商品销售平台方案设计

(一)接口设计

1. 用户信息及管理

为了实现用户注册登录功能、基本资料管理、订单的管理、相关的收货地址、针对商品的评价等功能，应当有一个用户实体类和其相应的方法：

#### **User类（父类）**

成员变量：

private Integer userId ; //User的id

private String username ; //用户名

private String password ; //用户密码

private String email ; //用户邮箱

private String phone ; //用户联系方式

Private String power; //用户种类（注册用户、管理员）

相关方法：

getters()和setters()方法，用来得到和设置成员变量的值

#### **Member类（注册用户）**

为User的子类，可添加特有的属性，如评价

成员变量：

private Integer userId ; //User的id

private String username ; //用户名

private String password ; //用户密码

private String email ; //用户邮箱

private String address ; //用户收货地址

private String phone ; //用户联系方式

private String comments; //用户评价

private int score;//用户积分

private int scorevalidtime; //积分有效期

相关方法：

getters()和setters()方法，用来得到和设置成员变量的值

#### **Admin类（管理员）**

为User的子类，可添加特有的属性

成员变量：

private Integer userId ; //User的id

private String username ; //用户名

private String password ; //用户密码

private String email ; //用户邮箱

private String phone ; //用户联系方式

Private String power; //用户种类（注册用户、管理员）

相关方法：

getters()和setters()方法，用来得到和设置成员变量的值

**IUserDAO接口**

void save(User user) ;//添加一个user

List<User> findAllUser() ;//查找出所有的会员

User findUserById(Integer userId) ;//根据某个id查找对应的user

void modify(User user) ;//修改相应的user

void delete(User user) ;//删除选中的user

User findUserByName(String username) ;//根据用户名称查找user

List findUserByPage(Page page) ;//根据页数号码列出用户

void modifyPasswordByUid(Integer uid) ;//根据id修改相应的用户密码

**UserDAO类**

实现IUserDAO接口。

**UserService类**

成员变量： IUserDAO userDao = new UserDAO() ;

相关方法：

public void save(User user){ }

public List<User> findAllUser(){}

public User findUserById(Integer userId){}

public User login(User user){}

public void modify(User user) {}

public void delete(User user){}

public User findUserByName(String username){}

public List findUserByPage(Page page){}

public void modifyPasswordByUid(Integer uid){}

1. 商品展示及管理

商品实体类和其相应的方法：

**Goods类**

private Integer gid ;//商品id

private String gname ;//商品名称

private Float price ;//商品价格

private String description ;//商品相关描述

private Integer cid ;//类别id

private String image ;//商品图片

**GoodsDAO接口**

void saveGoods(Goods goods) ;//添加商品

List listAllGoods(Page page) ;//列出所有的商品

List findGoodsByGid(Integer cid,Page page) ;//根据商品id找出相对的商品

void modifyGoodsByGid(Goods goods,Integer gid) ;//根据商品id修改某商品的信息

void deleteGoodsByGid(Integer gid) ;//根据商品id删除此商品

public void queryGoods() throws Exception;//使用sorlJ实现站内搜索

**GoodsDAO类**

实现了GoodsDAO接口的所有方法。

**GoodsService类**

成员变量**：**IGoodsDAO goodsDao = new GoodsDAO() ;

相关方法：

public void saveGoods(Goods goods){}

public List listAllGoods(Page page){}

public List findGoodsByGid(Integer cid,Page page{}

public void modifyGoodsByGid(Goods goods,Integer gid){}

public void deleteGoodsByGid(Integer gid){}

public void queryGoods() throws Exception{}

**SvImage类**

用来实现为商品上传图片，用到apache的commons-fileupload和commons-io组件，这是在Java的Web开发中最常用的文件上传组件。上传的图片会存放在服务器的目录下。

1. 购物车模块

**Car类**

private Integer carId ;//购物车id

private String gname ;//放入购物车某商品的名称

private Float price ;//放入购物车某商品的价格

private Integer userId ;//此购物车属于那个user的用户的id

**CarDAO接口**

void saveCar(Car car) ;//用户决定购买某件商品

void deleteCar();//把某件商品从购物车中移出来

List<Car> listCarsByUid(Integer uid) ;//列出某用户的购物车

void deleteCarByCarId(Integer carId) ;根据carid把某件商品从购物车中移出来

**CarDAO类**

实现了CarDAO接口的所有方法。

**CarService类**

成员变量：ICarDAO carDao = new CarDAO() ;

