**物流增值业务系统**

**详细设计说明书**

**班 级：07111707**

**小组成员：1120151874 管京渊**

**1120171189 崔程远**

**1120172765 曾煜瑾**

**1120173596 王西雨**

**1820171044 麦菲莉**

**1820171020格瑞斯**

**1820171015弗兰克**

**2020年11月**

目录

[1.引言 4](#_Toc529492280)

[1.1编写目的 4](#_Toc529492281)

[1.2适用范围 4](#_Toc529492282)

[1.3项目背景 4](#_Toc529492283)

[1.4术语、定义和缩写 4](#_Toc529492284)

[1.5参考资料 5](#_Toc529492285)

[2．车辆保险系统功能模块设计 5](#_Toc529492286)

[2.1车辆保险管理系统的类图、构件图和配置图 5](#_Toc529492287)

[2.2投保/退保/续保业务办理模块 6](#_Toc529492288)

[2.3查询模块 9](#_Toc529492289)

[3.车辆租赁系统功能模块设计 12](#_Toc529492290)

[3.1基本信息管理模块的类图、构件图和配置图 12](#_Toc529492291)

[3.2管理员添加信息模块 13](#_Toc529492292)

[3.3管理员添加汽车信息模块 15](#_Toc529492293)

[3.4管理员处理汽车信息（修改汽车信息、删除汽车信息）模块 17](#_Toc529492294)

[3.5汽车信息检索模块 19](#_Toc529492295)

[3.6车辆租赁管理系统服务端管理员登录模块 21](#_Toc529492296)

[3.7 管理员处理预定订单与托管申请模块 23](#_Toc529492297)

[3.8 管理员处理续借订单模块 26](#_Toc529492298)

[3.9 管理员提醒订单将截至信息 28](#_Toc529492299)

[3.10 管理员结束租赁到期订单模块 29](#_Toc529492300)

[3.11 管理员管理优惠活动模块 31](#_Toc529492301)

[3.12 管理员修改密码模块 32](#_Toc529492302)

[3.13 管理员退出管理模块 34](#_Toc529492303)

[4. 车辆租赁系统客户服务端设计 35](#_Toc529492304)

[4.1 用户登录和注销模块 37](#_Toc529492305)

[4.2 用户注册模块 39](#_Toc529492306)

[4.3 汽车信息检索模块 43](#_Toc529492307)

[4.4 查看汽车详细信息模块 47](#_Toc529492308)

[4.5在线客服模块 49](#_Toc529492309)

[4.6 预订汽车信息模块 50](#_Toc529492310)

[4.7 托管申请模块 53](#_Toc529492311)

[4.8 查看租赁订单模块 55](#_Toc529492312)

[4.9 个人信息面板模块 57](#_Toc529492313)

[5.进销台账模块 60](#_Toc529492314)

[5.1类图、构件图和配置图 60](#_Toc529492315)

[5.2功能描述 61](#_Toc529492316)

[5.3性能 61](#_Toc529492317)

[5.4输入项 61](#_Toc529492318)

[5.5输出项 61](#_Toc529492319)

[5.6算法 62](#_Toc529492320)

[5.7流程图 62](#_Toc529492321)

[5.8限制条件 62](#_Toc529492322)

[5.9测试条件 62](#_Toc529492323)

[5.10尚未解决的问题 63](#_Toc529492324)

[6.短信管理模块 63](#_Toc529492325)

[6.1类图、构件图和配置图 63](#_Toc529492326)

[6.2功能描述 64](#_Toc529492327)

[6.3性能 64](#_Toc529492328)

[6.4输入项 64](#_Toc529492329)

[6.5输出信息 64](#_Toc529492330)

[6.6算法 65](#_Toc529492331)

[6.7流程图 65](#_Toc529492332)

[6.8限制条件 65](#_Toc529492333)

[6.9测试条件 65](#_Toc529492334)

[6.10尚未解决的问题 65](#_Toc529492335)

[7.平台支付功能模块 65](#_Toc529492336)

[7.1类图、构件图和配置图 65](#_Toc529492337)

[7.2支付模块 66](#_Toc529492338)

[7.3后台管理（查看用户支付信息、支付安全检测）模块 67](#_Toc529492339)

[8.发票管理功能 68](#_Toc529492340)

[8.1类图、构件图和配置图 68](#_Toc529492341)

[8.2发票开具模块 69](#_Toc529492342)

[8.3发票查询模块 70](#_Toc529492343)

# 1.引言

## 1.1编写目的

本文档服务于道路运输车辆监控-增值业务系统的设计开发，通过对该系统进行需求工程，完成包括问题定义、可行性分析、需求分析等内容，回答“本系统必须做什么”的问题。本文档将对系统必须完成的任务内容、用户操作流程顺序、系统约束条件的规定等问题提供完整、准确、一致的描述。为开发团队后续进行概要设计、详细设计等工作进行指导和规范。

该文档详尽说明了这一软件产品的需求和规格，这些规格说明是进行设计的基础，也是编写测试用例和进行系统测试的主要依据。同时，该文档也是用户确定软件功能需求的主要依据。

## 1.2项目背景

随着运输企业业务范围的不断扩大和业务量的增加，现需要一种具有增值性功能的平台。通过增值业务，企业可以为这些注册车辆统一进行保险办理和租赁管理，并提供支付平台进行网络支付，车辆台帐管理实时显示每台车的进销情况。同时，该系统与税务系统、保险公司系统、车管所业务系统有相应的接口，提供发票数据、保险数据和车辆违章数据以供公司管理人员和车辆使用人员了解。根据业务需要，系统提供各种统计、分析和报表功能。

## 1.3术语、定义和缩写

1）MVC框架：是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

2）JT/T-796标准：是指道路运输车辆卫星定位系统平台技术要求，规定了道路运输车辆卫星定位系统架构，以及道路运输车辆卫星定位系统中政府监管平台和企业监控平台的功能要求/平台性能与技术要求等内容。

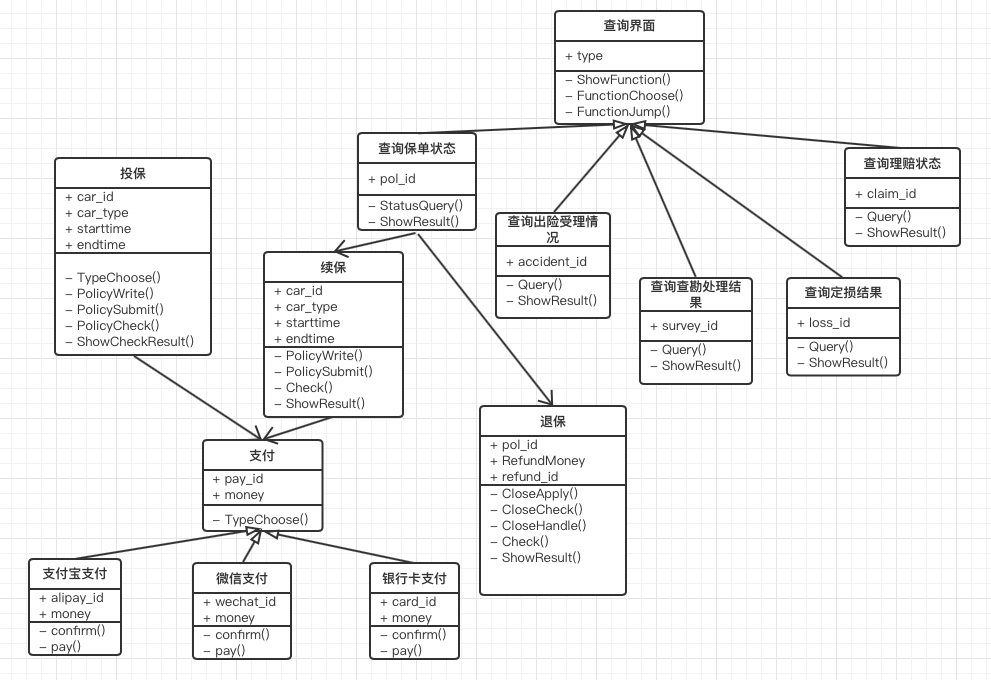
3）第三方保险公司：是指为本系统及客户提供保险业务服务的第三方公司，主要业务范围：保险业务推荐，理赔。

4）车辆租赁管理系统：作为道路车辆监控平台系统的增值业务平台实现车辆的租赁管理，通过一套平台全面对各区域、分店、分支机构、各网点车辆、订单、结算数据进行统一实时的管理，最终达到企业全面、统一、科学、有效管理车辆的目标。

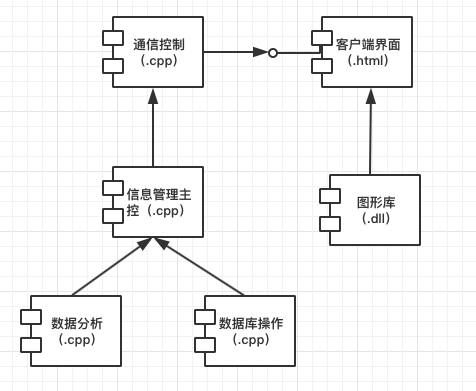
# 2．车辆保险系统功能模块设计

## 2.1车辆保险管理系统的类图、构件图和配置图

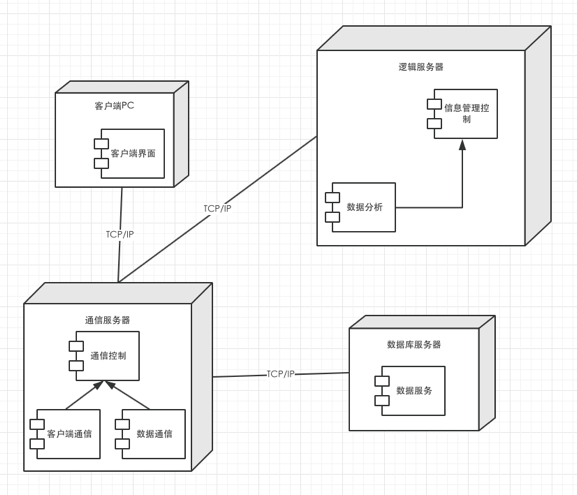
类图如下图所示：



构件图如下所示：



配置图如下所示：



## 2.2投保/退保/续保业务办理模块

### 2.2.1功能

保险业务办理模块主要完成用户进行投保、续保、退保等操作。在用户选择投保或者续保时，显示保险公司及险种的信息，在用户填写好保单以后，将保单传递到保险公司的业务接口，保险公司进行审核，若审核通过，则该子系统提供一个支付的接口，用户进行支付；若审核不通过，将审核信息反馈给用户。

### 2.2.2性能

（1）数据准确度

实时调用企业平台支持的互联网或专线网络进行查询相应账户信息及内容发布服务。通过企业自己的云端和数据库存储数据信息，系统服务器存储用户的个人信息等，数据在线存储时间不少于83天。系统支持至少1000终端接入可以同时跟踪运算10000个动态目标。

（2）时间特性

a．数据转换与传输时间（即与同步服务器与云端数据的速率）

每条记录文字部分300B，在目前3G/4G普遍的应用的情况下，每秒可传输达到平均600条/s记录，峰值1200条/s；云端数据同步时，要求在10s时间内完成；车载终端在非休眠状态下上报动态数据时间间隔不大于60s

b．数据存储和备份：

数据库每秒100w条数据更新：

在本机上的本地数据库更新，可在极短的时间内完成，同时在服务器端的数据更新，上传可在极短的时间内完成( < 5s)，下载在局部可在极短的时间内完成，整体在极端的条件下10s内完成。

数据库每周进行增量备份，系统数据恢复时间不超过5h

### 2.2.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 说明 |
| 客户编号 | Customer\_id | text | Int | 投保、续保、退保时输入 |
| 车牌号 | LicensePlateNumber | text | String | 投保、续保、退保时输入 |
| 车辆类型 | Car\_type | text | String | 投保、续保时输入 |
| 投保公司编号 | Company\_id | text | Int | 投保、续保时输入 |
| 投保险种编号 | Product\_id | text | Int | 投保、续保时输入 |
| 起保时间 | StartTime | text | Datetime | 投保、续保时输入 |
| 终保时间 | Endtime | text | Datetime | 投保、续保时输入 |
| 保险金额 | Money | text | Double | 投保、续保时输入 |

### 2.2.4输出项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输出 | 英文标识 | 数据格式 | 数据类型 | 说明 |
| 审核未通过信息 | Error | text | String | 投保或续保过程中，审核未通过时输出 |
| 支付单号 | Pay\_id | text | String | 投保或续保过程中，审核通过时输出 |
| 退保受理信息 | Exit\_msg | text | String | 退保时输出 |

### 2.2.5算法

从输入的文本框中取得数据，将数据通过接口传递给保险公司，保险公司进行审核，若审核成功，则生成支付单号，让用户支付保金；若审核不成功，则返回审核未通过的信息，用户根据反馈的信息进行修改重新提交或放弃。

### /Users/masiyi/Desktop/屏幕快照 2018-11-08 20.29.17.png2.2.6流程图

### 2.2.7限制条件

数据库在使用前要配置成功，且确保输入的车牌号、客户编号、保险公司编号和投保险种编号唯一。

### 2.2.8测试计划

1、分别输入错误的客户编号、车牌号、投保公司编号、投保险种编号观察运行结果。

2、输入不匹配的投保公司编号和投保险种编号，观察程序运行结果。

3、输入不匹配的车辆类型和投保类型，观察运行结果。

4、输入的起保时间在终保时间之后，观察运行结果。

5、输入的起保时间与终保时间的不符合投保险种的保险时长，观察运行结果。

6、输入正确的信息，观察运行结果。

## 2.3查询模块

### 2.3.1功能

查询模块主要是面向客户端的服务，用户查询保单状态、出险受理情况、查勘处理结果、定损结果和理赔状态等信息。当用户输入相应的编号时，通过sql语句对用户的保单状态、出险受理情况、查勘处理结果、定损结果和理赔状态进行查询和显示。

