**北京联合大学毕业设计（论文）期中报告**

题目： 易买网购物商城的设计与实现

专业： 计算机科学与技术 指导教师： 孙 悦

学院： 信息学院 学号： 2012080332077

班级： 计算机1303B 姓名： 宁 浩 声

1. 功能模块图

二、已完成的主要内容及所占比重：70%

1、完成数据流图；

2、完成系统概要设计（H图）；

3、完成详细设计（人机界面、数据库）；

4、完成了前后端代码的编写，调试；

5、完成了部分论文的撰写。

三、待完成主要内容及所占比重：30%

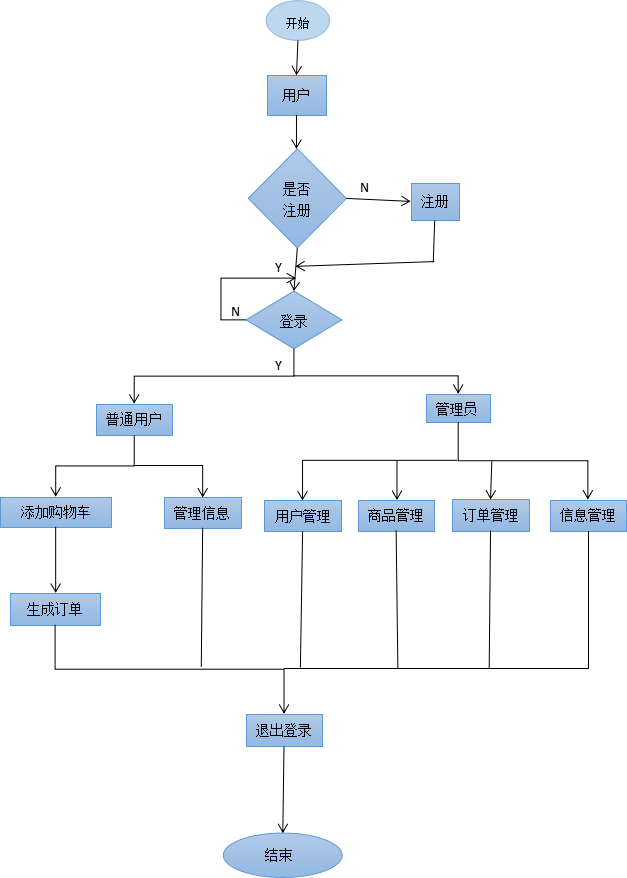
1、软件测试，完善代码。

2、完成论文的撰写。

四、存在的问题：

1、开发过程中存在死机现象，程序代码还待优化

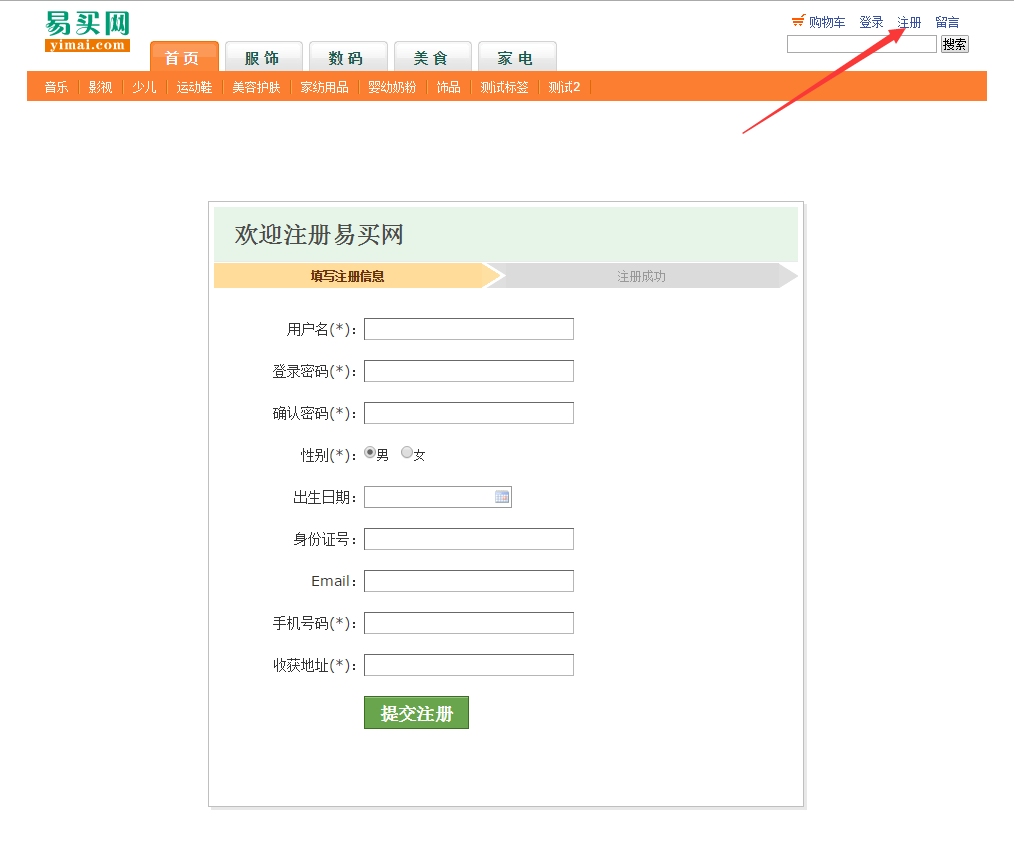
五、完成内容：



程序流程图

**1.用户注册、登录**

1.1注册效果图：



相关说明：

点击右上角“注册”，进入用户注册页面，用户输入用户名，密码，性别，出生日期，Email，手机号码，收货地址，然后点击“提交注册”，即完成用户注册；

关键代码实现：

/\*\*

\* 用户注册

\* @return

\* @throws Exception

\*/

public String register()throws Exception{

userService.saveUser(user);

return "register\_success";

}

1.2登录效果图：



说明：右上角点击“登录”，则进入用户登录页面，输入用户名和密码以及验证码，点击登录按钮。假如用户名和密码正确，则进入系统。假如错误，则登录失败；

关键代码实现：

