

成绩 评定	
----------	--



绵阳城市学院

设计报告

学 期:

项 目:

指导老师:

班 级:

学 号:

姓 名:

现代技术学院
****年**月**日

目录

第 1 章 系统分析	3
第 2 章 系统设计	4
2.1 系统架构设计	4
2.2 数据库设计	4
2.2.1 概念结构设计	4
2.2.2 逻辑结构设计	6
第 3 章 系统实现	8
3.1 用户登录	8
3.2 车辆管理	9
3.3 客户买车	10
3.4 订单管理	11
3.5 员工管理	12
第 4 章 开发总结	13

第 1 章 系统分析

1.1 需求分析

1.用户登录与注册：系统提供用户登录与注册功能，用户可以使用 用户名和密码进行登录，新用户可以通过注册功能创建新的账户。

2.角色分配：系统区分客户，店员以及管理员角色。根据用户角色，系统将提供不同的功能选项。

3.车辆管理：管理员和店员可以新增和修改车辆信息，客户可以在系统中浏览车辆列表，可以在系统中下单车辆。

4.订单管理：客户可以查看订单信息，店员可以查看自己客户的订单信息，管理员可以查看所有订单信息。

5.员工管理：系统提供员工管理，管理员可以对员工信息进行修改。

6.信息修改：用户可以在系统中修改自己的个人信息，如密码、手机号等。

7.权限管理：根据用户的角色，系统提供不同的操作权限，保证系统的安全性。

以上就是汽车销售系统的主要功能。通过这些功能，我们希望能够提高汽车销售的效率，使得汽车销售变得更加方便和高效。

第 2 章 系统设计

2.1 系统架构设计

本项目采用了经典的三层架构，即表现层、业务层和数据访问层。

1.表现层：也称为用户界面层，主要负责与用户的交互。我们在这一层使用了Bootstrap框架来构建用户界面。用户通过表现层进行操作，如登录、注册、下单车辆、确认、付款等。

2.业务层：也称为逻辑层，主要负责处理业务逻辑。例如，当用户下单车辆时，业务层会检查车辆库存是否符合要求，然后将订单存储到数据库中。

3.数据访问层：主要负责与数据库的交互，包括数据的查询、插入、更新和删除等操作。同时，我们使用 Druid 连接池和 DBUtils 工具来提高数据库操作的效率。