### 2.3.2性能

（1）数据准确度

实时调用企业平台支持的互联网或专线网络进行查询相应账户信息及内容发布服务。通过企业自己的云端和数据库存储数据信息，系统服务器存储用户的个人信息等，数据在线存储时间不少于83天。系统支持至少1000终端接入可以同时跟踪运算10000个动态目标。

（2）时间特性

a．数据转换与传输时间（即与同步服务器与云端数据的速率）

每条记录文字部分300B，在目前3G/4G普遍的应用的情况下，每秒可传输达到平均600条/s记录，峰值1200条/s；云端数据同步时，要求在10s时间内完成；车载终端在非休眠状态下上报动态数据时间间隔不大于60s

b．数据存储和备份：

数据库每秒100w条数据更新：

在本机上的本地数据库更新，可在极短的时间内完成，同时在服务器端的数据更新，上传可在极短的时间内完成( < 5s)，下载在局部可在极短的时间内完成，整体在极端的条件下10s内完成。

数据库每周进行增量备份，系统数据恢复时间不超过5h

### 2.3.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 说明 |
| 保单编号 | Policy\_id | text | Int | 查询保单状态时输入 |
| 出险单编号 | Accident\_id | text | Int | 查询出险受理情况时输入 |
| 查勘单编号 | Survey\_id | text | Int | 查询查勘处理结果时输入 |
| 定损单编号 | Loss\_id | text | Int | 查询定损结果时输入 |
| 理赔单编号 | Claim\_id | text | Int | 查询理赔状态时输入 |

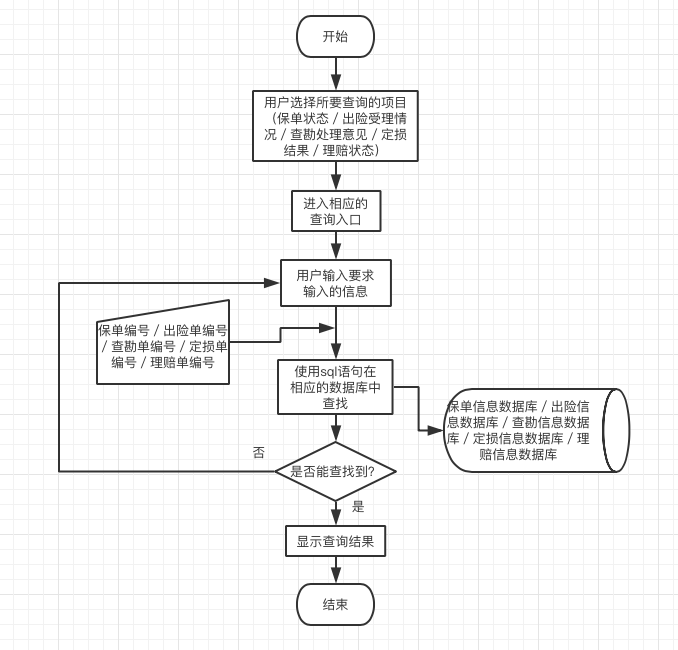
### 2.3.4输出项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输出 | 英文标识 | 数据格式 | 数据类型 | 说明 |
| 保单状态 | Policy\_status | text | String | 过期、在保或注销状态 |
| 出险受理情况 | Accident\_result | text | String | 予以/不予立案或注销 |
| 查勘处理意见 | Survey\_result | text | String | 包括责任判断、责任比例等信息 |
| 定损结果 | Loss\_result | text | String | 包括车损、物损、人伤明细 |
| 理赔状态 | Claim\_type | text | String | 待赔款、已赔款、注销状态 |

### 2.3.5算法

从输入的文本框中取得用户要查询的项目（保单状态／出险受理情况／查勘处理意见／定损结果／理赔状态），然后进入查询入口，让用户填写相应的信息（保单编号／出险单编号／查勘单编号／定损单编号／理赔单编号），在响应的数据库中使用sql语句进行查询，然后将查询结果反馈给用户。

### 2.3.6流程图



### 2.3.7 限制条件

数据库在使用前要配置成功，且确保输入的保单编号、出险单编号、查勘单编号、定损单编号和理赔单编号唯一。

### 2.3.8 测试计划

1、分别输入错误的保单编号、出险单编号、查勘单编号、定损单编号和理赔单编号，观察运行结果。

2、分别输入格式不规范的保单编号、出险单编号、查勘单编号、定损单编号和理赔单编号，观察运行结果。

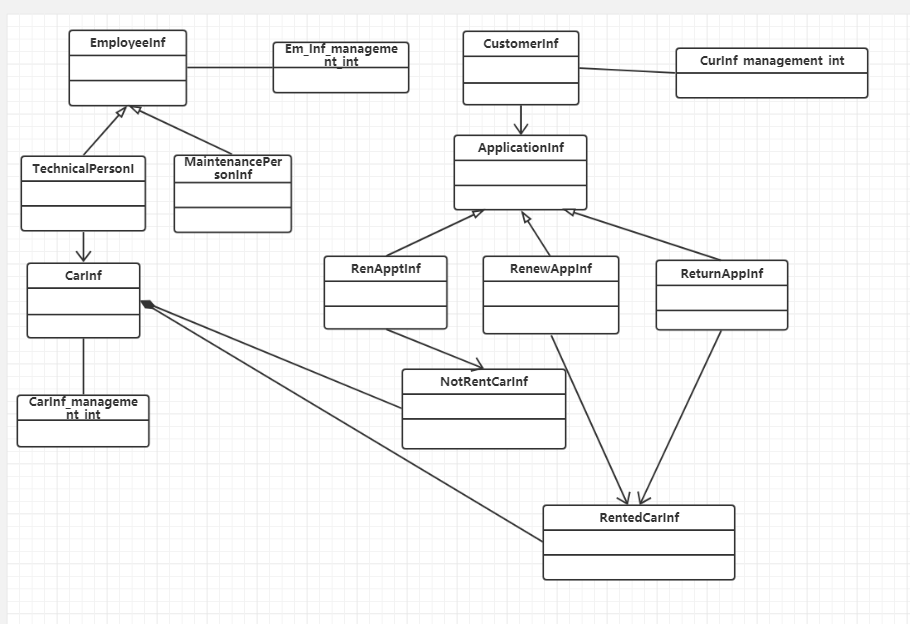
3、分别输入不存在或者已经注销的保单编号、出险单编号、查勘单编号、定损单编号和理赔单编号，观察运行结果。

4、分别输入正确的保单编号、出险单编号、查勘单编号、定损单编号和理赔单编号，观察运行结果。

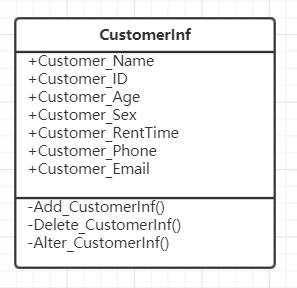
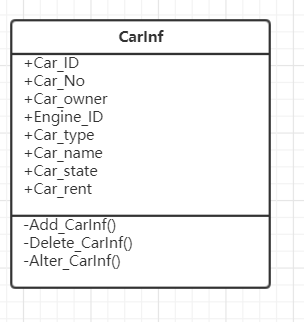
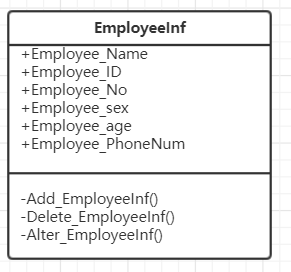
# 3.车辆租赁系统功能模块设计

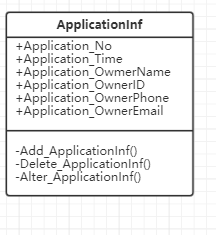
## 3.1基本信息管理模块的类图、构件图和配置图

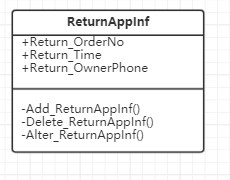
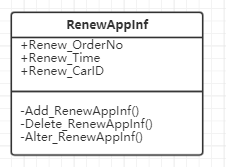
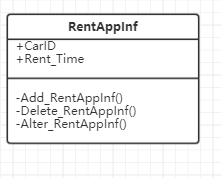
类图如下图所示：

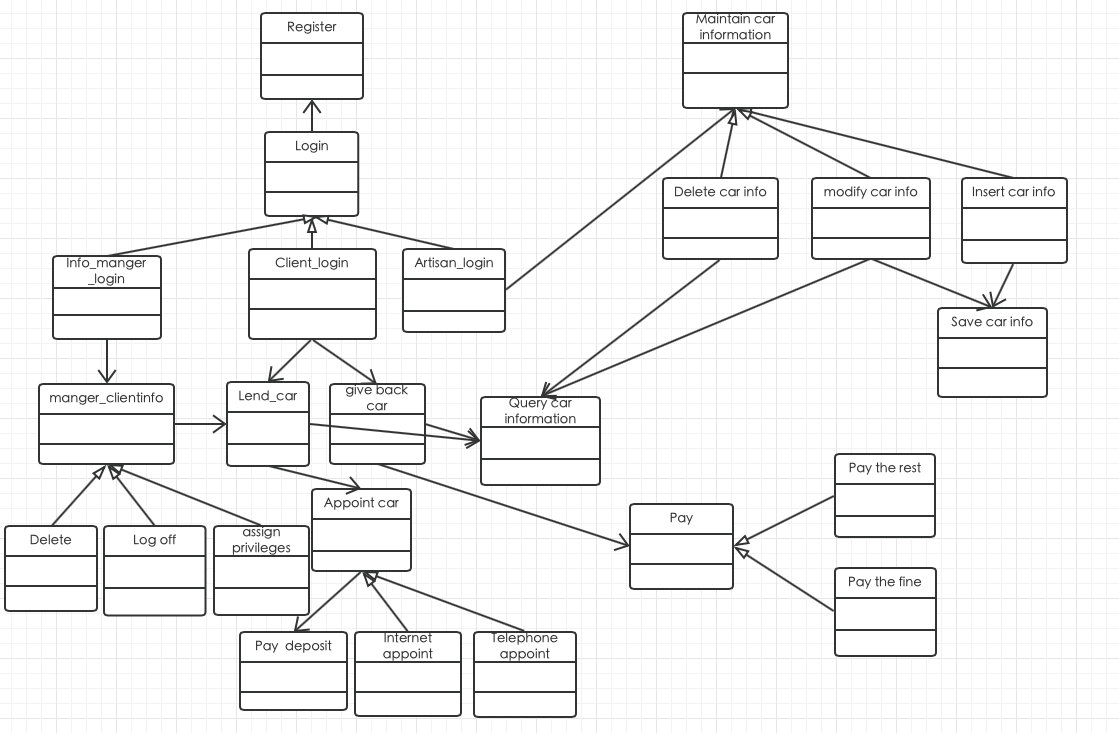


具体的类如下图所示：

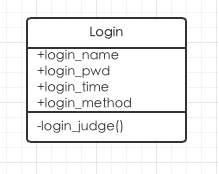
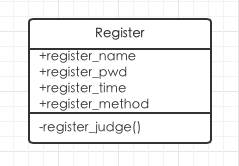


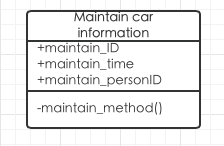


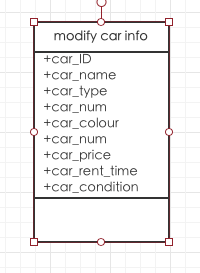
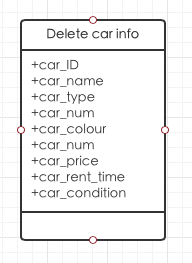


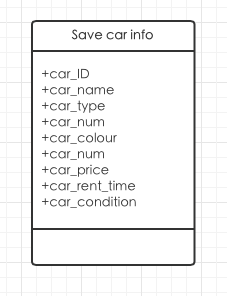
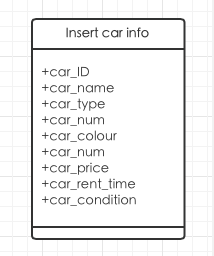


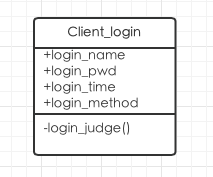
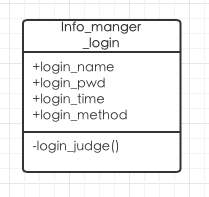
具体的类如下图所示：

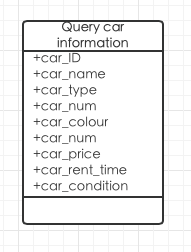
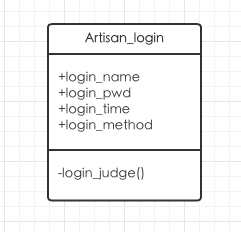


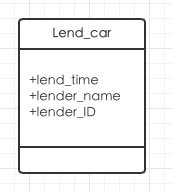
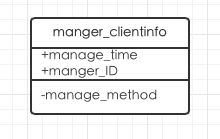


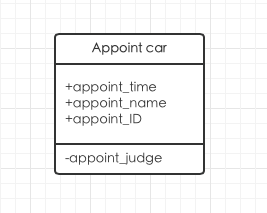
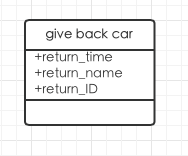


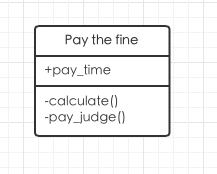
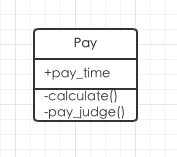


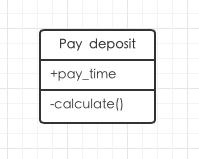
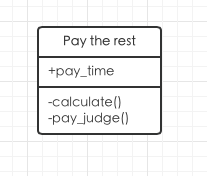