/\*\*

\* 用户登录

\* @return

\* @throws Exception

\*/

public String login()throws Exception{

HttpSession session=request.getSession();

User currentUser=userService.login(user);

if(!imageCode.equals(session.getAttribute("sRand"))){

error="验证码错误！";

if(user.getStatus()==2){

return "adminError";

}else{

return ERROR;

}

}else if(currentUser==null){

error="用户名或密码错误！";

if(user.getStatus()==2){

return "adminError";

}else{

return ERROR;

}

}else{

session.setAttribute("currentUser", currentUser);

}

if(user.getStatus()==2){

return "adminLogin";

}else{

return "login";

}

}

## 2购物功能实现

效果图：



说明：

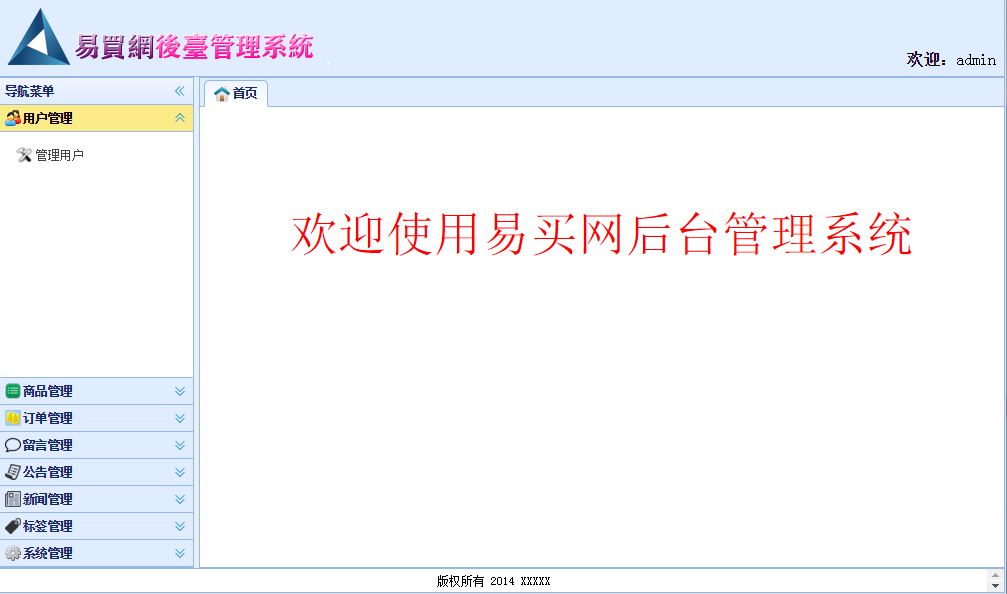
用户登录后，可以自行选择喜欢的商品，放入购物车内，然后点击“购物车”，则进入购车界面，该页面显示所有购买的商品，每条记录显示 商品名称，商品单价，购买数量，以及总金额，用户可以自行修改购买数量，也可以删除商品。点击“结算”，即可完成购买。