以上就是本项目的系统架构设计。这种架构将系统分解为几个独立的层次，每个层次负责处理特定的任务，这样可以提高系统的模块化程度，简化系统的开发和维护。

2.2 数据库设计

2.2.1 概念结构设计

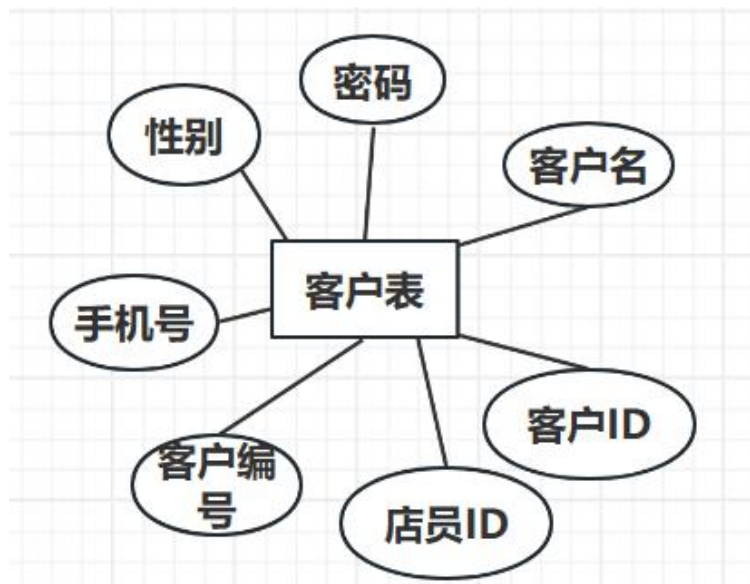


图2-1 客户信息实体属性图

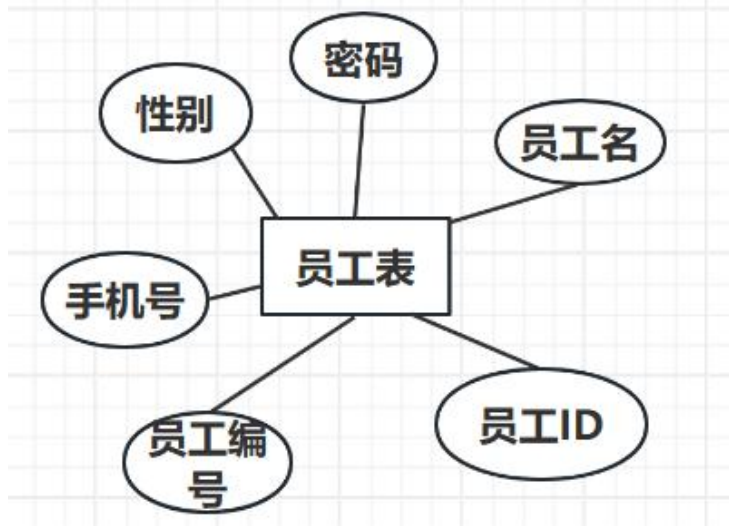


图2-2 员工信息实体属性图

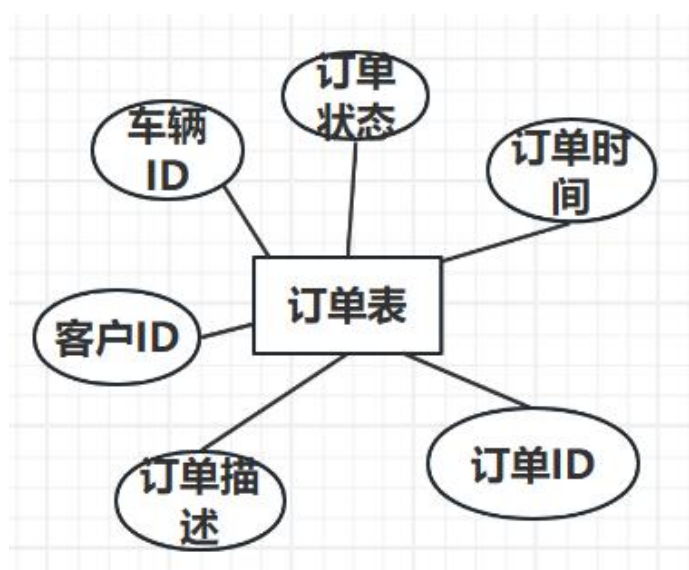


图2-3 订单信息实体属性图

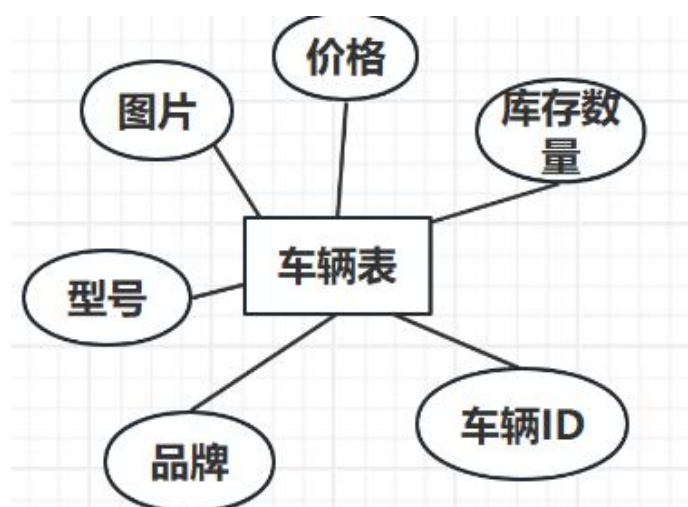


图2-4 车辆信息实体属性图

2.2.2 逻辑结构设计

表名：client

字段	类型	约束	不是null	备注
clientId	Int	PK	\	客户ID
clientNo	VARCHAR(20)	PK	\	客户编号
clientName	VARCHAR(20)	NOT NULL	\	客户名
password	VARCHAR(20)	NOT NULL		密码
sex	VARCHAR(20)			性别
phone	VARCHAR(20)			手机号
staffId	Int			员工ID

表 2-5 客户表

表名：staff

字段	类型	约束	不是null	备注
staffId	Int	PK	\	店员ID
staffNo	VARCHAR(20)	PK	\	店员编号
staffName	VARCHAR(20)	NOT NULL	\	店员名
password	VARCHAR(20)	NOT NULL		密码
sex	VARCHAR(20)			性别
phone	VARCHAR(20)			手机号

表 2-6 员工表

表名：order

字段	类型	约束	不是null	备注
orderId	Int	PK	√	订单ID