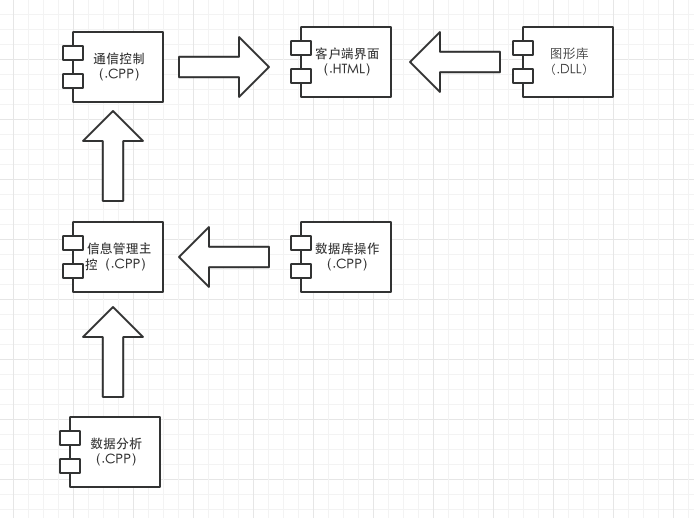




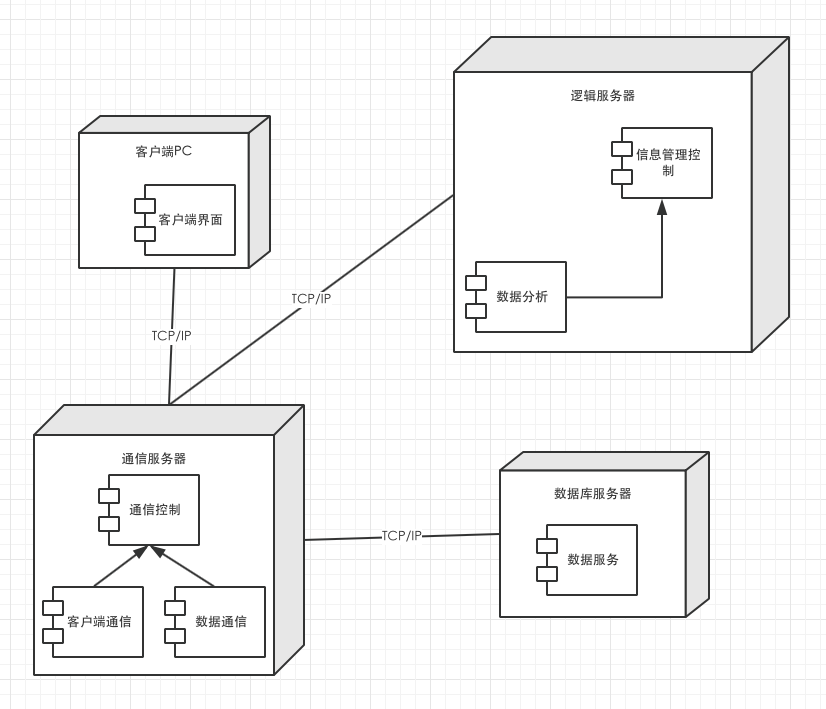




构件图如下所示：



配置图如下所示：



## 3.2管理员添加信息模块

### 3.2.1程序描述

管理员添加汽车信息模块主要完成管理员向数据库中添加新汽车的详细信息，系统先出现一个汽车信息提示界面，让管理员输入该汽车的相关信息，然后确定添加后，系统将该汽车的相关信息存入数据库中。

添加汽车的信息包含在如下文件中：

addCarInfo.htm（填写汽车信息）

addCarStore.jsp（添加汽车处理页面，主要完成向数据库插入汽车信息的功能）

### 3.2.2功能

在输入框中输入相应的汽车信息，然后单击添加汽车信息按钮，程序将插入的信息添加到数据库中，完成添加汽车信息的操作。

### 3.2.3输入项

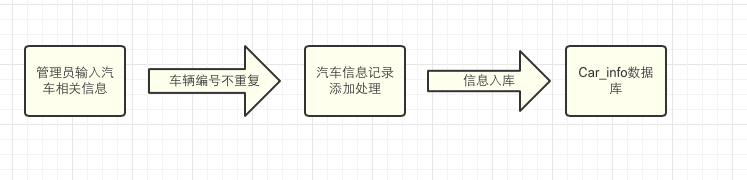
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 汽车编号 | Car\_Id | text | String | 1~20字符 |
| 车牌号 | Lic\_Id | text | String | 1~20字符 |
| 汽车品牌 | Car\_Brand | text | String | 1~20字符 |
| 入库时间 | Reg\_Date | text | Date |  |
| 汽车服务人员 | Car\_Servicer | text | String | 1~20字符 |
| 车库号 | Garage\_Id | text | String | 2字符 |
| 库存量 | Stock\_Num | text | Int | 1~65535 |
| 租赁价格 | Rent\_Price | text | Double |  |
| 备注 | Remarks | text | String | 1~200字符 |

### 3.2.4算法

先从输入的文本框中取得上述数据，然后以汽车车架号在数据库中查找，看看是否有重复的，如果有，则不能添加该汽车，因为每一本汽车的车架号是惟一的，如果数据库中不存在这一编号，则将数据用SQL语句insert into ……values……添加到数据库中。实现添加一条汽车信息记录的功能。

### 3.2.5流程图

模块流程图如下：



### 3.2.6限制条件

数据库在使用前要配置成功，且确保输入的车牌号唯一。

### 3.2.7测试计划

1、输入不完整的汽车信息观察运行结果。

2、输入完整的汽车信息且车牌号唯一，观察程序运行结果。

3、输入一个已有的车牌号编号，观察程序运行结果，看看是否出现出错信息。

### 3.2.8尚未解决的问题

暂无。

## 3.3管理员添加汽车信息模块

### 3.3.1 程序描述

管理员添加汽车信息模块主要完成管理员像数据库中添加新汽车的详细信息，系统先出现一个汽车信息输入提示界面，让管理员输入该汽车的相关信息，然后确定添加后，系统将该汽车的相关信息存入数据库。主要用于托管车辆和新车入库操作。

添加汽车信息模块包含如下文件：

addCarInfo.htm（填写汽车信息）

addCarStore.jsp（添加汽车处理页面，填写完成向数据库插入汽车信息的功能）

### 3.3.2 功能

在输入框中输入相应的汽车信息，然后单击添加汽车信息按钮，程序将输入的信息添加到数据库中，完成添加汽车信息操作。

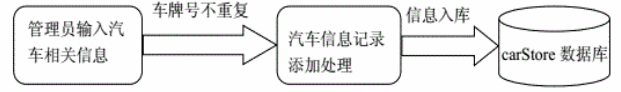
### 3.3.3 输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 车架号 | carNo | text | String | 1~20字符 |
| 车牌号 | carID | text | String | 1~20字符 |
| 汽车品牌 | carBrand | text | String | 1~20字符 |
| 行驶证号 | drivelicenseID | text | String | 1~20字符 |
| 车主身份证号 | ownerID | text | String | 1~20字符 |
| 汽车类型 | carType | text | String | 1~20字符 |
| 发动机号 | engineID | text | String | 1~20字符 |
| 入库时间 | regDate | text | date |  |
| 车库号 | garageNo | text | String | 1~20字符 |
| 租赁价格 | rentPrice | text | double |  |
| 汽车服务人员编号 | carServicer | text | String | 1~20字符 |
| 汽车状态 | carState | text | String | 1~20字符 |
| 备注 | remarks | text | String | 1~200个字符 |

### 3.3.4 算法

先从输入的文本框中取得上述数据，然后以车牌号在数据库中查找，看是否有重复的，如果有，则不能添加该汽车，因为每一辆汽车车牌号是唯一的，如果数据库中不存在这个车牌号，则将数据用SQL语句insert into...values...添加到数据库中。实现添加一条汽车信息记录的功能。

### 3.3.5 流程图



### 3.3.6 接口

该程序的接口为上述输入数据，使用request.getParameter方法从输入文本框中取得数据，然后将这些数据传递给汽车添加处理程序，完成向数据库中插入一条新汽车信息记录的功能。

### 3.3.7 限制条件

数据库在使用前要配置成功，且确保输入的车牌号唯一。

### 3.3.8 测试计划

1. 输入不完整的汽车信息观察运行结果

2. 输入完整的汽车信息且车牌号唯一，观察程序运行结果。

3. 输入一个已有的车牌号，观察程序运行结果，看看是否显示出错信息。

### 3.3.9 尚未解决的问题

暂无

## 3.4管理员处理汽车信息（修改汽车信息、删除汽车信息）模块

### 3.4.1 程序描述

管理员处理汽车信息模块主要完成对汽车信息修改、删除功能。在每条汽车信息后面有两个按钮，分别为修改和删除功能，单击修改按钮，即进入修改汽车页面，首先显示汽车原有信息，然后管理员对相应的项进行修改，修改完后，单机修改，程序收集修改后的信息，然后对原有数据库的相应项进行更新。即可完成对汽车信息的修改。如果单击删除按钮，及可将该汽车信息记录从数据库中删除。

修改汽车信息、删除汽车信息功能包含如下文件：

dealCarInfo.jsp：处理汽车信息页面，通过这个页面可以删除汽车、修改汽车信息。以及搜索汽车

deCar.jsp：根据车牌号在数据库中搜索该汽车，搜索到该汽车就将其从数据库中删除。

modifyCar.jsp：显示要修改汽车的原始信息，然后根据需要更改相应的项。

modifyCarS.jsp：修饰修改后的汽车信息

### 3.4.2 功能

实现管理员对汽车信息的删除、修改的功能。

### 3.4.3 输入

注：删除时不需要输入数据，而修改汽车信息需要对汽车相应信息进行修改，需要输入修改的数据，此时，汽车编号、车牌号、入库时间等不能被修改。

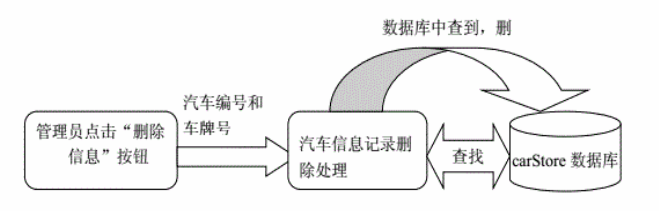
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 车库号 | garageNo | text | String | 1~20字符 |
| 租赁价格 | rentPrice | text | double |  |
| 汽车服务人员编号 | carServicer | text | String | 1~20字符 |
| 汽车状态 | carState | text | String | 1~20字符 |
| 备注 | remarks | text | String | 1~200个字符 |

### 3.4.4 算法

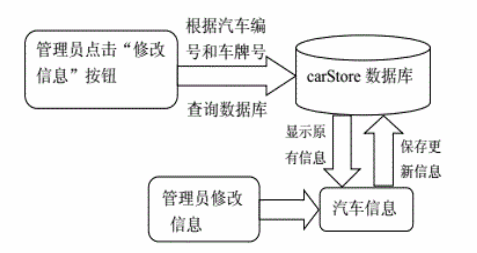
删除汽车时，通过点击汽车信息记录后面的删除按钮。系统将该汽车的编号传递到下一个页面进行处理，删除汽车处理页面先根汽车编号和车髀号在数库中进行查找，如果找到，则将该汽车从数据库中删除，页面进行自动刷新。而修改汽车信息时，也是首先将汽车编号和车牌号传递到下一个进行汽车修改操作的页面，程序先根据汽车编号在数据库查找该汽车，并把该汽车的原有信息输出，供管理员查看，管理员对汽车信息应修改的部分进行修改，然后点击确认后，程序对数据库进行更新，并显示修改成功后汽车的相应信息·。

### 3.4.5 流程逻辑

删除汽车信息：



修改汽车信息：



### 3.4.6 接口

汽车编号和车牌号：carNo、carID

方法：1. 通过URL来传递该参数给下一页面出路

2. 使用session变量保存

### 3.4.7 限制条件

1. 数据库在使用前要配置成功

2. 删除汽车时先会弹出一个确认删除框，只有点击确认后，才能对汽车进行删除操作，否则删除无法成功。

### 3.4.8 测试计划

1. 删除几条汽车记录，查看数据库中该汽车是否已经删除

2. 对汽车信息进行修改操作，修改后，查看输出修改后的结果和数据库中的显示是否一致

### 3.4.9 尚未解决的问题

暂无

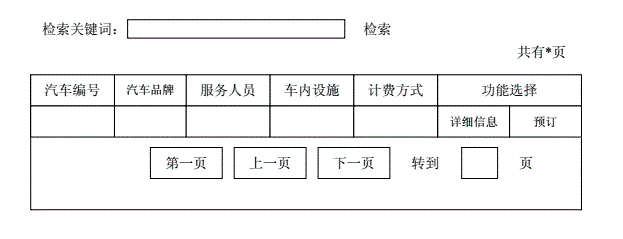
## 3.5汽车信息检索模块

### 3.5.1程序描述

主服务页面index.jsp中有汽车信息检索模块。通过检索可以查询到符合检索条件的所有汽车信息列表页sortList.jsp，并且可以链接到detailView.jsp页查看汽车详细信息。

### 3.5.2功能

按照汽车的信息检索，并可以查看汽车详细信息，界面大致如下：



### 3.5.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 关键字 | Key\_Word | text | string | 1~20字符 |
| 要转到的页码 | Page\_Id | text | Int | 1~65535 |

### 3.5.4输出项

汽车详细信息展示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 汽车编号 | Car\_Id | text | String | 1~20字符 |
| 汽车品牌 | Car\_Brand | text | String | 1~20字符 |
| 汽车服务人员 | Car\_Servicer | text | String | 1~20字符 |
| 车内设施 | Component | text | String | 1~20字符 |
| 租赁价格 | Rent\_Price | text | Double |  |
| 备注 | Remarks | text | String | 1~200字符 |

