、Acc、Admi是组成它的一部分
   2. 主要方法：create(), delect(), filter(), get()，update
8. Spierd类说明：
   1. 功能：实现Spierdx接口中的creTra()、creAir()、creAcc()方法，即通过网络爬虫获取车次、航班、酒店信息，并依赖于STra、ETra、SAir、EAir、ACC类进行存储
   2. 主要方法：creTra()、creAir()、creAcc()
9. Eleblist类说明：
   1. 功能：实现Elebx接口中的cre()、showele()、sele2()、ajax\_up\_ele()、ajax\_de\_ele()方法，即实现初始预算的创建，初始预算具体信息浏览、初始预算查询、初始预算信息修改、初始预算删除，并依赖于Eleb类进行存储

主要方法：cre()、showele()、sele2()、ajax\_up\_ele()、ajax\_de\_ele()

## 最终预算

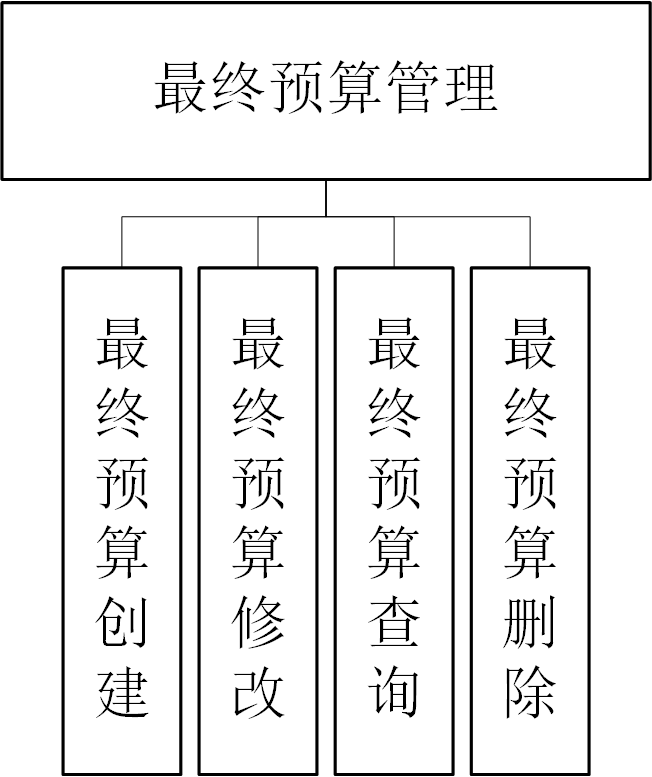


图3-6 最终预算模块机构图

### 模块编号和功能描述

YS\_zzys01：用户对最终预算进行创建

YS\_zzys02：用户对最终预算进行修改

YS\_zzys03：用户对最终预算进行查询

YS\_zzys04：用户对最终预算进行删除

### 操作者

教练，总教练

### 与本模块相关的码表和表

表3‑7 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| STra | 出发车次信息表 | √ |  | 提供数据 |
| ETra | 回程车次信息表 | √ |  | 提供数据 |
| SAir | 出发航班信息表 | √ |  | 提供数据 |
| EAir | 回程航班信息表 | √ |  | 提供数据 |
| CName | 竞赛账单信息表 | √ |  | 建立竞赛账单中的最终预算 |
| Acc | 酒店信息表 | √ |  | 提供数据 |
| Admi | 用户信息表 | √ |  | 建立者 |
| Eleb | 初级预算信息表 | √ |  | 提供数据 |
| Uleb | 最终预算表 | √ |  | 提供并显示最终预算信息 |

