

物流管理系统

设计报告

学院：软件学院

专业：软件工程

姓名：张启洋

学号：20301090

创建时间：2023/6/20

**目录**

[1 概要设计 3](#_Toc14601)

[1.1 功能设计 3](#_Toc18879)

[1.2 系统架构 4](#_Toc28512)

[2 接口设计 5](#_Toc12516)

[2.1 基础配置 5](#_Toc26639)

[（1）注册账户 5](#_Toc22168)

[（2）登录账号 6](#_Toc31061)

[（3）登出 7](#_Toc1732)

[（4）查询员工 8](#_Toc31604)

[（4）查询用户 9](#_Toc3232)

[（5）修改用户信息 11](#_Toc30933)

[（6）修改员工信息 12](#_Toc19640)

[（7）批量删除用户 14](#_Toc24699)

[（8）增加普通用户 15](#_Toc11296)

[（9）增加员工用户 16](#_Toc21720)

[（10）查找车辆列表 17](#_Toc5735)

[（11）新增车辆 19](#_Toc18715)

[（12）修改车辆 20](#_Toc22005)

[（13）删除车辆 21](#_Toc26585)

[2.2 订单管理 22](#_Toc10180)

[（1）创建订单 22](#_Toc22695)

[（2）计算价格，为订单分配车辆和驾驶人 24](#_Toc14442)

[（3）改变订单状态 25](#_Toc3927)

[（4）添加订单中转站 27](#_Toc68)

[（5）查找快递员负责的订单 28](#_Toc28323)

[（6）根据ID查找订单 30](#_Toc21412)

[（7）获取订单列表 32](#_Toc31925)

[（8）查询订单运输中转信息 33](#_Toc302)

[3 页面设计 35](#_Toc25491)

[4 单元测试 44](#_Toc28421)

[4.1 创建订单 44](#_Toc18884)

[4.2 更新订单状态 45](#_Toc9542)

[4.3 添加转运信息 46](#_Toc7054)

# 概要设计

# 功能设计

本系统是一个简易的物流系统，系统设置了三个角色——用户、承运人员和管理员，根据角色对页面访问权限进行控制。系统实现了用户管理、车辆管理、订单管理、员工管理。

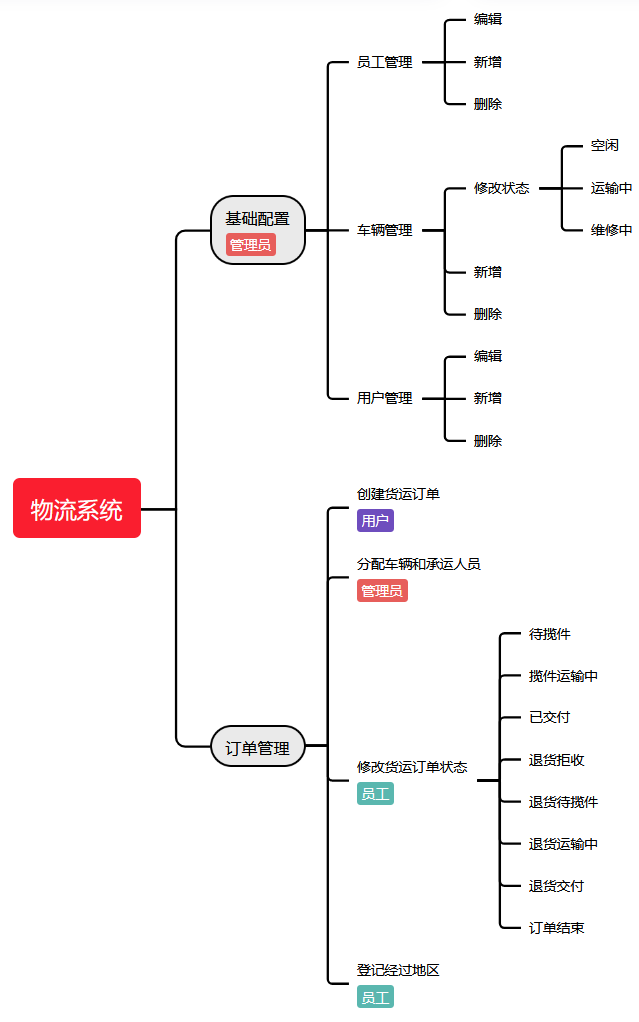
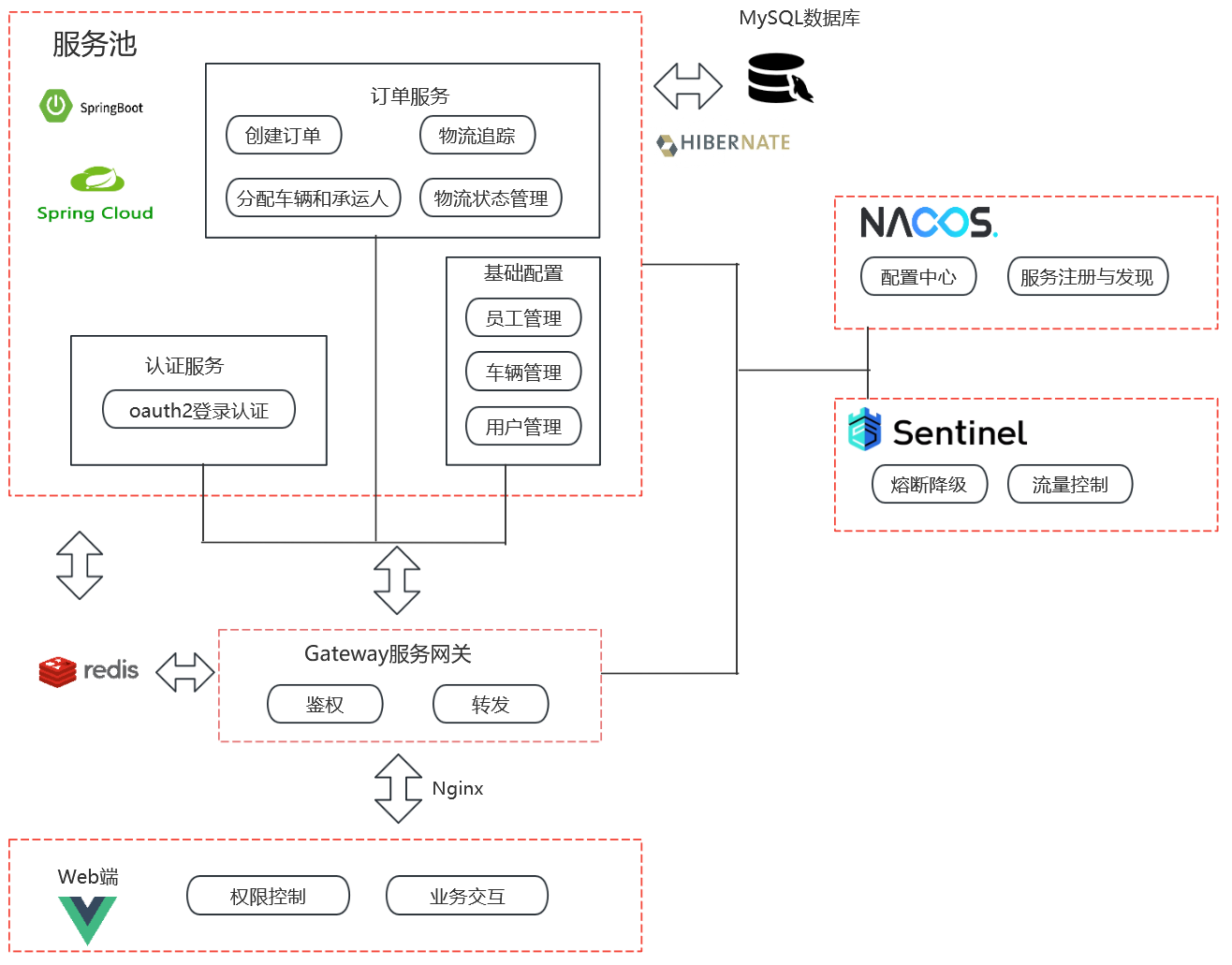