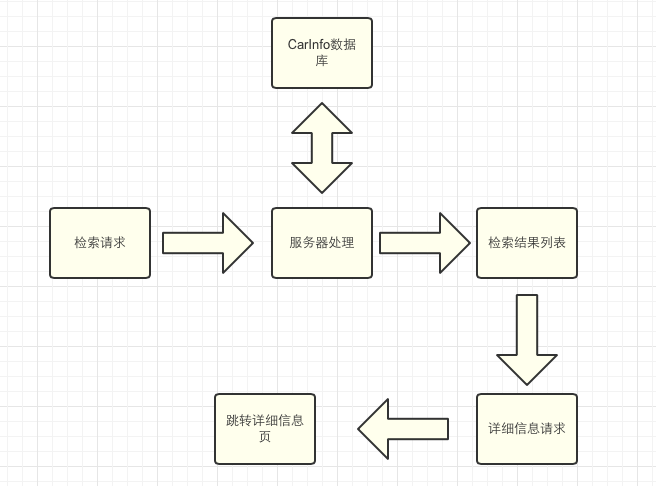
注：当查找汽车不存在的时候，给出出错提示。

### 3.5.5算法

1、利用数据库中查询语句进行查询。

2、分页算法：确定所有记录条数count，然后确定每页显示的记录条数pageSize，从而确定页数pageCount。显示第n页时，将数据库指针定位在要显示页的第一条记录，后面连续的pageSize-1条记录就是当前页应该显示的记录。

### 3.5.6流程图



### 3.5.7限制条件

数据库在使用前确保配置成功。

### 3.5.8测试计划

1、输入汽车的关键词，如类型，颜色。看是否可以查询到对应的车辆。

2、查询结果出来后，看看能否可以分页显示多条汽车信息。

### 3.5.9尚未解决的问题

暂无。

## 3.6车辆租赁管理系统服务端管理员登录模块

### 3.6.1程序描述

服务端管理员登陆模块主要完成对管理员登陆时ID和密码的验证，如果输入的ID和密码正确，则页面跳转到managerPannel.jsp，进行下一步的操作，并且用session变量保存管理员的ID，以方便修改密码验证使用，否则就不能进入服务端主界面，不能进行管理员对该系统的维护操作。

### 3.6.2功能

验证管理员输入的ID和密码是否合法

### 3.6.3输入项及格式

登录ID: （adminID）

标识：adminID

类型：字符串型

格式：字符、数字的组合（字母开头）均可

长度的有效范围：20

登录密码：

标识：password

类型：字符串型

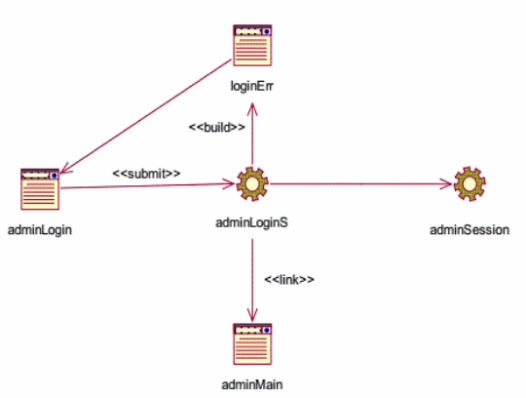
格式：字符、符号、数字的组合均可（区分大小写）

长度的有效范围：20

### 3.6.4 算法

系统先从登录界面的输入框中取得登录人员输入的ID和密码，然后将ID和密码与数据库中的ID和密码匹配，如果两者均和数据库中的相同，则验证通过，页面跳转到主界面，如果两者无法同时匹配，则验证不能通过，要求用户继续输入ID和密码进行验证。

### 3.6.5 流程逻辑



### 3.6.6 接口

ID：adminID

参数赋值：调用request.getParameter从输入文本框中得到

调用方式：使用session.setAttribute将ID存入session变量中

关联表：carRent数据库中的login表

### 3.6.7 限制条件

数据库在使用前要配置成功

### 3.6.8 测试计划

1. 输入一个错误的ID观察程序运行结果。

2. 输入一个正确的ID和密码观察程序运行的结果。

3. 不输入ID只输入密码观察程序结果。

4. 只输入ID而不输入密码观察程序运行结果

### 3.6.9 尚未解决的问题

暂无

## 3.7 管理员处理预定订单与托管申请模块

### 3.7.1 程序描述

1. 管理员处理订单模块主要完成管理员对订单的查看、取消以及个别特殊情祝的用户订单处理。首先显示所有订单信息，订单处理分为处理、查看、取消功能。首先查看租车用户以往的订单，根据其订单记录来对其进行租赁资格审查。其中处理订单功能可以由系统根据汽车租赁日程表安排来自动处理。如时间冲突，提示预订时间冲突，拒绝用户订单，系统将推荐相似度最接近的其他空闲车辆信息给用户。查看订单时，根据订单编号查找数据库，如果查到，显示该订单的相关信息，如果查不到，提示出错信息。取消订单时，先弹出一个确认窗口，以防止管理员误操作，确认取消后，将该订单标号传到取消订单处理页面，根据该订单号，查找数据库，如果查到，修改该订单的服务状态，如果查不到，提示错误信息。个别特殊情况用户订单，如系统提示用户时间冲突，但是用户坚持原有订单，那么如果其他用户取消了订单，而此时正好时间不再冲突，那么系统发送邮件提示用户是否需要继续服务，如果用户没有取消订单，则更改订单状态（“成功”），如果用户取消，则更改订单状态（“取消”）等个别特殊订单·。

2. 管理员根据当前公司状况对托管申请的订单进行处理，包括拒绝，同意。

处理订单包含如下文件：

dealOrder.jsp：订单处理主界面，可以查看、处理、取消订单

powerCheck.jsp：查看该订单用户历史订单记录，来判断其是否有租赁资格

orderLook.jsp：查看用户所有订单信息

orderDel.jsp：订单取消处理，更改用户订单状态

orderSucc.jsp：更改订单状态和相应汽车日程安排表

### 3.7.2 功能

管理员查看订单信息、取消用户订单以及根据用户预定时间进行订单预处理

### 3.7.3 输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 订单提交时间 | orderDate | Date | Date | 系统当前时间之前 |

### 3.7.4 算法

查看订单时，程序将订单编号传递给orderLook.jsp页面，该页面接受到订单编号后，首先从数据库查找，如果查到，就从数据库中取出该订单的信息项，在页面中输出供管理员查看。

管理员可以查看该订单用户历史订单记录，来判断其是否有租赁资格，从而对订单进行接受或者拒绝。

取消订单，程序首先显示是否确定取消订单，确认后，程序将订单编号传递给orderDel.jsp页面，该页面接收到订单编号后，在数据库中查找该订单，找到后就从数据库更改该订单状态。

处理订单时，首先将订单编号传递给orderPreproc.jsp页面，系统将根据汽车日程表安排是否已经安排任务来判断此次的订单是否可以接受，如果时间冲突，则订单将拒绝，如果时间空闲，则可以将订单状态更改“成功”，订单被接受。

### 3.7.5 流程逻辑

查看订单：

管理员点击“查看订单信息”按钮

提交订单编号

订单查看处理

carStore数据库

查找数据库，取出订单信息

取消订单：

管理员点击“取消订单”按钮

提交订单编号

订单取消处理

更改订单状态

carStore数据库

订单处理：

系统根据预订汽车日程表

客户提交汽车订单

查询汽车日程

订单预订汽车编号

carStore数据库

日程空闲订单成功

管理员特殊处理订单

日程安排冲突

冲突解决

### 3.7.6 接口

该功能模块所包含的程序之间的接口主要是用户ID和订单编号(recordNo)。

传参方法：

1.通过URL来传递该参数给下一页面处理

self.open('maintainCar.jsp? id=<%=ID%>& no =<%= recordNo %>');

2.使用session变量来保存

session.setAttribute(“id”，ID);

session.setAttribute(“recordNo”，no);

### 3.7.7 限制条件

数据库在使用前要配置成功。

### 3.7.8 测试计划

1.点击查看订单，观察显示出的订单信息和数据库中的是否一致。

2.点击取消订单，产看数据库中的该订单状态是否已经被修改。

3.分别针对算法中介绍的几种情况进行测试，查看数据库中订单的状态信息改变情况。

## 3.8 管理员处理续借/提前还车订单模块

### 3.8.1 程序描述

管理员对于客户提出的订单续借进行查看、处理、取消功能。如果该订单车辆在订单租借期限后已经被被的客户所租用，则拒绝客户的续借订单。如果该车辆在续借时间内未被租出，则同意续借申请。

处理续借订单包含以下文件：

dealreOrder.jsp：续借订单处理主界面，可以查看、处理、取消订单

reorderDel.jsp：订单取消处理，更改用户续借订单状态

reorderSucc.jsp：更改订单状态

reorderLook.jsp：根据要续借的车牌号在数据库中查找在续借时间内该车的租借订单

### 3.8.2 功能

管理员查看续借订单信息、取消用户续借以及根据用户续借时间对订单进行处理。

### 3.8.3 输入

|  |  |  |  |
| --- | --- | --- | --- |
| 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| carID | text | String | 1~20个字符 |
| renewDate | Date | Date |  |

### 3.8.4 算法

查看续借订单时，程序将订单编号传递给reorderLook.jsp页面，该页面接受到订单编号后，首先从数据库查找，如果查到，就从数据库中取出该订单的信息项，在页面中输出供管理员查看。

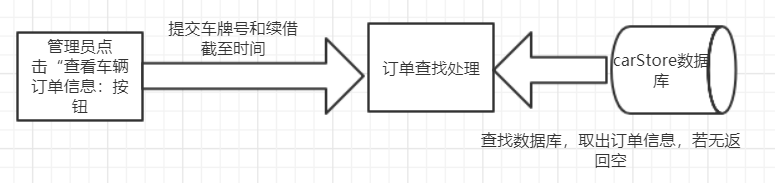
管理员可以查看在该续借订单截至时间内是否续借车辆有其他的租赁订单。

取消订单，程序首先显示是否确定取消订单，确认后，程序将订单编号传递给reorderDel.jsp页面，该页面接收到订单编号后，在数据库中查找该订单，找到后就从数据库更改该订单状态。

处理订单时，首先将订单编号传递给reorderPreproc.jsp页面，系统将根据汽车日程表安排是否已经安排任务来判断此次的订单是否可以接受，如果时间冲突，则订单将拒绝，如果时间空闲，则可以将订单状态更改“成功”，订单被接受。

### 3.8.5 流程逻辑

该模块的流程与管理员处理预定订单模块类似，只是多了一个根据车牌号查找租赁订单。



### 3.8.6 接口

该功能模块所包含的程序之间的接口主要是用户ID和续借/提前还车订单编号（renewNo）以及原租借订单编号(recordNo)。

### 3.8.7 限制条件

数据库在使用前要配置成功。

### 3.8.8 测试计划

1.点击查看订单，观察显示出的订单信息和数据库中的是否一致

2.点击取消订单，产看数据库中的该订单状态是否已经被修改。

3.根据续借订单截止时间和续借车牌号查看续借时间段内该车有无其他订单。

4.分别针对算法中介绍的几种情况进行测试，查看数据库中订单的状态信息改变情况。3.8.9 尚未解决的问题

暂无

## 3.9 管理员提醒订单将截至信息

### 3.9.1 程序描述

管理员对于租借时间即将到期的订单进行对订单客户发送平台提醒。其包含的文件如下：

remind.jsp：提醒处理主界面，向客户发送订单即将截至信息。

### 3.9.2 功能

实现管理员发送客户订单即将截至提醒。

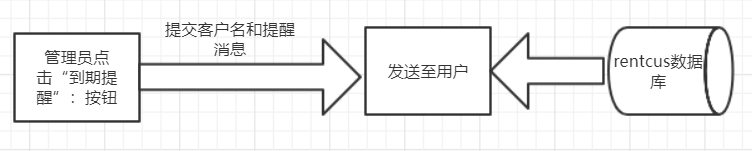
### 3.9.3 输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 客户登录名 | userName | string | string | 1~20字符 |
| 提醒消息 | remind | text | string | 1~20字符 |

### 3.9.4 算法

管理员在进入remind.jsp界面后，通过输入客户登录名，然后向该客户发送一则提醒通知。

### 3.9.5 流程逻辑



### 3.9.6 接口

客户的登录名

### 3.9.7 限制条件

数据库在使用前要配置成功

### 3.9.8 测试计划

是给用户发送一则“欢迎，收到请回复”提醒

### 3.9.9 尚未解决的问题

暂无

## 3.10 管理员结束租赁到期订单模块

### 3.10.1 程序描述

客户在租借到期后还车时，需要对该订单进行评估，包括车辆损坏程度，违章信息，同时根据优惠活动计算出该订单的租金，并且将该客户的订单记录到订单记录数据库。

结束租赁到期订单模块包含如下文件：

endorder.jsp：结束订单模块主界面，可以查看优惠活动，添加客户订单记录。

addorderrocord.jsp：添加客户订单记录页面，通过这个页面可以添加客户订单记录

historyorderLook.jsp：查看客户历史订单记录页面，通过这个页面可以查看用户的历史租借订单，便与参考优惠活动。

### 3.10.2 功能

1.查看优惠活动，计算租金

2.将客户的订单记录到数据库中

3.查看客户历史订单记录

### 3.10.3 输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 客户名 | username | text | String | 1~20字符 |
| 订单编号 | orderID | text | String | 1~20字符 |
| 订单租金 | orderMoney | text | double |  |
| 车辆损坏程度 | damagerDegree | text | String | 1~200字符 |
| 违章个数 | breakSum | text | int | >=0 |

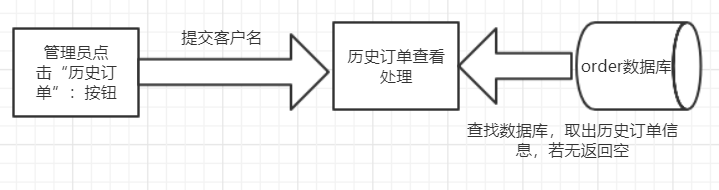
### 3.10.4 算法

查看历史订单时，程序将订单编号传递给historyorderLook.jsp界面，该界面接收到订单编号后，首先从数据库中查找，如果查到，就从数据库中取出该订单的信息项，在页面中输出供管理员查看。

