# 汽车销售管理系统

# 1.系统分析

## 1.1背景分析

## 随着现代电商行业的迅猛发展，不停的抨击传统销售行业，计算机技术已经渗透到各个领域，成为各行业必不可少的工具，特别是Internet技术的推广和信息高速公路的建立，使IT产业在市场竞争中越发显示出其独特的优势，步入信息化时代，有巨大的数据信息等待加工处理和传输，这使得对数据库的进一步开发和利用显得尤为迫切。汽车工业经历了一百多年的发展历史，对国家经济的发展和腾飞以及对人类社会的文明带来了巨大影响。在许多国家，汽车工业已成为支柱产业，随着人们生活水平以及汽车性能的不断提高，人们对汽车的消费和需求也越来越旺盛，世界汽车工业也保持庞大的市场需求和生产规模。

## **1.2系统需求分析**

## 近年来，中国汽车市场发展迅猛，政府也有新政策的出台，大大刺激了消费市场。巨大的市场容量和可观的经济效益，引起各汽车厂家(包括世界各著名汽车厂家的在华企业)纷纷加大投资，汽车市场烽烟四起，汽车销售策略和模式新招频出，竞争已是空前的激烈。在如此形式下，一种介于生产厂家和用户之间的桥梁，汽车销售公司正在不断的产生和发展壮大。汽车销售公司的汽车销售管理水平不高，直接关系着各汽车厂家的根本利益，但现有的汽车销售公司的汽车销售管理水平还停留在纸介质的基础上，这样的机制已经不能适应时代的发展，因为它浪费了许多的人力和物力。在信息时代这种传统的管理方法必然被计算机为基础的信息管理所取代。

## 1.4系统功能分析

本系统采用自定向下方法开发，其功能模块主要有如下几个部分：进货管理模块、库存管理模块、销售管理模块、客户管理模块和系统模块。具体的功能描述如下：

进货管理：能很好地完成汽车的进货管理。进货管理包括供应商信息，需求商信息和汽车基本信息的入库作业，并根据企业情况制定进货计划，实施产品采购，并进行进货查询。

销售管理：能详细地记录需求商与公司之间的交易过程，即订车和退车过程。能及时准确地获得汽车在某阶段的盈亏情况和市场的需求走势。即通过对汽车销售排行、汽车进货、汽车退货、订车、退车等信息查询获得相应的结果，以备公司作为参考，从而制定出相应的策略。

库存管理：其功能是对采购来的车辆进行企业内部编码入库，产品分类等活动，同时能够根据销售清单进行出货管理，以及库存统计。

客户管理：其功能是对公司的客户和供货商信息进行管理，包括查询，更新等。

系统管理：能够对不同权限的用户进行合理的管理。包括添加用户，删除用户，用户账号和密码管理以及系统各模块的维护等。功能结构图如下：

汽车销售管理系统

客户管理

销售管理

进货管理

库存管理

系统管理

客户信息管理

客户信息查询

销售信息登记

销售信息查询

销售信息盘点

进货信息登记

进货信息查询

库存信息登记

库存信息维护

客户权限管理

系统功能维护

## 2.系统业务流程图

汽车销售管理系统主要是是以先进的管理理念和科学的计算机技术来取代原始的手工操作。在新系统开发前，对这个汽车管理系统业务流程进行了系统的调查，得出了下面的业务流程图：

需

不需

否

否

是

是

是

否

公司总经理与供货商洽谈进货

是否是新供货商

《新供货商新车辆引进流程》

《订货管理作业流程》

《进货工作流程》

销售管理

采购与店面共同分析是否滞销等问题

采购部分析经营是否有问题

《新车辆转正流程》

《供应商分析淘汰流程》

《结算流程》

《旧供货商新车辆引进流程》

采购部处理解决

《退货管理流程》

新车辆是否已经到期

是否需新供应商

**系统数据流分析**

销售管理系统顶层图

公司经理

汽车销售管理系统

顾客

供货商

职工

人事信息操作

人事信息清单

车辆信息清单

车辆信息操作

销售单

进货单

采购单

职工信息

1：库存管理

2：销售管理

3：进货管理

4：人事管理

库存信息清单

已进货信息

进货信息查询

库存信息

进货信息请求

销售信息查询

库存信息查询

销售单

销售信息清单

前台销售信息

职工信息

人事信息清单

人事信息查询

采购单

进货信息清单

进货单

已有供货商信息

新供货商信息

销售信息

汽车销售管理系统第0层图









# 3.系统详细设计

## 3.1系统功能结构图

系统总体结构如图:

汽车销售管理系统

客户管理模块

销售管理模块

进货管理模块

库存管理模块

系统管理模块

客户信息管理

客户信息查询

销售信息登记

销售信息查询

销售信息盘点

进货信息登记

进货信息查询

库存信息登记

库存信息维护

客户权限管理

系统功能维护

## 3.2系统功能设计

系统功能设计是使整个系统能基本实现车辆的进、销、存等管理功能，能让公司管理者更直观的了解公司的经营状况，以做出相应的经营决策。管理信息系统(MIS)就是要对这些资源建立正确的数据库，并进行信息的收集、传送、存储、编制成各种凭证、报表、台帐、图形等信息，同时对这些信息进行维护，通过实测公司的各种运行情况，并利用过去的历史数据预测未来的发展，从公司全局的角度出发进行决策，利用信息控制公司的行为，辅助公司完成其销售目标。

本系统主要包括五大模块，分别是客户管理模块，销售管理模块，进货管理模块，库存管理模块，系统管理模块。每个模块的功能如下：

1. 客户管理模块功能：

(1)实现客户信息的查询；

(2)实现客户信息的更新，包括增加、删除、修改功能；

(3)实现供货商信息的查询；

(4)实现供货商信息的更新，包括增加、删除、修改功能。

2. 销售管理模块功能：

(1)实现销售信息的查询；

(2)实现车辆销售信息的盘点。

3. 进货管理模块功能：

(1)实现进货信息的查询；

(2)实现进货信息的更新，包括增加、删除、修改功能。

4. 库存管理模块功能：

(1)实现车辆信息的查询；

(2)实现车辆信息的更新，包括增加、删除、修改功能；

(3)实现库存信息的查询；

(4)实现库存信息的更新，包括增加、删除、修改功能。

5.系统管理模块功能：

(1) 实现对不同权限的用户进行合理的管理；

(2)实现对各个模块的维护和监控等功能。

## 3.3系统各模块功能设计

### 3.3.1. 客户管理模块

模块功能：该模块主要是实现对客户、供货商信息的查询；实现客户、供货商信息的更新，包括增加、删除、修改功能。

客户、供应商信息查询：根据该模块的信息公司可以制定营销策略和销售策略，完善公司服务，提高质量。更好的带动公司销售。

客户、供应商信息更新：更好的完善客户、供应商的信息，删除陈旧无用的信息。

输入项、输出项：该模块的输入项为客户、供应商信息。

### 3.3.2. 销售管理模块

模块功能：本模块的主要功能是把销售信息写入销售清单，同时对库存清单进行更新，以备用户将来对库存信息进行查询和打印，此外还可以对销售信息进行查询和盘点功能。

销售信息查询：根据车辆销售情况，按需要对销售情况进行查询。在该查询模块中，可以按照车辆的编号、销售日期等多种方式进行车辆销售信息查询。

销售信息盘点：可以按照需要对在一定时期内的销售情况进行盘点。

输入项、输出项：本模块的数据输入项主要是车辆的销售单号和车辆编号。其中还包括销售数量和销售日期。

### 3.3.3. 进货管理模块

模块功能：该模块主要是实现进行进货信息查询、进货计划制定两个功能。

进货信息查询：该模块为公司提供整个公司的进货情况查询，也可以进行进货信息的明细查询，并以进货报表形式打印出来。

进货计划制定：该模块可以为制定进货计划，并形成相应的采购订单。

输入项、输出项：该模块的输入项为公司的进货单，其中包括进货单的编号，车辆的编号，供货商号，进货价，数量，金额进货日期，备注。

### 3.3.4. 库存管理模块

模块功能：本模块的主要功能是车辆信息的查询和更新，库存信息的查询和更新。

查询车辆信息：主要功能查询车辆的明细信息，和它的库存信息。

添加车辆信息：主要功能添加新进的车辆。

修改车辆信息：主要功能是修改车辆的信息，包括他的库存数量。

删除车辆信息：主要功能是删除不再销售的车辆信息。

输入项、输出项：本模块的数据输入项主要是车辆编号。

### 3.3.5. 系统管理模块

系统用户管理：在系统实行多用户管理时，本模块将为系统开辟多个系统用户，并能对其进行简单的信息维护，用户名变更和密码修改。

退出：选择退出系统，将关闭整个系统，结束本次使用。

**3.4 程序结构**

这是一个针对汽车销售有关的后台管理系统。系统管理员登陆系统可以为用户设置不同权限，用户登录不同的页面实现不同的功能；项目中有报表生成，文件上传下载等功能。所涉及到的知识点有。

