

J2EE 课程设计总报告

项目名称: 日常业务管理模块

指导老师: 徐建

小组成员: 1102610118 闵大为
1102610110 郭一迪
1106840642 于新哲

目录

1 引言	2
2 需求分析	2
3 项目设计	2
3.1 相关技术简介	2
3.2 系统流程和系统结构设计	3
3.3 系统角色设计	5
3.4 数据库设计与构架	5
3.5 网站页面详细设计	10
4 项目开发	20
4.1 技术方案	20
4.2 关键模块的实现	21
5 个人体会	51
5.1 闵大为的个人小结	51
5.2 郭一迪的个人小结	52
5.3 于新哲的个人小结	54

1 引言

根据任课教师的要求：小组任选模块根据需求说明书完成课程设计。综合考虑了小组本身的条件及各个模块的特点，我们选择了制作日常事务管理模块。一方面这个模块和我们日常的学习生活息息相关，另一方面也具有一定的难度，可以提高我们的编程能力和项目设计能力，让我们能积累相关方面的知识。目前我们小组已经完成了模块的编译，基本上完成了需求说明书中的所有功能。项目主要完成了公告系统、会议系统、邮件系统和登录身份识别四个功能。界面简洁美观易操作，有自己独特的风格和特点。

2 需求分析

开始简化(Extjs 版本): [需求分析 V1.0.doc](#)

后来完善(现在版本): [需求分析 V2.0.doc](#)

3 项目设计

3.1 相关技术简介

1.Jsp

JSP (Java Server Pages)是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准。JSP 技术有点类似 ASP 技术，它是在传统的网页 HTML 文件(*.htm,*.html)中插入 Java 程序段(Scriptlet)和 JSP 标记(tag)，从而形成 JSP 文件,后缀名为(*.jsp)。用 JSP 开发的 Web 应用是跨平台的,既能在 Linux 下运行，也能在其他操作系统上运行。

2.Tomcat 服务器

Tomcat 是 Apache 软件基金会 (Apache Software Foundation) 的 Jakarta 项目中的一个核心项目，由 Apache、Sun 和其他一些公司及个人共同开发而成。由于有了 Sun 的参与和支持，最新的 Servlet 和 JSP 规范总是能在 Tomcat 中得到体现，Tomcat 5 支持最新的 Servlet 2.4 和 JSP 2.0 规范。因为 Tomcat 技术先进、性能稳定，而且免费，因而深受 Java 爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的 Web 应用服务器。目前最新版本是 8.0。

3.MySql 数据库

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 公司。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高

了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策（本词条“授权政策”），它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。由于其社区版的性能卓越，搭配 PHP 和 Apache 可组成良好的开发环境。

4.Hibernate

Hibernate 是一个开放源代码的对象关系映射框架，它对 JDBC 进行了非常轻量级的对象封装，使得 Java 程序员可以随心所欲的使用对象编程思维来操纵数据库。Hibernate 可以应用在任何使用 JDBC 的场合，既可以在 Java 的客户端程序使用，也可以在 Servlet/JSP 的 Web 应用中使用，最具革命意义的是，Hibernate 可以在应用 EJB 的 J2EE 架构中取代 CMP，完成数据持久化的重任。