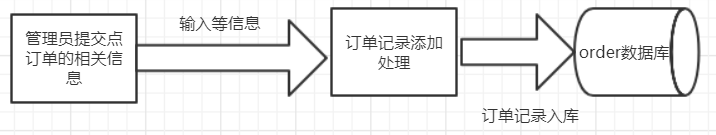
添加客户订单记录时，程序首先获取记录的输入数据，然后将其添加数据表中。

### 3.10.5 流程逻辑

查看历史订单：



添加订单记录：



### 3.10.6 限制条件

数据库在使用前要配置正确

### 3.10.7 测试计划

1.输入一个有租借历史的客户，看是否能查看其历史订单

2.添加一条租借订单记录，查看数据库中是否成功添加该记录。

### 3.10.8 尚未解决的问题

暂无

## 3.11 管理员管理优惠活动模块

### 3.11.1 程序描述

管理员添加，删除近段时间的租车优惠活动信息。其包含的文件如下：

adddiscount.jsp：管理优惠活动界面，主要发布，删除优惠活动功能。

writediscount.jsp：填写优惠活动

### 3.11.2 功能

在输入框中输入优惠活动信息后，然后单击添加又优惠活动按钮，程序将优惠活动信息添加到数据库中。

### 3.11.3 输入

输入：优惠活动内容

标识：discount

数据类型：text

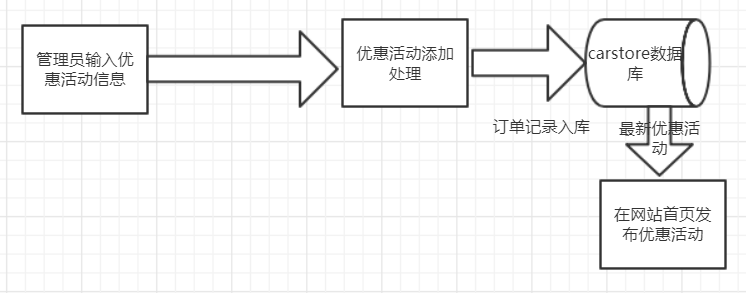
数据格式：String

数据有效范围：1~1000字符

### 3.11.4 算法

先从输入的文本框中取得上述数据，将数据用SQL语句insert into……values…添加到数据库中或delete删除优惠活动。实现添加优惠活动的功能。

### 3.11.5 流程图



### 3.11.6 接口

该程序的接口为上述输入数据，使用request.getParameter方法从输入文本框中取得数据，然后将这些数据传递给优惠活动添加处理程序，完成向数据库中插入优惠活动的功能。

### 3.11.7 限制条件

数据库在使用前要配置成功

### 3.11.8 测试条件

添加一条优惠活动，查看数据库中是否有，首页上是否显示

### 3.11.9 尚未解决的问题

暂无

## 3.12 管理员修改密码模块

### 3.12.1 程序描述

管理员修改密码模块主要完成管理员对密码的修改。其包含的文件如下：

updatePwd.jsp：对管理员的密码进行验证以及更改操作。

### 3.12.2 功能

实现管理员更该密码功能。

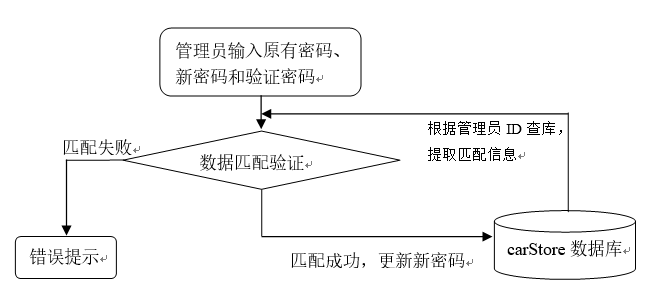
### 3.12.3 输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 原有密码 | password | text | String | 1~20字符 |
| 新密码 | newPassword | text | String | 1~20字符 |
| 验证密码 | confirmPassword | text | String | 1~20字符 |

### 3.12.4 算法

管理员根据需要可以修改登录密码，实现方法如下：首先通过session变量取得管理员登录服务端所输入正确的adminID，通过管理员的登录ID查找数据库，取出相应密码，即为原密码，然后将该密码与修改密码时需要输入的原来密码项、新密码和验证密码项作比较，如果不同，显示出错信息，如果相同，则意味着通过验证，将管理员输入的新密码写入数据库，更新数据库中相应的项。

### 3.12.5 流程逻辑



### 3.12.6 接口

该程序的接口为上述输入数据，使用request.getParameter方法从输入文本框中取得数据，然后将数据传递到密码更新处理程序，完成在数据库中对管理员密码更新功能。

### 3.12.7 限制条件

数据库在使用前要配置成功。而且在操作前数据库一定要打开，否则会报出错信息。

### 3.12.8 测试计划

1.不输入原始密码，观察程序运行结果。

2.输入原始密码，不输入新密码观察程序运行结果。

3.输入原始密码和新密码，不输入验证密码，观察程序运行结果。

4.输入原始密码、新密码和验证密码，但是新密码和验证密码输入不一致，观察程序运行结果。

5.新密码和验证密码输入一致，但是原始密码输入错误时，观察程序运行结果。

6.新密码和验证密码输入一致，原始密码输入正确，查看数据库中管理员的密码是否已经更改。

### 3.12.9 尚未解决的问题

暂无

## 3.13 管理员退出管理模块

### 3.13.1 程序描述

当管理员按下退出管理的按钮时，通过response方法将页面定向到login.jsp页面，并使所有的session变量失效。从而完成退出功能。

### 3.13.2 功能

实现管理员退出服务端的功能

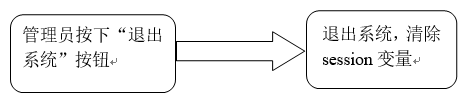
### 3.13.3 输入

暂无

### 3.13.4 算法

当管理员按下“退出系统”按钮时，通过response方法将页面定向到login.jsp页面，并使所有的session变量失效，从而完成退出功能。

### 3.13.5 流程逻辑



### 3.13.6 接口

接口为一个可以改变的转向地址，通过这个地址，可以确定管理员退出服务端时，程序页面将转向哪个页面，此时暂定为login.jsp（管理员登录界面）

### 3.13.7 限制条件

暂无

### 3.13.8 测试计划

1.点击退出按钮，观察程序是否按照预先设定的转向地址转向

2．随便输入一个页面地址，观察是否可以访问，如果能访问，说明退出服务端时session变量未清除成功，如果清除成功，不经过登录界面，不能访问服务端任何一个页面，页面会自动跳转到错误提示页面。

### 3.13.9 尚未解决的问题

暂无

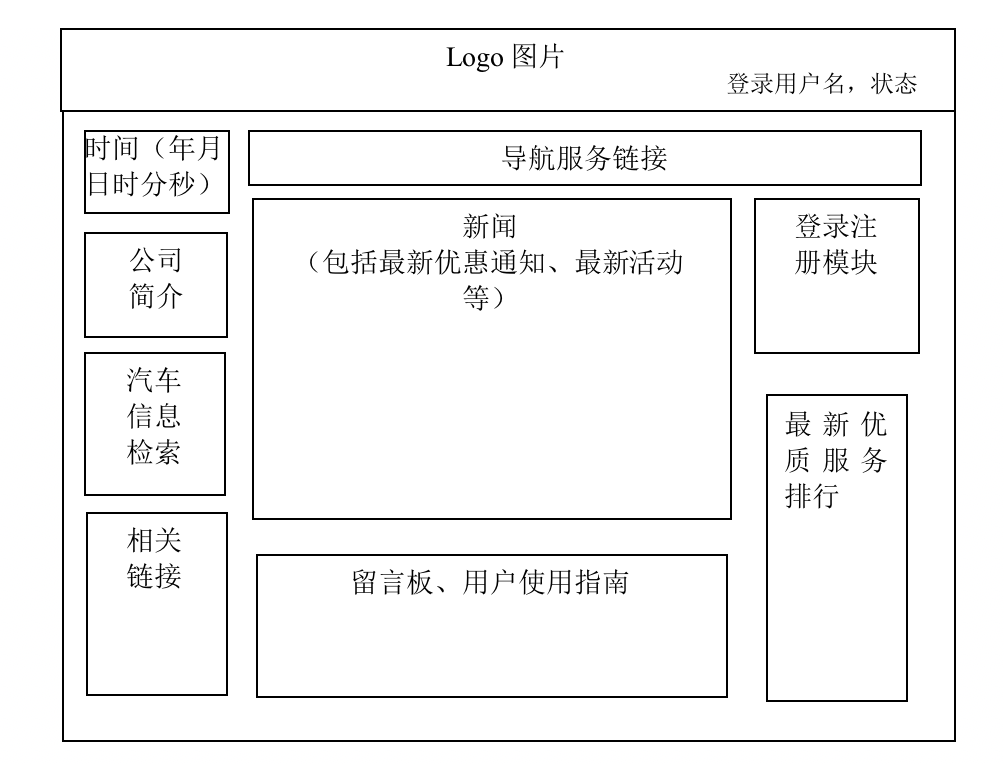
# 4. 车辆租赁系统客户服务端设计

客户服务端功能列表：

|  |  |  |
| --- | --- | --- |
| 序号 | 功能 | 执行者 |
| 1 | 注册 | 所有用户 |
| 2 | 登录 | 注册用户 |
| 3 | 注销 | 注册用户 |
| 4 | 分类浏览汽车信息 | 注册用户、游客 |
| 5 | 最新优惠活动通知 | 注册用户、游客 |
| 6 | 查看汽车详细信息 | 注册用户、游客 |
| 7 | 查询公司简介 | 注册用户、游客 |
| 8 | 在线客服 | 注册用户、游客 |
| 9 | 汽车托管预订服务 | 注册用户 |
| 10 | 托管预定状态查询服务 | 注册用户 |
| 11 | 预订汽车服务 | 注册用户 |
| 12 | 查看订单 | 注册用户 |
| 13 | 修改订单 | 注册用户 |
| 14 | 取消订单 | 注册用户 |
| 15 | 个人信息面板 | 注册用户 |
| 16 | 查看个人信息 | 注册用户 |
| 17 | 修改个人信息 | 注册用户 |

客户服务端提供给用户详细的服务，完成了整个基于WEB的汽车租赁系统的主要功能，是整个工程实现的重点和难点。基于WEB的汽车租赁系统服务端提供给用户网上预订服务的接口，用户可以注册成为会员后进行预订汽车服务订单的相关服务请求，也可以以游客身份浏览相关查询汽车信息，但此时不能预订汽车服务。

主服务模块大致界面如下：



## 4.1 **用户登录和注销模块**

### 4.1.1 功能

用户登录模块主要完成用户输入用户名和密码后验证数据库中是否有及是否正确，登录失败后提示错误信息。登录成功后链接进登录后的主服务界面。模块界面大致如下：

登录前：

用户名：

密 码：

登录

注册

登录后：

Hi ! ××× 用户

修改个人信息

最近订单

相关最新推荐

注销

### 4.1.2 性能

该模块主要是一个嵌在主服务页面index.jsp中的一个表单，由用户输入ID和密码后，点击《登录》，会现调用JavaScript函数验证是否满足基本输入要求，确定正确后将表单提交给服务端。登录成功后链接到userPannel.jsp，若用户不存在或密码不正确，将链接到登录失败页。登录后点击“注销”相当于请求未登录时的主服务页面index.jsp。链接关系类图如下：



### 4.1.3 输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| ID | logName | String | 字符、符号、数字的组合均可 | 20 |
| 密码 | password | String | 字符、符号、数字的组合均可 | 20 |

附加限制说明：

输入为空时，在客户端通过脚本程序（javascript）即可给出出错提示信息，信息不会提交到服务端。

### 4.1.4 输出项

输出变量：

无

附加限制说明：

ID输入为空或所输入的用户不存在时，给出出错提示信息，提示用户错误原因。

### 4.1.5 算法

目前不涉及加密算法。

### 4.1.6 流程逻辑

输入ID和密码并提交

服务端查询数据库并验证

正确则链接到userPannel.jsp，否则返回登录错误页面

carStore数据库

### 4.1.7 接口

登录成功后，将ID保存到session变量中，提供给其他页面做验证用。

### 4.1.8 测试计划

登录输入：

|  |  |  |
| --- | --- | --- |
| ID | 密码 | 预期结果 |
| user0（数据库中有该项） | user0 | 登录成功 |
| user0（数据库中有该项） | 123 | 登录失败，密码错误 |
| user0（数据库中有该项） |  | 提示“密码不能为空” |
| user1(数据库中没有该项) | 123 | 登录失败，报用户不存在 |

在登录后，点击 “注销”，预期结果：进入未登录的主服务页面，同时将Session变量清空。

### 4.1.9 尚未解决的问题

登录信息加密传送。

## 4.2 用户注册模块

### 4.2.1 功能

主要完成用户输入注册信息并提交注册。大致界面如下：

注册信息提示页：

注册信息输入页：

会员权利和义务说明

我同意

我不同意

用户名：

密 码：

确认密码：

性 别：

真实姓名：

联系方式：

E-mail：

提交

重写

男

女

### 4.2.2 性能

包括注册信息提示页registerInfo.htm，注册信息输入页register.jsp，主要包含一个输入用户注册信息的表单。输入完成后现调用JavaScript函数验证基本的正确性，确定正确后提交给服务端。注册成功后进入未登录的主服务端页面index.jsp。关系类图如下：



### 4.2.3 输入项

输入项：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| ID | userName | String | 字符、符号、数字的组合均可 | 20 |
| 登录密码 | password | String | 字符、符号、数字的组合均可 | 20 |
| 确认密码 | confirmPassw | String | 字符、符号、数字的组合均可 | 20 |
| 真实姓名 | tureName | String | 字符、符号、数字的组合均可 | 20 |
| 性别 | sex |  | 单选 |  |
| 联系方式 | tel | String | 数字（如需要，加区号） | 20 |
| E-mail | email | String | 按照邮箱的格式输入 | 20 |

