目录

[1. 系统分析： 1](#_Toc20654)

[1.1. 课题背景 1](#_Toc11752)

[1.2. 开发环境和软件 2](#_Toc11276)

[1.3. 目的和意义 2](#_Toc16407)

[1.4. 系统设计与功能分析：（大致功能规划） 2](#_Toc10434)

[1.5. 可行性分析（业务流程介绍） 2](#_Toc20425)

[2. 总体设计： 3](#_Toc2380)

[2.1. 数据设计： 3](#_Toc7153)

[2.1.1. 逻辑设计： 3](#_Toc1447)

[2.1.2. 物理设计： 4](#_Toc1538)

[2.2. 对外接口设计： 4](#_Toc9511)

[2.3. 详细设计： 4](#_Toc7435)

[2.3.1. 位于服务器端的关系数据库表的设计 4](#_Toc21402)

[2.3.2. 对数据库本身的物理设计（表内详细信息） 5](#_Toc15020)

[2.3.3. 网页的详细设计（功能模块叙述） 5](#_Toc29926)

[2.3.4. 数据库系统接口的详细设计。 5](#_Toc2265)

# 系统分析：

## 课题背景

中国拥有总里程超过五万公里的铁路线，是世界上最大的铁路运输网之一，而铁路客运服务在其中又占有非常重要的地位。其中有5000多个车站承办业务，日开列车2000多列。为了在日益加剧的客户运输竞争服务中确保优秀，改善铁路客户的服务质量，铁道部门一直在寻找提高竞争力、改善服务的途径。计算机应用火车站售票的日常管理为火车站售票的现代化带来了前所未有的动力和机遇，为火车站票务管理领域的飞速发展提供了无限潜力。能给火车站票务带来明显的经济效益和社会效益。

## 开发环境和软件

## 目的和意义

火车票票务管理的全部数据处理都由人工操作，工作量大，工作效率低，错误率高，信息反馈不及时。为了解决上述问题，要根据目前火车票的管理模式和方法利用计算机开发基于browser的火车票订票系统，可以票额、坐席、计费、结算、统计等工作的实现。形成统一的客票信息源，实现信息共享，同时方便客户购票。

## 系统设计与功能分析：（大致功能规划）

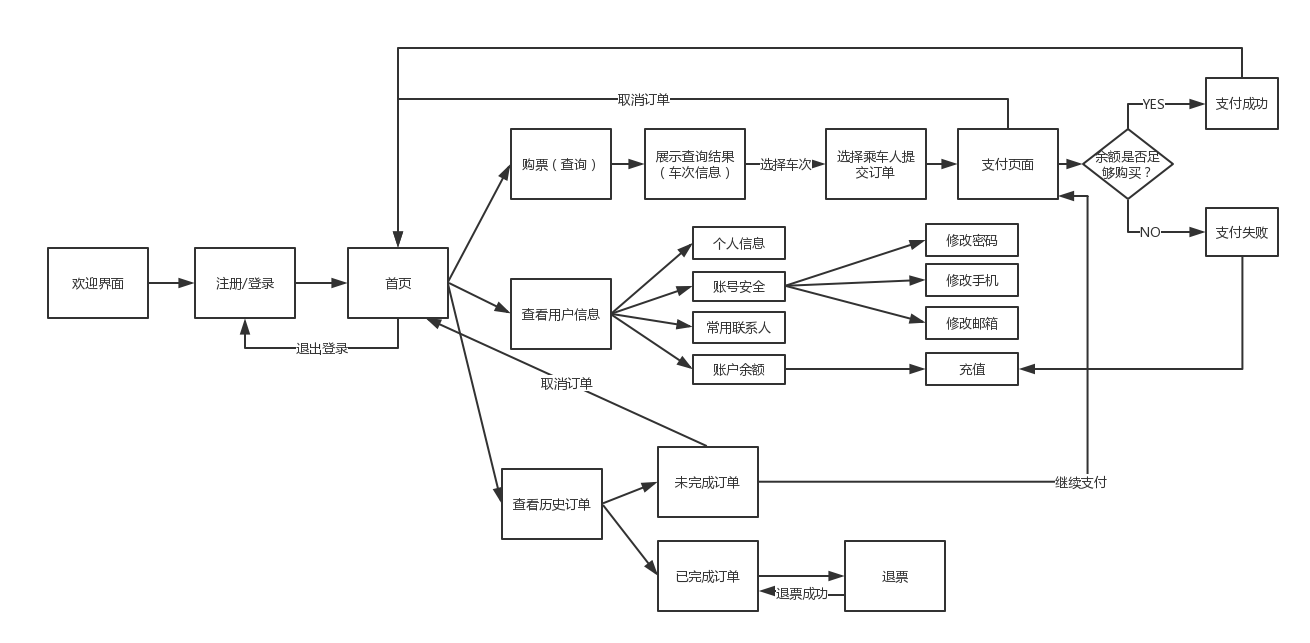
1. 查询。为对车次信息的查询：车次信息包括： 日期、出发站、目的站、车次、类型、出发时间、到达时间、票价、余票；对历史订单的车票信息的查询：订单信息包括：车次信息、席位信息、旅客信息、票款金额、车票状态。客户经过查询结果来进行想要的互动操作。查看个人信息：