### 界面设计与说明

网页端，水平表单，颜色对比明显

### 输入信息

初始预算

选择，字符串

老师数量

选择，整型，大于零

队伍数量

选择，整型，大于零

女性数量

选择，整型

出发/回程车次

选择，字符串型

车次或航班

选择，字符串型

酒店

选择，字符串型

报名费

浮点型

补贴费

浮点型

### 输出信息

比赛名称：

字符串型，最大值30，包含地点。

老师数量：

整型，不得小于零。

学生数量：

整型，不得小于3

学生数量 = 队伍数量 \* 3

队伍数量：

整型，不得小于零。

女性数量：

整型

初始出发/回程时间

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm

学生/老师外出天数

整型，大于零

学生/老师外出天数=学生/老师回程到达时间 - 学生/老师出发时间 + 1

学生/老师停留天数

整型，大于零

学生/老师停留天数=学生/老师回程出发时间 - 学生/老师到达时间 + 1

队伍/老师出发/回程车次：

字符串型，最大值15。

如果老师的交通工具为飞机，就不用输出老师出发/回程车次

队伍/老师出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

队伍/老师/来/回票价：

浮点型

队伍/老师/来/回交通费

浮点型

队伍/老师/来/回交通费=交通票价 \* 学生/老师数量

报名费

浮点型

补贴费

浮点型

酒店名称

字符串，最大值15

老师/学生房间数量

整型

房间数量需要根据老师到达和回程时间及女性数量进行判断

老师/学生酒店价格

浮点型

老师/学生酒店总费用

浮点型

老师/学生酒店总费用需根据老师/学生酒店数量及停留时间灵活判断

出发/回程航班：

字符串型，最大值15。

出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

老师的交通工具是火车，就不需要输出该数据

老师/队伍总预算

浮点型

老师/队伍总预算=交通费用+酒店总费用

最终总预算

浮点型，队伍总预算+老师总预算。

创建人

字符串，最大值为15

### 算法

车次出发时间-车次回程到达时间=天数

### 处理流程

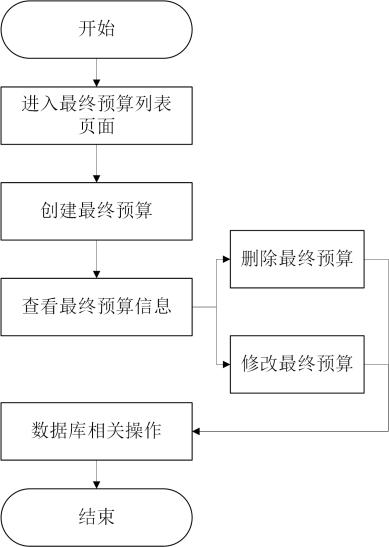


图3-8 最终预算流程图

### 类设计

#### 类图

|  |
| --- |
| **最终预算类图** |

图3‑9 类图

#### 类说明

1. STra类说明：
   1. 功能：出发车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
2. ETra类说明：
   1. 功能：回程车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
3. SAir类说明：
   1. 功能：出发航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
4. EAir类说明：
   1. 功能：回程航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
5. Acc类说明：
   1. 功能：酒店信息的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
6. Admi类说明：
   1. 功能：用户信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
7. Eleb类说明：
   1. 功能：初始预算信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
8. Ultb类说明：
   1. 功能：最终预算信息的创建，删除，修改，查询多条或得到一条，其中STra、ETra、SAir、EAir、Acc、Admi、Eleb是组成它的一部分
   2. 主要方法：create(), delect(), filter(), get()，update
9. Ultblist类说明：
   1. 功能：实现Ultbx接口中的showult()、ul\_list()、ajax\_cre\_ul()、ajax\_up\_ult()、ajax\_de\_ult()、ajax\_up\_ult()方法，即实现最终预算具体信息浏览、最终预算浏览、最终预算的创建，最终预算信息修改、最终预算删除、最终预算查询，并依赖于Ultb类进行存储

主要方法：showult()、ul\_list()、ajax\_cre\_ul()、ajax\_up\_ult()、ajax\_de\_ult()、ajax\_up\_ult()

## 实际账单



图3-10 实际账单模块机构图

### 模块编号和功能描述

YS\_zm01：用户对实际账单的创建

YS\_zm02：用户对实际账单的修改

YS\_zm03：用户对实际账单的查询

YS\_zm04：用户对实际账单的删除

YS\_zm05：用户对实际账单的总结

### 操作者

教练、总教练

### 与本模块相关的码表和表

表3‑11 模块功能表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | | | **作用** |
| **码表** | | **表** | |
| STra | 出发车次信息表 | √ |  | | 提供数据 | |
| ETra | 回程车次信息表 | √ |  | | 提供数据 | |
| SAir | 出发航班信息表 | √ |  | | 提供数据 | |
| EAir | 回程航班信息表 | √ |  | | 提供数据 | |
| CName | 竞赛账单信息表 | √ |  | | 建立竞赛账单中的实际账单 | |
| Acc | 酒店信息表 | √ |  | | 提供数据 | |
| Admi | 用户信息表 | √ |  | | 建立者 | |
| Account | 实际账单信息表 | √ |  | | 提供并显示最终预算信息 | |
| Uleb | 最终预算表 | √ |  | | 提供数据 | |