Spring MVC+Spring+MyBatis+BootsStrap+Ajax+JQuery。

**3.4.1程序applicationContext配置文件**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xmlns:aop=*"http://www.springframework.org/schema/aop"*

xmlns:context=*"http://www.springframework.org/schema/context"*

xmlns:mvc=*"http://www.springframework.org/schema/mvc"*

xmlns:tx=*"http://www.springframework.org/schema/tx"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-4.1.xsd*

*http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.1.xsd*

*http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc-4.1.xsd*

*http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.1.xsd"*>

<!-- 配置文件上传 -->

<bean class=*"org.springframework.web.multipart.commons.CommonsMultipartResolver"* id=*"multipartResolver"*>

<property name=*"defaultEncoding"* value=*"utf-8"*></property>

<property name=*"maxUploadSize"* value=*"-1"*></property>

</bean>

<!-- 文件上传异常 -->

<!-- <bean class="org.springframework.web.multipart.MaxUploadSizeExceededException"></bean> -->

<!-- JSON -->

<mvc:annotation-driven></mvc:annotation-driven>

<mvc:default-servlet-handler/>

<!-- 配置M层 -->

<bean class=*"org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter"*></bean>

<!-- 配置V层 -->

<bean class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>

<property name=*"prefix"* value=*"/"*></property>

<property name=*"suffix"* value=*".jsp"*></property>

<property name=*"viewClass"* value=*"org.springframework.web.servlet.view.JstlView"*></property>

</bean>

<!-- 配置C层 -->

<context:annotation-config></context:annotation-config>

<context:component-scan base-package=*"com.svse"*></context:component-scan>

<!-- 配置数据源 -->

<bean class=*"com.mchange.v2.c3p0.ComboPooledDataSource"* id=*"dataSource"*>

<property name=*"driverClass"* value=*"com.mysql.jdbc.Driver"*></property>

<property name=*"jdbcUrl"* value=*"jdbc:mysql://localhost:3306/car\_database"*></property>

<property name=*"user"* value=*"root"*></property>

<property name=*"password"* value=*"HAI740520"*></property>

</bean>

<!-- 配置sqlsession -->

<bean class=*"org.mybatis.spring.SqlSessionFactoryBean"* id=*"sqlSessionfactory"*>

<property name=*"dataSource"* ref=*"dataSource"*></property>

<property name=*"configLocation"* value=*"classpath:conf.xml"*></property>

<property name=*"mapperLocations"* value=*"classpath:com/svse/mapping/\*.xml"*></property>

</bean>

<!-- 配置dao -->

<bean class=*"org.mybatis.spring.mapper.MapperScannerConfigurer"*>

<property name=*"sqlSessionFactory"* ref=*"sqlSessionfactory"*></property>

<property name=*"basePackage"* value=*"com.svse.dao"*></property>

</bean>

<!-- 配置事务 -->

<bean class=*"org.springframework.jdbc.datasource.DataSourceTransactionManager"* id=*"transactionManager"*>

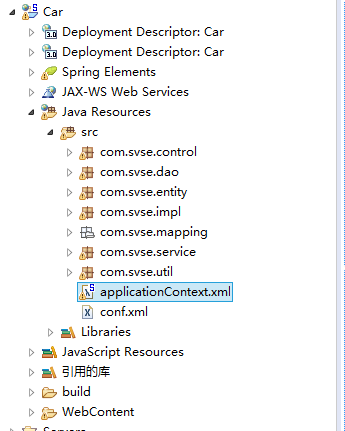
<property name=*"dataSource"* ref=*"dataSource"*></property>