图 1 功能设计图

# 系统架构

本系统使用springboot+mybatis-plus+hibernate+vue进行Web应用程序开发，使用Restful API和Ajax设计开发系统控制层。在此基础上进行微服务拆分，使用nacos注册中心实现服务的注册与发现，并通过nacos-config实现配置中心；使用ouath2实现单点登录与授权；通过gateway网关对外部客户端进行请求转发与权限校验。系统微服务包括服务网关，认证服务器和外部服务、库管服务两个资源服务，架构图如下所示。



该项目选择IDEA作为开发工具，数据库使用MySQL进行搭建，并利用Redis进行访问权限资源缓存。

# 接口设计

# 基础配置

（1）注册账户

创建个人账户，可以选择注册成为哪种角色。

|  |  |
| --- | --- |
| URL | http://localhost:8080/auth/signup |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必需 | 缺省值 |
| username | 用户名 | 无 | 是 | 无 |
| phone | 电话号码 | 11位数字串 | 是 | 无 |
| password | 密码 | 至少为六位字符串 | 是 | 无 |
| role | 角色 | 数组类型，可选值0或1。  1：员工  0：普通用户 | 是 | 无 |
| realName | 真实姓名 | 中文字符 | 否，员工必需 | 无 |
| sex | 性别 | 数字，可选值1或0。  1：女  0：男 | 否，员工必需 | 无 |
| entryTime | 入职时间 | 格式：“YYYY-MM-DD” | 否，员工必需 | 无 |
| birthday | 生日 | 格式：“YYYY-MM-DD” | 否，员工必需 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | roles | | Array | | 返回用户的角色信息 | |
|  |  | role | | String | | 用户角色名 | |
|  | name | | | String | | 用户名 | |
|  | Phone | | | String | | 电话号码 | |

截图：



（2）登录账号

登录个人账号。

登录个人账号API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/auth/login |
| 请求方式 | POST |

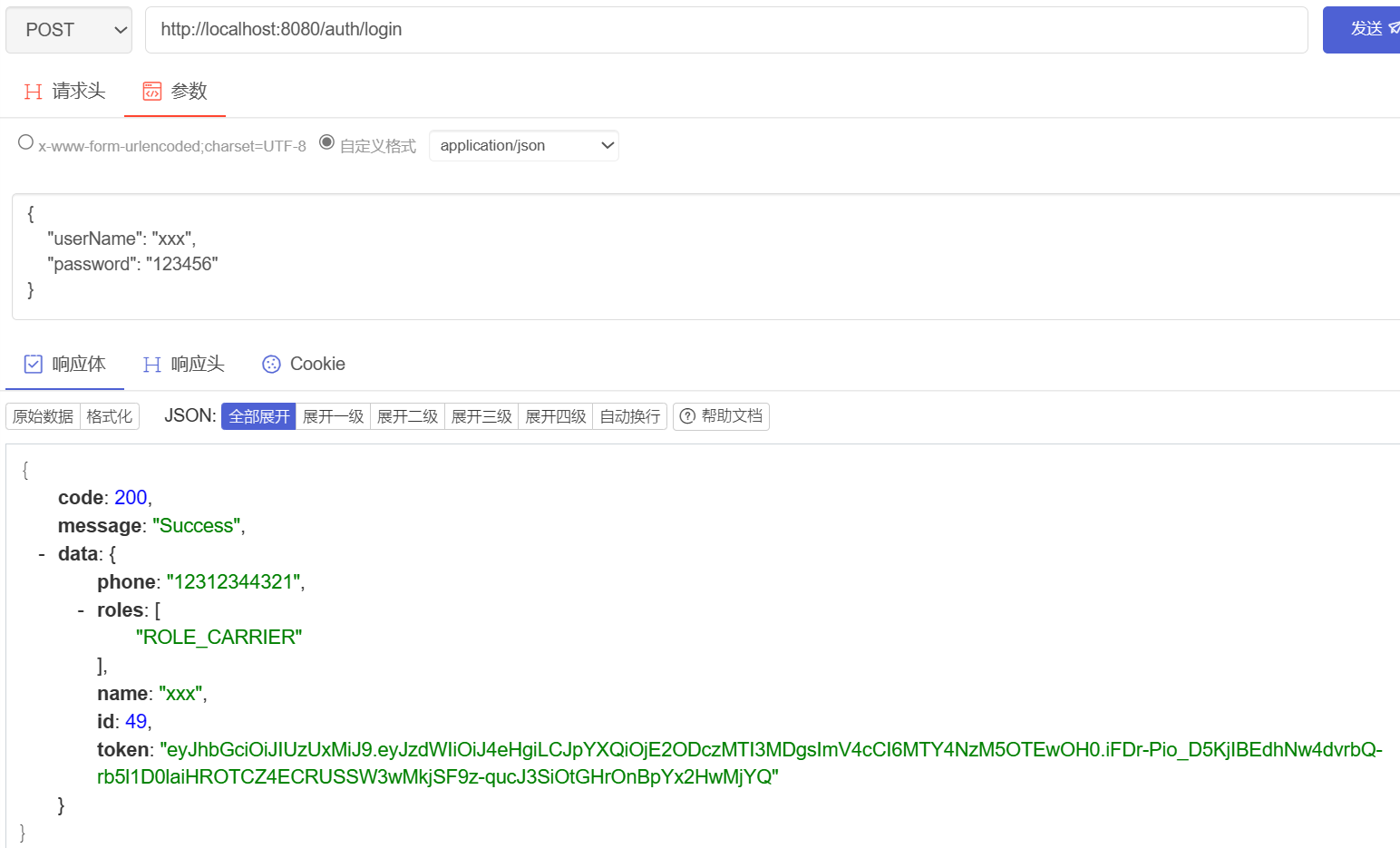
请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| username | 用户名 | 无 | 是 | 无 |
| password | 密码 | 至少为六位字符串 | 是 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | roles | | Array | | 返回用户的角色信息 | |
|  |  | role | | String | | 用户角色名 | |
|  | name | | | String | | 用户名 | |
|  | phone | | | String | | 电话号码 | |
|  | id | | | String | | 用户ID | |
|  | token | | | String | | token编码 | |

截图：



（3）登出

登出个人账号。

登出个人账号API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/auth/logout |
| 请求方式 | POST |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |

截图：



（4）查询员工

根据用户名和真实姓名模糊查询普通用户。

登出个人账号API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/carrierList |
| 请求方式 | GET |