附加限制说明：

必须要的输入的项为空，或密码与确认密码不一致时，在客户端即可判断并给出出错报告信息，信息不会提交到服务端。

### 4.2.4 输出项

输出变量：

无

附加限制说明：

必须输入的项为空或密码与确认密码不一致时，给出出错报告信息，提示用户输入错误原因。所注册ID已经存在，则详细错误提示。

### 4.2.5 算法

系统先从登录界面的输入框中取得登录人员输入的ID和密码，然后将ID和密码与数据库中的ID和密码匹配，如果两者均和数据库中的相同，则验证通过，页面跳转到主界面，如果两者无法同时匹配，则验证不能通过，要求用户继续输入ID和密码进行验证。

### 4.2.6 流程逻辑

主服务页

注册请求

注册提示信息

注册信息输入

注册成功，跳转个人信息页

详细错误提示页

carStore数据库

服务端验证

注册成功

失败

### 4.2.7 接口

数据库接口:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 变量名称 | 变量符号 | 数据库中符号 | 变量类型 | 限制说明 |
| ID | logName | logName | String | 不能为空 |
| 登录密码 | password | password | String |

### 4.2.8 测试计划

注册信息输入：

测试1：

|  |  |  |
| --- | --- | --- |
| 输入项名称 | 输入信息 | 预期结果 |
| ID | User111 | 数据库中没有此ID，注册成功. |
| 登录密码 | 123 |
| 确认密码 | 123 |
| 真实姓名 | User111 |
| 性别 | 男 |
| 联系方式 |  |
| E-mail | 111@sohu.com |

测试2：

|  |  |  |
| --- | --- | --- |
| 输入项名称 | 输入信息 | 预期结果 |
| ID | User111 | 该ID已经被用过，提示注册失败. |
| 登录密码 | 321 |
| 确认密码 | 321 |
| 真实姓名 | 123 |
| 性别 | 女 |
| 联系方式 |  |
| E-mail | 222@yahoo.com.cn |

## 4.3 汽车信息检索模块

### 4.3.1 功能

按汽车信息检索，并可以查看汽车详细信息。界面大致如下：

汽车检索模块：

汽车编号

汽车品牌

服务人员

车内设施

计费方式

功能选择

详细信息

预订

共有\*页

第一页

上一页

下一页

转到

页

检索关键词： 检索

### 4.3.2性能

主服务页面index.jsp中有汽车信息检索模块。通过检索可以查询到符合检索条件的所有汽车信息列表页sortList.jsp，并且可以链接到detailView.jsp页查看汽车详细信息。关系类图如下：



### 4.3.3 输入项

分页信息中，当有多页显示时，可以输入要转到的页数:

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制说明 |
| 要转到的页码 | ToPage | int | 通过下拉列表选择 |

### 4.3.4输出项

分类浏览汽车

输出列表信息包括：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | **20** |
| 汽车品牌 | brand | String | 字符、符号、数字的组合均可 | **20** |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | **20** |
| 车内设施 | component | String | 字符、符号、数字的组合均可 | **20** |
| 计费方式 | carPrice | double | 数值 |  |

附加限制说明：

检索汽车不存在时，给出出错提示。

### 4.3.5 算法

分页算法：确定所有记录条数count，然后确定每页显示的记录条数pageSize，从而确定共有页数pageCount。要显示第k页时，将数据库指针定位到要显示页的第一条记录，其下面连续的pageSize条记录即为当前页的记录。

### 4.3.6 流程逻辑

检索请求

服务器处理

检索结果列表

跳转详细信息页

详细信息请求

carStore数据库

### 4.3.7接口

内部接口：

sortList.jsp和index.jsp之间：

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 汽车类别 | sort | String |  |

外部接口：

sortList.jsp与数据库之间：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | **20** |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | **20** |
| 车内设施 | component | String | 字符、符号、数字的组合均可 | **20** |
| 计费方式 | carPrice | double |  |  |

### 4.3.8 测试计划

在index.jsp页面检索模块中填写检索关键字，单击“检索”，查看结果，对结果列表中的某些项单击“详细信息”，查看详细信息。

## 4.4 查看汽车详细信息模块

### 4.4.1 功能

根据传入的汽车编号查询数据库并显示查询到的汽车的详细信息，包括所有对用户可见的信息。如：汽车编号、汽车品牌、服务人员、车内设施、计费方式。

### 4.4.2 性能

对应文件detailView.jsp,主要是显示汽车的详细信息，包括所有对用户可见的信息。同时还可以链接到预订订单库将预订租赁汽车存入其中。

### 4.4.3 输入项

无

### 4.4.4 输出项

查看汽车详细信息

输出变量：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | **20** |
| 汽车品牌 | brand | String | 字符、符号、数字的组合均可 | **20** |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | **20** |
| 车内设施 | component | String | 字符、符号、数字的组合均可 | **20** |
| 计费方式 | carPrice | double | 数值 |  |

附加限制说明： 该类汽车不存在时，给出错误提示。

### 4.4.5 算法

暂无

### 4.4.6 流程逻辑

4.4.7 接口

查看详细信息请求

服务端查询数据库

显示详细信息

carStore数据库

外部请求接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 汽车编号 | carID | String | 长度 < 20 |

数据库接口：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | 20 |
| 汽车品牌 | brand | String | 字符、符号、数字的组合均可 | **20** |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | **20** |
| 车内设施 | component | String | 字符、符号、数字的组合均可 | **20** |
| 计费方式 | carPrice | double | 数值 |  |

### 4.4.8 测试计划

在不同的页面中点击查看汽车详细信息，看显示的结果。

## 4.5在线客服模块

### 4.5.1 功能

通过对话框在线服务。

### 4.5.2 性能

主要是链接聊天对话框，由汽车租赁公司服务人员直接在线进行解答疑问。

### 4.5.3 输入项

暂无

### 4.5.4 输出项

显示聊天对话框。

### 4.5.5 算法

暂无

### 4.5.6 流程逻辑

暂无

### 4.5.7 接口

暂无

### 4.5.8 测试计划

在各个页面上点击“在线帮助”，查看运行结果。

## 4.6 预订汽车信息模块

### 4.6.1 功能

具体包括将汽车租赁记录放入预订订单库中，生成订单和提交订单。

界面大致如下：

预订订单库页面：

汽车编号

服务人员

计费方式

优惠方式

预订车辆数

用户信息:

修 改

生成订单

生成订单页面：

汽车编号

服务人员

计费方式

优惠方式

预订车辆数

用户帐户信息:

提交订单

返回主页

总金额：

提交订单页面：

订单提交结果

返回首页

### 4.6.2 程序描述

汽车详细信息页面中有提交租赁订单的按钮，此功能仅对登录用户有用。orderView.jsp根据提交的汽车编号将信息加入预订订单库中，并保存在Session中，提交订单时再将Session中预订汽车信息全部提交。用户也可以修改订单的信息。

### 4.6.3 输入项

修改预订汽车信息时输入：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 租赁汽车数 | num | int | 数值 | 车场现有车辆 |

### 4.6.4 输出项

订单信息列表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | 20 |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | 20 |
| 计费方式 | calculateWay | int | 数值 | **系统提供** |
| 优惠方式 | discount | Int | 数值 | **系统提供** |
| 订购辆数 | carNum | int | 数值 | **系统提供** |

### 4.6.5 算法

生成订单是要给订单一个编号，生成这个编号的方法是构造一个字符串，它由一下两部分组成:

1.ID

2.订单提交的时间，包括年月日和时分秒。

这样，可以保证一个用户在较短时间那连续提交不同的订单不会造成数据库出错.

### 4.6.6 流程逻辑

**4.**4.6.7 接口

放入预订订单库请求

订单信息显示

生成订单显示

服务器处理订单提交

显示订单提交结果

CarStore数据库

订单修

改结果

订单修改时orderView.jsp与modify.jsp之间的接口:

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 操作类型 | type | int | 1 表示修改  2 表示清空购物车 |
| 预订订单库 | Store | Vector | 保存在Session变量中 |

订单提交时submitOrder.jsp 与submitResult.jsp之间的接口:

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 登录名 | logName | String | 由Session变量中取得 |
| 预订订单库 | Store | Vector | 保存在Session变量中 |

submitResult.jsp与数据库之间的接口:

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 订单号 | ordered | String | 由特定算法构成 |
| ID | logName | String | 由Session变量中取得 |
| 总消费金额 | totalCost | double |  |
| 日期 | orderDate | Date |  |

### 4.6.8 测试计划

预订多辆汽车，然后修改订单，改变所预订租赁汽车数包括将其改为0看是否会把该项删除.然后再提交订单，查看数据库结果。

## 4.7 托管申请模块

### 4.7.1 功能

具体包括将汽车托管记录放入托管申请订单库中，生成订单和提交订单

### 4.7.2 性能

在系统首页中有托管申请按钮，此功能仅对登录用户有用。tuoorderView.jsp根据提交的订单编号将信息加入但托管申请库中，并保存在Session中，提交订单时再将Session中托管申请中信息全部提交。

### 4.7.3 输入项

填写申请时输入：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 标识 | 数据类型 | 数据格式 | 数据有效范围 |
| 申请人编号 | userName | text | String | 1~20字符 |
| 车牌号 | carID | text | String | 1~20字符 |
| 汽车品牌 | carBrand | text | String | 1~20字符 |
| 行驶证号 | drivelicenseID | text | String | 1~20字符 |
| 车主身份证号 | ownerID | text | String | 1~20字符 |
| 汽车类型 | carType | text | String | 1~20字符 |
| 发动机号 | engineID | text | String | 1~20字符 |
| 申请时间 | regDate | text | date |  |
| 备注 | remarks | text | String | 1~200个字符 |

### 4.7.4 输出项

输出订单提交结果

### 4.7.5 算法

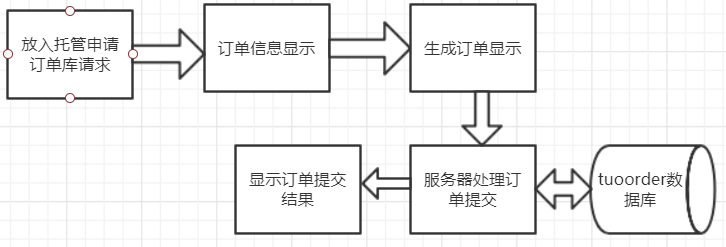
生成订单是要给订单一个编号，生成这个编号的方法是构造一个字符串，它由一下两部分组成:

1.ID

2.订单提交的时间，包括年月日和时分秒。

这样，可以保证一个用户在较短时间那连续提交不同的订单不会造成数据库出错.

### 4.7.6 流程逻辑



### 4.7.7 接口

订单提交时submittuoOrder.jsp与submitResult.jsp之间的接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 登录名 | logName | String | Session变量中取得 |
| 托管申请订单库 | tuoStore | Vector | 保存在Session变量 |

### 4.7.8 测试计划

托管多辆汽车，提交订单，查看数据库结果

## 4.8 查看租赁订单模块

### 4.8.1功能

用户查看本次所预订的汽车信息，以修改所预订汽车的数量，可以删除某种汽车的预订。最后可以生成本次预订的订单.界面大致如下:

汽车编号

服务人员

计费方式

优惠方式

预订车辆数

用户信息:

修 改

生成订单

### 4.8.2 性能

查看订单模块主要就是查看预订订单库内容页面orderView.jsp和修改订单页面modify.jsp.

### 4.8.3输入项

修改订单时输入项：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 预订车数 | carNum | int | 数值 | **系统现有车辆** |

附加限制说明：预订车数输入为0时，该汽车的预订信息将被删除。

### 4.8.4 输出项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | 20 |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | 20 |
| 计费方式 | calculateWay | int | 数值 | **系统提供** |
| 优惠方式 | discount | Int | 数值 | **系统提供** |
| 订购辆数 | carNum | int | 数值 | **系统提供** |

### 4.8.5 算法

暂无

### 4.8.6 流程逻辑

4.8.7 接口

查看订单请求或提交预订汽车订单

从Session中取得预订汽车信息

显示预订汽车信息列表

用户修改预订汽车数并单击修改按钮

链接到modify.jsp

用户单击生成订单按钮

链接到

orderSubmit.jsp

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 预订订单库 | Store | Vector |  |
| 登录名 | logName | String |  |
| 汽车编号 | carID | String | 查看订单时carID为－1 |

### 4.8.8 测试计划

在服务主模块中点击“查看订单”，预期结果为显示用户预订的汽车信息。修改预订汽车数后，点击“修改”，看是否链接到modify.jsp，点击“生成订单”，看是否链接到orderSubmit.jsp中。

## 4.9 个人信息面板模块

### 4.9.1 功能

用户可以查看个人信息，修改个人信息，查看个人定汽车情况。

### 4.9.2 性能

主要用于用户查看个人信息，修改个人信息。包括modUser.jsp ，nowOrder.jsp，userView.jsp和head.jsp三个页面。modUser.jsp用于修改个人信息；nowOrder.jsp用于显示用户订单；head.jsp用于显示页面框架并判断用户是否登录而userView.jsp用于显示用户信息。

### 4.9.3 输入项

修改个人信息时的输入项：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 登录密码 | password | String | 字符、符号、数字的组合均可 | 20 |
| 确认密码 | confirmPassw | String | 字符、符号、数字的组合均可 | 20 |
| 真实姓名 | tureName | String | 字符、符号、数字的组合均可 | 20 |
| 性别 | sex |  | 单选 |  |
| 联系方式 | tel | String | 数字（如需要，加区号） | 20 |
| E-mail | email | String | 按照邮箱的格式输入 | 20 |

附加限制说明：必要的输入项为空，或密码与确认密码不一致时，在客户端即可判断并给出出错报告信息，信息不会提交到服务端。

### 4.9.4输出项

查看个人信息：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| ID | userName | String | 字符、符号、数字的组合均可 | 20 |
| 登录密码 | password | String | 字符、符号、数字的组合均可 | 20 |
| 确认密码 | confirmPassw | String | 字符、符号、数字的组合均可 | 20 |
| 真实姓名 | tureName | String | 字符、符号、数字的组合均可 | 20 |
| 性别 | sex |  | 单选 |  |
| 联系方式 | tel | String | 数字（如需要，加区号） | 20 |
| E-mail | email | String | 按照邮箱的格式输入 | 20 |