status	VARCHAR(20)	PK	√	订单状态
desc	VARCHAR(255)	NOT NULL	√	订单描述
clientId	Int	NOT NULL		客户ID
carId	Int			车辆ID
orderTime	TimeStamp			订单时间

表 2-7 订单表

表名：car

字段	类型	约束	不是null	备注
carId	Int	PK	√	车辆ID
brand	VARCHAR(40)	PK	√	品牌
model	VARCHAR(100)	NOT NULL	√	型号
count	Int	NOT NULL		库存数量
price	Int			价格（万）
pic	VARCHAR(100)			图片地址

表 2-7 车辆表

第 3 章 系统实现

3.1 用户登录



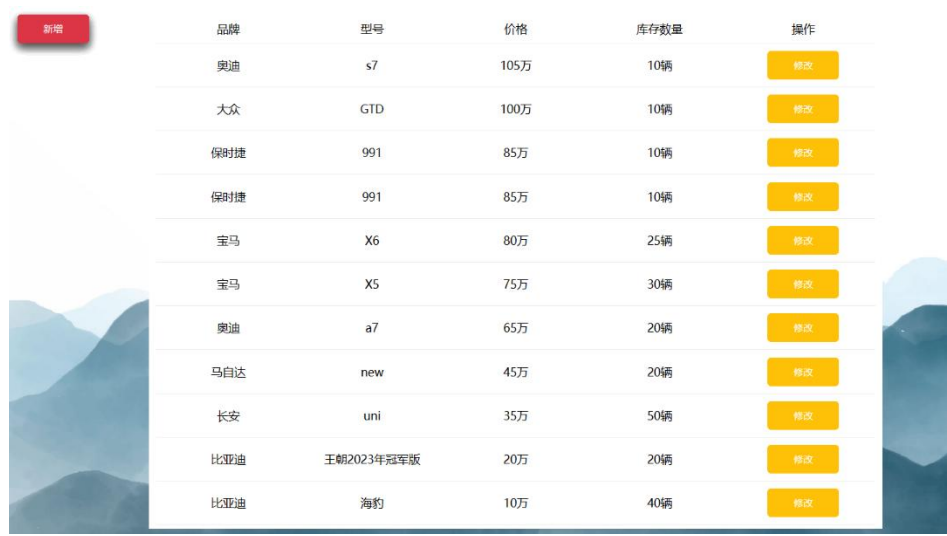
图3-1 登录界面图

核心代码截图如图 3-2 所示：

```
// 登录请求
$.ajax({
    url:"<%=path%>/login",
    data: JSON.stringify(loginDataInfo),
    dataType: 'json',
    type: 'post',
    contentType: 'application/json',
    success:function (data){
        if (data.code === 0){
            alert(data.msg)
        } else {
            window.location.replace("<%=path%>/View/clientCarList.jsp")
        }
    },
    error: function () {
        alert("请求失败")
    }
})
```