5.Jquery

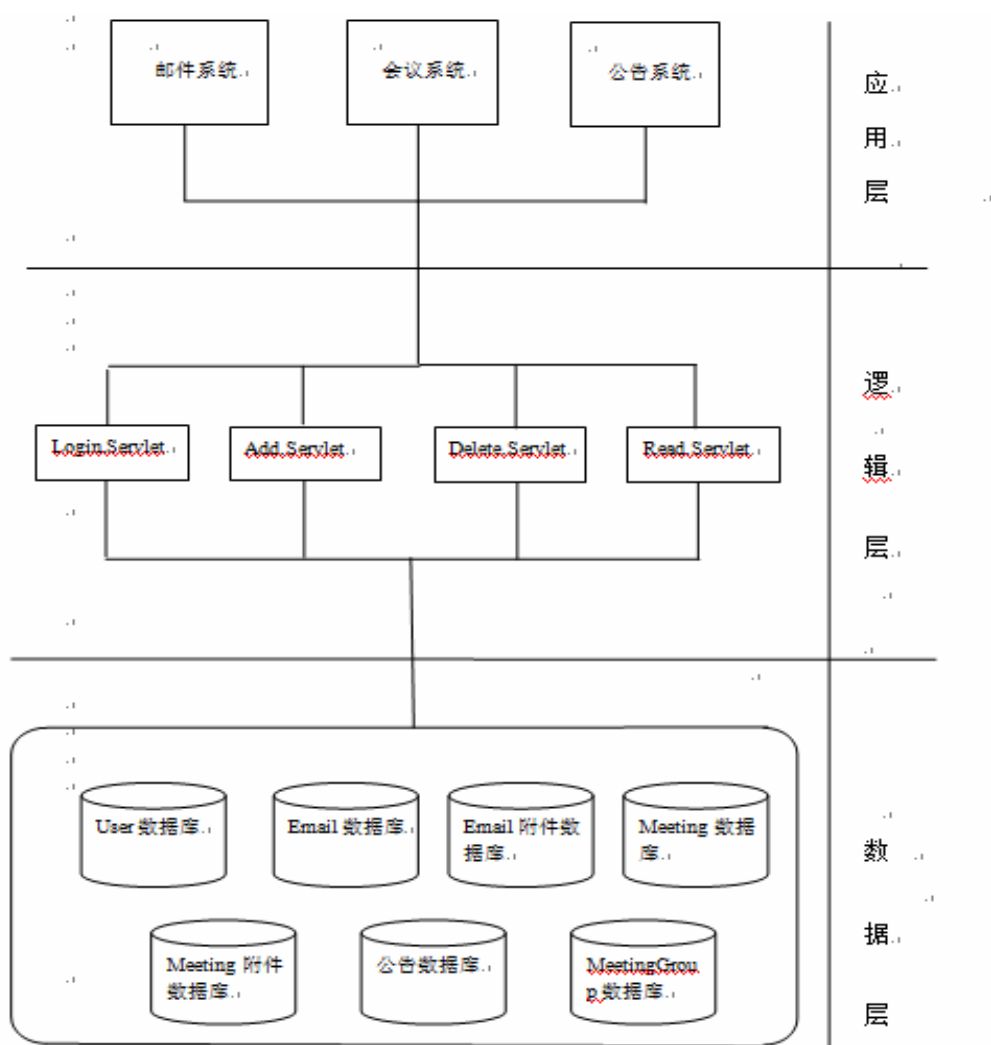
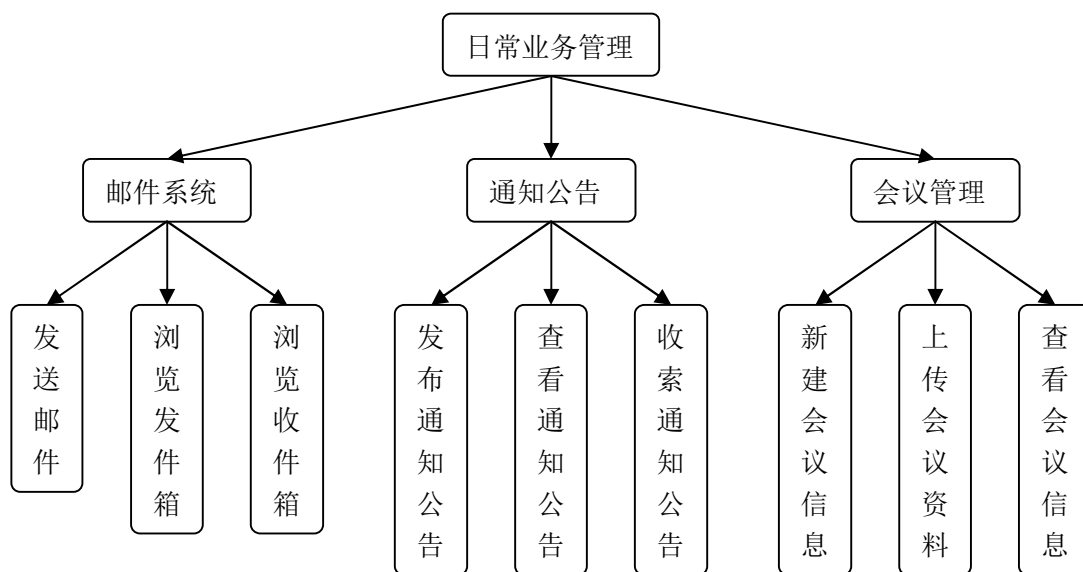
Jquery 是继 prototype 之后又一个优秀的 Javascript 框架。它是轻量级的 js 库，它兼容 CSS3，还兼容各种浏览器（IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+），jQuery2.0 及后续版本将不再支持 IE6/7/8 浏览器。jQuery 使用户能更方便地处理 HTML documents、events、实现动画效果，并且方便地为网站提供 AJAX 交互。jQuery 还有一个比较大的优势是，它的文档说明很全，而且各种应用也说得很详细，同时还有许多成熟的插件可供选择。jQuery 能够使用户的 html 页面保持代码和 html 内容分离，也就是说，不用再在 html 里面插入一堆 js 来调用命令了，只需定义 id 即可。

6.Ajax

一种网页开发技术，这个术语源自描述从基于 Web 的应用到基于数据的应用的转换。在基于数据的应用中，用户需求的数据如联系人列表，可以从独立于实际网页的服务端取得并且可以被动态地写入网页中，给缓慢的 Web 应用体验着色使之像桌面应用一样。

