|  |  |
| --- | --- |
| **课 程 编 号：** | D04810156 |

|  |  |
| --- | --- |
| **学 号：** |  |

****

**实 训 报 告**

|  |  |
| --- | --- |
| **教 学 院** | 计算机学院 |
| **课程名称** | UML建模技术 |
| **专业 班级** | 15级计算机科学与技术（2）班 |
| **姓 名** | 黄知琼 徐颖 |
| **指导教师** | 伍红华 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2018 | 年 | 4 | 月 | 28 | 日 |

**目 录**

**[1系统需求分析](#_Toc513801630)** [1](#_Toc513801630)

[1.1新闻管理系统的需求分析 1](#_Toc513801631)

[1.2新闻管理系统的功能模块 2](#_Toc513801632)

**[2系统的用例图](#_Toc513801633)** [4](#_Toc513801633)

[2.1 前台用例图 4](#_Toc513801634)

[2.2 后台用例图 6](#_Toc513801635)

**[3系统中的类图](#_Toc513801636)** [11](#_Toc513801636)

[3.1类图的生成 11](#_Toc513801637)

[3.2 类之间的关系图 16](#_Toc513801638)

**[4系统中的顺序图](#_Toc513801639)** [17](#_Toc513801639)

[4.1 前台顺序图 17](#_Toc513801640)

[4.2后台顺序图 18](#_Toc513801641)

**[5系统中的通信图](#_Toc513801642)** [21](#_Toc513801642)

[5.1前台通信图 21](#_Toc513801643)

[5.2后台通信图 23](#_Toc513801644)

**[6系统状态机图](#_Toc513801645)** [25](#_Toc513801645)

[6.1前台状态机图 25](#_Toc513801646)

[6.2后台状态机图 26](#_Toc513801647)

**[7系统的活动图](#_Toc513801648)** [29](#_Toc513801648)

[7.1前台活动图 29](#_Toc513801649)

[7.2 后台活动图 31](#_Toc513801650)

**[8系统的配置和实现](#_Toc513801651)** [35](#_Toc513801651)

**[9系统的部署图](#_Toc513801652)** [36](#_Toc513801652)

**[10生成相关语言代码](#_Toc513801653)** [37](#_Toc513801653)

# 

# 1系统需求分析

## 1.1新闻管理系统的需求分析

新闻管理系统主要是对网站实时新闻动态的显示及管理，能系统安全的对新闻信息进行显示与管理。一个完整的新闻管理系统可分为前台显示和后台管理，从前台功能上看包括新闻分类显示，新闻搜索等功能。从后台功能上看包括对新闻的增加，修改，删除，审核等功能。

（1）新闻查看

在新闻页面显示中，能根据信息标题对新闻进行分类显示，用户在进入新闻系统后，能查看某个类别的所有新闻，在每个新闻下有详细内容的跳转链接，可对单个新闻进行详细了解。

（2）新闻搜索

用户在进入系统后，可根据关键字查找新闻记录，系统将含有关键字的所有新闻记录显示出来。

（3）用户注册登录

在进入系统后，由于相关权限需要用户登录后才能进行操作，所有需要用户在系统中进行注册及登录，用户的注册负责处理用户信息的提交，用户的登录负责处理用户的登录与退出。

（4）用户评论

用户在登录后，在查看新闻时，能对新闻进行评论，并能对该条新闻下其他用户的评论进行查看。

（5）新闻管理

在系统后台中，对新闻记录进行列表显示，根据不同用户权限，如超级管理员与普通管理员可对所有新闻进行增加，修改，审核，删除，查询；新闻编辑只能对自己发布的新闻进行增加，修改，删除及对新闻的查询。

（6）后台用户管理

后台用户可分为超级管理员，普通管理员，新闻编辑，超级管理员的权限比普通管理员具有更多的权限，可对普通管理员及新闻编辑进行管理。

（7）新闻类别管理

超级管理员可在此对新闻的类别进行增加，修改，删除操作。

## 1.2新闻管理系统的功能模块

新闻管理系统主要包括两大模块：前台功能模块和后台管理模块。如图1-1所示。

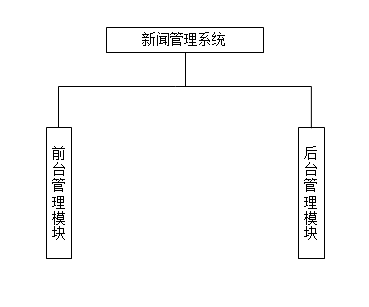


图1-1 系统功能模块

两大模块说明如下：

前台功能模块主要包括新闻分类查看、新闻详细信息查看、新闻搜索、用户登录、用户注册、对新闻发表发表和删除评论等，如图1-2所示。

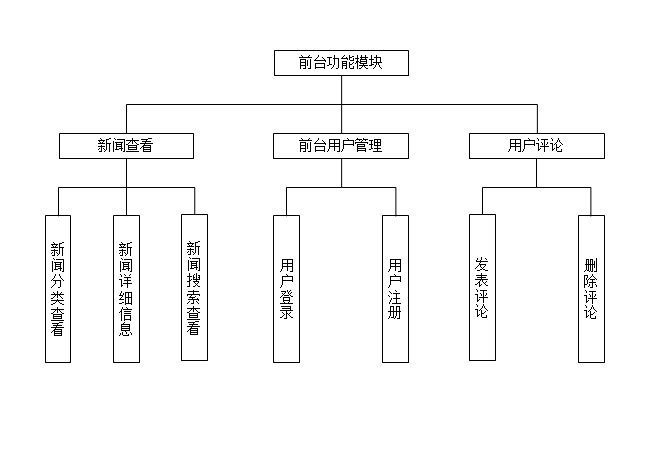


图1-2 前台管理模块

后台管理主要包括对新闻进行增加、修改、审核、删除、查找，对新闻类别进行增加、修改、删除，根据用户权限对后台用户进行分类管理，如图1-3所示。

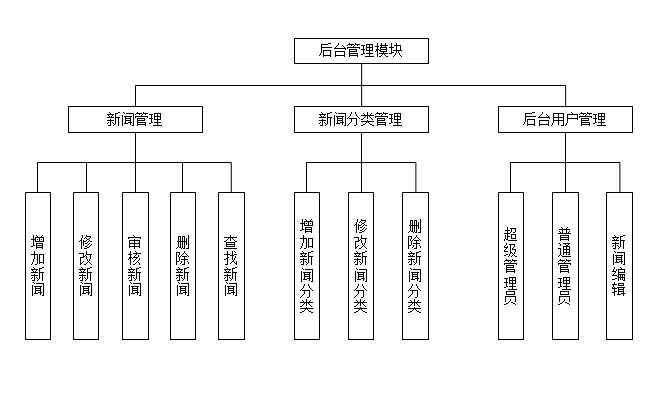


图1-3 后台管理模块

# 2系统的用例图

用例图是从用户角度描述系统功能，是用户所能观察到的系统功能的模型图，用例是系统中的一个功能单元。用例图作为参与者的外部用户所能观察的系统功能的模型图，在需求分析阶段起着重要的作用，整个开发过程都是围绕需求分析阶段的用例进行的，总体分为前台用例图与后台用例图。