### 界面设计与说明

网页端，水平表单，颜色对比明显

### 输入信息

最终预算

选择，字符串，最大值为30

老师数量

选择，整型，大于零

队伍数量

选择，整型，大于零

女性数量

选择，整型

出发/回程车次

选择，字符串型

车次或航班

选择，字符串型

酒店

选择，字符串型

报名费

浮点型

补贴费

浮点型

备注

文本型，最大值为300。

### 输出信息

比赛名称：

字符串型，最大值30，包含地点。

老师数量：

整型，不得小于零。

学生数量：

整型，不得小于3

学生数量 = 队伍数量 \* 3

队伍数量：

整型，不得小于零。

女性数量：

整型

初始出发/回程时间

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm

学生/老师外出天数

整型，大于零

学生/老师外出天数=学生/老师回程到达时间 - 学生/老师出发时间 + 1

学生/老师停留天数

整型，大于零

学生/老师停留天数=学生/老师回程出发时间 - 学生/老师到达时间 + 1

队伍/老师出发/回程车次：

字符串型，最大值15。

如果老师的交通工具为飞机，就不用输出老师出发/回程车次

队伍/老师出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

队伍/老师/来/回票价：

浮点型

队伍/老师/来/回交通费

浮点型

队伍/老师/来/回交通费=交通票价 \* 学生/老师数量

报名费

浮点型

补贴费

浮点型

酒店名称

字符串，最大值15

老师/学生房间数量

整型

房间数量需要根据老师到达和回程时间及女性数量进行判断

老师/学生酒店价格

浮点型

老师/学生酒店总费用

浮点型

老师/学生酒店总费用需根据老师/学生酒店数量及停留时间灵活判断

出发/回程航班：

字符串型，最大值15。

出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

老师的交通工具是火车，就不需要输出该数据

老师/队伍总费用

浮点型

老师/队伍总预算=交通费用+酒店总费用

实际总费用

浮点型，队伍总预算+老师总预算。

创建人

字符串，最大值为15

备注

文本型，最大值为300。

### 算法

车次出发时间-车次回程到达时间=天数

### 处理流程

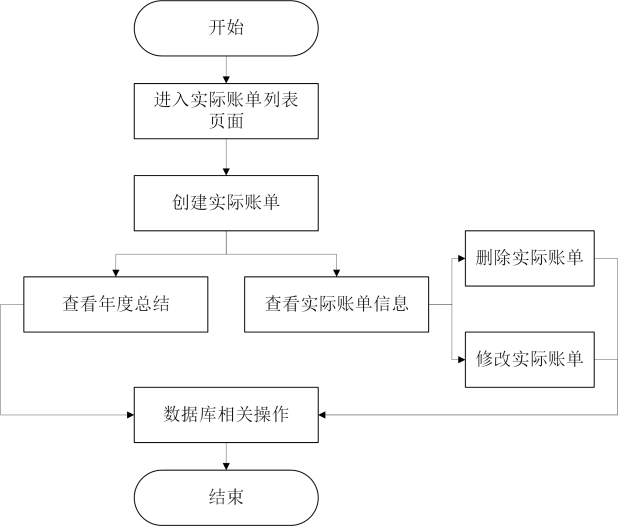


图3-12 实际账单流程图

### 类设计

#### 类图

|  |
| --- |
| **实际账单类图** |

图3‑13 类图

#### 类说明

1. STra类说明：
   1. 功能：出发车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
2. ETra类说明：
   1. 功能：回程车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
3. SAir类说明：
   1. 功能：出发航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
4. EAir类说明：
   1. 功能：回程航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
5. Acc类说明：
   1. 功能：酒店信息的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
6. Admi类说明：
   1. 功能：用户信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
7. Eleb类说明：
   1. 功能：初始预算信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
8. Account类说明：
   1. 功能：最终预算信息的创建，删除，修改，查询多条或得到一条，其中STra、ETra、SAir、EAir、Acc、Admi、Ultb是组成它的一部分
   2. 主要方法：create(), delect(), filter(), get()，update
9. Accountlist类说明：
   1. 功能：实现Accountx接口中的show\_ac(), accoun\_list(), ajax\_cre\_ac(), ajax\_up\_acc(), ajax\_de\_acc(), ajax\_se\_acc(), accyear()方法，即实现实际账单具体信息浏览、实际账单浏览、实际账单的创建，实际账单信息修改、实际账单删除、实际账单查询、实际账单总结，并依赖于Account类进行存储
   2. 主要方法：show\_ac(), accoun\_list(), ajax\_cre\_ac(), ajax\_up\_acc(), ajax\_de\_acc(), ajax\_se\_acc(), accyear()

## 用户管理

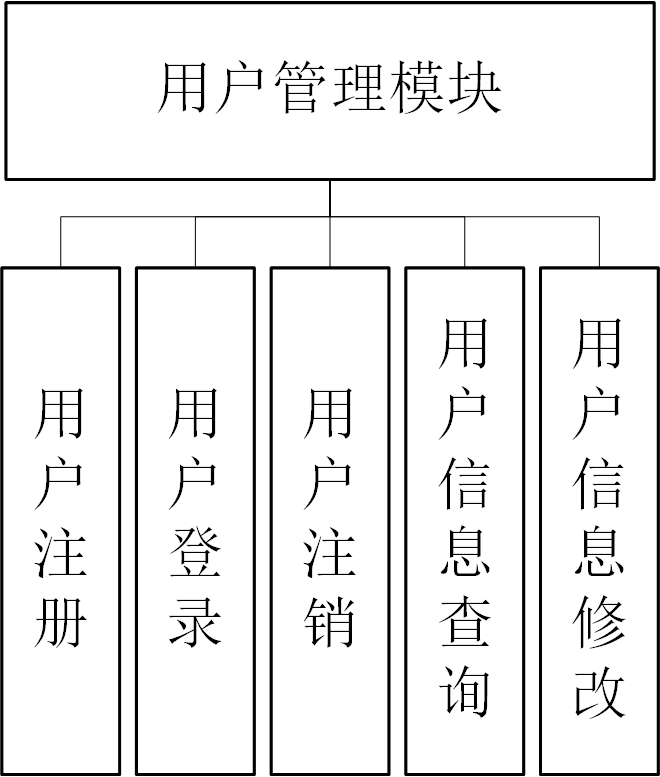


图3-14 用户管理模块机构图

### 模块编号和功能描述

YS\_adm01：用户注册

YS\_adm02：用户登录

YS\_adm03：用户注销

YS\_adm04：用户信息查询

YS\_adm05：用户信息修改

### 操作者

总教练

### 与本模块相关的码表和表

表3‑15 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| Adm | 用户信息表 | √ |  | 提供并显示数据 |