# 3后台系统设计实现

# 3.1管理员登录功能

管理员在浏览器地址栏输入：<http://localhost:8080/Ebuy/admin/login.jsp>登录后台



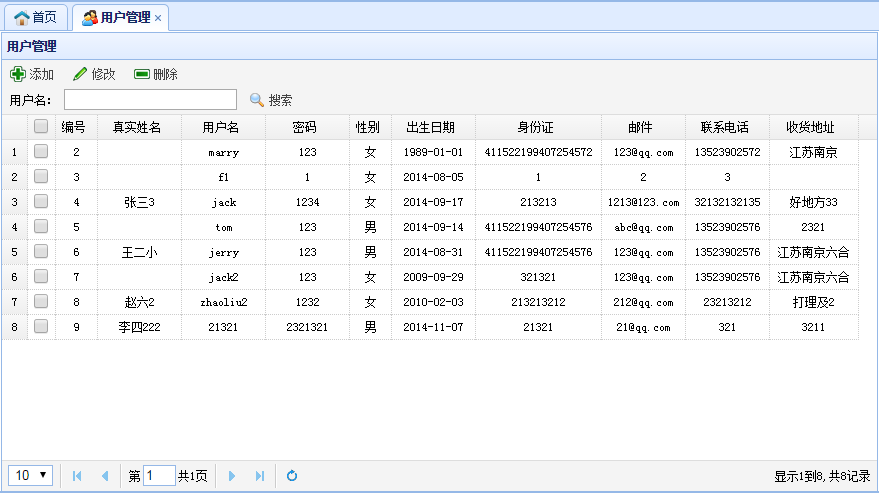


说明：

管理员输入正确的用户名和密码，点击“登录”按钮，则进入系统后台主界面；

3.2用户管理

点击“管理用户”菜单按钮，显示：



说明：

用户管理模块包括用户信息的查询，用户添加，用户修改，用户删除；

关键代码实现：

添加或者修改用户信息：

@Override

public void saveUser(User user) {

// TODO Auto-generated method stub

baseDAO.merge(user);

}

删除用户信息：

@Override

public void delete(User user) {

// TODO Auto-generated method stub

baseDAO.delete(user);

}

分页查询获取所有用户信息;

@Override

public List<User> findUserList(User s\_user, PageBean pageBean) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("from User");

if(s\_user!=null){

if(StringUtil.isNotEmpty(s\_user.getUserName())){

hql.append(" and userName like ? ");

param.add("%"+s\_user.getUserName()+"%");

}

}

hql.append(" and status=1");

if(pageBean!=null){

return baseDAO.find(hql.toString().replaceFirst("and", "where"), param, pageBean);

}else{

return null;

}

}

@Override

public Long getUserCount(User s\_user) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("select count(\*) from User");

if(s\_user!=null){

if(StringUtil.isNotEmpty(s\_user.getUserName())){

hql.append(" and userName like ? ");

param.add("%"+s\_user.getUserName()+"%");

}

}

hql.append(" and status=1");

return baseDAO.count(hql.toString().replaceFirst("and", "where"), param);