## 2.1 前台用例图

新闻的前台浏览模块主要的参与者是系统的使用者，即游客与用户。

游客主要进行的是注册用户与新闻查看，具体如图2-1所示：

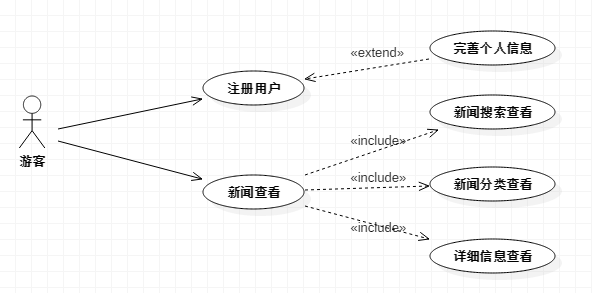


图2-1 游客用例图

用户是进入系统注册后，登录系统进行一系列操作的，用户在登录后可在查看新闻后对新闻发表评论，在与用户有关的用例图中，用户可进行的操作是用户登录，新闻查看，新闻评论，具体用例图如图2-2所示。

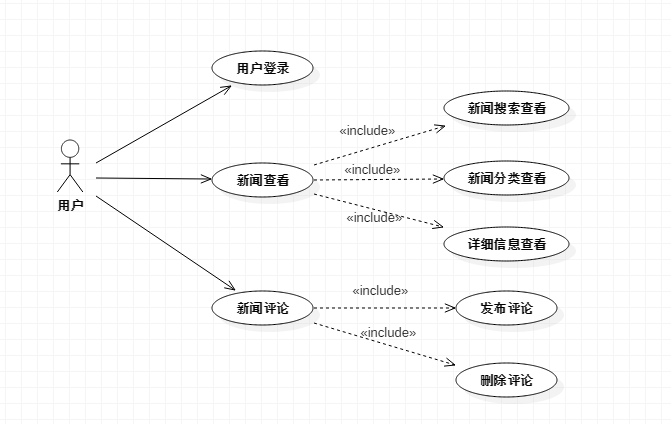


图2-2 用户用例图

## 2.2 后台用例图

新闻的后台管理模块的参与者为超级管理员，普通管理员，新闻编辑。

超级管理员的主要操作是新闻管理，新闻分类管理，用户管理，管理员管理员，具体的用例图如图2-3所示。

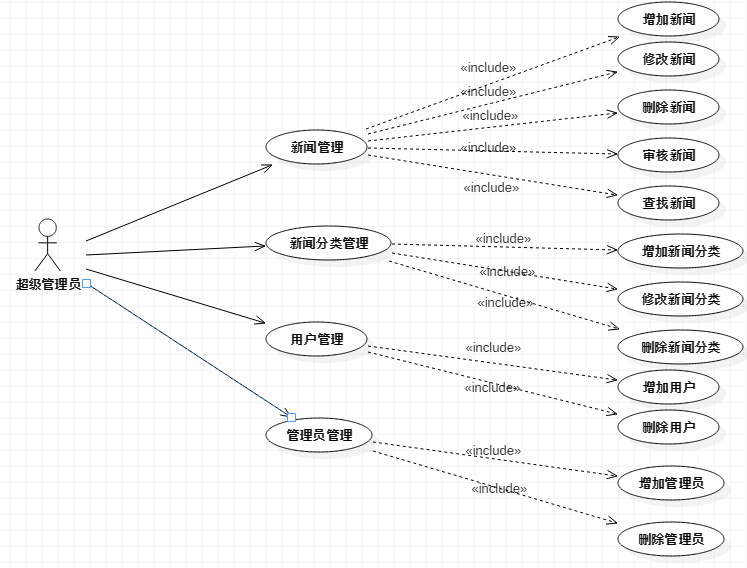


图2-3 超级管理员用例图

在与普通管理员有关的用例图中，普通管理员主要完成的操作是新闻管理，新闻分类管理，用户管理，与超级管理员相同，所以为关联关系。具体的用例图如图2-4所示。

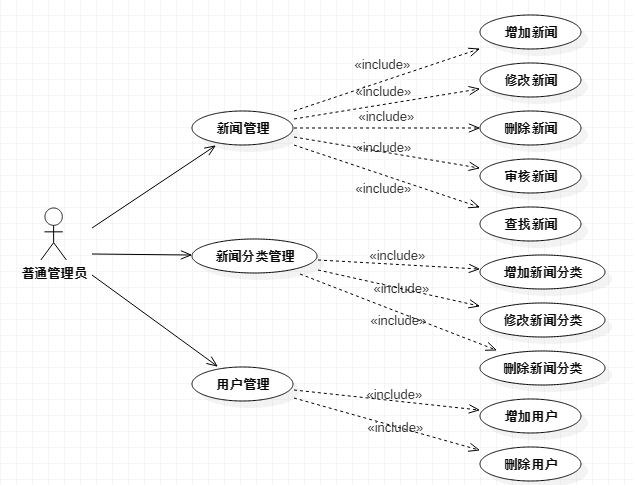


图2-4 普通管理员用例图

在与新闻编辑有关的用例图中，新闻编辑只能增加新闻，并且只能对自己所添加的新闻进行修改和删除，同时可对新闻进行查找，具体的用例图如图2-5所示。

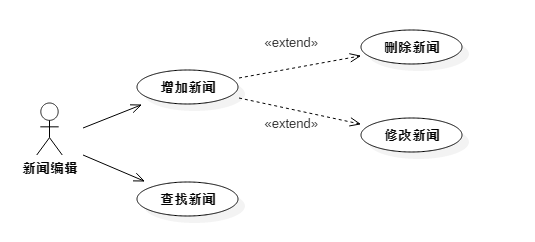
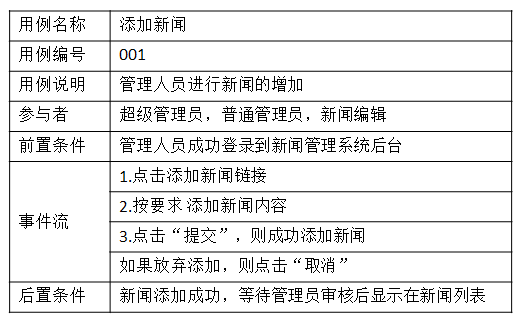