### 界面设计与说明

网页端，水平表单，颜色对比明显

### 输入信息

用户名

字符串型，最大值15。

用户密码：

字符串型，最大值20。

用户类型：

布尔型

### 输出信息

用户名

字符串型，最大值15。

用户密码：

字符串型，最大值20。

用户类型：

布尔型

### 算法

数据库的增删改查

### 处理流程

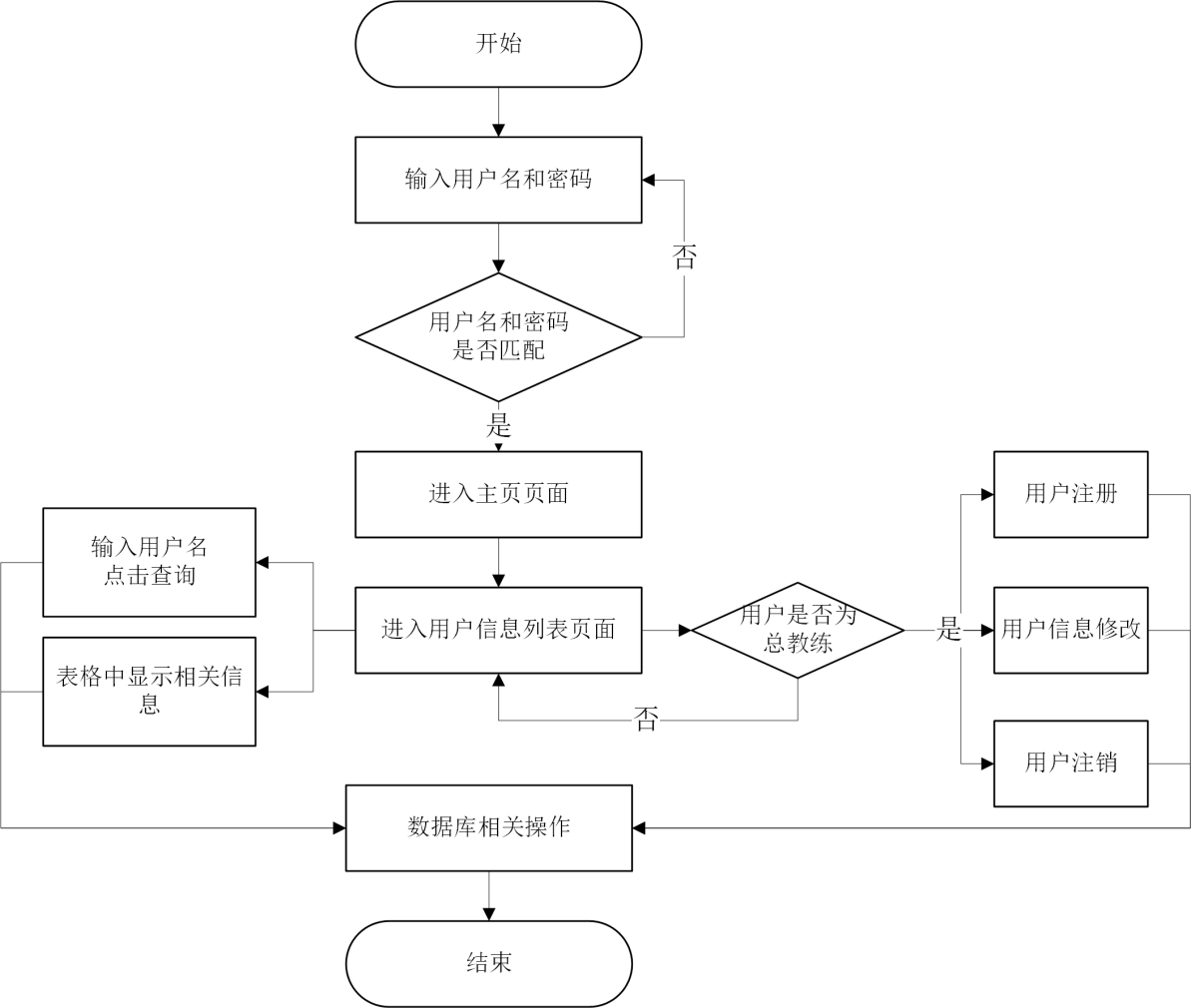


图3-16 用户管理流程图

### 类设计

#### 类图

|  |
| --- |
| **用户管理类图** |

图3‑17 用户管理类图

#### 类说明

1. Admi类说明：
   1. 功能：用户信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
2. Admilist类说明：
   1. 功能：实现Admix接口中的login(), showuser(), ajax\_cre\_user(), ajax\_up\_user(), ajax\_se\_user(), ajax\_de\_user()方法，即实现用户登录，用户信息浏览、用户注册，用户查询，用户注销，并依赖于Admi类进行存储
   2. 主要方法：login(), showuser(), ajax\_cre\_user(), ajax\_up\_user(), ajax\_se\_user(), ajax\_de\_user()