相关方法：

public void saveCar(Car car){}

public List listCarByUid(Integer uid){}

public void deleteCarByCarId(Integer carId){}

1. 订单模块

**Orders类**

成员变量：

private Integer orderId ;//订单id

private Integer userId ;//下此订单的用户id

private String gname ;//此订单中某商品的名称

private Float price ;//此订单中某商品的价格

private Date orderDate ;//订单日期

private String username ;//下此订单的用户名称

private String state ;//订单的状态

方法：

set和get方法。

**IOrdersDAO 接口**

void saveOrder(Orders order) ;//保存订单

List<Orders> listOrdersByUid(Integer userId) ;//列出某用户的订单

List<Orders> listAllOrders() ;//列出所有订单

void handleOrderByOrderId(String state,Integer orderId) ;//处理订单

void deleteOrderByOrderId(Integer orderId) ;//删除某订单

List<Orders> statisticOrdersBydate(Datetime from\_datetime,Datetime to\_datetime) ;

//根据日期或者日期统计订单

void commentOrder(Orders order);//评价订单

**OrdersDAO 类**

实现了IOrderDAO接口的所有方法。

**OrderService类**

成员变量： IOrdersDAO orderDao = new OrderDAO() ;

相关方法：

public void saveOrder(Orders order){}

public List<Orders> listOrdersByUid(Integer userId){}

public List<Orders> listAllOrder(){}

public void handleOrderByOrderId(String state,Integer orderId){ }

public void deleteOrderByOrderId(Integer orderId) {}

public List<Orders> statisticOrdersBydate(Datetime from\_datetime,Datetime

to\_datetime) {}

public void commentOrder(Orders order);{}

1. 积分模块

**Increase\_Score接口**

Void getScore(int score) //赚取积分

**Increaseby\_order类**

通过下订单赚取积分，实现Increase\_Score接口中的方法，并在方法中定义积分获取规则。

**Increaseby\_sign类**

通过每日签到赚取积分，实现Increase\_Score接口中的方法，并在方法中定义积分获取规则。

**Increaseby\_evaluate类**

通过评论赚取积分，实现Increase\_Score接口中的方法，并在方法中定义积分获取规则。

**Decrease\_Score 接口**

Void reduceScore(int score) //消费积分

**Reduceby\_deduct类**

通过抵扣订单金额消费积分，实现Decrease\_Score接口中的方法，并在方法中定义积分兑换规则。

**Reduceby\_coupon类**

通过兑换优惠券消费积分，实现Decrease\_Score接口中的方法，并在方法中定义积分兑换规则。

**Reduceby\_redem类**

通过积分换购消费积分，实现Decrease\_Score接口中的方法，并在方法中定义积分兑换规则。

1. 连接数据库

**DataBase类**

方法：

public static Connection getConn() ;//取得用于连接数据库的Connection。

public static void close(Connection conn);//关闭Connection。

public static void close(Statement stmt);//关闭Statement。

public static void close(ResultSet rs);//关闭ResultSet。

1. 分页展示

**Page类**

成员变量：

private Integer pageNo;//当前是第几页

private Integer pageSize;//当前总共多少页

private Long totalPage ;//总共多少页

private String url;//页面地址

private String method ;//相关方法名

private String index ;//用来作标记的

方法：

getters()和setters()方法

1. 广告新闻模块

**Advers类**

成员变量：

private Integer adverId ;//广告id

private Date adverDate ;//广告发布日期

private String advercategory ;//广告分类

方法：

set和get方法。

**IAdversDAO 接口**

ViewAdver listAdversByid(Integer adverId) ;//根据广告id列出广告

ViewAdver listAdversBycategory(String advercategory) ;//根据广告分类列出广告

ViewAdver listAdversBydate(Date adverDate) ;//根据广告发布日期列出广告

ViewAdver listAllAdvers() ;//列出所有广告

void addAdver(Orders order) ;//发布广告

void editAdver(Orders order) ;//编辑广告

void deleteAdverByAId(Integer adverId) ;//删除广告

**AdversDAO 类**

实现了IAdversDAO接口的所有方法。

**AdverService类**

成员变量： AdversDAO adversDao = new AdversDAO () ;