图2-5 新闻编辑用例图

在对本系统进行相关的用例描述时，在后台管理模块中，主要选取了添加新闻，修改新闻，审核新闻进行了用例描述。

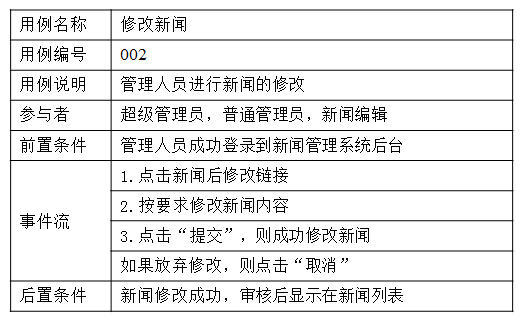
在添加新闻的用例描述中，管理人员在成功登录系统后台后，可进行添加新闻的操作，具体用例描述如表2-1所述。

表2-1 添加新闻用例描述



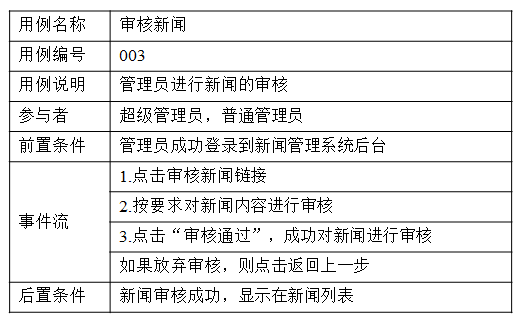
在修改新闻的用例描述中，管理人员在成功登录系统后台后，可进行修改新闻的操作，具体用例描述如下表2-2所述。

表2-2 修改新闻用例描述



在审核新闻的用例描述中，管理员在成功登录系统后台后，可进行审核新闻的操作，具体用例描述如表2-3所述。

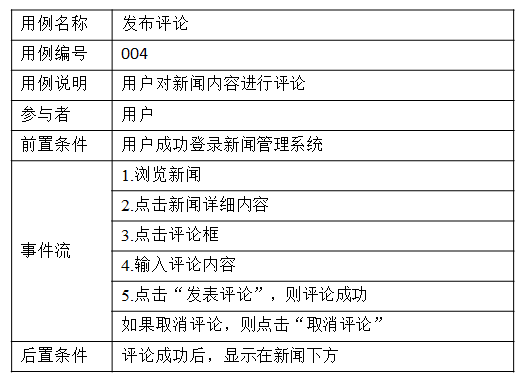
表2-3 审核新闻用例描述



在前台浏览模块中，主要选取了发布评论与新闻查询这两个用例进行了用例描述：

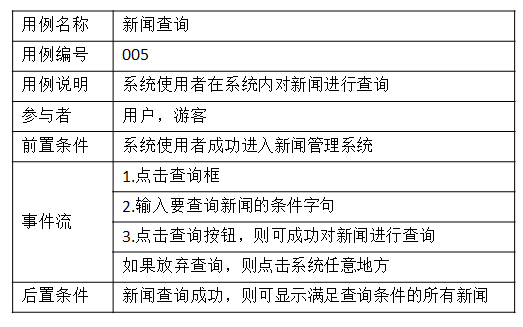
在发布评论的用例描述中，用户在成功登录到系统后可在浏览新闻后对新闻进行评论，具体的用例描述如表2-4所示。

表2-4 发布评论用例描述



在新闻查询的用例描述中，用户在成功登录到系统后可在系统首页通过条件语句对新闻进行查询，具体的用例描述如表2-5所示。

表2-5 新闻查询用例描述



# 3系统中的类图

类图描述系统中的静态结构。不仅定义系统中的类，表示类之间的联系如关联、依赖、聚合等，也包括类的内部结构（类的属性和操作）。类图是以类为中心组织的，类图中的其他元素或属于某个类或类相关联。

## 3.1类图的生成

对于新闻的浏览者没有什么特殊要求，也就是说可以是任何人，所以这里只考虑系统管理员。那么该新闻中心管理系统中的参与者（Actor）相关的类图只有一个即NewsAdmin类。NewsAdmin类是管理员类，如图3-1所示。

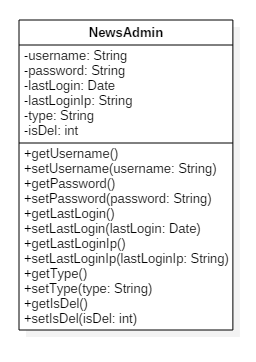


图3-1 NewsAdmin类图

1. News类

表示基本新闻信息的类，包含的属性有新闻编号（newsId）、新闻分类编号（classId）、新闻标题（headTitle）、新闻内容（content）、新闻作者（author）、新闻编辑者（editor）、新闻来源（newsFrom）、是否置顶（top）、新闻发布时间（newsTime）、新闻点击量（hits）、是否审核（state）、新闻图片（pictures）、是否被删除（isDel）、新闻分类名称（className）。News的类图如图3-2所示。

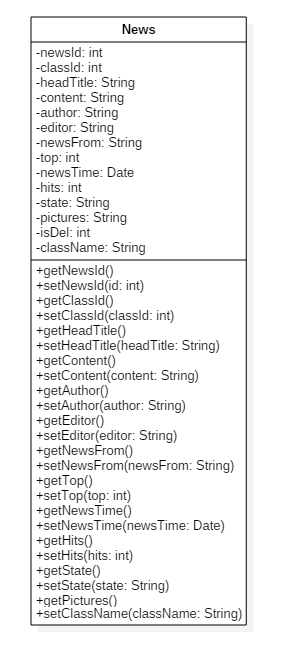


图3-2 News类图

1. NewsClass类

表示新闻分类的类，包含的属性有新闻分类编号（classId）、分类内容（content）和是否被删除（isDel）。NewsClass的类图如图3-3所示。

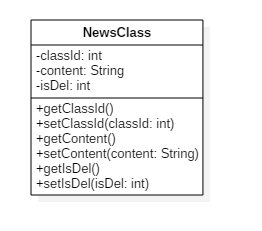


图3-3 NewsClass类图

1. NewsUser类

表示新闻用户的类，包含的属性有用户名（username）、用户密码（password）、性别（sex）、密码提示问题（question）、密码提示答案（answer）、电子邮件地址（emailAddress）、QQ号（qq）和验证码（reg）。NewsUser类的类图如图3-4所示。