(2)售票。通过查询系统，可以根据客户的需求找到车次，再选择客户信息后确定售票，订票信息应包括：。

(3)退票。可退票，通过查询系统，根据客户的名字找到购票信息，通过退票模块退去已购车票。

(4)修改个人信息：

## 可行性分析（业务流程介绍）



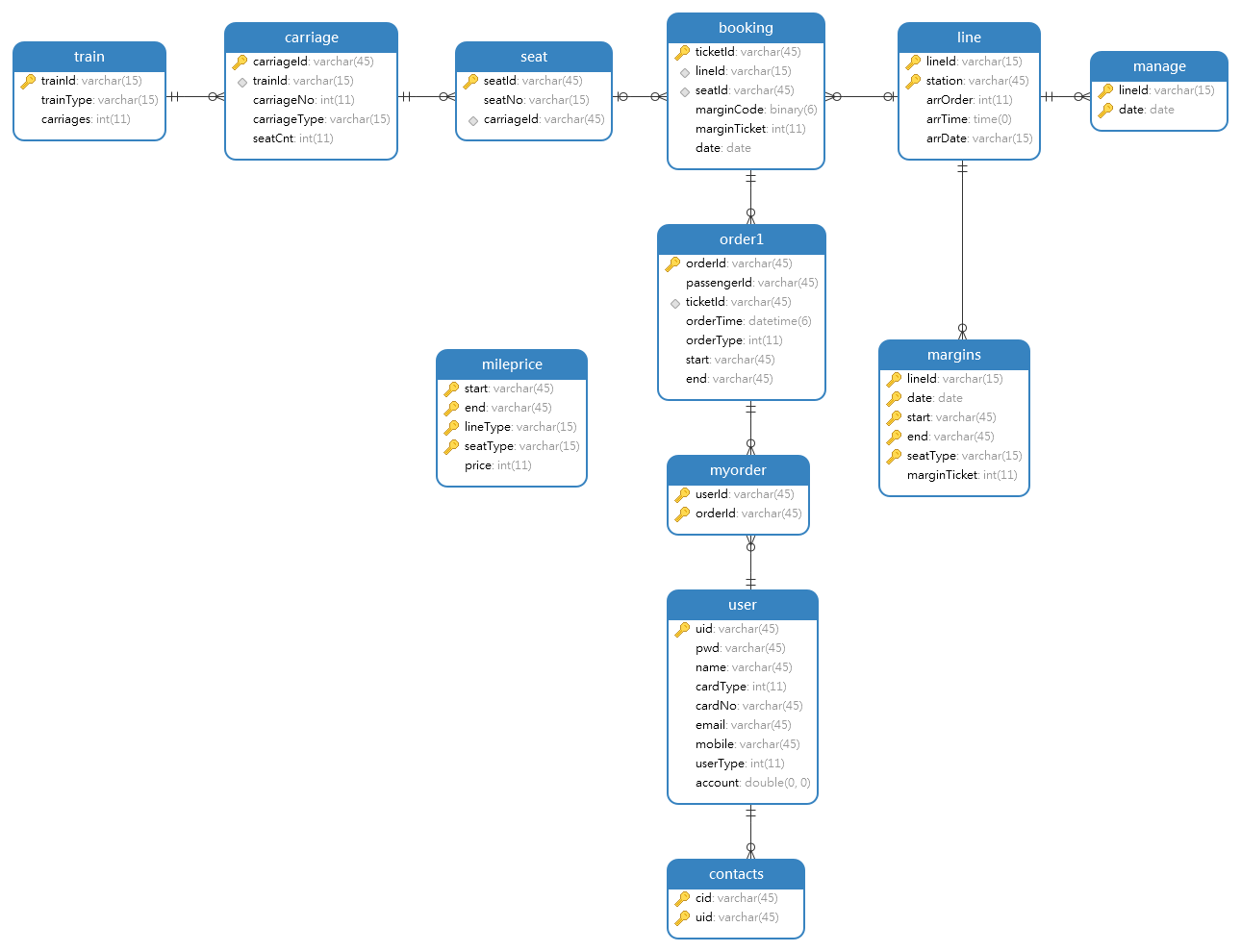
# 总体设计：

## 数据设计：

解决数据库中有哪些数据，这些数据是怎样分类的，数据之间有什么联系，怎样把数据转化为关系表的方式组织，数据怎样存储以方便操作建立系统信息模型、根据管理信息模型建立系统的E-R模型、**关系表文件怎样存储**

### 逻辑设计：

#### ER图如下：



#### 关系模式：

train (trainId , trainType , carriages)

carriage (carriageId , trainId , carriageNo , carriageType , seatCnt)

seat (seatId , seatNo , carriageId)

booking (ticketId , lineId , seatId , marginCode , marginTicket , date)

line (lineId , station , arrOrder , arrTime , arrDate)

margins (lineId , date , start , end , seatType , marginTicket)

manage (lineId , date)

order1 (orderId , passengerId , ticketId , orderTime , orderType , start , end)

myorder (userId , orderId)

user (uid , pwd , name , cardType , cardNo , email , mobile , userType , account)

contacts (cid , uid)

mileprice (start , end , lineType , seatType , price)

### 物理设计：

共12个表分为：

用户信息表(user)

常用联系人表(contacts)

价目表(mileprice)

路线表(line)

列车表(train)

车厢表(carriage)

座位表(seat)

运营表(manage)

订单表(order)

用户订单表(myorder)

座位对应订单表(booking)

余票表(margin)

## 对外接口设计：

前端提交表单、后端PHP....

## 详细设计：

### 位于服务器端的关系数据库表的设计

**(详细叙述2.1)**

### 对数据库本身的物理设计（表内详细信息）

#### 用户信息表(user)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 表中存储名 | 变量类型 | 完整性约束条件 | 补充说明 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

#### 常用联系人表(contacts)

#### 价目表(mileprice)

#### 路线表(line)

#### 列车表(train)

#### 车厢表(carriage)

#### 座位表(seat)

#### 运营表(manage)

#### 订单表(order)

#### 用户订单表(myorder)

#### 座位对应订单表(booking)

#### 余票表(margin)

### 网页的详细设计（功能模块叙述）

### 数据库系统接口的详细设计。

前端提交表单，如： <form id="click" method="post" action="">

<input type="submit" id="button" value="查 询" />

</form>

后端PHP.....