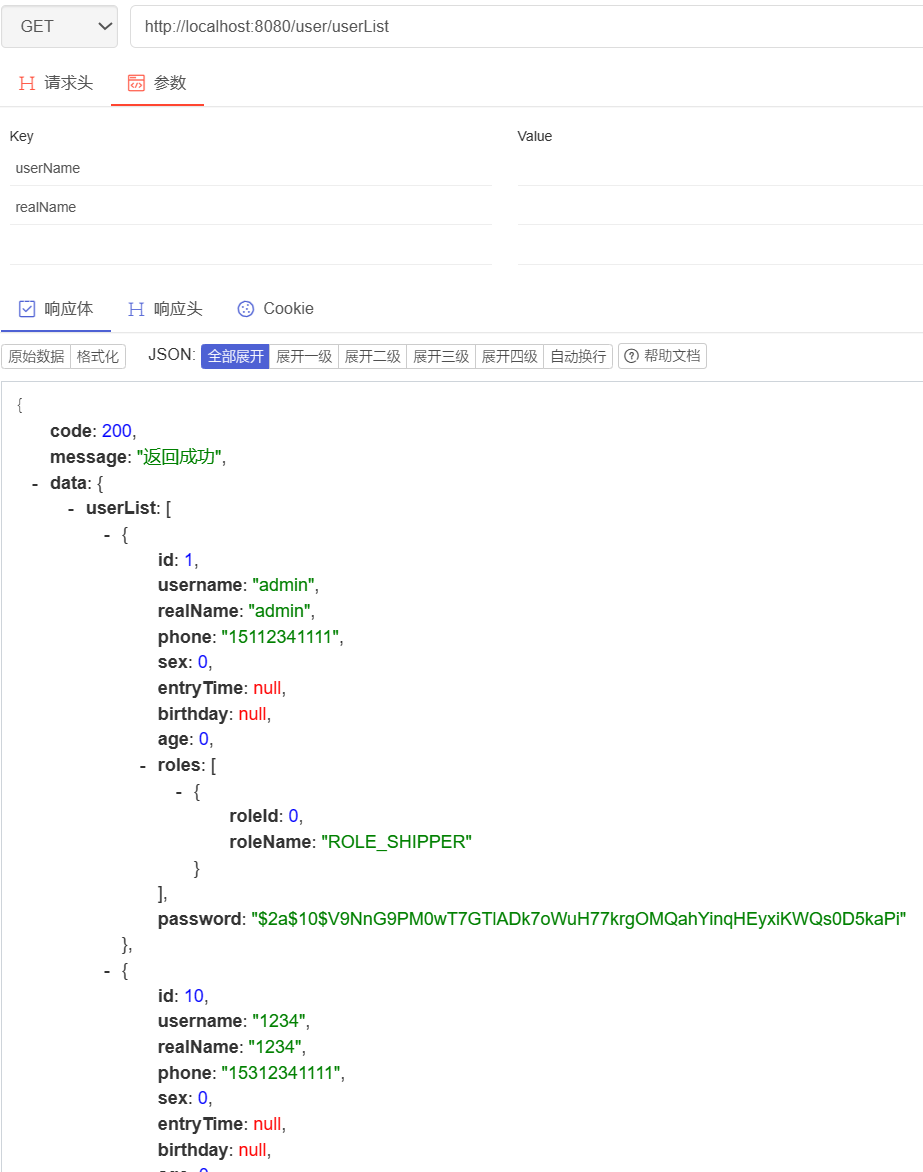
请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| userName | 用户名 | 无 | 否 | 无 |
| realName | 真实姓名 | 中文字符串 | 否 | 无 |

返回结果：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | | 类型 | 说明 |
| code | | String | 访问状态码 |
| message | | String | 状态码说明 |
| data | | Object | 返回的用户数据 |
|  | carrierList | Array | 返回员工列表 |

截图：



（4）查询用户

根据用户名和真实姓名模糊查询普通用户。

登出个人账号API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/userList |
| 请求方式 | GET |

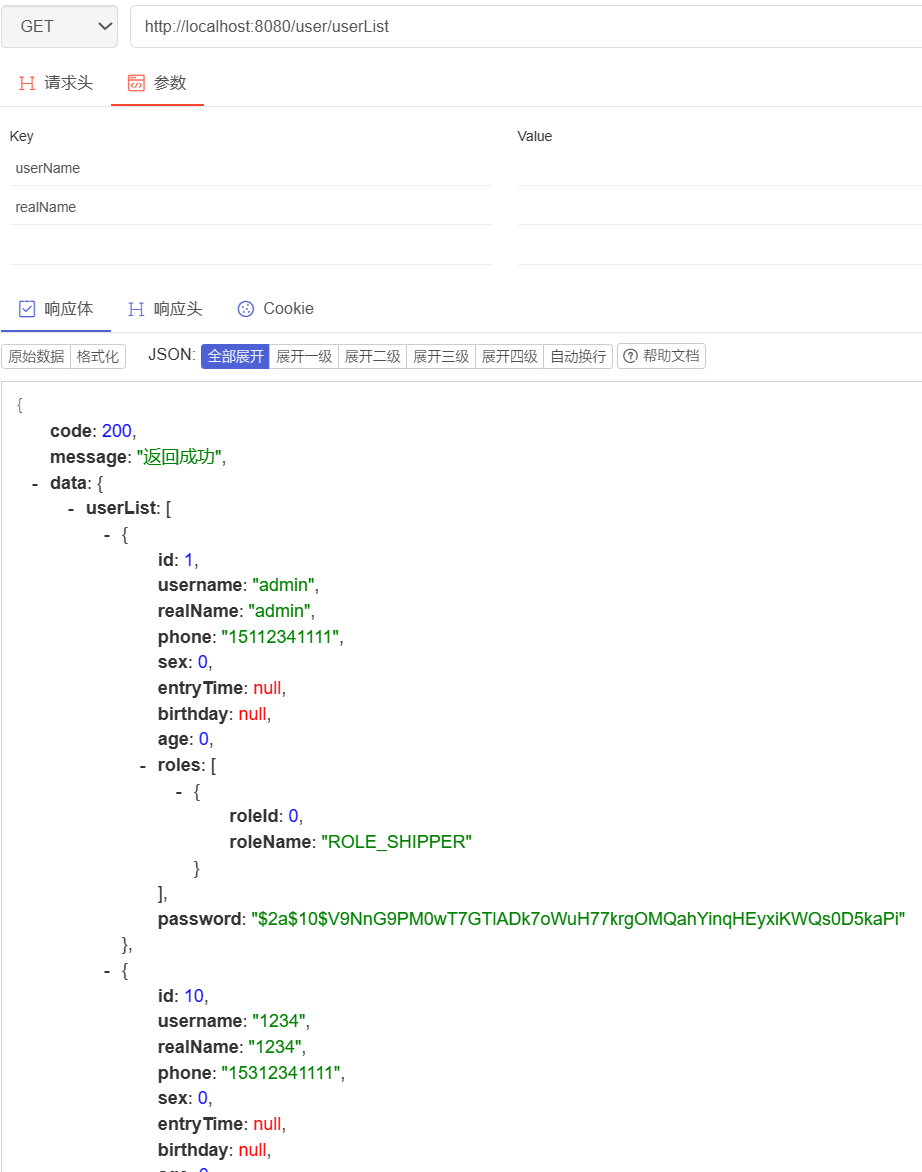
请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| userName | 用户名 | 无 | 否 | 无 |
| realName | 真实姓名 | 中文字符串 | 否 | 无 |

返回结果：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | | 类型 | 说明 |
| code | | String | 访问状态码 |
| message | | String | 状态码说明 |
| data | | Object | 返回的用户数据 |
|  | userList | Array | 返回用户的角色信息 |

截图：



（5）修改用户信息

修改普通用户信息。

修改用户信息API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/modifyUser |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| id | 用户ID | 无 | 是 | 无 |
| username | 用户名 | 无 | 是 | 无 |
| realName | 真实姓名 | 中文字符串 | 是 | 无 |
| phone | 电话号码 | 11位数字串 | 是 | 无 |
| password | 密码 | 不少于6位字符串 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（6）修改员工信息

修改员工信息。

修改员工信息API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/modifyCarrier |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| id | 用户ID | 无 | 是 | 无 |
| username | 用户名 | 无 | 是 | 无 |
| realName | 真实姓名 | 中文字符串 | 是 | 无 |
| phone | 电话号码 | 11位数字串 | 是 | 无 |
| password | 密码 | 不少于6位字符串 | 是 | 无 |
| entryTime | 入职时间 | 格式：“YYYY-MM-DD” | 是 | 无 |
| birthday | 生日 | 格式：“YYYY-MM-DD” | 是 | 无 |
| sex | 性别 | 数字 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（7）批量删除用户