3.2 系统流程和系统结构设计

首先用户可以在主页中查看相关公告，可以通过搜索寻找自己所需要的公告，当用户在未登陆的情况下点击邮件系统，会议系统或者公告系统，则都会跳转到登陆界面，当用户登陆成功后，根据用户的权限，获得他所拥有的权限，从而进行操作。



3.3 系统角色设计

1.游客

只能进行浏览主页和查询公告的操作

2.普通用户

能进行浏览主页和查询公告，还可以查看自己的邮件，发/收邮件。

3.公告管理员

除了拥有普通用户的权限，还可以进入公告系统，进行管理公告，发布公告的操作。

4.会议管理员

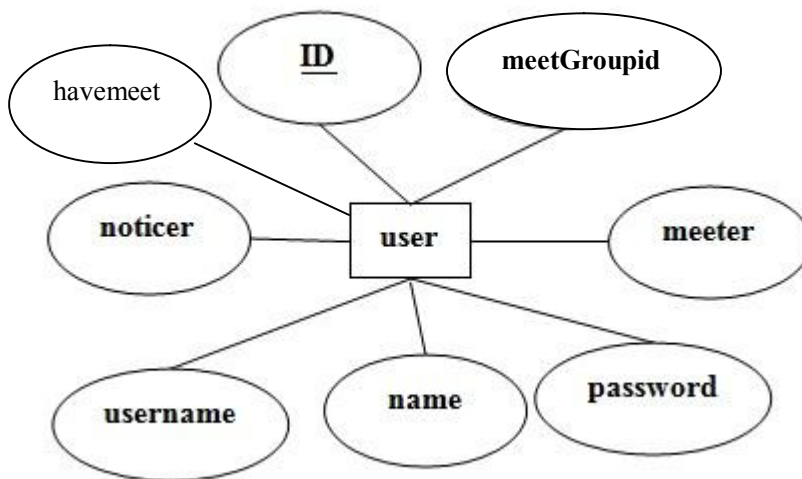
除了拥有普通用户的权限，还可以进入会议系统，进行管理会议，发布会议的操作。

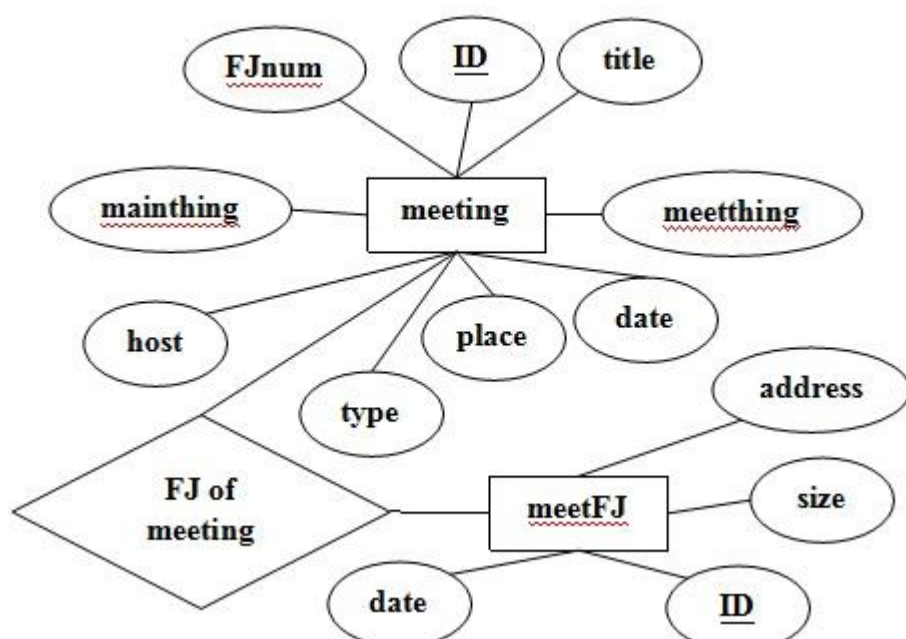
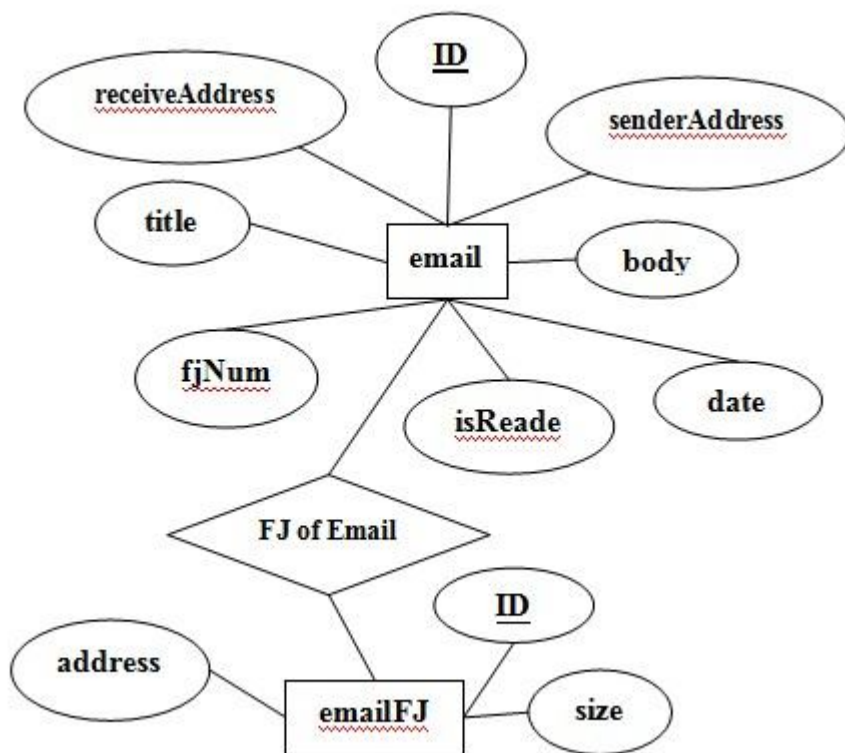
5.最高权限的管理员