查看预订汽车情况：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入项名称 | 标识 | 数据类型 | 数据格式 | 数值有效范围 |
| 汽车编号 | carNo | String | 字符、符号、数字的组合均可 | 20 |
| 服务人员 | ServicerID | String | 字符、符号、数字的组合均可 | 20 |
| 计费方式 | calculateWay | int | 数值 | **系统提供** |
| 优惠方式 | discount | Int | 数值 | **系统提供** |
| 订购辆数 | carNum | int | 数值 | **系统提供** |

### 4.9.5算法

暂无

### 4.9.6 流程逻辑

用户个人信息面板

用户查看个人信息

用户修改个人信息

用户查看预订信息

显示个人信息

等待输入个人信息

显示预订信息

### 4.9.7 接口

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量符号 | 变量类型 | 限制条件 |
| 预订订单库 | Store | Vector |  |
| 登录名 | logName | String |  |

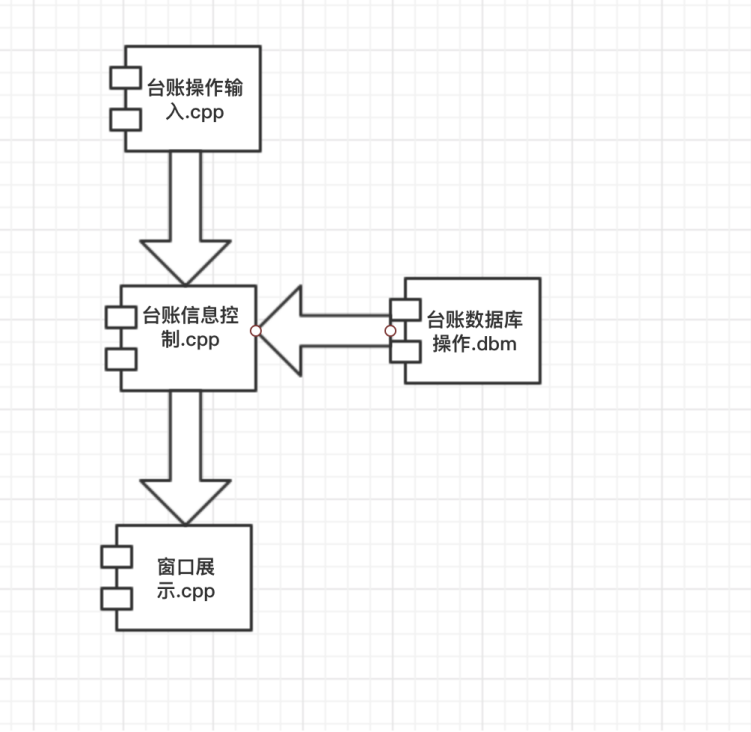
### 4.9.8 测试计划

进入用户面板，点击“查看个人信息”，“修改个人信息”和“查看预订信息”按钮然后再观察。在修改个人信息页中输入不同的值检查不同的输入的情况。

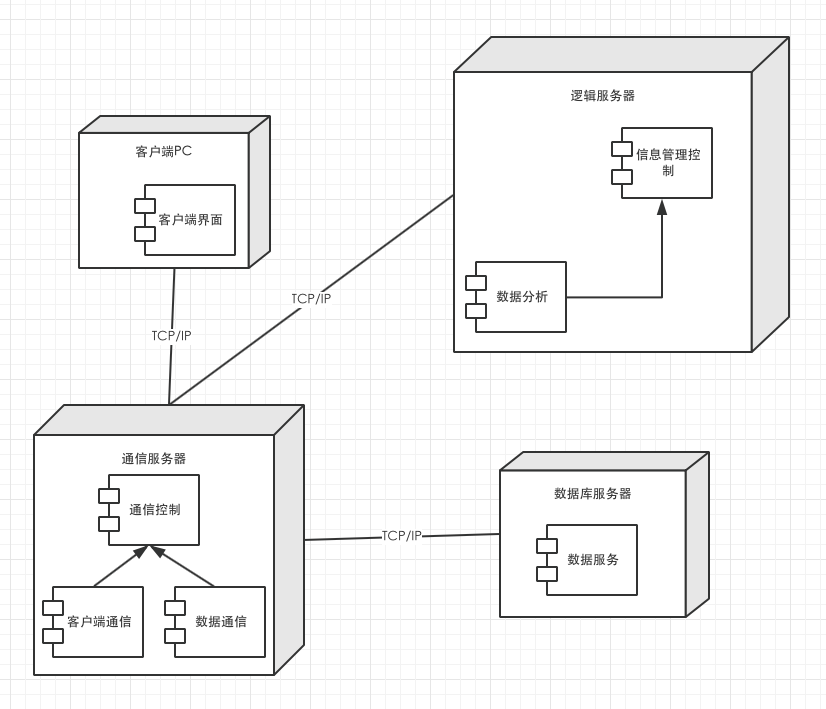
# 5.进销台账模块

## 5.1类图、构件图和配置图

构件图如下所示：



配置图如下所示：



## 5.2功能描述

输出车辆台账明细或者是显示公司台账报表，台账进销模块主要进行每一台车辆收支细节的存储统计并显示，同时可以实现报表功能。程序接受车辆id的输入和操作输入，如果验证正确则跳转到DisplayWindow.cpp进行信息显示，所有车辆台账信息存储在cartrans.mdb中。否则显示错误信息。

## 5.3性能

支持在需求设计时峰值时连续工作且具有鲁棒性，对于数据错误可以在规定时间生成错误信息并处理

## 5.4输入项

车辆ID: (Car\_id)

标识: Car\_id

类型: 字符串类型

格式: 字符、符号、数字的按照车牌规定的组合

长度的有效范围: 11

台账操作: (Transsum\_type)

标识: Transsum\_type

类型: int

格式: 数字

时间段: (Timeperiod)

标识: Timeperiod

类型: time

格式: yy-mm-dd-hh-mm-ss

长度: 固定时间格式长度

## 5.5输出项

台账报表: (report\_from)

标识: (report\_form)

类型: struct 类型

格式: string 车牌号，time 时间，type交易类型，money交易金额 ，string 报表

长度: 变长

操作错误: (errcode)

标识: errcode

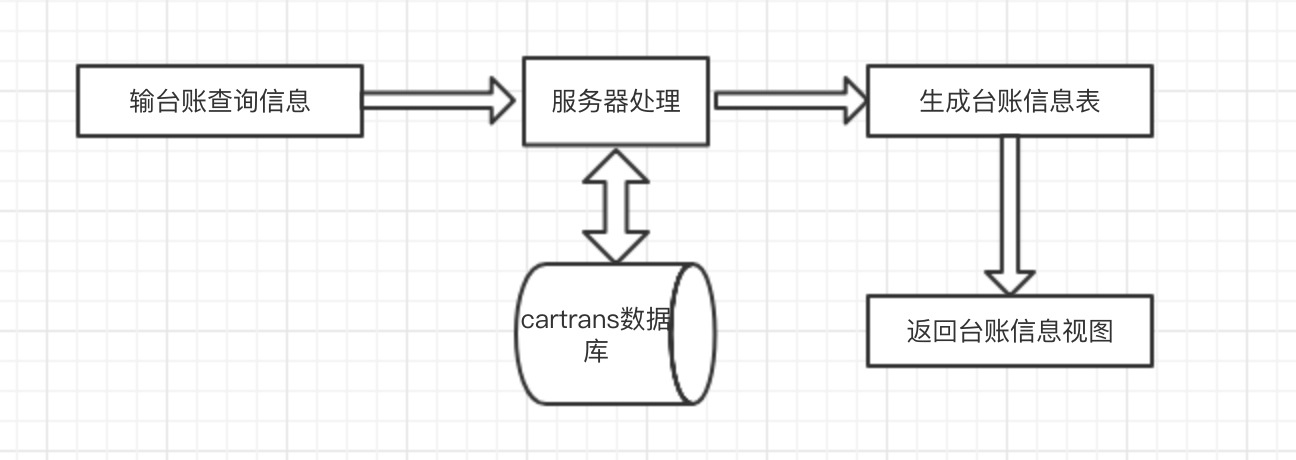
类型: int

格式: 数字

## 5.6算法

系统通过输入的参数个数识别操作类型，如果传入的参数只有一个且是车辆信息，那么可以判定为单个车辆台账操作，如果传入的参数是两个: 1 如果是车辆编号和时间段，那么判定为单个车辆台账汇总。2 如果是时间段和汇总操作，那么判定为公司台账报表汇总。此时转入DisplayWindow.cpp中进行相应的操作，首先进入cartrans.mdb通过获取数据，生成相应的数据库视图，转化成软件页面输出，输出单个车辆的台账汇总或者是台账明细或者是公司一个时间段之内的台账报表。

## 5.7流程图



## 5.8限制条件

数据库在使用前要配置成功,要保证数据库中的车牌号是唯一的

## 5.9测试条件

1.输入正确车牌号，输出车牌对应的台账明细

2.输入错误的车牌号，输出错误信息: 车牌号错误

3.输入正确车牌号，和时间段，输出一段时间内的车辆台账

4.输入时间段，汇总操作，输出一段时间内的公司台账汇总报表

5.输入错误时间段，汇总操作，输出错误信息:时间错误

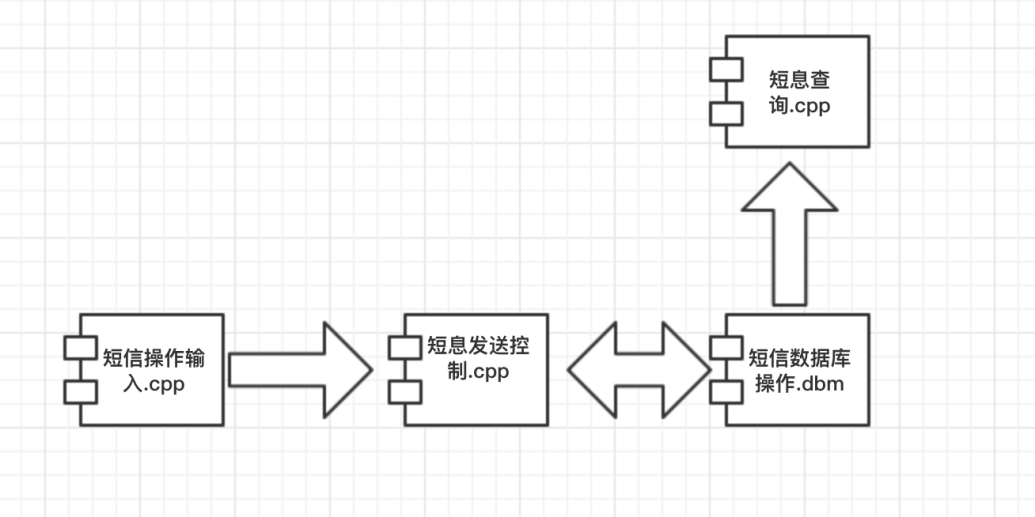
## 5.10尚未解决的问题

需要总的公司平台提供权限接口，使得不同操作人员的操作行为对应于其匹配的权限级别进行操作，这样不出现越权操作现象。

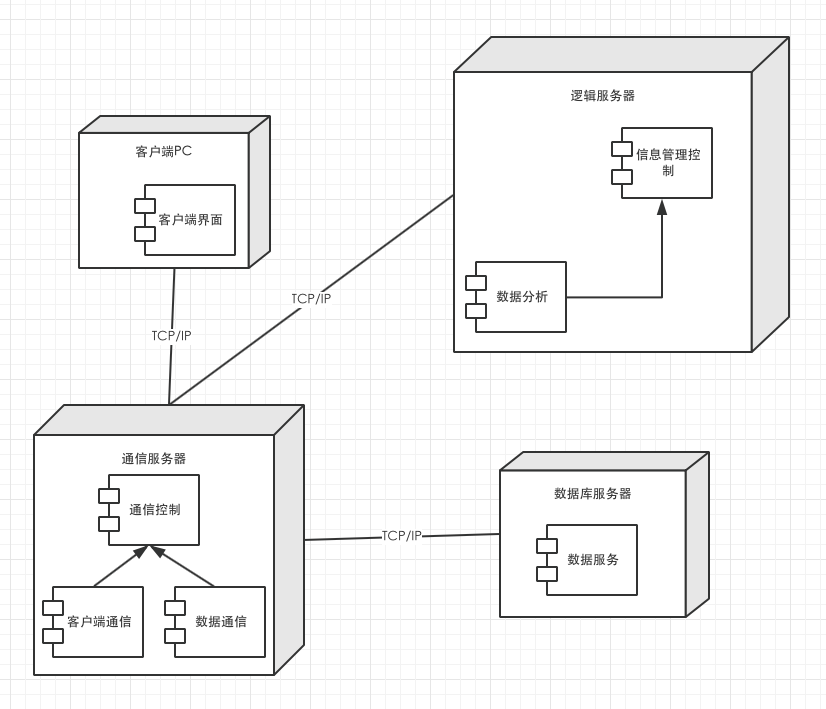
# 6.短信管理模块

## 6.1类图、构件图和配置图

构件图如下所示：



配置图如下所示：



## 6.2功能描述

在管理端页面index.jsp中可以进行短信业务，输入客户端id以及短信内容msg\_context。输入的内容可以是业务功能提醒和违章信息警示。

## 6.3性能

峰值可以在10秒钟对于所有的在线车辆进行发送短信操作，并在网络允许并稳定的情况下，可以在使用拥塞队列完成1分钟之内完成信息到达车载终端。

## 6.4输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 汽车编号 | Car\_Id | text | String | 1~20字符 |
| 短信内容 | Msg\_context | text | String | 1~200字符 |

## 6.5输出信息

提示信息: (rt\_info)

标识: (rt\_info)