图3-2 登录核心代码图

3.2 车辆管理



新增	品牌	型号	价格	库存数量	操作
	奥迪	s7	105万	10辆	修改
	大众	GTD	100万	10辆	修改
	保时捷	991	85万	10辆	修改
	保时捷	991	85万	10辆	修改
	宝马	X6	80万	25辆	修改
	宝马	X5	75万	30辆	修改
	奥迪	a7	65万	20辆	修改
	马自达	new	45万	20辆	修改
	长安	uni	35万	50辆	修改
	比亚迪	王朝2023年冠军版	20万	20辆	修改
	比亚迪	海豹	10万	40辆	修改

图3-3 车辆界面图

核心代码截图如图 3-4 所示：

```
// 设置编码格式
req.setCharacterEncoding("UTF-8");
res.setCharacterEncoding("UTF-8");

// 解析请求
Car car = JSON.parseObject(Utils.parse(req), Car.class);

// 设置响应头的编码
res.setHeader( s: "Content-Type", s1: "application/json;charset=utf-8");

// 调用dao层方法
try {
    carDao.update(car);
    Utils.returnJson(res, Result.success());
} catch (Exception e) {
    Utils.returnJson(res, Result.error( msg: "修改失败"));
}
```

图3-4 车辆管理核心代码图

3.3 客户买车



图3-5 客户买车界面图

核心代码截图如图 3-6 所示：

```
<div class="main">
  <div class="img-contain"></div>
  <div class="main-content">
    <div style="..."><%=car.getModel()%></div>
    <div class="main-content-piece">
      <div>价格:</div>
      <div style="..."><%=car.getPrice()%>0000元</div>
    </div>
    <div class="main-content-piece">
      <div>库存:</div>
      <div style="..."><%=car.getCount()%>辆</div>
    </div>
    <div class="main-content-piece">
      <div></div>
      <div style="...">
        <button id="buyBtn" >
          立即购买
        </button>
      </div>
    </div>
  </div>
</div>
```

图3-6 客户买车核心代码图

3.4 订单管理

订单详情	订单时间	订单状态	操作
客户：客户1; 店员：zhangsan; 车辆：保时捷 991	2023-12-14T18:41:41	待确认	取消订单



图3-7 订单管理界面图

核心代码截图如图 3-8 所示：

```
<div>
  <!-- 只有客户有支付权限 -->
  <%
    if ("待付款".equals(order.getStatus()) && "client".equals(userName)) {
  %>
  <button class="button-color-yellow" onclick="payOrder(<%=order.getOrderId()%>)">
    立即支付
  </button>
  <%
    }
  %>

  <!-- 只有员工有确认订单权限 -->
  <%
    if ("待确认".equals(order.getStatus()) && !"client".equals(userName)) {
  %>
  <button class="button-color-yellow" onclick="affirm(<%=order.getOrderId()%>)">
    确认订单
  </button>
  <%
    }
  %>
</div>
```

图3-8 订单管理核心代码图

3.5 员工管理

新增	用户名	姓名	密码	性别	电话	操作
	000001	zhangsan	123456	女	12355556666	修改
	000002	lisi	123456	女	12344445555	修改
	staff1	店员1	1234	男	12344445555	修改
	staff5	市高业业东	1234	男	18121353620	修改

图3-9 员工管理界面图

核心代码截图如图 3-10 所示：

```
function modifyEmploy(id){
    console.log(id)
    $.ajax({
        url: "<%=path%>/detail?id=" + id,
        dataType: 'json',
        type: 'get',
        contentType: 'application/json',
        success: function (data) {
            if (data.code === 0) {
                alert(data.msg)
            } else {
                $("#modify-name").val(data.data.staffName)
                $("#modify-password").val(data.data.password)
                $("#modify-sex").val(data.data.password)
                $("#modify-phone").val(data.data.phone)
                modifyInfo.staffNo = data.data.staffNo;
                modifyInfo.staffId = data.data.staffId;
            }
        },
        error: function () {
            alert("修改失败")
        }
    })
}
```