}

## 3.3 商品管理

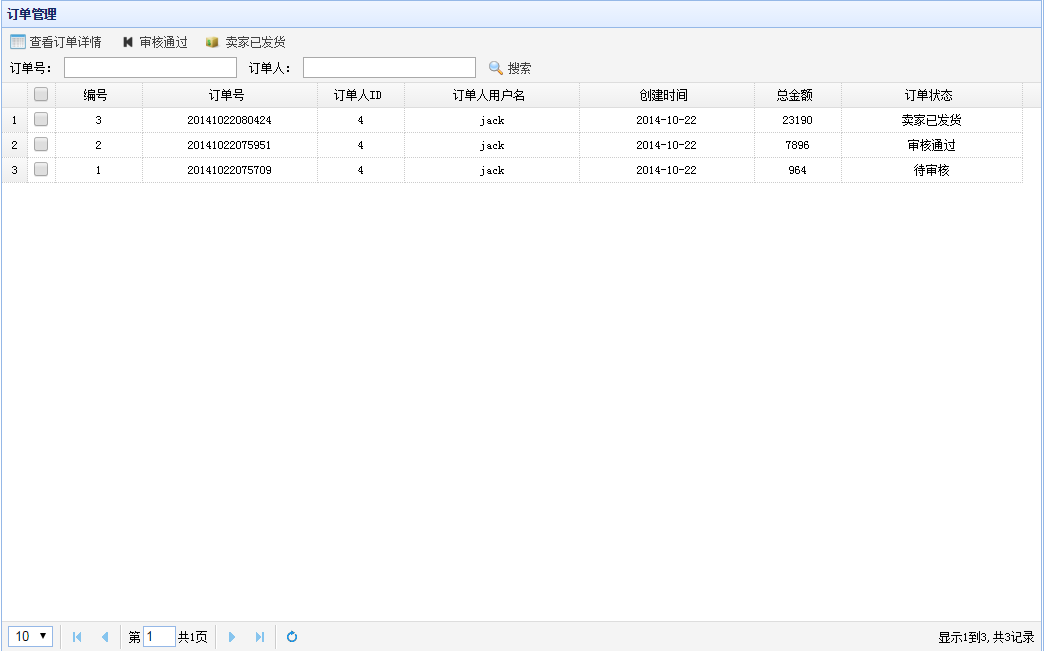
点击“商品管理”菜单按钮，显示：



说明：商品管理模块里，有商品分页显示功能，按照商品名称查询功能，以及商品添加，商品修改，商品删除，设置热卖商品，设置特价商品；

## 3.4 订单管理

点击“管理订单”菜单按钮，显示：



说明：进入订单管理界面，可以通过订单号，订单人来查询订单，可以查看订单，审核订单；

关键代码实现：

分页查询订单功能实现

@Override

public List<Order> findOrder(Order s\_order, PageBean pageBean) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("from Order");

if(s\_order!=null){

if(s\_order.getUser()!=null&&s\_order.getUser().getId()!=0){

hql.append(" and user.id=?");

param.add(s\_order.getUser().getId());

}

if(s\_order.getUser()!=null&&StringUtil.isNotEmpty(s\_order.getUser().getUserName())){

hql.append(" and user.userName like ?");

param.add("%"+s\_order.getUser().getUserName()+"%");

}

if(StringUtil.isNotEmpty(s\_order.getOrderNo())){

hql.append(" and orderNo like ?");

param.add("%"+s\_order.getOrderNo()+"%");

}

}

hql.append(" order by createTime desc");

if(pageBean!=null){

return baseDAO.find(hql.toString().replaceFirst("and", "where"), param,pageBean);

}else{

return baseDAO.find(hql.toString().replaceFirst("and", "where"), param);

}

}

查看订单功能实现：

@Override

public Order getOrderById(int id) {

return baseDAO.get(Order.class, id);

}

审核订单和修改订单状态功能实现：

@Override

public void updateOrderStatus(int status, String orderNo) {

// TODO Auto-generated method stub

List<Object> param=new LinkedList<Object>();

String hql="update Order set status=? where orderNo=?";

param.add(status);