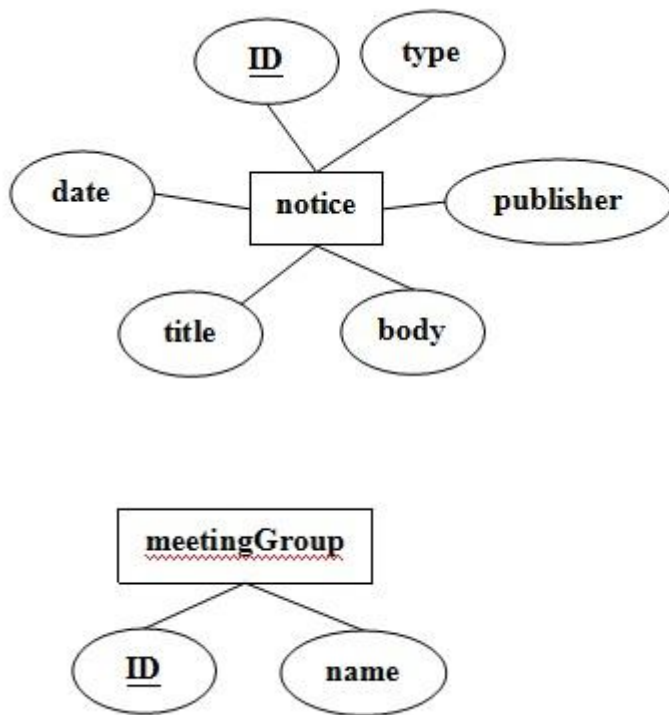
拥有公告管理员和会议管理员的全部权限

3.4 数据库设计与构架

1.E-R 图







2.物理设计

表一： User 表格 用户表格

id 为主键，非空，用于标记用户 INT PK,NN,AI
 userName 为用户登录名 VARCHAR(45) NN
 password 为用户登录密码 VARCHAR(45) NN
 noticer 判断用户是否有公告管理权利 VARCHAR(45) NN
 meeter 判断用户是否有会议管理权利 VARCHAR(45) NN
 name 为用户中文名 VARCHAR(45) NN
 meetGroupId 为用户所属组别 VARCHAR(45) NN
 havemeet 为判断当前用户是否有会议 VARCHAR(45) NN

名	类型	长度	小数点	不是 null	
► id	int	11	0	<input checked="" type="checkbox"/>	🔑 1
meeter	varchar	45	0	<input checked="" type="checkbox"/>	
meetGroupId	int	11	0	<input checked="" type="checkbox"/>	
name	varchar	45	0	<input checked="" type="checkbox"/>	
noticer	varchar	45	0	<input checked="" type="checkbox"/>	
password	varchar	45	0	<input checked="" type="checkbox"/>	
userName	varchar	45	0	<input checked="" type="checkbox"/>	
havemeet	varchar	45	0	<input checked="" type="checkbox"/>	

表二： Email 表格 邮件表格

id 唯一标记 INT PK,NN,AI
 receiverAdress 接收人地址 VARCHAR(45) NN
 senderAdress 发件人地址 VARCHAR(45) NN
 title 邮件标题 VARCHAR(45) NN
 body 邮件内容 TEXT NN
 fjNum 附件个数 INT NN
 date 邮件发送日期 DATE NN
 isReaded 是否有邮件未读 VARCHAR(45) NN
 recDelete 被删除的收到邮件 VARCHAR(45) NN
 sendDelete 被删除的发送邮件 VARCHAR(45) NN
 sender 邮件发送者 VARCHAR(45) NN