</bean>

<tx:annotation-driven transaction-manager=*"transactionManager"*/>

</beans>

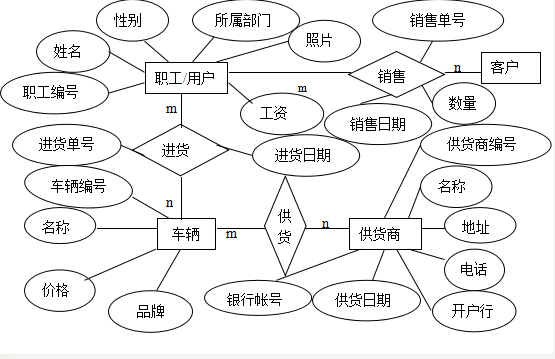
**3.4.2项目框架图**



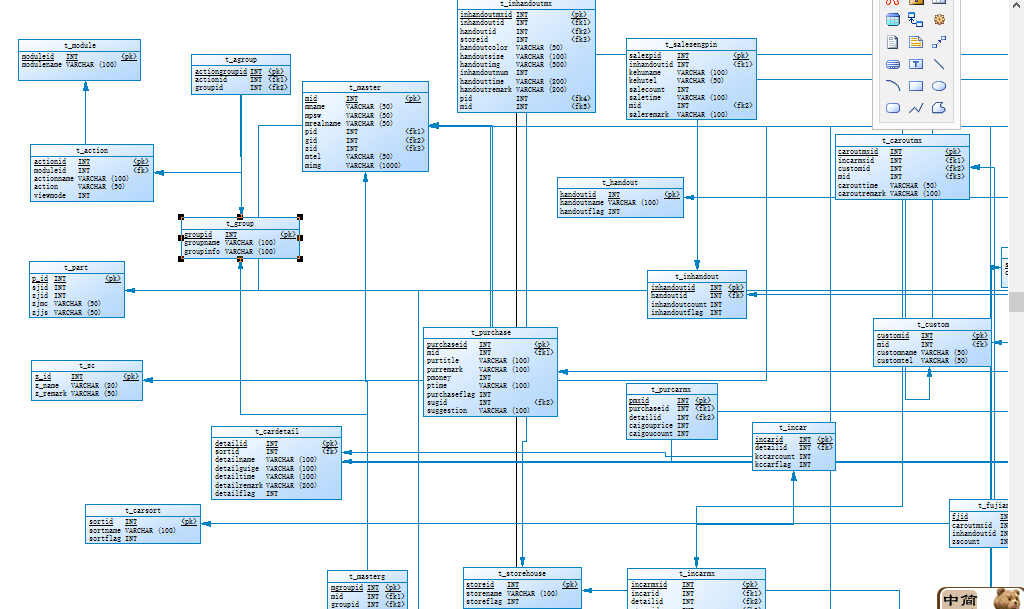
## 3.5系统数据库设计

### 3.5.1. E-R模型

汽车销售管理系统数据库E-R模型，如图所示：



**3.5.2数据库物理模型图**

****

**3.5.3数据库初始化代码**

create database car\_database; #创建数据库test\_database

use car\_database; #使用创建好的数据库

#1左侧模块表

CREATE TABLE t\_module (

moduleid INT auto\_increment PRIMARY KEY NOT NULL, #模块编号

modulename VARCHAR (100) #模块名称

) ENGINE = INNODB DEFAULT charset = utf8;

#2模块具体动作表

CREATE TABLE t\_action ( actionid INT auto\_increment PRIMARY KEY NOT NULL, #动作编号

moduleid INT, #模块编号（外键1）

actionname VARCHAR (100), #动作名称

action VARCHAR (50), #动作对应的字符串

viewmode INT #是否可见

) ENGINE = INNODB DEFAULT charset = utf8;

#3管理组表

CREATE TABLE t\_group ( groupid INT auto\_increment PRIMARY KEY NOT NULL, #管理组编号(主键)

groupname VARCHAR (100), #管理组名称

groupinfo VARCHAR (100) #管理级信息

) ENGINE = INNODB DEFAULT charset = utf8;

#4动作所在组记录表

CREATE TABLE t\_agroup ( actiongroupid INT auto\_increment PRIMARY KEY NOT NULL, #动作所在组编号(主键)

actionid INT, #编号所属模块具体动作ID（外键1）

groupid INT #动作所属管理组ID（外键2）

) ENGINE = INNODB DEFAULT charset = utf8;

CREATE TABLE t\_part ( p\_id INT auto\_increment NOT NULL PRIMARY KEY, sjid INT, zjid INT, zjmc VARCHAR (50), zjjs VARCHAR (50));

CREATE TABLE t\_gw (

g\_id INT auto\_increment PRIMARY KEY NOT NULL,

g\_name VARCHAR (50),

p\_id INT,

g\_remark VARCHAR (50)