批量删除用户，可删除普通用户和员工。

批量删除用户API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/deleteUsers |
| 请求方式 | DELETE |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| ids | 需要删除的用户ID | 数组 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（8）增加普通用户

增加普通用户。

增加普通用户API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/addUser |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| username | 用户名 | 无 | 是 | 无 |
| phone | 电话号码 | 11位数字串 | 是 | 无 |
| password | 密码 | 至少为六位字符串 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（9）增加员工用户

增加员工用户。

增加员工用户API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/user/addCarrier |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| id | 用户ID | 无 | 是 | 无 |
| username | 用户名 | 无 | 是 | 无 |
| realName | 真实姓名 | 中文字符串 | 是 | 无 |
| phone | 电话号码 | 11位数字串 | 是 | 无 |
| password | 密码 | 不少于6位字符串 | 是 | 无 |
| entryTime | 入职时间 | 格式：“YYYY-MM-DD” | 是 | 无 |
| birthday | 生日 | 格式：“YYYY-MM-DD” | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（10）查找车辆列表

模糊查询车辆列表。

查找车辆列表API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/van/vanList |
| 请求方式 | GET |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| license | 牌照 | 无 | 否 | 无 |
| state | 状态 | 数字  0：空闲  1：使用中  2：损坏 | 否 | 无 |

返回结果：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | | 类型 | 说明 |
| code | | String | 访问状态码 |
| message | | String | 状态码说明 |
| data | | Object |  |
|  | vanList | Array | 返回车辆信息 | |

截图：



（11）新增车辆

新增车辆。

新增车辆API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/van/addVan |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| maxLoad | 最大载重 | 数字 | 是 | 无 |
| license | 牌照 | 字符串 | 是 | 无 |
| state | 状态 | 数字  0：空闲  1：使用中  2：损坏 | 是 | 无 |
| info | 备注 | 字符串 | 否 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（12）修改车辆

修改车辆。

修改车辆API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/van/modifyVan |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| maxLoad | 最大载重 | 数字 | 是 | 无 |
| license | 牌照 | 字符串 | 是 | 无 |
| state | 状态 | 数字  0：空闲  1：使用中  2：损坏 | 是 | 无 |
| info | 备注 | 字符串 | 否 | 无 |
| id | 车辆ID | 表vans中已有记录的ID | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（13）删除车辆

根据汽车ID删除车辆。

删除车辆API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/van/deleteVans |
| 请求方式 | DELETE |

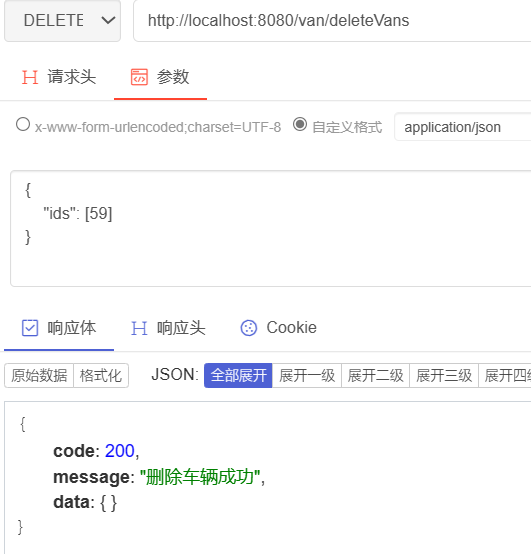
请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| ids | 车辆ID | 数组 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



# 订单管理

（1）创建订单

用户创建订单。

创建订单API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/order/create |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| shipperId | 创建订单用户ID | 必须是users表中已有的记录id | 是 | 无 |
| start | 出发地 | 无 | 是 | 无 |
| destination | 目的地 | 无 | 是 | 无 |
| receiver | 收件人 | 无 | 是 | 无 |
| weight | 重量 | 数字，单位kg | 是 | 无 |
| info | 信息 | 无 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（2）计算价格，为订单分配车辆和驾驶人

为用户创建的订单分配车辆和驾驶人，统计所需运费。

添加取件人API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/order/update |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| id | 订单ID | 必须是orders表中已有的记录id | 是 | 无 |
| carrierIds | 驾驶员ID | 必须是users表中已有的记录id且角色为carrier | 是 | 无 |
| money | 运费 | 数字 | 是 | 无 |
| vanIds | 货车ID | 必须是vans表中已有记录id | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（3）改变订单状态

改变订单的状态。

创建订单API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/order/changeState |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| id | 订单ID | 必须是orders表中已有的记录id | 是 | 无 |
| state | 状态值 | 可选值如下：  1：已付款待揽件  2：揽件运输中  3：已交付  4：退货拒收  5：订单结束  40：退货待揽件  41：退货运输中  42：退货交付 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（4）添加订单中转站

为运输中的订单添加其到达的中转站。

添加订单中转站API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/order/transfer |
| 请求方式 | POST |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| Id | 订单ID | 必须是orders表中已有的记录id | 是 | 无 |
| loca | 中转站地名 | 无 | 是 | 无 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | String | 访问状态码 |
| message | String | 状态码说明 |
| data | Object |  |

截图：



（5）查找快递员负责的订单

根据快递员id查找快递员负责的订单记录。

查找快递员负责的订单API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/api/order/searchCarrier/{carrierId} |
| 请求方式 | GET |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| carrierId | 驾驶人ID | 必须是users表中已有的记录id且角色为carrier | 是 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | totalMoney | | | String | | 总金额 | |
|  | orders | | Object | | 返回运输订单信息 | |
|  |  | money | | String | | 用户角色名 | |
|  |  | from | | String | | 订单出发地 | |
|  |  | to | | String | | 订单目的地 | |
|  |  | id | | String | | 订单ID | |

截图：



（6）根据ID查找订单

根据订单id查找订单。

查找订单API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/api/order/searchId/{orderId} |
| 请求方式 | GET |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| orderId | 订单ID | 必须是orders表中已有的记录id | 是 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | totalMoney | | | String | | 总金额 | |
|  | orders | | Object | | 返回运输订单信息 | |
|  |  | money | | String | | 用户角色名 | |
|  |  | from | | String | | 订单出发地 | |
|  |  | to | | String | | 订单目的地 | |
|  |  | id | | String | | 订单ID | |

截图：



（7）获取订单列表

根据发货人用户名、驾驶人用户名、汽车牌照模糊查询订单。

查询订单运输中转信息API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/api/order/orderList |
| 请求方式 | GET |

请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| shipperName | 发货人用户名 | 必须是users表中已有的记录id且角色为shipper | 否 | 无 |
| carrierName | 驾驶人用户名 | 必须是users表中已有的记录id且角色为carrier | 否 | 无 |
| vanLicense | 汽车牌照 | 必须是vans表中已有的记录license | 否 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | shipments | | Object | | 返回运输订单中转信息 | |
|  |  | transferTime | | String | | 中转时间 | |
|  |  | location | | String | | 中转地名 | |
|  |  | id | | String | | 中转记录ID | |

截图：



（8）查询订单运输中转信息

根据订单ID查询订单的运输中转信息。

查询订单运输中转信息API服务地址：

|  |  |
| --- | --- |
| URL | http://localhost:8080/api/order/searchShipment/{orderId  } |
| 请求方式 | GET |