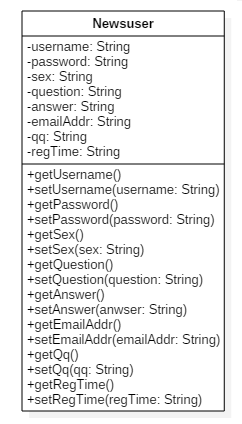


图3-4 NewsUser类图

1. Page类

为了使新闻列表、新闻分类列表和管理员列表实现分页功能而创建的类，Page类的类图如图3-5所示。

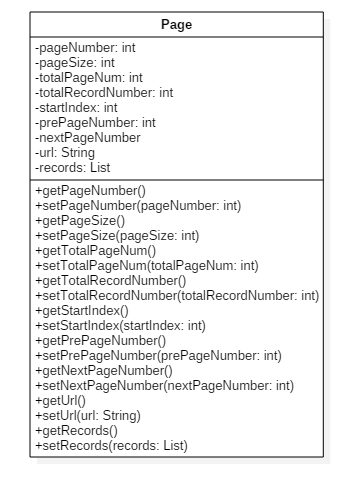


图3-5 Page类图

1. ListNews类

表示新闻的添加、删除、修改、审核和新闻分类的添加、删除操作的类，主要提供了业务逻辑的方法。ListNews类的类图如图3-6所示。

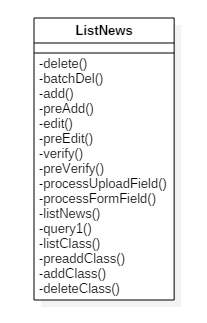


图3-6 ListNews类图

1. ListManager类

表示后台管理员的添加、删除、修改操作的类，主要提供了业务逻辑的方法。ListManager类的类图如图3-7所示。



图3-7 ListManager类图

1. ListUSer类

表示新闻用户的添加和删除操作的类，主要提供了业务逻辑的方法。ListUserr类的类图如图3-8所示。



图3-8 ListUser类图

1. AdminLogin类

表示管理员登录后台时的类，包含的属性有登录用户名（username）、登录密码（password）和session。AdminLogin类来的类图如图3-9所示。

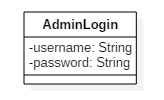


图3-9 AdminLogin类图

1. Front类

为前台提供服务的类，使得前台可以获得新闻列表、分类显示新闻、查询新闻、登录、注册和发表评论，该类执行具体的业务逻辑。Front类的类图如图3-10所示。

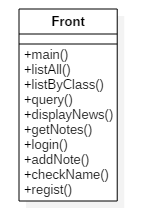


图3-10 Front类图

## 3.2 类之间的关系图

在系统中存在的各类之间的关系图如图3-11所示。

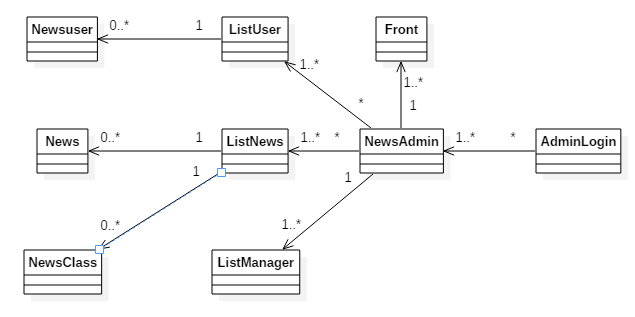


图3-11 类之间的关系图

* AdminLogin——NewsAdmin：多对多的关系，多个管理员可以在新闻后台登录。
* NewsAdmin——ListUser：多对多的关系，多个管理员可以对一个新闻用户进行管理，一个新闻用户也可以被多个管理员进行管理。
* NewsAdmin——ListNews：多对多的关系，多个管理员可以对一条新闻进行管理，一条新闻也可以被多个管理员进行管理。
* NewsAdmin——ListManager：一对多的关系，超级管理员可以对多个普通管理员和新闻编辑进行管理。

# 4系统中的顺序图

顺序图是强调消息时间顺序的交互图，它描述的是对象之间传送消息的时间顺序，用于表示用例中的行为顺序。顺序图在记录一个未来系统的行为应该如何表现时具有非常大的作用，它能把用例表达的需求，转化为进一步，更加正式层次的精细表达。