## 其它信息管理

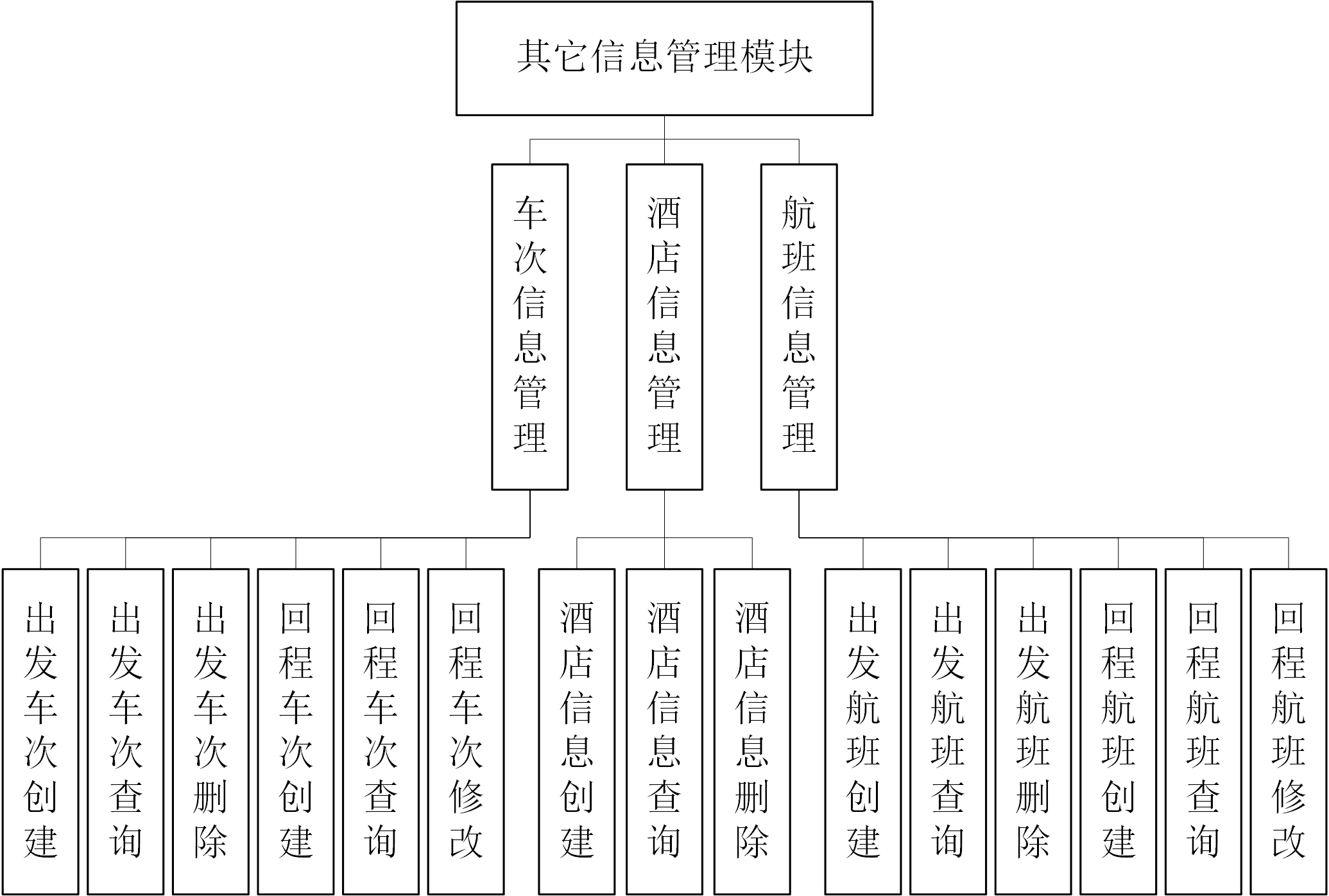


图3-18 用户管理模块机构图

### 模块编号和功能描述

YS\_qt01：车次信息管理，即出发车次和回程车次的创建、查询、删除

YS\_qt02：酒店信息管理，即酒店信息的创建、查询、删除

YS\_qt01：航班信息管理，即出发航班和回程航班的创建、查询、删除

### 操作者

教练、总教练

### 与本模块相关的码表和表

表3‑19 模块功能表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **中文注释** | **类型** | | **作用** |
| **码表** | **表** |
| STra | 出发车次信息表 | √ |  | 显示并提供数据 |
| ETra | 回程车次信息表 | √ |  | 显示并提供数据 |
| SAir | 出发航班信息表 | √ |  | 显示并提供数据 |
| EAir | 回程航班信息表 | √ |  | 显示并提供数据 |
| Acc | 酒店信息表 | √ |  | 显示并提供数据 |