相关方法：

public ModelAndView listAdversByid(Integer adverId) {}

public ViewAdver listAdversBycategory(String advercategory) {}

public ViewAdver listAdversBydate(Date adverDate) {}

public ModelAndView listAllAdvers() {}

public addAdver(Advers adver) {}

public void editAdver(Advers adver) {}

public void deleteAdverByAId(Integer adverId) {}

**News类**

成员变量：

private Integer newsId ;//新闻id

private Date newsDate ;//新闻发布日期

private String newscategory;//新闻类别

方法：

set和get方法。

**INewsDAO接口**

继承了IAdversDAO 接口，并且新增了void topAdver(Advers adver) ;//置顶新闻

**NewsDAO类**

实现了INewsDAO接口中所有的方法。

**NewService类**

成员变量： NewsDAO newsDao = new NewsDAO();

相关方法：

public ModelAndView listNewsByid(Integer newsId) {}

public ViewAdver listAdversBycategory(String newscategory) {}

public public ViewAdver listAdversBydate(Date adverDate) {}

public ModelAndView listAllNews() {}

public addNews(News news) {}

public void editNews(News news) {}

public void deleteNewsBynId(Integer newsId) {}

public void topAdver(Advers adver) {}

1. 支付模块

**PayMent接口**

String PayOrders(Orders order) ;//支付订单

**PayBy\_zfb类**

通过支付宝支付订单，实现PayMent接口中的方法。

**PayBy\_wx类**

通过微信支付订单，实现PayMent接口中的方法。

1. 客服模块

实现在线客服功能，将信息存储到Application域里面。然后使用json格式的数据进行前后台交互。

具体实现是使用两个Action，分别为用户和管理员服务。

**AdminMessageAction类**

public class AdminMessageAction extends ActionSupport

成员变量：

private String to;//接受用户

private String result; //返回消息

private String content;//消息内容

private String username;//昵称

方法：

set、get方法。

public String execute();//执行

public String send();//发送消息

**AdminListAction类**

public class AdminListAction extends ActionSupport

成员变量：

private List<Admin> list;

方法：

public List<Admin> getList();//获取管理员列表

public String execute();//得到当前的管理员

1. 数据库设计

**member表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| userId | Int(11) | YES | NO | 用户id |
| username | varchar(40) |  | YES | 用户名 |
| password | varchar(20) |  | YES | 密码 |
| email | varchar(40) |  | YES | 用户邮箱 |
| address | varchar(255) |  | YES | 送货地址 |
| phone | varchar(50) |  | YES | 联系方式 |
| power | varchar(10) |  | YES | 用户种类 |
| score | Int(8） |  | YES | 账户积分 |
| scorevalidtime | Int(8） |  | YES | 积分有效期 |

**admin表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| userId | Int(11) | YES | NO | 管理员id |
| username | varchar(40) |  | YES | 管理员用户名 |
| password | varchar(20) |  | YES | 密码 |
| email | varchar(40) |  | YES | 管理员邮箱 |
| phone | varchar(50) |  | YES | 联系方式 |
| power | varchar(10) |  | YES | 用户种类 |

**goods表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| gid | Int(11) | YES | NO | 商品id |
| gname | varchar(255) |  | YES | 商品名称 |
| price | Decimal(6,2) |  | YES | 商品价格 |
| image | varchar(255) |  | YES | 图片名称 |
| description | Varchar(100) |  | YES | 描述 |
| cid | double |  | YES | 商品类别 |

**car表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| car\_id | Int(11) | YES | NO | 购物车id |
| user\_id | Int(11) |  | YES | 用户id（外键） |
| gname | varchar(255) |  | YES | 商品名称 |
| price | datetime |  | YES | 商品价格 |

**orders表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| order\_id | Int(11) | YES | NO | 订单项id |
| user\_id | Int(11) |  | NO | 用户id（外键） |
| gname | Varchar(20) |  | YES | 商品名称 |
| price | Decimal(6,2) |  | YES | 商品价格 |
| order\_date | Datetimr |  | YES | 下单日期 |
| uename | Varchar(20) |  | YES | 用户名称 |
| state | Varchar(20) |  | YES | 订单状态 |

**Advers表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| adverId | Int(11) | YES | NO | 广告id |
| adverDate | datetime |  | YES | 发布日期 |
| advercategory | varchar(20) |  | YES | 广告分类 |

**News类**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否为主键 | 是否可为空 | 描述 |
| newsId | Int(11) | YES | NO | 新闻id |
| newsDate | datetime |  | YES | 新闻发布日期 |
| newscategory | varchar(20) |  | YES | 新闻类别 |