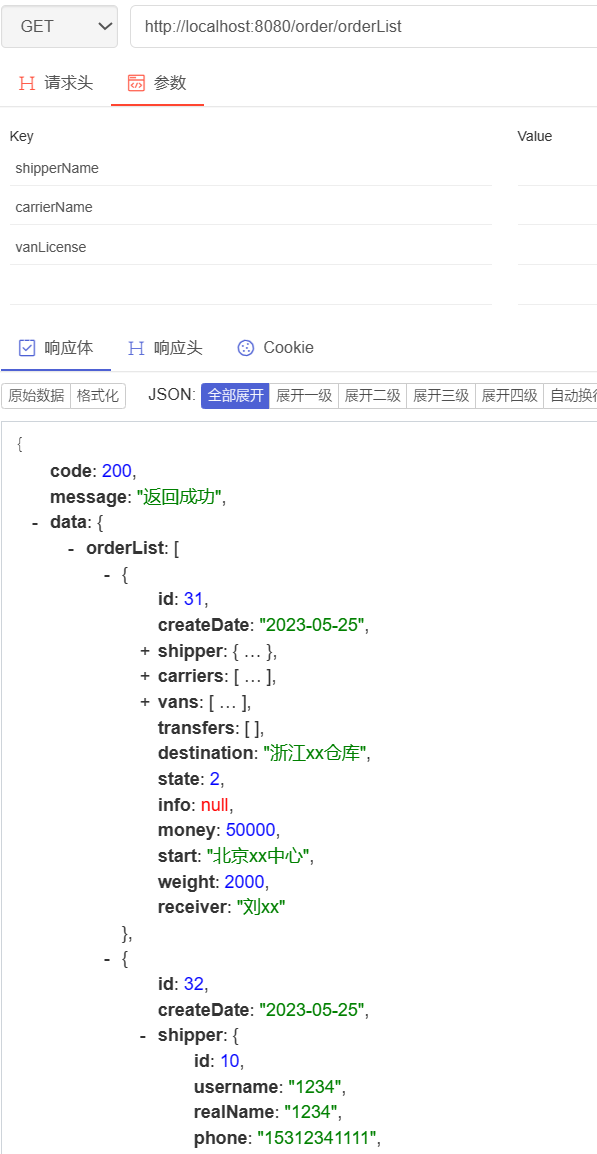
请求参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 规则说明 | 是否必须 | 缺省值 |
| orderId | 订单ID | 必须是orders表中已有的记录id | 是 | 无 |

返回结果：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | | | 类型 | | 说明 | |
| code | | | String | | 访问状态码 | |
| message | | | String | | 状态码说明 | |
| data | | | Object | | 返回的用户数据 | |
|  | shipments | | Object | | 返回运输订单中转信息 | |
|  |  | transferTime | | String | | 中转时间 | |
|  |  | location | | String | | 中转地名 | |
|  |  | id | | String | | 中转记录ID | |