### 界面设计与说明

网页端，水平表单，颜色对比明显

### 输入信息

出发/回程车次/航班号

字符串型，最大值15。

如果不是酒店信息添加

出发/到达时间：

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

如果不是酒店信息

地点

字符串型，最大为15

价格

浮点型

酒店名称

字符串型，最大为50

如果是酒店信息

### 输出信息

出发/回程车次/航班号

字符串型，最大值15。

如果不是酒店信息

出发/到达时间

日期型，具体到年、月、日、时、分，格式为：YYYY-MM-DD hh:mm。

如果不是酒店信息

地点

字符串型，最大为15

价格

浮点型

酒店名称

字符串型，最大为50

如果是酒店信息

### 算法

数据库的增删改查

### 处理流程

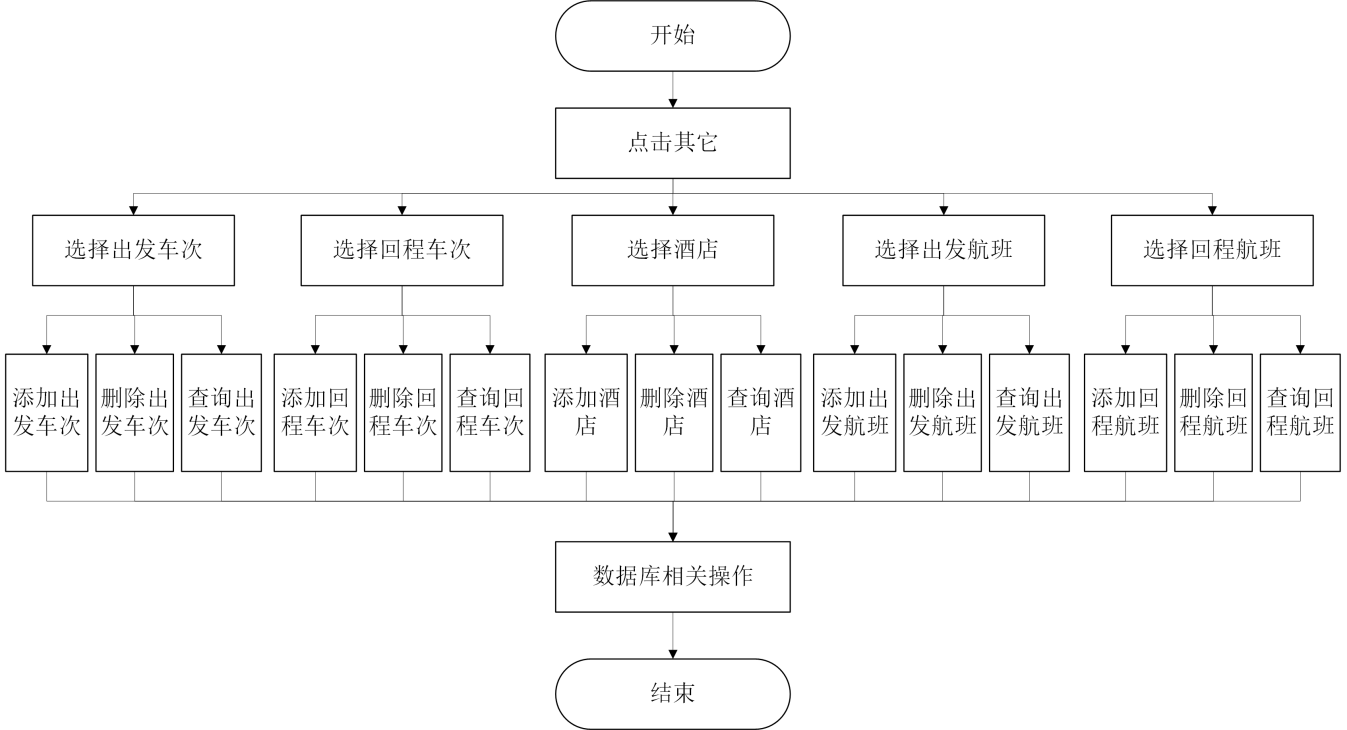


图3-20 其它信息流程图

### 类设计

#### 类图

|  |
| --- |
| **其它信息管理类** |

图3‑21 其它信息管理类图

#### 类说明

1. STra类说明：
   1. 功能：出发车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
2. ETra类说明：
   1. 功能：回程车次的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
3. SAir类说明：
   1. 功能：出发航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
4. EAir类说明：
   1. 功能：回程航班的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
5. Acc类说明：
   1. 功能：酒店信息的创建，删除，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()
6. Admi类说明：
   1. 功能：用户信息的创建，删除，修改，查询多条或得到一条
   2. 主要方法：create(), delect(), filter(), get()，update
7. Qtlist类说明：
   1. 功能：实现Qtx接口中的show\_qt(), ajax\_cre\_qt(), ajax\_de\_qt(), ajax\_se\_qt()方法，即实现其它信息浏览、其它信息的添加，其它信息的删除、其它信息查询并依赖于STra、ETra、SAir、EAir、Acc类进行存储
   2. 主要方法：show\_qt(), ajax\_cre\_qt(), ajax\_de\_qt(), ajax\_se\_qt()