);

CREATE TABLE t\_zc (

z\_id INT auto\_increment PRIMARY KEY NOT NULL,

z\_name VARCHAR (20),

z\_remark VARCHAR (50)

);

#5用户表

CREATE TABLE t\_master ( mid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

mname VARCHAR (50), #用户登陆名

mpsw VARCHAR (50), #密码

mrealname VARCHAR (50), #用户真实姓名

pid INT, #所属部门编号(外键1)

gid INT, #所属岗位编号（外键2）

zid INT, #职称编号（外键3）

mtel VARCHAR (50), #电话号码

mimg VARCHAR (1000) #相片

) ENGINE = INNODB DEFAULT charset = utf8;

#6用户所在管理组表

CREATE TABLE t\_masterg (

mgroupid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

mid INT, #用户信息表id(外键1)

groupid INT #动作所属管理组ID(外键2)

) ENGINE = INNODB DEFAULT CHARset = utf8;

#1-5汽车品牌表(carsort)

CREATE TABLE t\_carsort (

sortid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

sortname VARCHAR (100), #汽车品牌名称

sortflag INT

) ENGINE = INNODB DEFAULT CHARset = utf8;

#1-6汽车系列表(cardetail)

CREATE TABLE t\_cardetail (

detailid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

sortid INT, #汽车品牌编号

detailname VARCHAR (100), #汽车系列名称

detailguige VARCHAR (100), #汽车车型

detailtime VARCHAR (100), #汽车出厂时间

detailremark VARCHAR (200), #汽车系列备注说明

detailflag INT #是否停产

) ENGINE = INNODB DEFAULT charset = utf8;

#1-7赠品表(handout)

CREATE TABLE t\_handout (

handoutid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

handoutname VARCHAR (100), #赠品名称

handoutflag INT

) ENGINE = INNODB DEFAULT charset = utf8;

#1-8仓库字典(storehouse)

CREATE TABLE t\_storehouse (

storeid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

storename VARCHAR (100), #仓库名称

storeflag INT

) ENGINE = INNODB DEFAULT charset = utf8;

#1-9采购申请表(purchase)

CREATE TABLE t\_purchase (

purchaseid INT auto\_increment PRIMARY KEY NOT NULL, #采购申请单编号(主键)

mid INT, #申请人编号(外键)

purtitle VARCHAR (100), #申请标题

purremark VARCHAR (100), #备注

pmoney INT, #预算资金

ptime VARCHAR (100), #申请时间

purchaseflag INT, #是否审批

sugid INT, #审批人编号(外键#员工表)

suggestion VARCHAR (100) #审批意见

) ENGINE = INNODB DEFAULT charset = utf8;

1-10汽车采购明细表(purcarmx)

CREATE TABLE t\_purcarmx (

pmxid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

purchaseid INT, #申请单编号（外键1）

detailid INT, #汽车系列表编号（外键2）

caigouprice INT, #汽车采购单价

caigoucount INT #数量

) ENGINE = INNODB DEFAULT charset = utf8;

#1-11赠品入库表(inhandout)

CREATE TABLE t\_inhandout (

inhandoutid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

handoutid INT, #赠品编号(外键1)

inhandoutcount INT, #库存量

inhandoutflag INT #1—代表有库存量 0—无库存量

) ENGINE = INNODB DEFAULT charset = utf8;

#1-12赠品入库明细表(inhandoutmx)

CREATE TABLE t\_inhandoutmx (

inhandoutmxid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

inhandoutid INT, #赠品入库编号(外键1)

handoutid INT, #赠品编号（外键2）

storeid INT, #仓库编号（外键3）

handoutcolor VARCHAR (50), #赠送品颜色

handoutsize VARCHAR (100), #赠送品尺寸

handoutimg VARCHAR (500), #赠品图片

inhandoutnum INT, #入库数量

handouttime VARCHAR (200), #入库时间

handoutremark VARCHAR (200), #备注说明

pid INT, #入库经办人所在部门（外键5）

mid INT #入库经办人(外键4)

) ENGINE = INNODB DEFAULT charset = utf8;

#1-13汽车入库表(incarhandout)

CREATE TABLE t\_incar (

incarid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

detailid INT, #汽车系列编号(外键1)

kccarcount INT, #库存量

kccarflag INT #1—代表有库存 0—无库存

) ENGINE = INNODB DEFAULT charset = utf8;