截图：

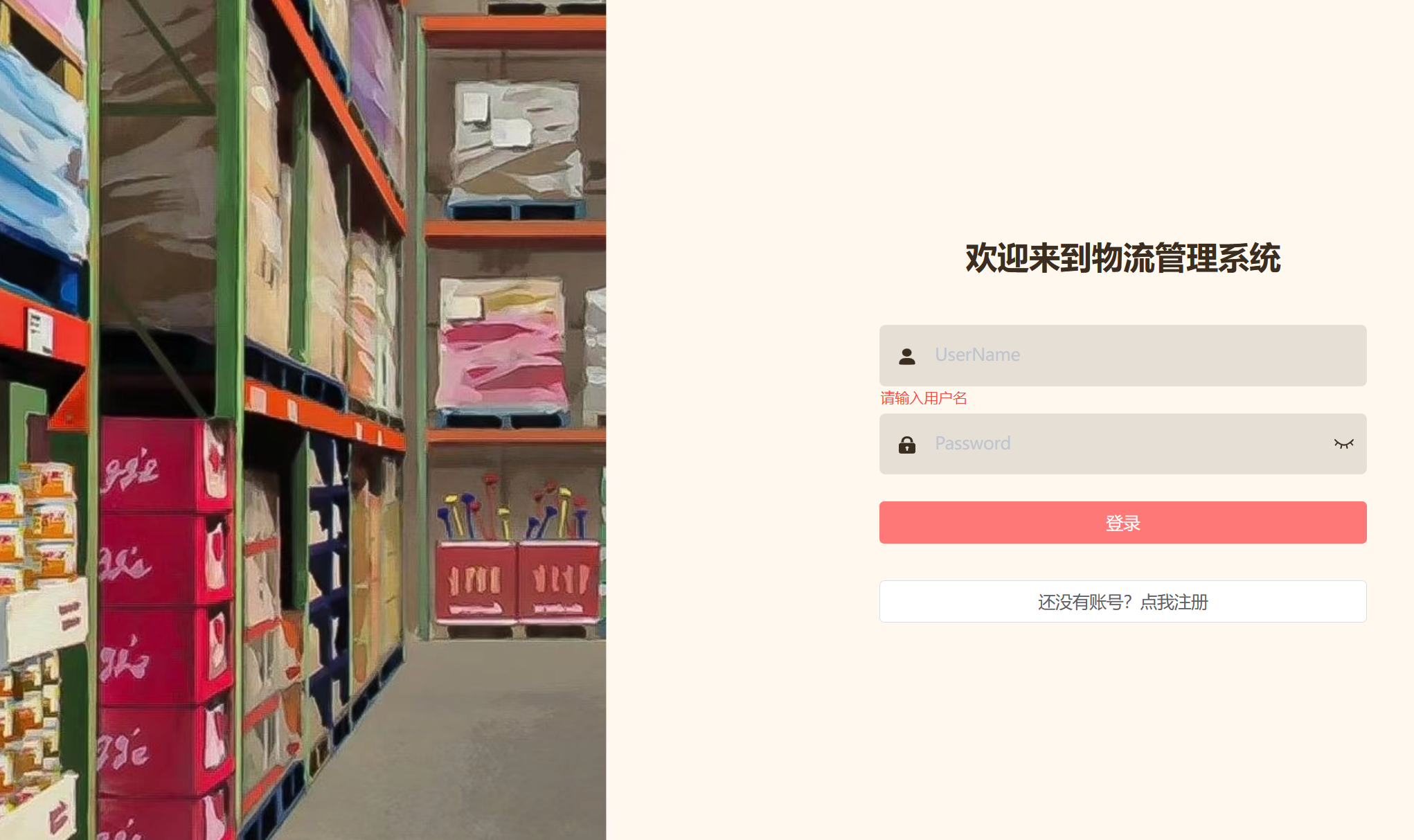


# 页面设计

主要设计了登录注册页面、首页、用户管理、员工管理、车辆管理、全部订单、承运员订单、用户订单页面。

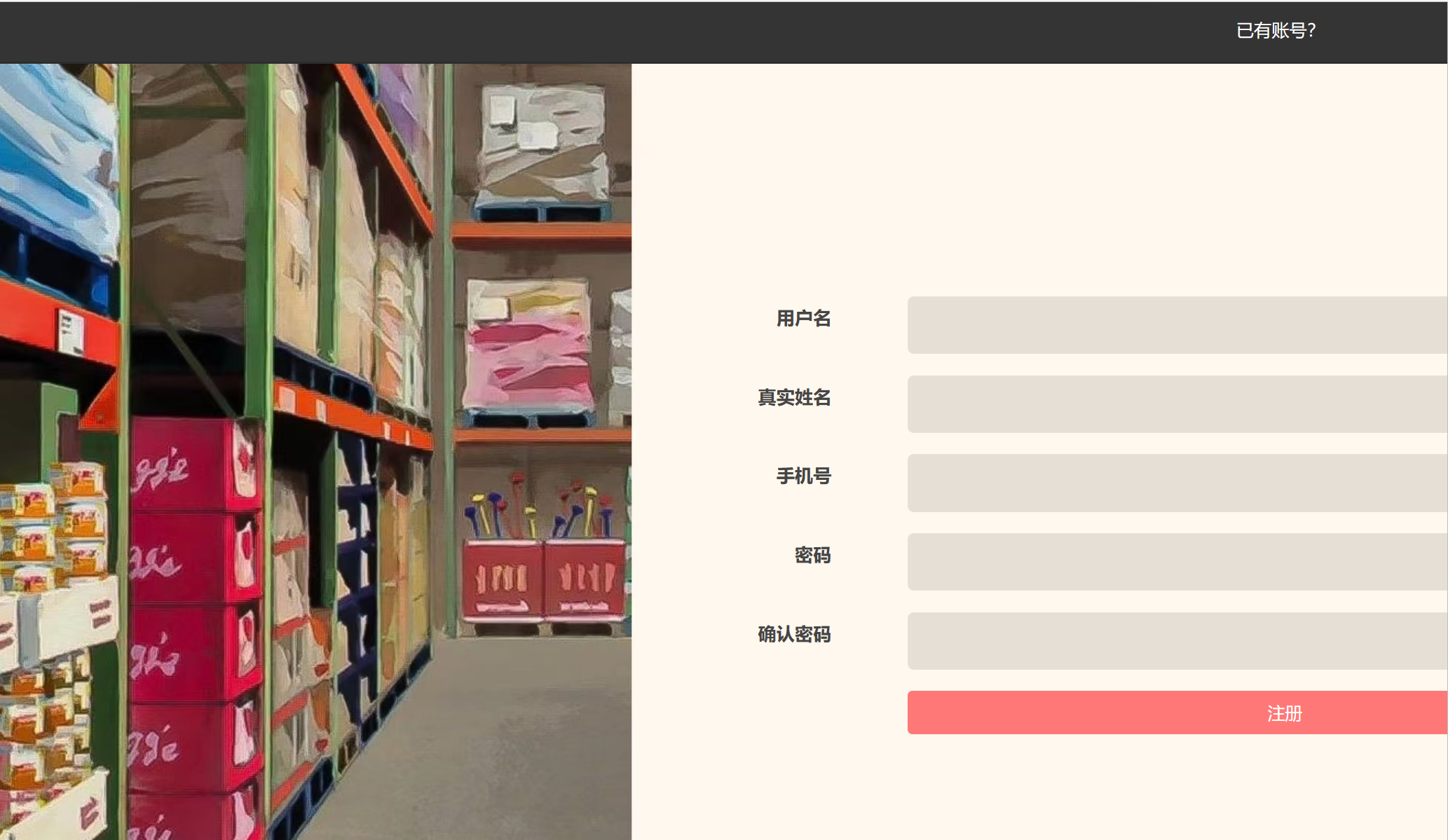
1. 登录页面：

员工、用户、管理员可从此登录进入系统。



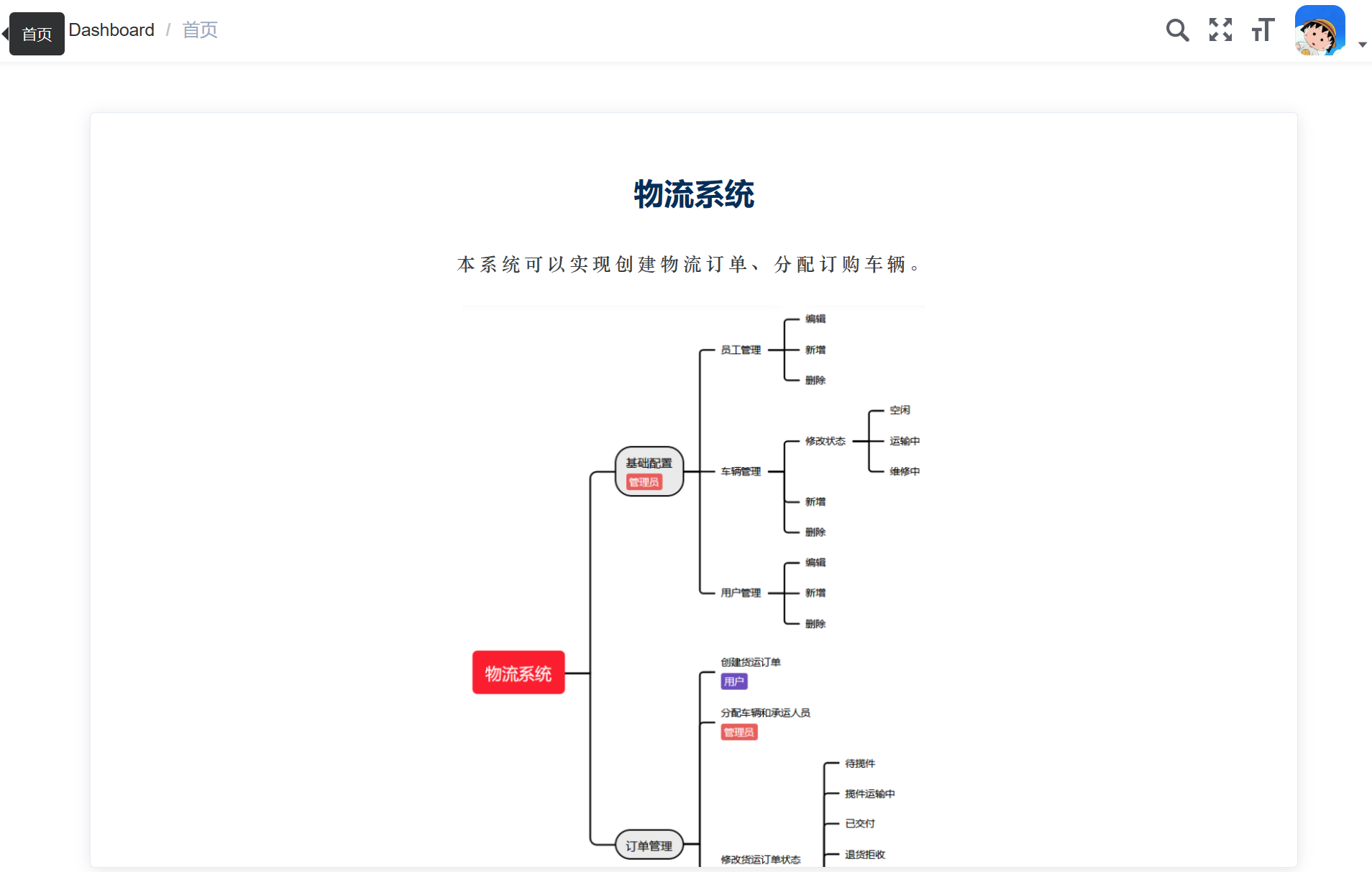
1. 注册页面：

普通用户能在此页面进行注册。



1. 首页：

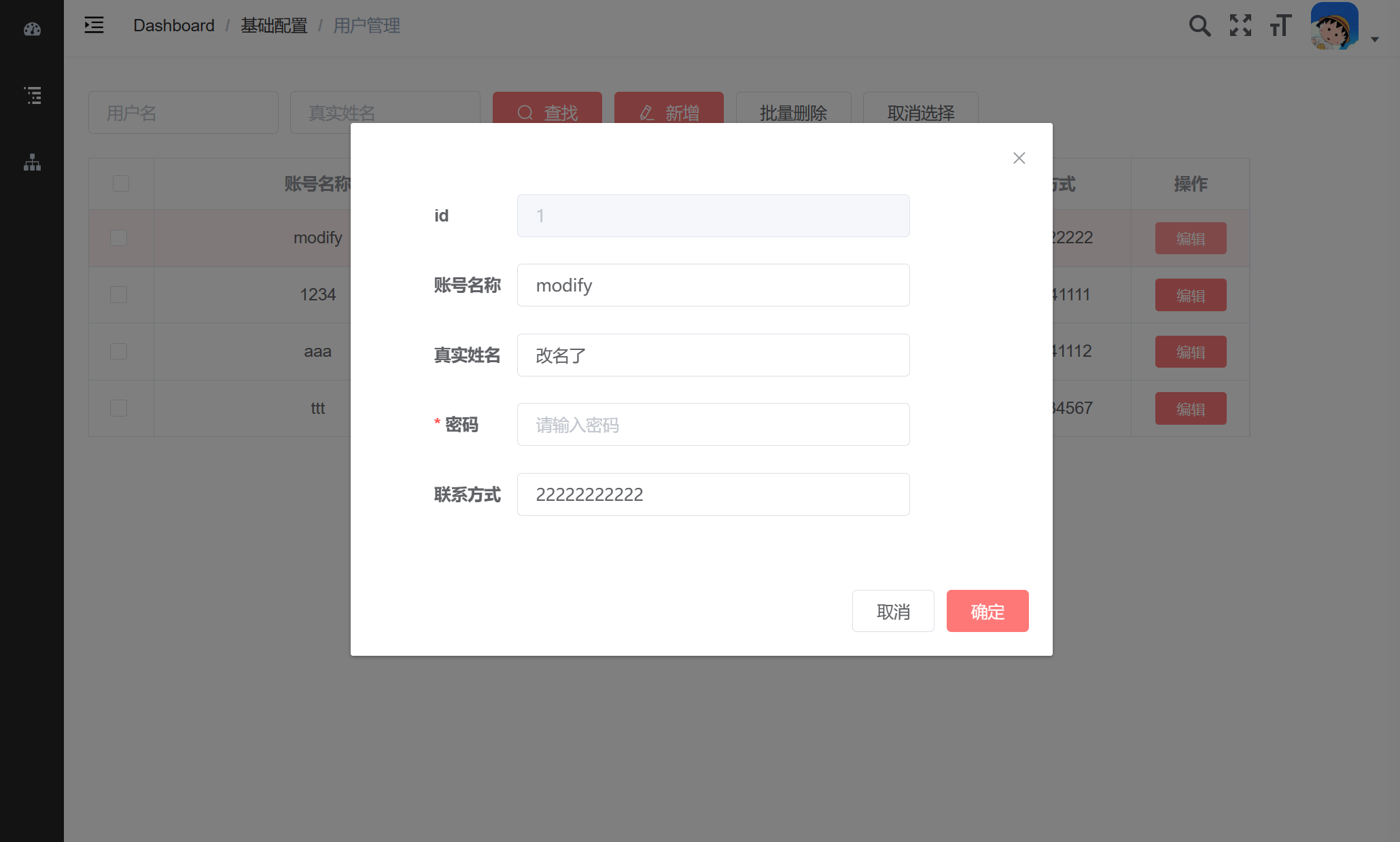
该系统的简单介绍。



1. 用户管理：

管理员可在此页面进行用户的增删查改。

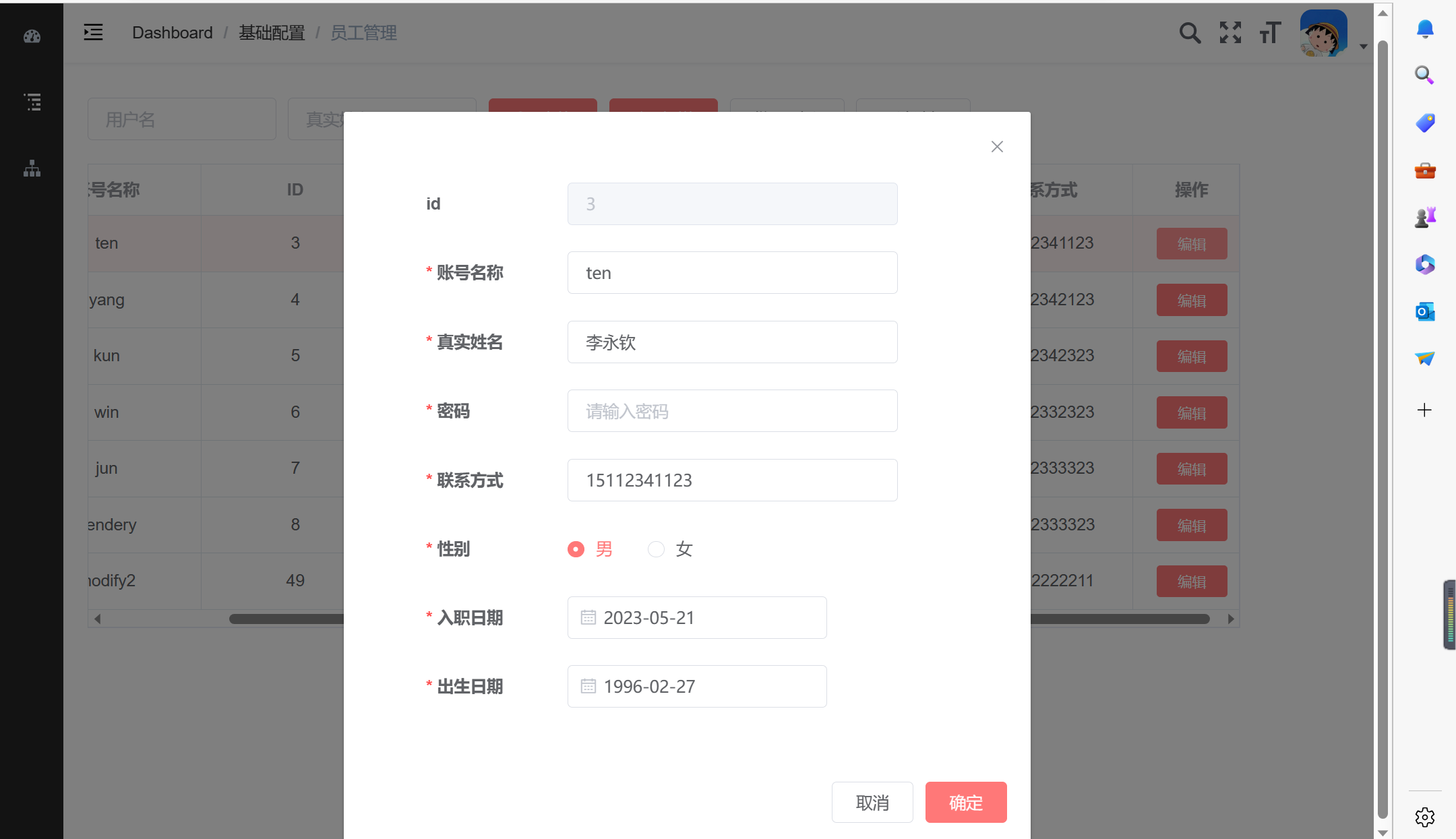




1. 员工管理：

管理员可在此页面进行员工用户的增删查改。





1. 车辆管理：

管理员可在此页面进行车辆管理。



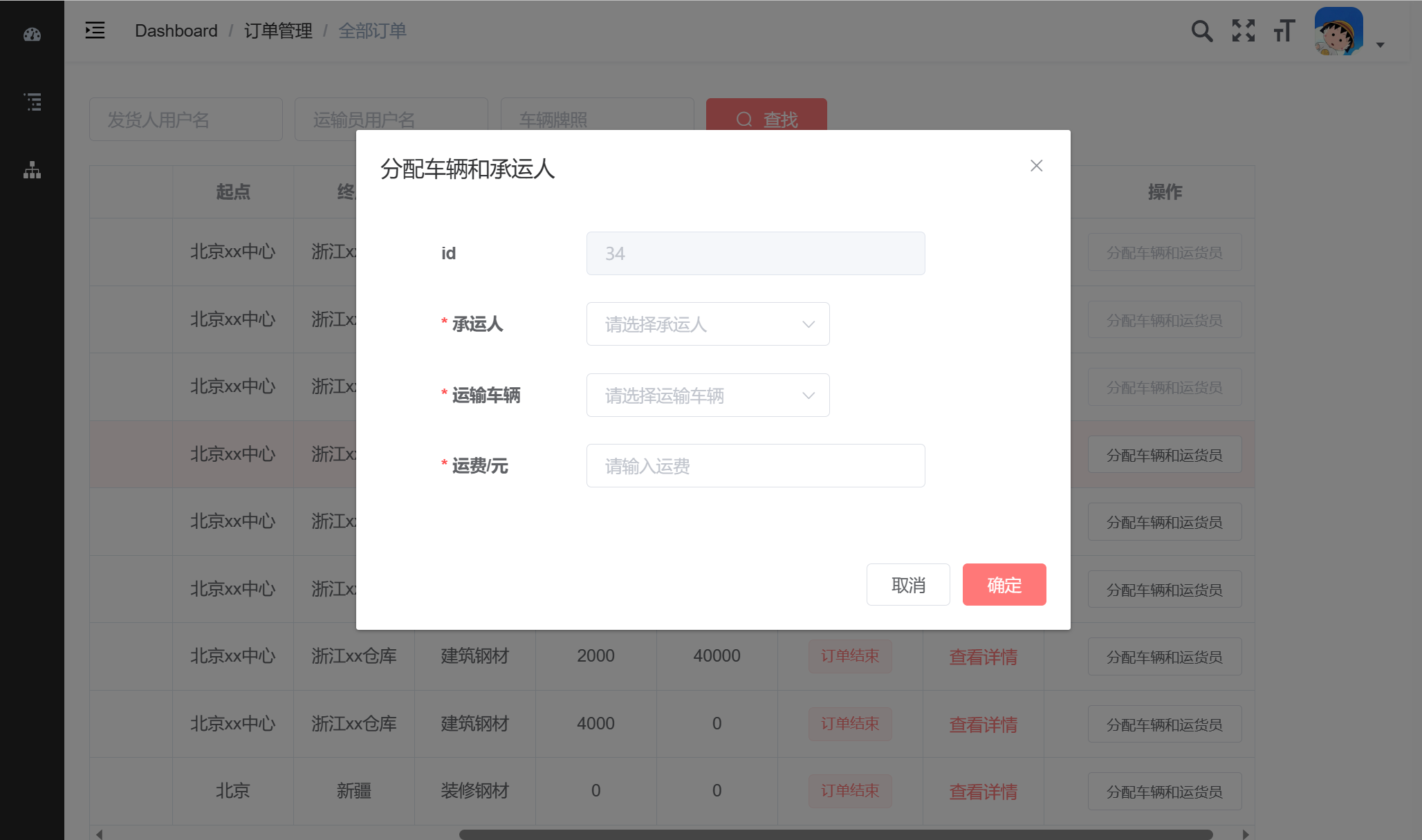


1. 全部订单：

管理员可在此页面给订单分配车辆和承运员、确定运费，分配车辆的数量不能多于承运人数量，车辆的总载重不能小于货物总重量。管理员还可在次页面查看订单详情，包括订单的运输记录。







1. 承运员订单：

员工可在此页面对自己负责的订单的地点进行登记，更新订单的状态，未分配车辆和承运人的订单无法进行地点登记和状态更新，处于“待揽件”状态的订单无法进行地点登记。员工还可在此页面查看自己负责订单的总运费金额。









1. 用户订单：

用户可在此页面创建订单，查看自己订单的物流记录。







# 单元测试

主要测试了订单模块Controller层的创建订单、更新订单状态、添加转运信息的功能。单元测试采用JUnit+ Mockito实现，生成代码测试覆盖分析报告。

# 创建订单

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例编号** | Order-01 | **用例类型** | Controller层 |
| **用例目的** | 测试创建订单的功能 | | |