# 内部接口设计

表4‑1 构件接口列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块名称** | **接口编号** | **接口名称** | **接口类型** | **说明** |
| 初始预算管理 | JK\_cjys01 | Spierdx | 内部 | 通过网络爬虫实现出发车次、回程车次、出发航班、回程航班、酒店信息的存储 |
| Jk\_cjys02 | Elebx | 外部 | 通过用户请求，实现初始预算的创建、浏览、修改、删除和查询 |
| 最终预算管理 | JK\_zzys | Ultbx | 外部 | 通过用户请求，实现最终预算的创建、浏览、修改、删除和查询 |
| 实际账单管理 | JK\_sjzd | Accountx | 外部 | 通过用户请求，实现实际账单的创建、浏览、修改、删除、查询和年度总结 |
| 用户信息管理 | JK\_user | Admix | 外部 | 通过用户请求，实现用户注册、用户登录、用户信息修改、用户查询、用户注销和用户信息浏览 |
| 用户信息管理 | JK\_qt | Qtx | 外部 | 通过用户请求，实现其它信息浏览、其它信息查询、其它信息添加、其它信息删除和其它信息查询 |

## 接口1

1. 接口属性设计

表4‑2Spierdx接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_cjys01 |
| **接口名称** | Spierdx |
| **接口说明** | 通过网络爬虫实现出发车次、回程车次、出发航班、回程航班、酒店信息的存储 |
| **数据来源** | 用户输入和网络爬虫 |
| **调用者** | 教练和总教练 |
| **输入** | 接口调用，出发时间，到达时间，地点 |
| **输出** | 出发车次、回程车次、出发航班、回程航班、酒店信息 |

1. 类设计

表4‑3 STra类

|  |  |
| --- | --- |
| **类名称** | 出发车次信息 |
| **分类** | 实体 |
| **描述** | 含所有通过网络爬虫的出发车次信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：T\_id, id, stime, etime, price, site  方法：create(), delect(), filter(), get() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑4 ETra类

|  |  |
| --- | --- |
| **类名称** | 回程车次信息 |
| **分类** | 实体 |
| **描述** | 含所有通过网络爬虫的回程车次信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：T\_id, id, stime, etime, price, site  方法：create(), delect(), filter(), get() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑5 SAir类

|  |  |
| --- | --- |
| **类名称** | 出发航班信息 |
| **分类** | 实体 |
| **描述** | 含所有通过网络爬虫的出发航班信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：A\_id, id, stime, etime, price, site  方法：create(), delect(), filter(), get() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑6 EAir类

|  |  |
| --- | --- |
| **类名称** | 回程航班信息 |
| **分类** | 实体 |
| **描述** | 含所有通过网络爬虫的回程航班信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：A\_id, id, stime, etime, price, site  方法：create(), delect(), filter(), get() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑7 Acc类

|  |  |
| --- | --- |
| **类名称** | 酒店信息 |
| **分类** | 实体 |
| **描述** | 含所有通过网络爬虫的酒店信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：name, id, price, site  方法：create(), delect(), filter(), get() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

## 接口2

1. 接口属性设计

表4‑8 Elebx接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_cjys02 |
| **接口名称** | Elebx |
| **接口说明** | 通过用户请求，实现初始预算的创建、浏览、修改、删除和查询 |
| **数据来源** | 用户输入和数据库 |
| **调用者** | 教练和总教练 |
| **输入** | 表单请求，老师数量，队伍数量，出发时间，到达时间等信息 |
| **输出** | 初始预算信息表 |

1. 类设计

表4‑9 Eleb类

|  |  |
| --- | --- |
| **类名称** | 初始预算信息 |
| **分类** | 实体 |
| **描述** | 含所有初始预算必须含有的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：id, name, tecnum, teanum, funm, stime, etime, site, allo, apply, str\_id, etr\_id, sai\_id, eai\_id, ac\_id, ad\_id  方法：create(), delect(), filter(), get()，update |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑10 Eleblist类

|  |  |
| --- | --- |
| **类名称** | 初始预算管理 |
| **分类** | 实体 |
| **描述** | 含所有初始预算表的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：name, teachernum, teamnum, studentnum, feanum, studentdays, stuholdays, ty, teacherdays, teajoldays, teas\_air, teacs\_pri, teas\_tra, teae\_tra, teace\_pri, stus\_tra, stue\_tra, stus\_pri, stue\_pri, acc, stu\_acc, ea\_acc, stu\_accpr, tea\_accpr, allo, apply, stu\_pri, tea\_pri, price  方法：cre()、showele()、sele2()、ajax\_up\_ele()、ajax\_de\_ele() |
| **使用/交互** | 用户交互，请求url，调用方法 |
| **其他** | 无 |

## 接口3

1. 接口属性设计

表4‑11 Ultbx接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_zzys |
| **接口名称** | Ultbx |
| **接口说明** | 通过用户请求，实现最终预算的创建、浏览、修改、删除和查询 |
| **数据来源** | 用户输入和数据库 |
| **调用者** | 教练和总教练 |
| **输入** | Url请求，初始预算信息等 |
| **输出** | 最终预算信息表 |