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	1
body	text	0	0	<input checked="" type="checkbox"/>	
date	datetime	0	0	<input checked="" type="checkbox"/>	
fjNum	int	11	0	<input checked="" type="checkbox"/>	
isReaded	varchar	45	0	<input checked="" type="checkbox"/>	
recDelete	varchar	45	0	<input checked="" type="checkbox"/>	
receiverAdress	varchar	45	0	<input checked="" type="checkbox"/>	
sendDelete	varchar	45	0	<input checked="" type="checkbox"/>	
sender	varchar	45	0	<input checked="" type="checkbox"/>	
senderAdress	varchar	45	0	<input checked="" type="checkbox"/>	
▶ title	varchar	100	0	<input checked="" type="checkbox"/>	

表三： EmailFj 表格 用来记录附件


id 唯一标号 INT PK,NN,AI
 address 存放地址 VARCHAR(45) NN
 name 附件名字 FLOAT NN
 emailId 所属邮件的 Id INT NN
 addTime 邮件发送的时间 VARCHAR(45)

名	类型	长度	小数点	不是 null	
▶ id	int	11	0	<input checked="" type="checkbox"/>	1
adress	varchar	400	0	<input checked="" type="checkbox"/>	
emailId	int	11	0	<input checked="" type="checkbox"/>	
name	varchar	100	0	<input checked="" type="checkbox"/>	
addTime	varchar	45	0	<input type="checkbox"/>	

表四： Meet 表格 用来记录会议表格


id 会议唯一标号 INT PK,NN,AI
 date 会议时间 DATE NN
 endTime 会议结束时间 VARCHAR(45) NN

fjNum 会议附件个数 INT NN
 host 会议主持者 VARCHAR(45) NN
 mainThing 会议主要事项 TEXT NN
 meetEnd 会议结束时间的完整表示 VARCHAR(45) NN
 meetGroupId 收件人所在组 INT NN
 Meetstart 会议开始时间的完整表示 VARCHAR(45) NN
 meetThing 会议纪要 TEXT NN
 place 会议地点 VARCHAR(45) NN
 startTime 会议开始时间 VARCHAR(45) NN
 title 会议议题 VARCHAR(100) NN
 type 会议类型 VARCHAR(45) NN
 daiban 代办事项 VARCHAR(45) NN

名	类型	长度	小数点	不是 null	
► id	int	11	0	<input checked="" type="checkbox"/>	 1
date	varchar	100	0	<input checked="" type="checkbox"/>	
endTime	varchar	45	0	<input checked="" type="checkbox"/>	
fjNum	int	11	0	<input checked="" type="checkbox"/>	
host	varchar	45	0	<input checked="" type="checkbox"/>	
mainThing	text	0	0	<input checked="" type="checkbox"/>	
meetEnd	varchar	45	0	<input checked="" type="checkbox"/>	
meetGroupId	int	45	0	<input checked="" type="checkbox"/>	
meetstart	varchar	45	0	<input checked="" type="checkbox"/>	
meetThing	text	0	0	<input checked="" type="checkbox"/>	
place	varchar	45	0	<input checked="" type="checkbox"/>	
startTime	varchar	45	0	<input checked="" type="checkbox"/>	
title	varchar	100	0	<input checked="" type="checkbox"/>	
type	varchar	100	0	<input checked="" type="checkbox"/>	
daiban	varchar	45	0	<input checked="" type="checkbox"/>	


表五： meetFj 表格 会议附件表格

id 唯一标号 INT PK,NN,AI
 address 会议附件地址 VARCHAR(200) NN
 meetId 附件所属会议的 Id INT NN
 name 附件名字 VARCHAR(100) NN

名	类型	长度	小数点	不是 null	
► id	int	11	0	<input checked="" type="checkbox"/>	 1
adress	varchar	400	0	<input checked="" type="checkbox"/>	
meetId	int	11	0	<input checked="" type="checkbox"/>	
name	varchar	100	0	<input checked="" type="checkbox"/>	


表六 notice 表格 公告表格

id 唯一标号 INT NN
 type 公告类型 VARCHAR(45) NN
 title 公告主题 VARCHAR(45) NN
 body 主要内容 TEXT NN
 publisher 发表者中文姓名 VARCHAR(45) NN
 date 发布日期 DATE NN

名	类型	长度	小数点	不是 null	
▶ id	int	11	0	<input checked="" type="checkbox"/>	 1
body	text	0	0	<input checked="" type="checkbox"/>	
date	datetime	0	0	<input checked="" type="checkbox"/>	
publisher	varchar	45	0	<input checked="" type="checkbox"/>	
title	varchar	100	0	<input checked="" type="checkbox"/>	
type	varchar	45	0	<input checked="" type="checkbox"/>	


表七 meetingGroup 表格 组别表格

id 唯一标识 INT PK,NN,AI
 name 组别名字 VARCHAR(45) NN

名	类型	长度	小数点	不是 null	
▶ id	int	11	0	<input checked="" type="checkbox"/>	 1
name	varchar	100	0	<input checked="" type="checkbox"/>	

表八 noticeType 表格 公告类型

id 唯一标识 INT PK,NN,AI
 name 公告名字 VARCHAR(45) NN

名	类型	长度	小数点	不是 null	
▶ id	int	11	0	<input checked="" type="checkbox"/>	 1
name	varchar	45	0	<input checked="" type="checkbox"/>	