| **测试步骤** | | | |
| 创建一个模拟的用户对象 shipper，并使用 Mockito 配置 userRepository.findById() 方法，以便在调用时返回这个用户对象。  创建一个 OrderRequest 对象，该对象包含创建订单所需的信息。  配置 orderRepository.save() 方法，以便在调用时返回一个新的 Order 对象，并将该对象的 ID 设置为 1L。  使用 MockMvc 发送 POST 请求，将 OrderRequest 对象作为 JSON 内容发送到 /order/create 路径。  使用 MockMvcResultMatchers 验证响应中的状态码和其他属性，例如响应体中的 code 字段是否为 200。  验证订单是否已创建并保存到数据库中。为此，创建一个 Order 对象，然后将其 ID 设置为 1L，并使用 Mockito 配置 orderRepository.findById() 方法，以便在调用时返回该对象。最后，使用 orderRepository.findAll() 方法获取数据库中的所有订单，并验证其长度是否为 1。 | | | |
| **代码覆盖分析** | | | |
|  | | | |

# 更新订单状态

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例编号** | Order-02 | **用例类型** | Controller层 |
| **用例目的** | 测试更新订单状态的功能 | | |
| **测试步骤** | | | |
| 创建一个模拟的用户对象 shipper，并使用 Mockito 配置 userRepository.findById() 方法，以便在调用时返回这个用户对象。  创建一个订单对象，并使用 Mockito 配置 orderRepository.findById() 方法，以便在调用时返回该订单对象。  创建一个 OrderRequest 对象，用于指定要更改的订单的 ID 和状态。  使用 MockMvc 发送 POST 请求，将 OrderRequest 对象作为 JSON 内容发送到 /order/changeState 路径。  