1. 类设计

表4‑9 Ultb类

|  |  |
| --- | --- |
| **类名称** | 最终预算信息 |
| **分类** | 实体 |
| **描述** | 含所有最终预算必须含有的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：id, name, tecnum, teanum, funm, stime, etime, site, allo, apply, str\_id, etr\_id, sai\_id, eai\_id, ac\_id, ad\_id, el\_id  方法：create(), delect(), filter(), get()，update |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑10 Ultblist类

|  |  |
| --- | --- |
| **类名称** | 最终预算管理 |
| **分类** | 实体 |
| **描述** | 含所有最终预算表的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：name, teachernum, teamnum, studentnum, feanum, studentdays, stuholdays, ty, teacherdays, teajoldays, teas\_air, teacs\_pri, teas\_tra, teae\_tra, teace\_pri, stus\_tra, stue\_tra, stus\_pri, stue\_pri, acc, stu\_acc, ea\_acc, stu\_accpr, tea\_accpr, allo, apply, stu\_pri, tea\_pri, price  方法：showult()、ul\_list()、ajax\_cre\_ul()、ajax\_up\_ult()、ajax\_de\_ult()、ajax\_up\_ult() |
| **使用/交互** | 用户交互，请求url，调用方法 |
| **其他** | 无 |

## 接口4

1. 接口属性设计

表4‑11 Accountx接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_sjzd |
| **接口名称** | Accountx |
| **接口说明** | 通过用户请求，实现实际账单的创建、浏览、修改、删除、查询和年度总结 |
| **数据来源** | 用户输入和数据库 |
| **调用者** | 教练和总教练 |
| **输入** | Url请求，最终预算信息等 |
| **输出** | 实际账单信息 |

1. 类设计

表4‑12 Account类

|  |  |
| --- | --- |
| **类名称** | 实际账单信息 |
| **分类** | 实体 |
| **描述** | 含所有实际账单必须含有的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：id, name, tecnum, teanum, funm, stime, etime, site, allo, apply, str\_id, etr\_id, sai\_id, eai\_id, ac\_id, ad\_id, ul\_id, comm  方法：create(), delect(), filter(), get()，update |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑13 Accountlist类

|  |  |
| --- | --- |
| **类名称** | 实际账单管理 |
| **分类** | 实体 |
| **描述** | 含所有实际账单表的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：name, teachernum, teamnum, studentnum, feanum, studentdays, stuholdays, ty, teacherdays, teajoldays, teas\_air, teacs\_pri, teas\_tra, teae\_tra, teace\_pri, stus\_tra, stue\_tra, stus\_pri, stue\_pri, acc, stu\_acc, ea\_acc, stu\_accpr, tea\_accpr, allo, apply, stu\_pri, tea\_pri, price  方法：show\_ac(), accoun\_list(), ajax\_cre\_ac(), ajax\_up\_acc(), ajax\_de\_acc(), ajax\_se\_acc(), accyear() |
| **使用/交互** | 用户交互，请求url，调用方法 |
| **其他** | 无 |

## 接口5

1. 接口属性设计

表4‑14 Admix接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_user |
| **接口名称** | Admix |
| **接口说明** | 通过用户请求，实现用户注册、用户登录、用户信息修改、用户查询、用户注销和用户信息浏览 |
| **数据来源** | 用户输入和数据库 |
| **调用者** | 总教练 |
| **输入** | Url请求，用户信息 |
| **输出** | 用户信息 |

1. 类设计

表4‑15 Admi类

|  |  |
| --- | --- |
| **类名称** | 用户信息 |
| **分类** | 实体 |
| **描述** | 含所有用户的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：id, name, passwrd, ty  方法：create(), delect(), filter(), get()，update() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑16 Admilist类

|  |  |
| --- | --- |
| **类名称** | 用户信息管理 |
| **分类** | 实体 |
| **描述** | 含所有用户信息的操作方法 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：name, passwrd, ty  方法：login(), showuser(), ajax\_cre\_user(), ajax\_up\_user(), ajax\_se\_user(), ajax\_de\_user() |
| **使用/交互** | 用户交互，请求url，调用方法 |
| **其他** | 无 |

## 接口5

1. 接口属性设计

表4‑17 Qtx接口说明

|  |  |
| --- | --- |
| **接口编号** | JK\_Qt |
| **接口名称** | Qtx |
| **接口说明** | 通过用户请求，实现其它信息浏览、其它信息查询、其它信息添加、其它信息删除和其它信息查询 |
| **数据来源** | 用户输入和数据库 |
| **调用者** | 教练或总教练 |
| **输入** | Url请求，出发车次，回程车次，出发航班，回程航班，酒店信息等 |
| **输出** | 出发车次，回程车次，出发航班，回程航班，酒店信息 |

1. 类设计

表4‑17 Qt类

|  |  |
| --- | --- |
| **类名称** | 其它信息 |
| **分类** | 实体 |
| **描述** | 含所有出发车次、回程车次、出发航班、回程航班、酒店的信息 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：id, str, etr, sai, eai, acc  方法：create(), delect(), filter(), get()，update() |
| **使用/交互** | 直接调用方法 |
| **其他** | 无 |

表4‑18 Qtlist类

|  |  |
| --- | --- |
| **类名称** | 其它信息管理 |
| **分类** | 实体 |
| **描述** | 含所有其它信息的操作方法 |
| **使用到的其他类** | 无 |
| **属性及方法描述** | 属性：str, etr, sai, eai, acc  方法：show\_qt(), ajax\_cre\_qt(), ajax\_de\_qt(), ajax\_se\_qt() |
| **使用/交互** | 用户交互，请求url，调用方法 |
| **其他** | 无 |