3.5 网站页面详细设计

本系统主要有：主界面模块，邮件系统模块，通知公告模块，会议管理模块，用户登陆模块。

主界面模块：能够显示一些通知公告，然后提供一些模块的进入按钮。

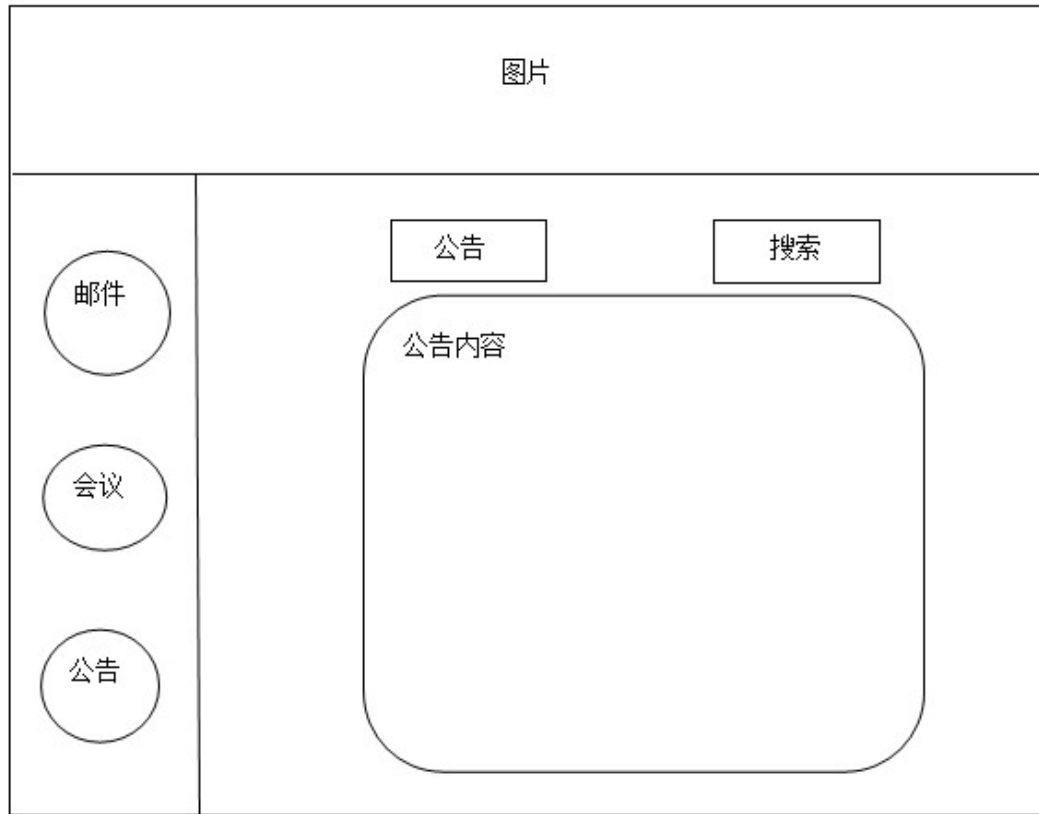
邮件系统模块：一般用户登进去以后能够发送邮件，查看收件箱，查看发件箱。如果有未读邮件就会自动提醒。发送邮件时能上传附件，接收邮件时能下载附件。