#1-14汽车入库明细(incarhandoutmx)

CREATE TABLE t\_incarmx (

incarmxid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

incarid INT, #汽车入库表编号（外键1）

detailid INT, #汽车系列编号(外键2)

storeid INT, #原始存放仓库编号(外键3)

newstoreid INT, #现存放仓库编号

incarprice VARCHAR (100), #入库价格

outcarprice VARCHAR (100), #市场指导价

carshouzi VARCHAR (100), #是否自动挡

#carxhname varchar(100),#车辆型号

carcjh VARCHAR (100), #车架号

carhgz VARCHAR (100), #合格证

carfdjh VARCHAR (100), #发动机号

carcolor VARCHAR (100), #车身颜色

carimg VARCHAR (1000), #车辆图片

carchanshu VARCHAR (1000), #详细配置参数

carrukucount INT, #入库数量

carrukutime VARCHAR (100), #入库时间

pid INT, #入库经办人所在部门（外键5）

mid INT, #入库经办人

carkucunflag INT #1—没有出售 0—已经出售

) ENGINE = INNODB DEFAULT charset = utf8;

#1-15客户信息表(custom)

CREATE TABLE t\_custom (

customid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

mid INT, #员工编号(外键)

customname VARCHAR (50), #客户姓名

customtel VARCHAR (50) #客户电话

) ENGINE = INNODB DEFAULT charset = utf8;

#1-16汽车出库记录明细表(caroutmx)

CREATE TABLE t\_caroutmx (

caroutmxid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

incarmxid INT, #汽车表编号（外键）

customid INT, #客户编号

mid INT, #经办人编号（外键）

carouttime VARCHAR (50), #出售时间

caroutremark VARCHAR (100) #备注说明

) ENGINE = INNODB DEFAULT charset = utf8;

#1-17出售赠品表(salezengpin)

CREATE TABLE t\_salezengpin (

salezpid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

inhandoutid INT, #赠品编号（外键）

kehuname VARCHAR (100), #零售客户姓名

kehutel VARCHAR (50), #零售客户电话

salecount INT, #出售数量

saletime VARCHAR (100), #卖出时间

mid INT, #经办人

saleremark VARCHAR (100) #备注

) ENGINE = INNODB DEFAULT charset = utf8;

#1-18汽车赠品(附件记录)表(fujian)

CREATE TABLE t\_fujian (

fjid INT auto\_increment PRIMARY KEY NOT NULL, #附件编号(主键)

caroutmxid INT, #汽车出售记录编号(外键)

inhandoutid INT, #赠品入库编号（外键）

zscount INT #赠送数量

) ENGINE = INNODB DEFAULT charset = utf8;

#1-19汽车售后记录表(shouhou)

CREATE TABLE t\_shouhou (

shouhouid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

customid INT #客户编号（外键）

) ENGINE = INNODB DEFAULT charset = utf8;

ALTER TABLE t\_shouhou ADD CONSTRAINT caroutmxidk712\_check FOREIGN KEY (customid) REFERENCES t\_custom (customid);

#1汽车售后记录表明细(shouhoumx)

CREATE TABLE t\_shouhoumx (

shouhoumxid INT auto\_increment PRIMARY KEY NOT NULL, #编号(主键)