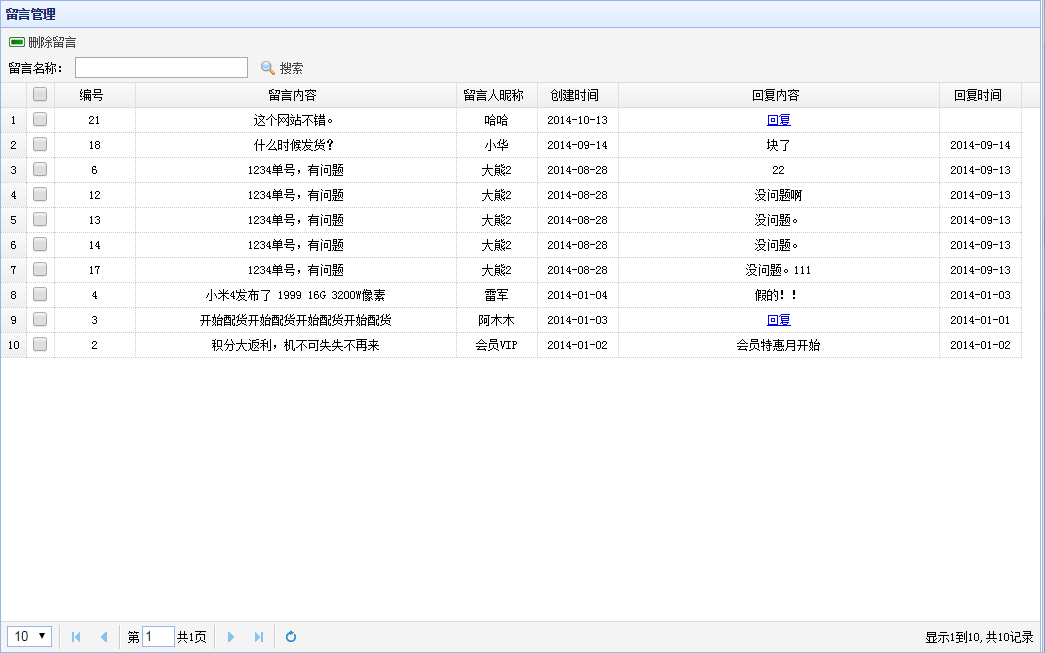
param.add(orderNo);

baseDAO.executeHql(hql, param);

}

## 3.5 留言管理

点击“留言”菜单按钮，显示：



说明：进入留言管理页面，有留言搜索功能，留言回复功能，还有留言删除功能；

关键代码实现：

分页查询留言信息：

@Override

public List<Comment> findCommentList(Comment s\_Comment, PageBean pageBean) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("from Comment");

if(s\_Comment!=null){

if(StringUtil.isNotEmpty(s\_Comment.getContent())){

hql.append(" and content like ?");

param.add("%"+s\_Comment.getContent()+"%");

}

}

hql.append(" order by createTime desc ");

if(pageBean!=null){

return baseDAO.find(hql.toString().replaceFirst("and", "where"), param, pageBean);

}else{

return null;

}

}

回复留言功能实现：

@Override

public void saveComment(Comment comment) {

baseDAO.merge(comment);

}

@Override

public Comment getCommentById(int commentId) {

return baseDAO.get(Comment.class, commentId);

}

删除留言功能实现：

@Override

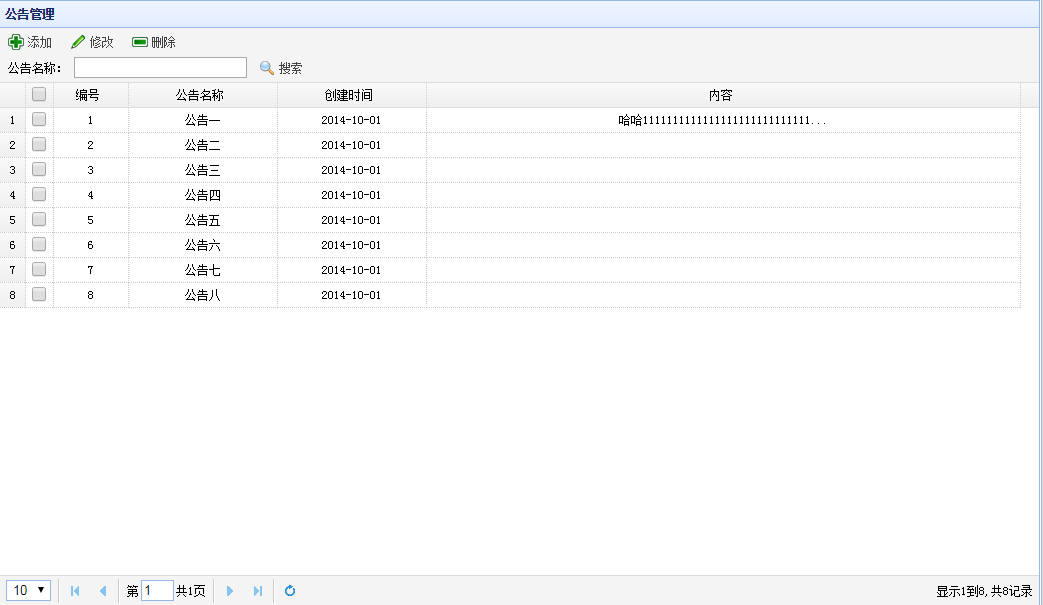
public void delete(Comment comment) {

baseDAO.delete(comment);