类型: string

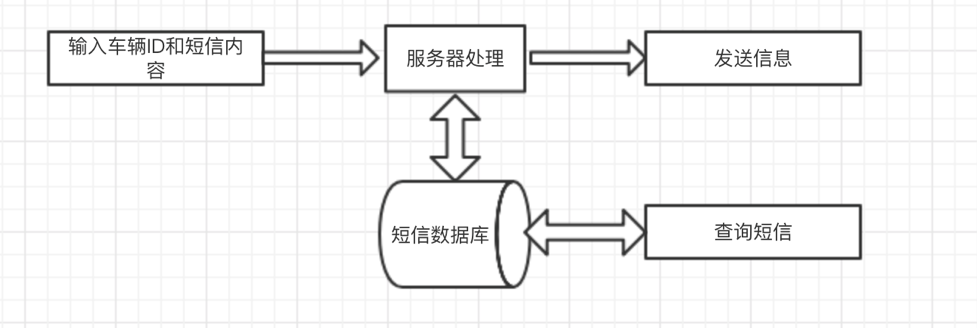
格式: text

长度: 1-10字符

## 6.6算法

短信系统模块接受输入车牌号Car\_id, 短信内容Msg\_context, 向网路传输模块API提供参数，网络传输模块将参数达成数据包并放入发送队列发送。

## 6.7流程图



## 6.8限制条件

数据库在使用前要配置成功,要保证数据库中的车牌号是唯一的

## 6.9测试条件

1.输入正确车牌号，短信内容: 发送成功

2.输入错误的车牌号，输出错误信息: 车牌号错误

3.输入正确车牌号，不输入短信内容，输出错误提示:内容不可为空

## 6.10尚未解决的问题

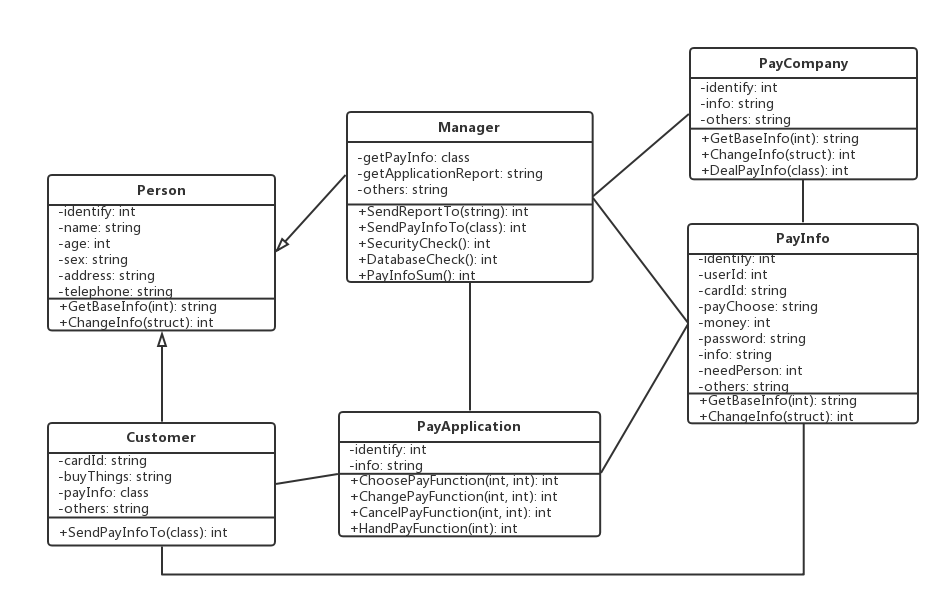
需要总的公司平台提供权限接口，使得不同操作人员的操作行为对应于其匹配的权限级别进行操作，这样不出现越权操作现象。

车载终端最好可以提供反馈信息以显示发送信息时候接受到。

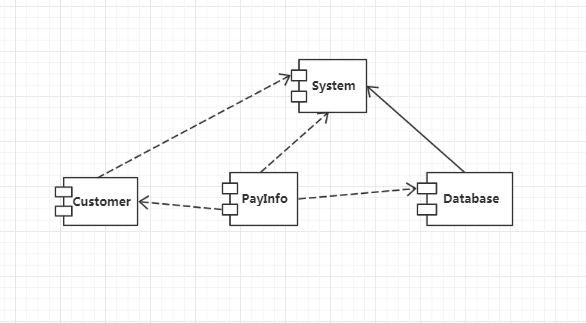
# 7.平台支付功能模块

## 7.1类图、构件图和配置图

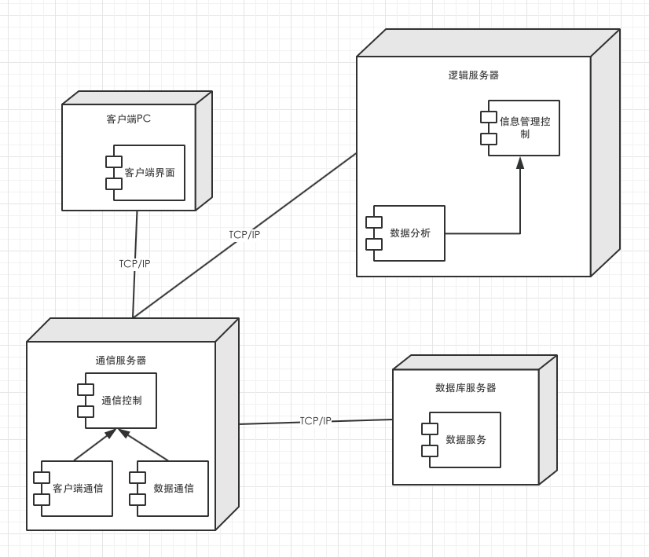
类图如下图所示：



构件图如下所示：



配置图如下所示：



## 7.2支付模块

### 7.2.1程序描述

支付模块用于用户完成支付功能，当用户选择好支付平台和方式后，系统会提交相应用户填写的表格，并返回第三方平台的支付界面，同时在数据库中保存用户此次操作。

提交支付的信息包含在如下文件中：

PayInfo.htm（填写支付信息）

PayReturn.htm（返回的支付界面信息）

CustomerPay.jsp（添加支付处理页面）

### 7.2.2功能

在页面选择或输入相应支付选项，确认信息无误后，点击确定按钮，提交相应表单到后台数据库。

### 7.2.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 支付编号 | Pay\_Id | text | Int | 1~20字符 |
| 金额信息 | Money\_Info | text | String | 1~20字符 |
| 用户id | Customer\_Id | text | Int | 1~20字符 |
| 用途信息 | Payfor\_Info | text | String |  |
| 选择支付方式 | Pay\_Way | text | String | 1~20字符 |

### 7.2.4算法

用户提交页面后，系统根据用户id和支付方式，选择相应平台支付界面信息返回用户。

### 7.2.5限制条件

数据库在使用前要配置成功，且确保输入的支付号、用户id唯一。

### 7.2.6测试计划

输入有误的支付界面信息，如缺少用户id、平台选择等。

### 7.2.7尚未解决的问题

暂无。

## 7.3后台管理（查看用户支付信息、支付安全检测）模块

### 7.3.1程序描述

后台管理模块主要是管理员用来管理用户支付信息，防止用户填写的支付订单有误，同时检测支付渠道是否安全。

包含以下文件：

CheckCustomer.jsp:处理用户支付订单信息。

SecurityCheck.jsp:检测安全信息

### 7.3.2功能

实现管理员对支付安全的管理i。

### 7.3.3输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 管理员标识 | Manager\_id | text | Int | 1~20字符 |
| 支付标识 | Pay\_id | text | Int | 2字符 |

### 7.3.4算法

检测用户支付订单时，程序主要是检测格式是否有误，是否用户进行了非法的操作。

### 7.3.5限制条件

用户要有订单完成

### 7.3.6测试计划

根据系统漏洞进行测试

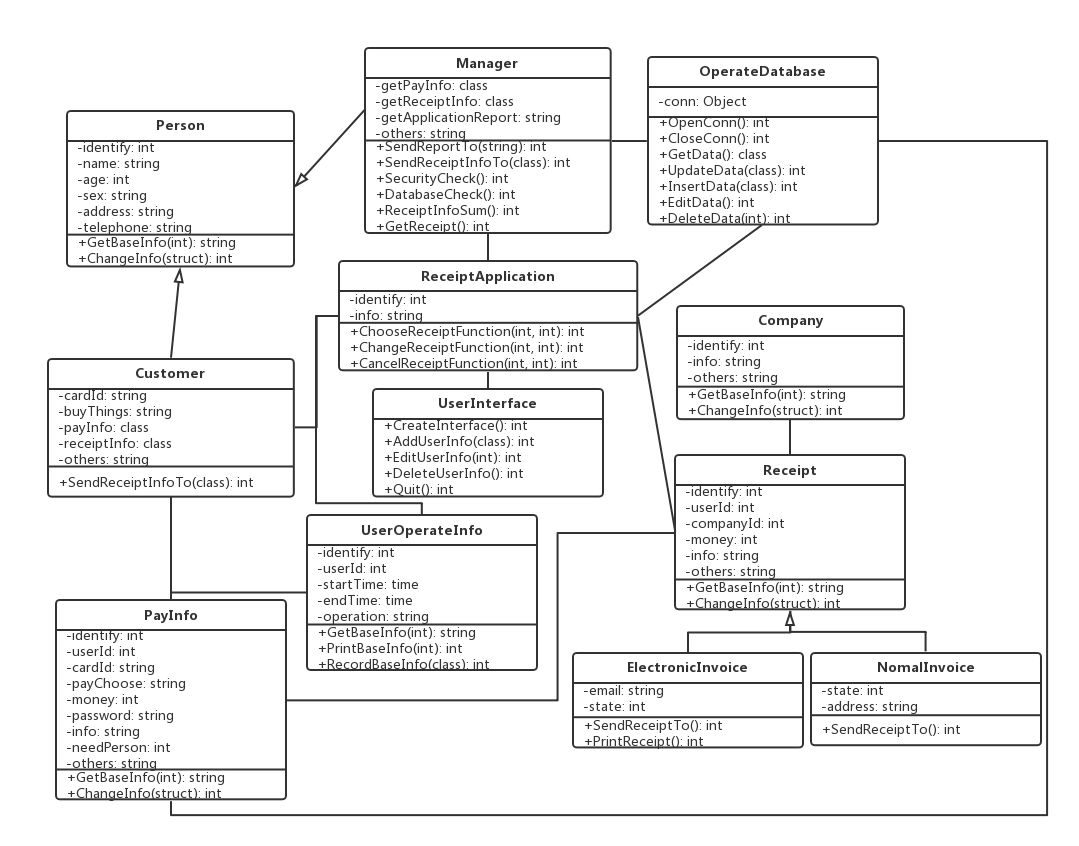
### 7.3.7尚未解决的问题

暂无。

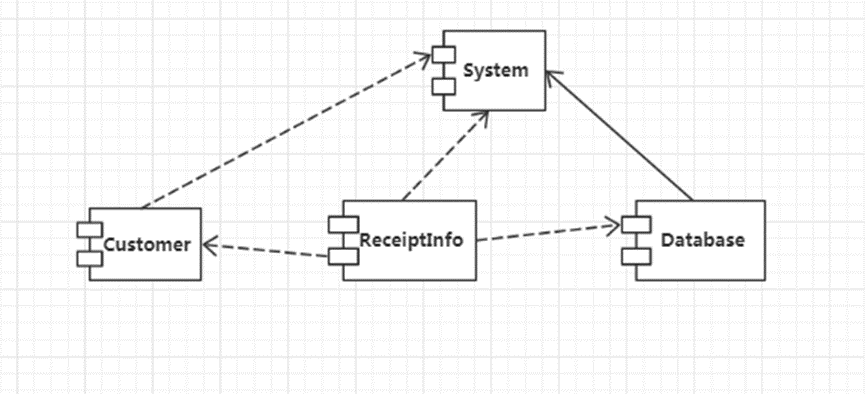
# 8.发票管理功能

## 8.1类图、构件图和配置图

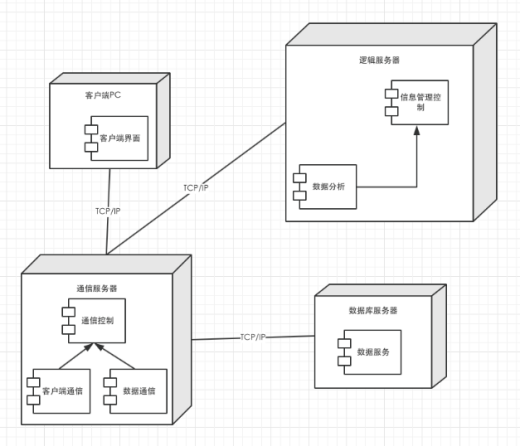
类图如下图所示：



构件图如下所示：



配置图如下所示：



## 8.2发票开具模块

### 8.2.1程序描述

此模块主要是用来开具发票，ReceiptGet.jsp 是用来根据公司的发票存量来开具相应的发票。

### 8.2.2功能

实现电子发票的开具。

### 8.2.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 关键字 | Key\_Word | text | string | 1~20字符 |
| 开具的发票号 | Receipt\_Id | text | Int | 1~100 |

### 8.2.4算法

利用数据库中查询语句进行查询发票信息，并填写开具。

### 8.2.5限制条件

1.数据库在使用前确保配置成功。

2.公司已经购进了足够的发票

### 8.2.6测试计划

让测试人员点击发票开具，看是否能开具完成。

### 8.2.7尚未解决的问题

暂无。

## 8.3发票查询模块

### 8.3.1程序描述

此模块主要是用来查询发票，ReceiptCheck.jsp 是用来根据公司的发票存量来开具相应的发票。

### 8.3.2功能

实现电子发票的查询。

### 8.3.3输入项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | 英文标识 | 数据格式 | 数据类型 | 数据有效范围 |
| 关键字 | Key\_Word | text | string | 1~20字符 |
| 开具的发票号 | Receipt\_Id | text | Int | 1~100 |

8.3.4算法

利用数据库中查询语句进行查询发票信息。

### 8.3.5限制条件

1.数据库在使用前确保配置成功。

2.公司已经购进了足够的发票。

### 8.3.6测试计划

让测试人员点击发票开具，看是否能开具完成。

### 8.3.7尚未解决的问题

暂无。