## 4.1 前台顺序图

在本系统的前台浏览模块中，主要选取了新闻查看与用户评论进行了顺序图的展示。

用户在进行新闻查看时，顺序图中涉及的对象有显示新闻标题，新闻详细信息和相关新闻。

新闻查看的顺序图如下图4-1所示。

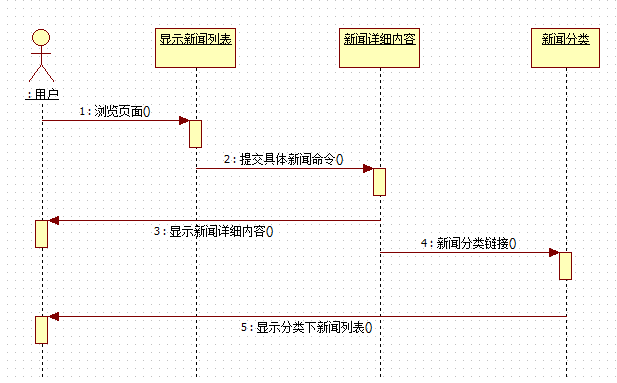


图4-1 新闻查看顺序图

用户进行评论时，顺序图中涉及的对象主要有登录，显示新闻标题，新闻详细信息，发表评论和数据库模块。

发布评论的顺序图如图4-2所示。

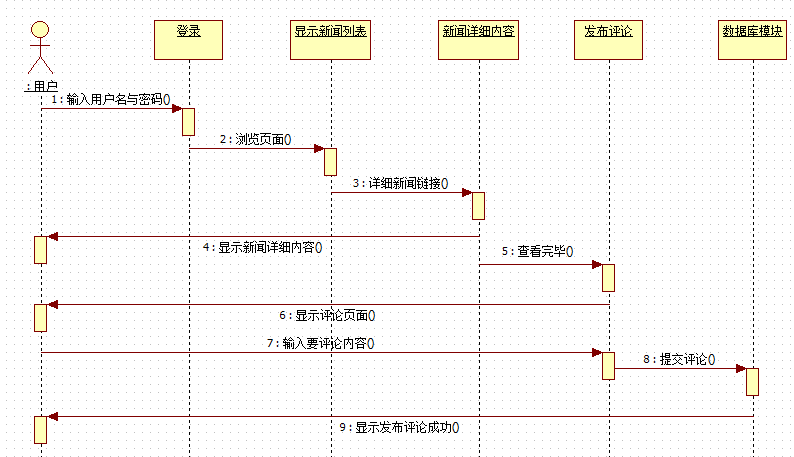


图4-2 发布评论顺序图

## 4.2后台顺序图

在本系统的后台管理模块中，主要选取的是添加新闻与删除新闻进行了顺序图的展示。

管理员在进行新闻的添加时，顺序图中涉及的对象主要有三个，即登录，添加新闻和数据库模块。添加新闻的顺序图如图4-3所示。

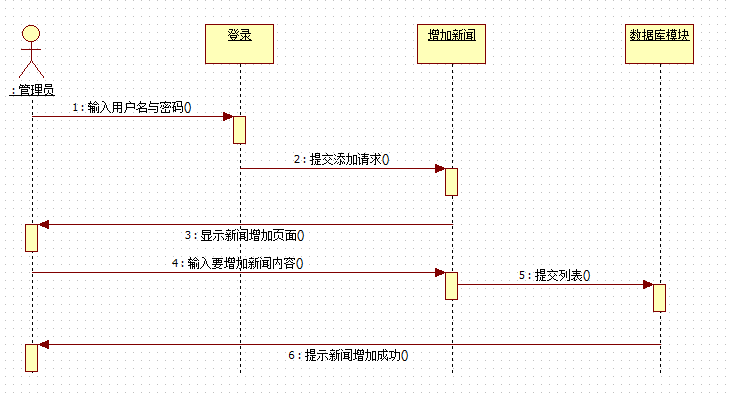


图4-3 添加新闻顺序图

在删除新闻的顺序图中，涉及的对象有登录，删除新闻，数据库模块。

删除新闻的顺序图如4-4图所示。

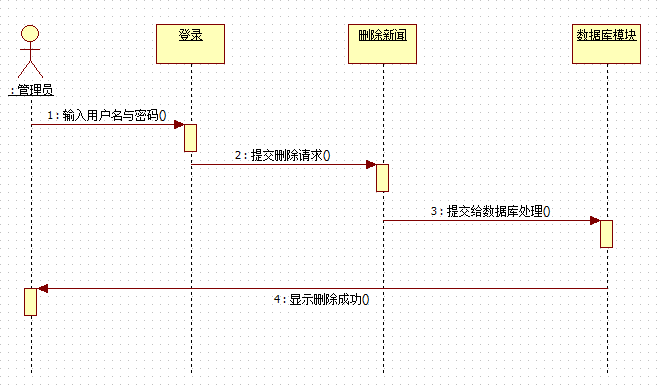


图4-4 删除新闻顺序图

# 5系统中的通信图

通信图也叫合作图，是一种交互图，强调的是发送和接收消息的对象之间的组织结构，一个通信图显示了一系列对象和在这些对象之间的联系以及对象间发送和接收的消息。一般使用通信图来说明系统的动态情况，通信图使描述复杂的程序逻辑或多个平行事务变得容易。与顺序图相比，顺序图是按照时间顺序描述的对象间的交互，但是由于它过于强调交互的时间，而削弱了对象间的静态连接关系的重视，通信图则强调了参与交互作用的对象的组织，显示对象间如何进行交互以执行特定用例或用例中特定部分的行为，设计人员使用通信图和顺序图确定并阐明对象的角色，这些对象执行用例的特定事件流，是主要的信息来源，用于确定类的职责和接口。在系统中，可根据顺序图转化为通信图。

## 5.1前台通信图

在前台中，主要展示的是新闻查看与用户评论的顺序图，根据两者的顺序图可转化成为通信图。

新闻查看的通信图如图5-1所示。

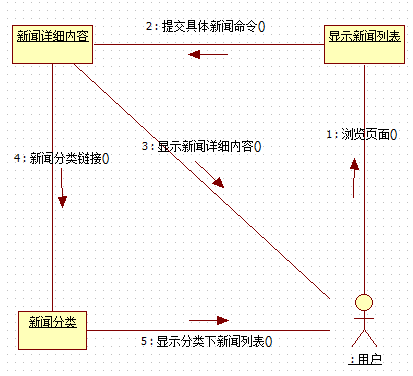


图5-1 新闻查看通信图

用户发表评论的通信图如图5-2所示。

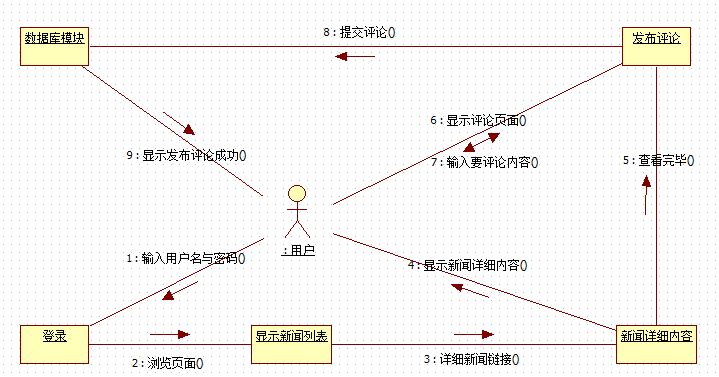


图5-2 发表评论通信图

## 5.2后台通信图

在后台中，我主要展示的是新闻添加与新闻删除的顺序图，根据二者的顺序图转化可得到它们的通信图。新闻添加通信图如图5-3所示。

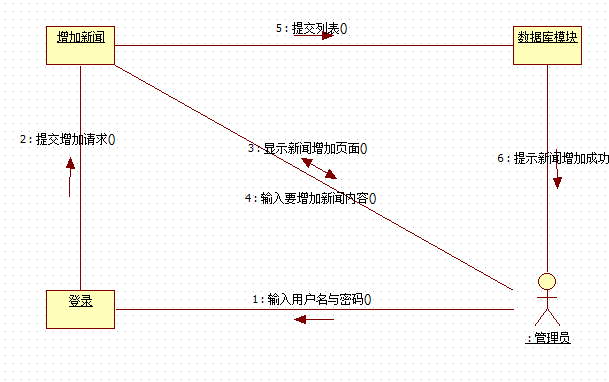


图5-3添加通信图

新闻删除的通信图如图5-4所示。

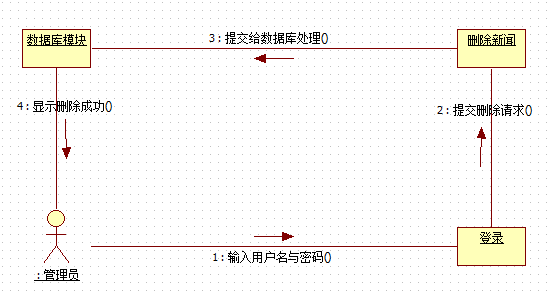


图5-4 新闻删除通信图

# 6系统状态机图

状态机图是系统分析的常用工具之一，它通过建立类对象的生存周期模型来描述对象随时间变化的动态行为。所有对象都有状态，状态是对象执行了一系列活动的结果，当某个事件发生后，对象的状态将发生变化。对象从产生到结果，可以处于一系列不同状态。在任一给定的时刻，一个对象总是处于某一特定的状态。