}

## 3.6 公告管理

点击“公告管理”菜单按钮，显示：



说明：进入公告管理主界面，有公共查询，公告添加，修改，删除功能；

关键代码实现

公告信息添加和修改功能实现：

公告信息删除功能实现：

@Override

public void saveNotice(Notice notice) {

// TODO Auto-generated method stub

baseDAO.merge(notice);

}

@Override

public void delete(Notice notice) {

// TODO Auto-generated method stub

baseDAO.delete(notice);

}

分页查询公告信息：

@Override

public List<Notice> findNoticeList(Notice s\_notice, PageBean pageBean) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("from Notice");

if(s\_notice!=null){

if(StringUtil.isNotEmpty(s\_notice.getTitle())){

hql.append(" and title like ?");

param.add("%"+s\_notice.getTitle()+"%");

}

}

if(pageBean!=null){

return baseDAO.find(hql.toString().replaceFirst("and", "where"), param, pageBean);

}else{

return null;

}

}

@Override

public Long getNoticeCount(Notice s\_notice) {

List<Object> param=new LinkedList<Object>();

StringBuffer hql=new StringBuffer("select count(\*) from Notice");

if(s\_notice!=null){

if(StringUtil.isNotEmpty(s\_notice.getTitle())){

hql.append(" and title like ?");

param.add("%"+s\_notice.getTitle()+"%");

}

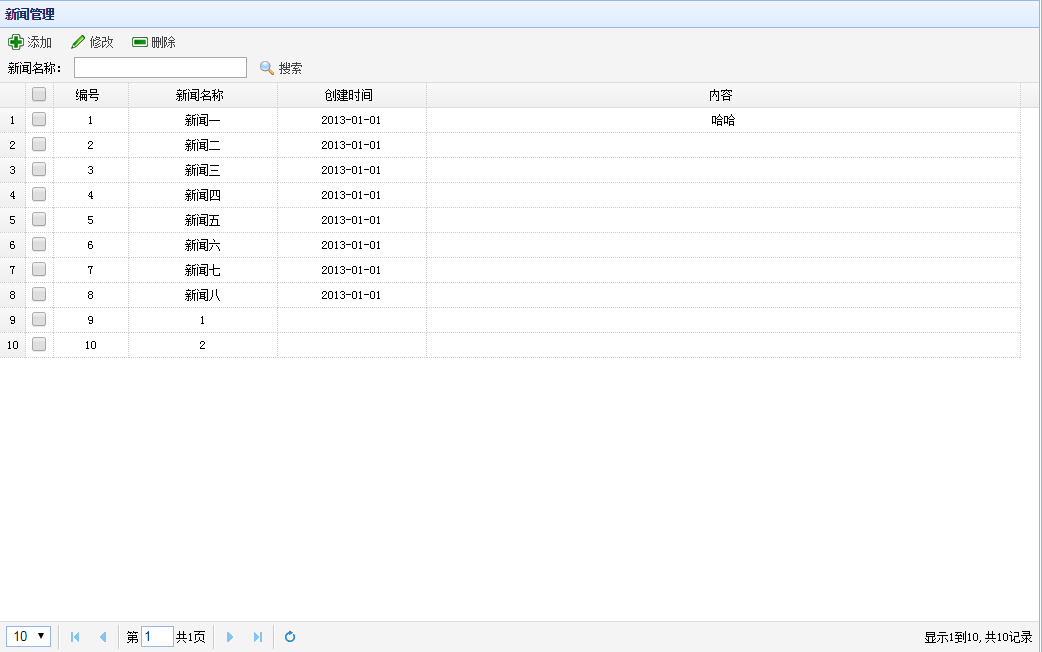
}

return baseDAO.count(hql.toString().replaceFirst("and", "where"), param);

}

## 3.7 新闻管理

点击“新闻管理”菜单按钮，显示：



说明：进入新闻管理页面，可以查询新闻信息，以及添加，修改，删除新闻；

以上是中期进度报告，接下来的时间本人将完成上文中提到的未完成的内容。