通知公告模块：具有通知公告管理权限的用户登进来之后，能够进行发布通知公告，查看通知公告，搜索通知公告等操作。

会议管理模块: 具有会议管理权限的用户登进来之后, 能够新建会议信息, 上传会议资料, 查看会议信息等操作。

用户登陆模块: 用户进行一些必要的操作需要首先进行登陆。

1.主界面模块设计

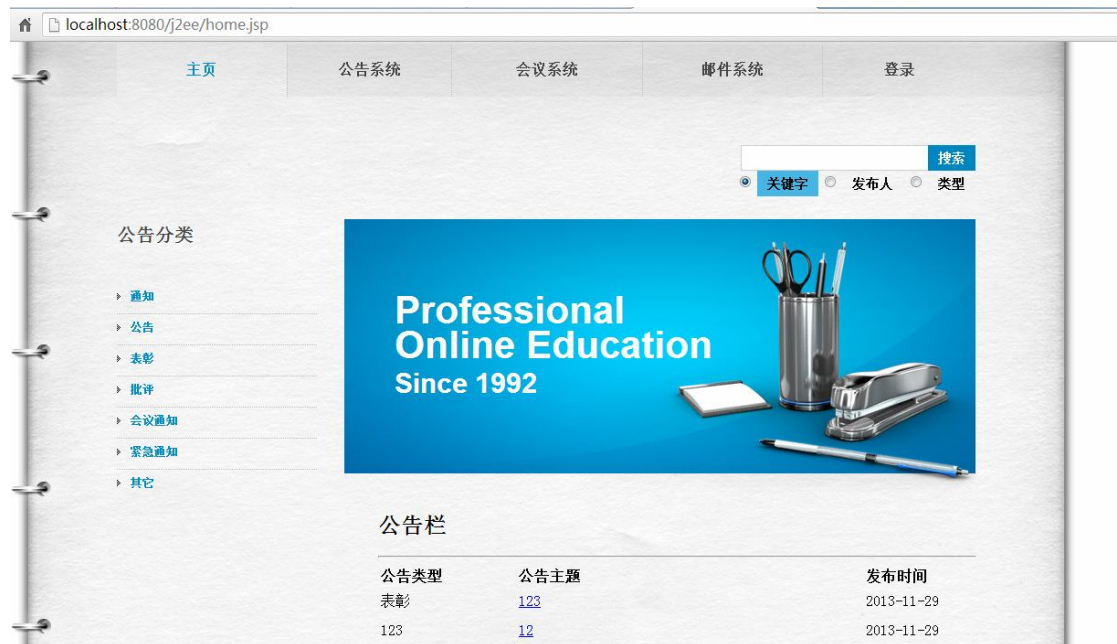


最终成果

Extjs 版:



后来版：



2.邮件系统模块设计

主页面

写信	_____

收信	
发件箱	

写信

	收件人	<input type="text"/>
	主题	<input type="text"/>
	正文	<div></div>
	附件	<input type="text"/>
		<div>发送</div>

收信

发件人	主题	时间		
			阅读	删除

最终成果

Extjs 版:



后来版:



3.通知公告模块设计

新建公告

查看公告

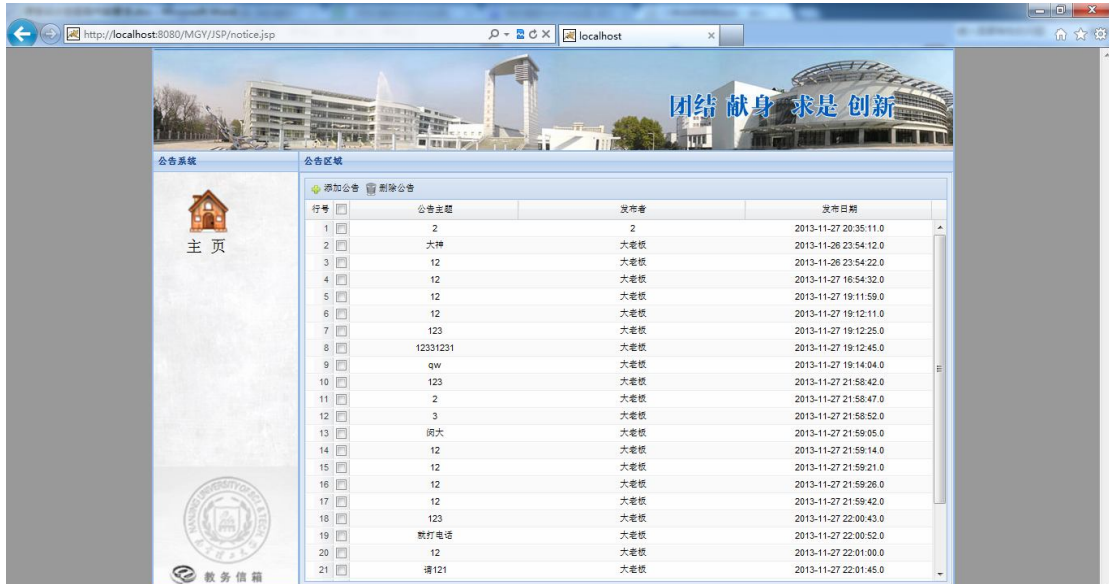
☐

☐

☐

最终成果

Extjs 版:



行号	公告主题	发布者	发布日期
1	2	2	2013-11-27 20:35:11.0
2	大神	大老铁	2013-11-26 23:54:12.0
3	12	大老铁	2013-11-26 23:54:22.0
4	12	大老铁	2013-11-27 16:54:32.0
5	12	大老铁	2013-11-27 19:11:59.0
6	12	大老铁	2013-11-27 19:12:11.0
7	123	大老铁	2013-11-27 19:12:25.0
8	12331231	大老铁	2013-11-27 19:12:45.0
9	qw	大老铁	2013-11-27 19:14:04.0
10	123	大老铁	2013-11-27 21:58:42.0
11	2	大老铁	2013-11-27 21:58:47.0
12	3	大老铁	2013-11-27 21:58:52.0
13	周大	大老铁	2013-11-27 21:59:05.0
14	12	大老铁	2013-11-27 21:59:14.0
15	12	大老铁	2013-11-27 21:59:21.0
16	12	大老铁	2013-11-27 21:59:26.0
17	12	大老铁	2013-11-27 21:59:42.0
18	123	大老铁	2013-11-27 22:00:43.0
19	群打电话	大老铁	2013-11-27 22:00:52.0
20	12	大老铁	2013-11-27 22:01:00.0
21	请121	大老铁	2013-11-27 22:01:45.0