使用 MockMvcResultMatchers 验证响应中的状态码和其他属性，例如响应体中的 code 字段是否为 200。  验证订单状态是否已更改但未保存到数据库中。为此，创建一个 Order 对象，该对象包含新状态，然后使用 Mockito 配置 orderRepository.findById() 方法，以便在调用时返回该对象。 | | | |
| **代码覆盖分析** | | | |
|  | | | |

# 添加转运信息

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例编号** | Order-03 | **用例类型** | Controller层 |
| **用例目的** | 测试添加转运信息的功能 | | |
| **测试步骤** | | | |
| 创建一个模拟的用户对象 shipper，并使用 Mockito 配置 userRepository.findById() 方法，以便在调用时返回这个用户对象。  创建一个模拟的用户对象 carrier，并使用 Mockito 配置 userRepository.findById() 方法，以便在调用时返回这个用户对象。  创建一个模拟的 Van 对象，并使用 Mockito 配置 vanRepository.findById() 方法，以便在调用时返回这个 Van 对象。  创建一个订单对象，并使用 Mockito 配置 orderRepository.findById() 方法，以便在调用时返回该订单对象。  创建一个 OrderRequest 对象，用于指定要转移的订单的 ID、运输公司 ID 和货车 ID。  使用 Mockito 配置 transferRepository.save() 方法，以便在调用时返回一个新的 Transfer 对象。  使用 MockMvc 发送 POST 请求，将 OrderRequest 对象作为 JSON 内容发送到 /order/transfer 路径。  使用 MockMvcResultMatchers 验证响应中的属性，例如响应体中的 code 字段是否为 102。 | | | |
| **代码覆盖分析** | | | |
|  | | | |