shouhouid INT, #售后编号（外键）

fwnr VARCHAR (100), #服务内容

shouhouflag INT, #是否收费

shmoney VARCHAR (100), #费用

shtime VARCHAR (100), #售后时间

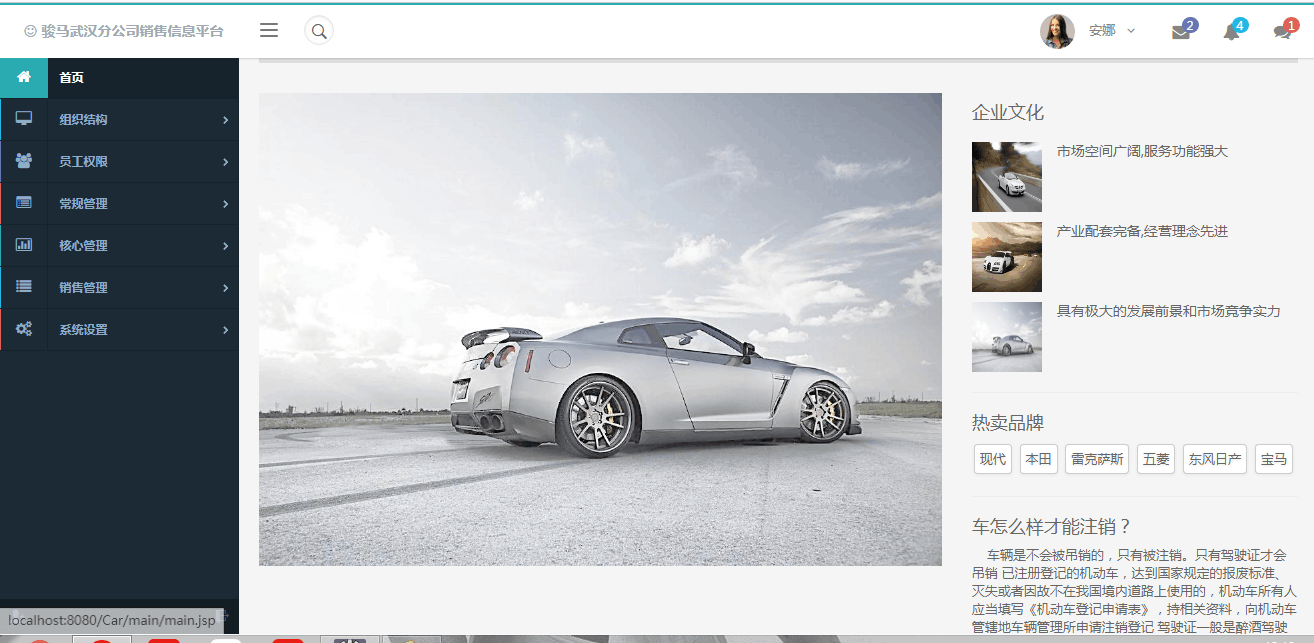
mid INT #经办人编号（外键）

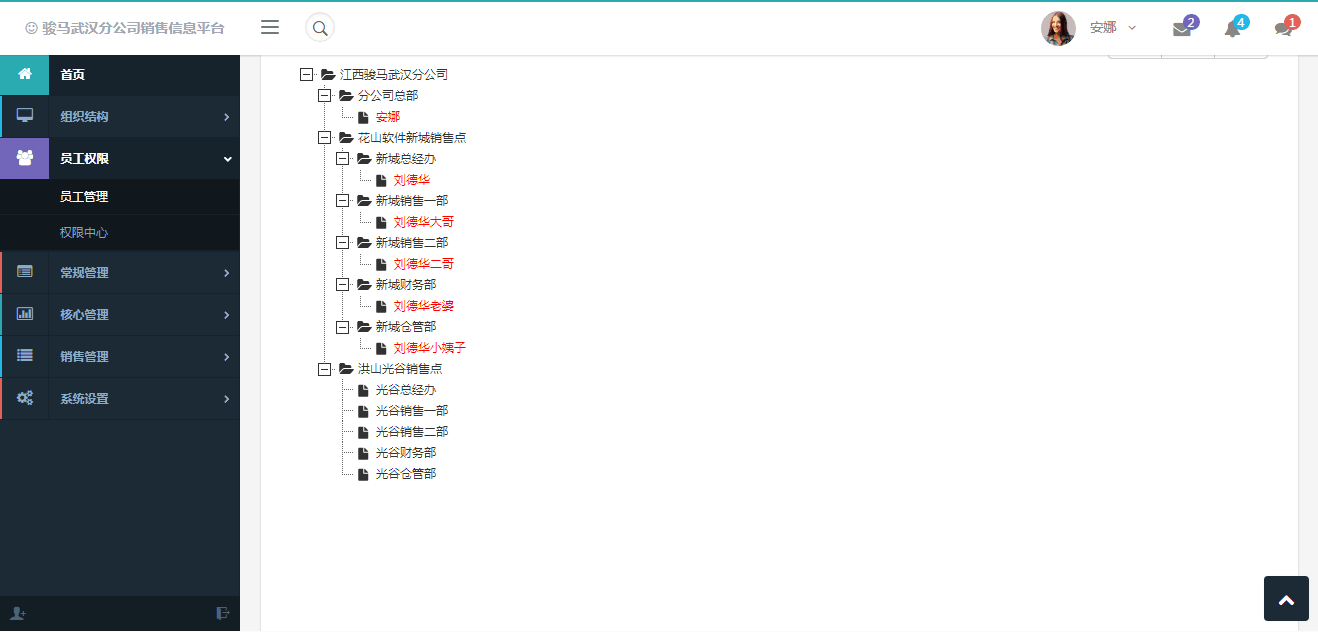
) ENGINE = INNODB DEFAULT charset = utf8;