状态机图是一个类对象所可能经历的所有过程的模型图。状态机图由对象的各个状态和连接这些状态的转换组成。

## 6.1前台状态机图

（1）用户注册状态机图

游客在浏览新闻的过程中，可以在本网站进行注册，用户注册的状态机图如图6-1所示。

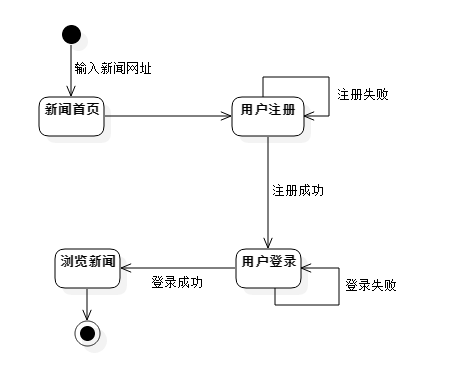


图6-1 用户注册状态机图

1. 用户评论状态机图

用户在浏览新闻时，可以进行注册和登录，登录成功之后便可发表评论，用户评论的状态机图如图6-2所示。

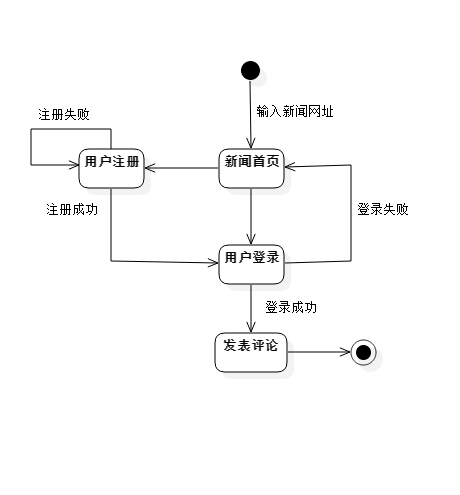


图6-2 用户评论的状态机图

## 6.2后台状态机图

1. 新闻管理的状态机图

在系统的后台中，管理员可以对新闻进行添加、修改、删除和审核等操作，新闻管理的状态机图如图6-3所示。

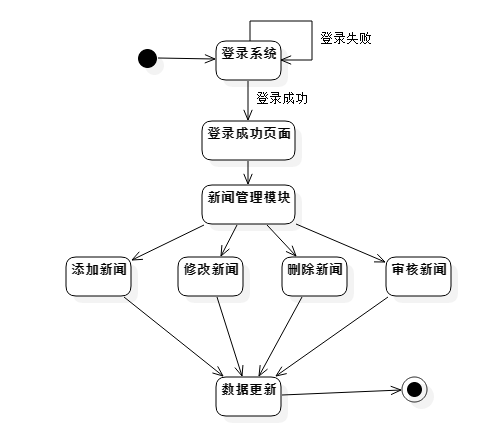


图6-3 新闻管理的状态机图

（2）新闻分类管理的状态机图

在系统的后台中，管理员可以对新闻的分类进行添加和删除等操作，新闻分类管理的状态机图如图6-4所示。

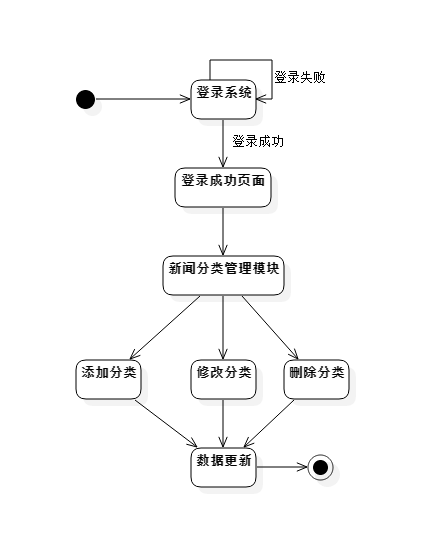


图6-4 新闻分类的状态机图

（3）管理员管理的状态机图

在系统的后台中，超级管理员可以对管理员和新闻用户进行添加和删除等操作，管理员管理的状态机图如图6-5所示。

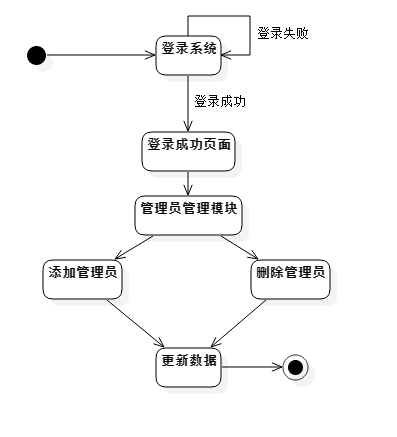


图6-5 管理员管理的状态机图

# 7系统的活动图

活动图表示在处理某个活动时，两个或者更多类对象之间的过程控制流。它用来描述采取何种动作、做什么（对象状态改变）、何时发生（动作序列）以及何处发生（泳道）。活动图可用于在业务单元的级别上对更高级别的业务过程进行建模，或者对低级别的内部类操作进行建模。