图3-10 员工管理核心代码图

第 4 章 开发总结

学生 1:

在开发汽车销售系统的期末作业中，使用 Java 和 MySQL 构建系统是一次富有挑战性而又充实的经历。通过 Java 编程语言，我能够实现系统的后端逻辑，处理业务逻辑和数据库交互，同时充分利用 Java 的面向对象特性提高代码的可维护性。

MySQL 数据库的运用为数据的存储和管理提供了强大支持。通过设计合理的数据库结构，我能够高效地存储汽车信息、用户数据和销售记录。同时，通过利用 MySQL 的事务处理机制，确保数据的一致性和完整性。

在项目中，团队协作也是不可或缺的一环。通过合理的任务分工和沟通，我们能够充分发挥各自的优势，提高开发效率。总的来说，通过这次项目，我更加深入理解了 Java 和 MySQL 在实际项目中的应用，也更加熟练地掌握了团队协作的重要性。

学生 2:

在开发汽车销售系统的期末项目中，运用 Java 和 MySQL 技术带来了丰富的学习体验。Java 的多线程编程和异常处理为系统的并发性和稳定性提供了有力支持，使得用户能够流畅地进行多项操作。通过使用 MySQL 数据库，我深入了解了数据库设计和查询优化，确保系统能够高效地处理大量数据。

在开发过程中，前端采用了 Bootstrap，这使得用户界面更加友好和直观。通过设计交互式的界面，用户能够轻松浏览汽车信息、进行购车操作。同时，通过引入设计模式和代码重构，我提高了代码的可读性和可维护性，使得未来的系统升级更为便捷。

这次项目锻炼了我的团队协作能力，了解了如何有效地分工合作，从而提高项目的整体质量。通过这次实践，我不仅加深了对 Java 和 MySQL 的理解，也更自信地应对了复杂项目的挑战。

学生 3:

在汽车销售系统的开发中，Java 和 MySQL 的应用为期末项目增色不少。通过 Java 编程，我深入研究了系统的业务逻辑和后端处理，借助多态和继承等特性提高了代码的灵活性。MySQL 数据库的运用使我更加熟练地处理数据存储和检索，通过索引和优化查询语句，提高了系统的响应速度。

前端方面，采用了 Bootstrap 技术，这为系统用户界面的设计提供了更多可能性。通过实现动态可视化效果和用户友好的交互，我让用户更直观地了解汽车信息，提高了系统的易用性。

在团队协作方面，我们通过版本控制工具有效地协同工作，充分利用各自的专长，高效地完成各个模块。这次项目让我对软件开发的整个流程有了更清晰的认识，同时也提升了解决问题和团队协作的能力。通过这次实践，我在 Java 和 MySQL 应用上迈出了坚实的一步。

学生 4:

在汽车销售系统的开发中，Java 和 MySQL 的结合为期末项目提供了强大的技术支持。通过 Java 的面向对象编程，我能够将系统划分为模块，提高了代码的可维护性和可扩展性。使用 MySQL 作为数据库管理系统，我深入研究了数据表设计和优化查询语句，确保系统在处理大量数据时保持高效性。

前端方面采用了 Bootstrap，这为用户提供了直观且友好的界面。通过设计吸引人的用户界面和实现交互性功能，我致力于提升用户体验，使其更容易浏览汽车信息和完成购车流程。

在项目开发中，团队的协作是成功的关键。通过分工明确、及时沟通，我们能够克服挑战，高效推进项目进度。整个过程让我深感团队合作的重要性，并对 Java 和 MySQL 的实际应用有了更深入的理解。通过这次项目，我不仅提高了技术水平，还培养了团队协作和问题解决的能力。