1. **项目运行效果**

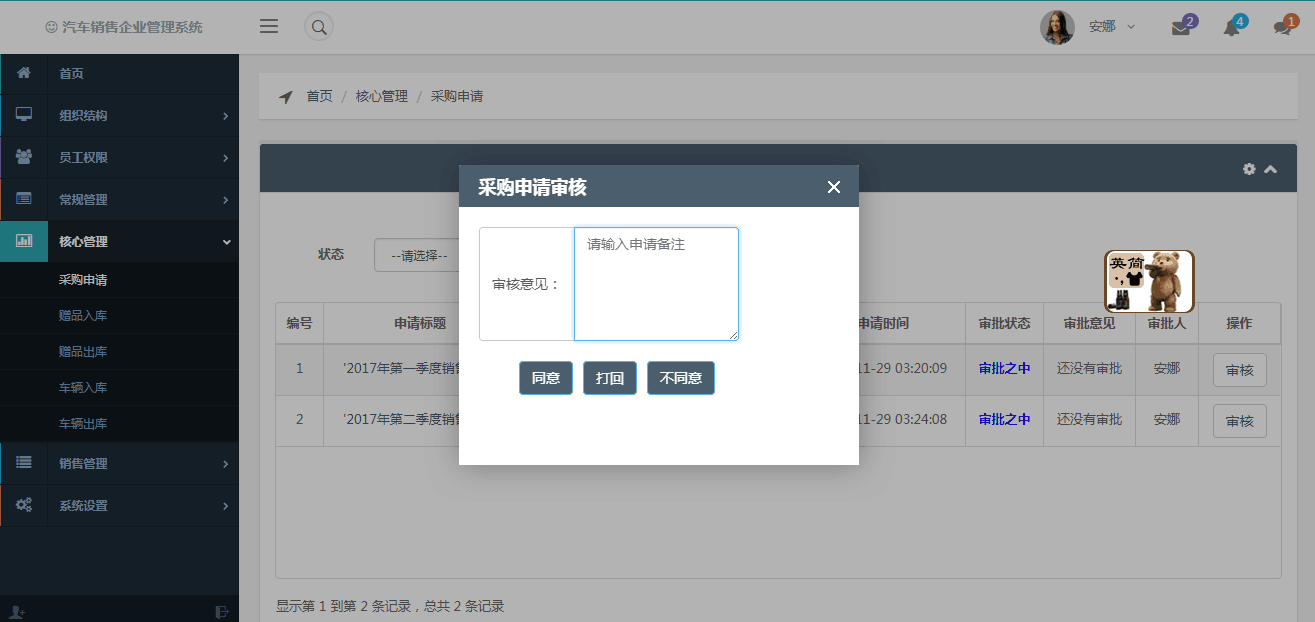
部分截图如下(详细请参考附录文件)：

主页：









1. **总结**

本次课程设计，我们组是三个人，比其他组多一个人，所以我们的项目要比别人考虑多一点。在项目中我们尽量用了比较新的知识点，当然也用了自己熟悉一点的过了时的知识点。在本项目中我们三个队数据库的设计下了重点，也用了很多精力。当然在项目后台逻辑和界面方面我们查了很多资料，也借鉴了别人很多的代码。做出来的项目可能还是很差，有很多bug但是我们每个人都很认真的对待了这一次项目设计。我们都觉得在javaEE这门课中，我们学到最多的是思想，是去学习的能力，老师教会我们怎样去学习最前沿的技术，怎样去看帮助文档，给我们把知识框架建立起来了，我们以后会更有方向，也更好的确定我们的职业方向。这次课程设计项目估计做的不理想，很多代码都是百度看别人的博客借鉴过来的，想想也有点愧对老师这学期的努力，但是我们也通过这门课程设计的制作更加；了解了javEE项目的开发流程和相关知识点。在这里我们也清醒后有这么一帮好队友，我们一起敲代码做设计，查资料相互团结解决知识难点，盲点，相互进步着。