## 7.1前台活动图

1. 游客活动图

游客可在新闻首页浏览和查询新闻，同时也可查看新闻详情，并可在网站中进行注册等功能。游客活动图如图7-1所示。

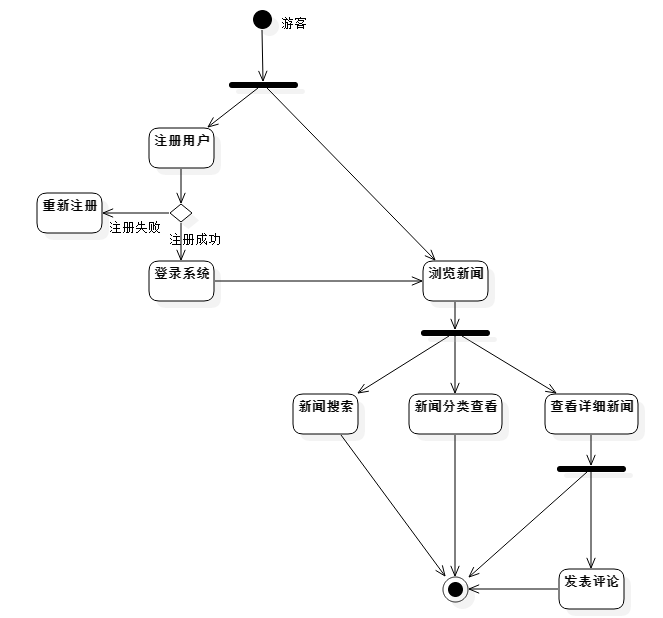


图7-1 游客活动图

1. 普通用户活动图

普通用户可在新闻首页浏览和查询新闻，同时也可查看新闻详情，并可在网站中进行登录等功能。普通用户活动图如图7-2所示。

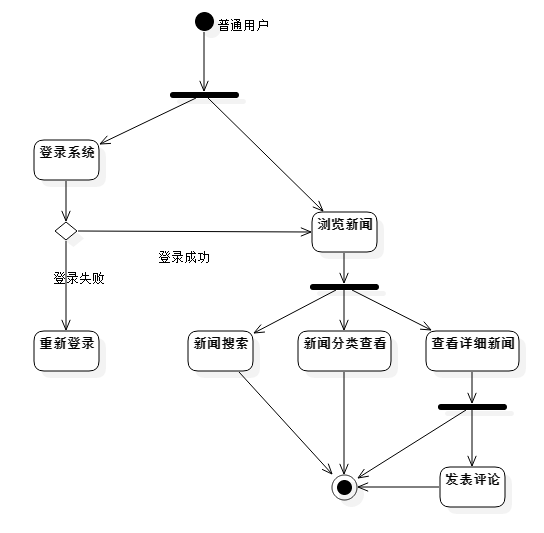


图7-2 普通用户的活动图

## 7.2 后台活动图

（1）新闻编辑者的活动图

在系统的后台中，新闻编辑者可添加新闻，并对自己发布的新闻进行修改和删除，新闻编辑者的活动图如图7-3所示。

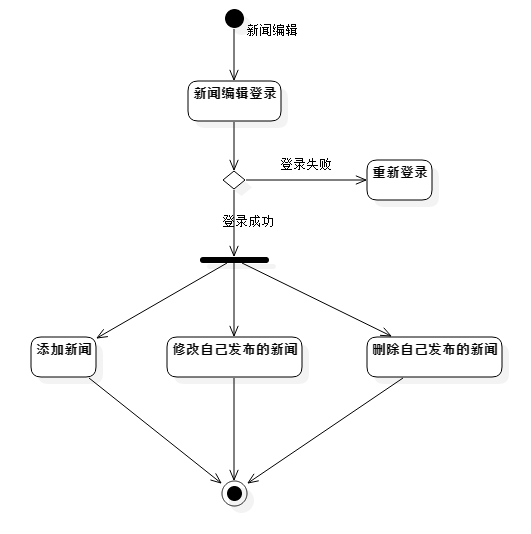


图7-3 新闻编辑者的活动图

（2）普通管理的活动图

普通管理员在后台可以添加、修改、删除和审核所有新闻，也可以添加和删除新闻分类，普通管理员的活动图如图7-4所示。

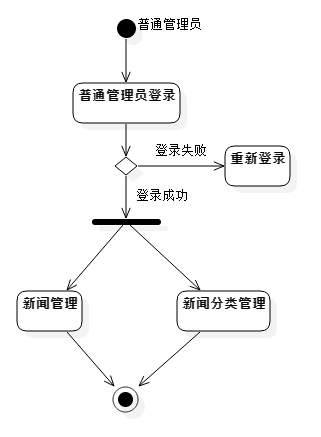


图7-4 普通管理员的活动图

（3）超级管理员的活动图

超级管理员在后台可以添加、修改、删除和审核所有新闻，也可以添加、删除新闻分类、普通管理员和新闻编辑者，超级管理员的活动图如图7-5所示。

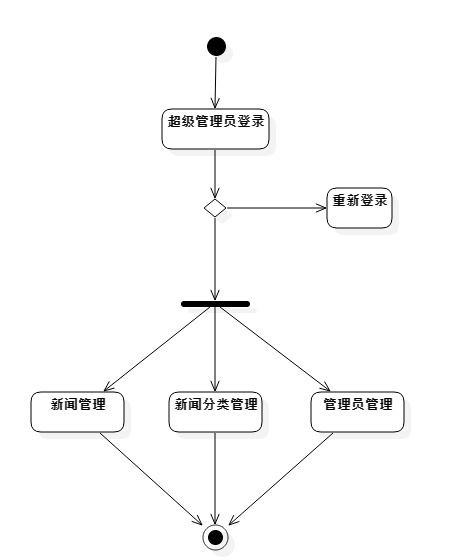


图7-5 超级管理员的活动图

# 8系统的配置和实现

构件图是表示组件类型的组织以及各种组件之间依赖关系的图，它通过对组件间依赖关系的描述来估计对系统组件的修改给系统可能带来的影响。可以使用构件图来可视化物理组件与它们之间的关系并描述其构造细节。

新闻管理系统中，组成Web应用程序页面的包括：前台浏览页面与后台管理页面，前台浏览页面由新闻查询页面，新闻评论页面，新闻查看页面，注册页面，登录页面组成，后台管理页面由新闻管理与管理员管理两部分组成，新闻管理有新闻审核页面，新闻删除页面，新闻修改页面，新闻添加页面组成，管理员管理由管理员添加页面与管理员删除页面组成。

新闻管理系统的构件图如图8-1所示。

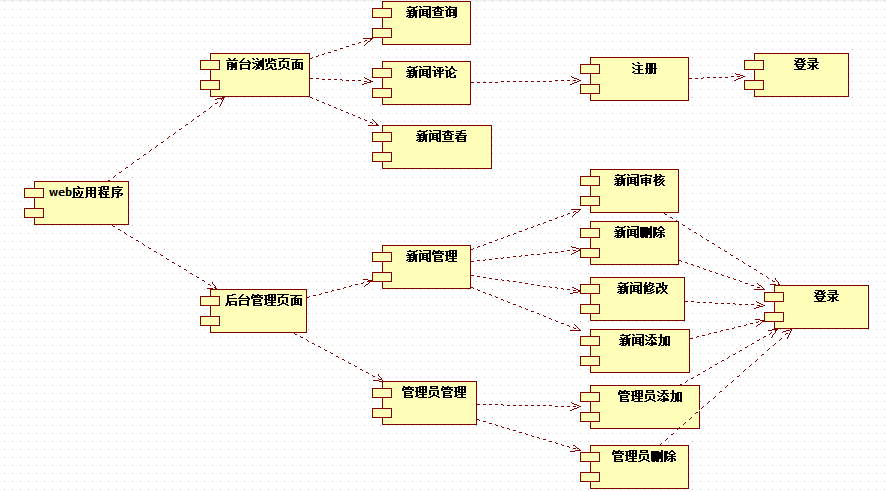


图8-1 构件图

# 9系统的部署图

新闻管理系统的部署图如图9-1所示。

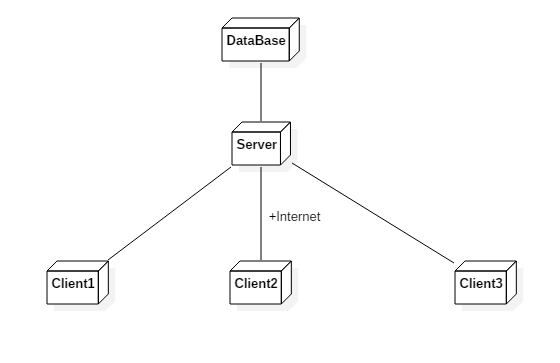


图9-1 系统的部署图

# 10生成相关语言代码

NewAdmin.java代码

package cn.edu.hbpu.news.model;

import java.sql.Date;

public class NewsAdmin {

private String username;

private String password;

private Date lastLogin;

private String lastLoginIP;

private String type;

private int isDel;

private int adminId;

public int getAdminId() {

return adminId;

}

public void setAdminId(int adminId) {

this.adminId = adminId;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public Date getLastLogin() {

return lastLogin;

}

public void setLastLogin(Date lastLogin) {

this.lastLogin = lastLogin;

}

public String getLastLoginIP() {

return lastLoginIP;

}

public void setLastLoginIP(String lastLoginIP) {

this.lastLoginIP = lastLoginIP;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public int getIsDel() {

return isDel;

}

public void setIsDel(int isDel) {

this.isDel = isDel;

}

/\*\*

\* @param args

\*/

public static void main(String[] args) {

// TODO Auto-generated method stub

}

}

ListManager.java代码

public class ListManager extends BaseServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

doPost(request, response);

}

public void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String op = request.getParameter("op");

if (op.equals("list")) {

listManager(request, response);

} else if (op.equals("delete")) {

delete(request, response);

} else if (op.equals("add")) {

add(request, response);

}

}

public void listManager(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

PrintWriter out = response.getWriter();

NewsAdminDAO newsAdminDAO = NewsAdminFactory.getNewsAdminDAOInstance();

HttpSession session = request.getSession();

String type = newsAdminDAO.getAdminType((String) session.getAttribute("adminUser"));

if (!type.equals("1")) {

out.print("<script>alert('您不是超级管理员，无法访问！');window.location.href='index.jsp'</script>");

return;

}

String num = request.getParameter("num");

int pageNum = 1;

if (num != null && !num.equals("")) {

pageNum = Integer.parseInt(num);

}

int totalRecordNum = newsAdminDAO.getTotalRecordNum();// 获取总的记录数

Page page = new Page(pageNum, totalRecordNum);// 构建分页对象

List<NewsAdmin> newsList = newsAdminDAO.getListByPage(page.getStartIndex(), page.getPageSize());// 分页查询

page.setRecords(newsList);

page.setUrl("/manager/ListManager?op=list");

request.setAttribute("page", page);

try {

request.getRequestDispatcher("changeManager.jsp").forward(request, response);

} catch (ServletException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

}

public void delete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String adminId = request.getParameter("adminId");

PrintWriter out = response.getWriter();

NewsAdminDAO newsAdminDAO = NewsAdminFactory.getNewsAdminDAOInstance();

int result = newsAdminDAO.delete(adminId);

if (result == 0) {

out.print("新闻删除失败");

} else {

out.print("新闻删除成功");

}

}

public void add(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String username = request.getParameter("username");

String password = request.getParameter("password");

String type = request.getParameter("type");

NewsAdminDAO newsAdminDAO = NewsAdminFactory.getNewsAdminDAOInstance();

NewsAdmin newsAdmin = new NewsAdmin();

newsAdmin.setUsername(username);

newsAdmin.setPassword(password);

newsAdmin.setType(type);

PrintWriter out = response.getWriter();

int result = newsAdminDAO.insert(newsAdmin);

if (result == 0) {

out.print("<script>alert('管理员信息插入失败');window.location.href='addManager.jsp'</script>");

} else {

out.print("<script>alert('管理员信息插入成功!');window.location.href='index.jsp'</script>");

}

}

}

AdminLogin.java代码

public class AdminLogin extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

//处理get请求

doPost(request,response);//如果是get请求，执行doPost方法

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

NewsAdminDAO a=new NewsAdminDAO();

PrintWriter out1=response.getWriter();

String username=request.getParameter("username");

String password=request.getParameter("password");

if(username!=null&&!username.equals("")&&!password.equals("")&&password!=null){

String passString=MD5Util.encoder(password);

if(a.isExist(username, password)){

request.getRequestDispatcher("main.jsp").forward(request, response);

HttpSession session=request.getSession();

session.setAttribute("adminUser", username);

}

else{

request.setAttribute("msg", "你输入的用户名或密码有误");

request.getRequestDispatcher("login.jsp").forward(request, response);

}

}else{

out1.print("<script>alert('请您输入用户名和密码！');" +

"window.location.href='login.jsp'</script>");

}

}

}

**实训成绩评定表**

|  |
| --- |
| 实训总结：在此次实训过程中，使我从书本上获得的理论知识真正的得到了学以致用。在对类图、活动图和状态机图等的绘制过程中，我认识到了UML的作用以及方法。同时也让我了解到需求分析和系统分析在项目开发中的重要。在实训中我学到了许多新的知识和方法。是一个让我把书本上的理论知识运用于实践中的好机会，学的时候感叹学的内容太难懂，现在想来，有些其实并不难，关键还是在于理解。同时也锻炼了我做项目的能力，也提高了独立思考问题和自己动手操作的能力。 |
| 教师评语： |
| 考勤成绩： ， 占总成绩10%  平时成绩： ， 占总成绩20%  答辩成绩： ， 占总成绩30%  报告成绩： ， 占总成绩40%  实训成绩： 。  指导老师签字：  年 月 日 |