后来版:

行号	公告主题	公告类型	发布者	发布日期
1	123	表彰	大老板	2013-11-29
2	12	123	大老板	2013-11-29
3	111111111111111	表彰	大老板	2013-11-29
4	12	1212	大老板	2013-11-29
5	12	什么	大老板	2013-11-28
6	请121	表彰	大老板	2013-11-27
7	12	紧急通知	大老板	2013-11-27
8	就打电话	表彰	大老板	2013-11-27
9	123	会议通知	大老板	2013-11-27
10	12	表彰	大老板	2013-11-27
11	12	会议通知	大老板	2013-11-27
12	12	会议通知	大老板	2013-11-27
13	12	批评	大老板	2013-11-27
14	闲大	表彰	大老板	2013-11-27
15	3	表彰	大老板	2013-11-27
16	2	公告	大老板	2013-11-27
17	123	通知	大老板	2013-11-27

4.会议管理模块设计

新建

上传附件

最终成果

Extjs 版:



后来版:

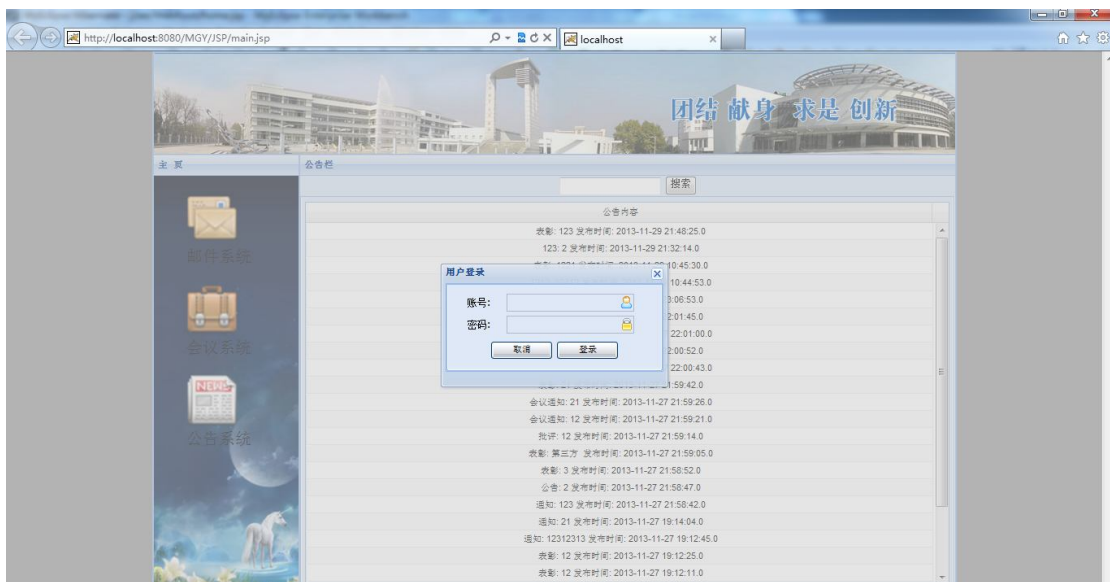


5.登陆模块设计

The diagram illustrates the layout for a login module. It consists of a large rectangular container. Inside, there are four rectangular input fields arranged in a 2x2 grid. At the bottom right of the container, there are two buttons: one labeled '登陆' (Login) and one labeled '取消' (Cancel).

最终成果

Extjs 版



后来版

localhost:8080/j2ee/login.jsp

主页 公告系统 会议系统 邮件系统 登录

用户名:

密码:

登录

4 项目开发

4.1 技术方案

1.项目模型：采用 b/s 模型，前台是浏览器，后台是服务器。

2.项目技术：

用途	采用的技术	采用原因
前台页面	主要 Jsp，部分 HTML	Jsp 可以根据数据库的变化动态显示页面，html 不用编译，减少服务器载荷
页面显示	Css	Css 可以使页面更加美观，而且一个 css 文件可以给多个页面同时使用，便于修改和样式同一
与用户交（信息提示）	Js 和 JQuery	Js 可以很方便地控制页面元素，实现与用户的交互，而 JQuery 提供了很多功能强大的函数调用，可以方便我们调用
为用户提供服务（上传附件等）	Extjs	Extjs 中有一个功能很强大的上传附件组件，可以先实现多个邮件同时上传，文件格式限制，文字大小限制等
前台与后台数据交互	主要: JQuery 中的 Ajax 辅助: Form 表单	Ajax 能实现与后台的异步交互，不用刷新页面，能提供及时信息更新，Form 表单比较传统方便
后台响应	Servlet	尽量不在 jsp 中写数据，实现表示层与业务层的分离
后台与数据库的交互	Hibernate, JavaBean	Hibernate 为我们数据库中的每个表单新建了一个 JavaBean，这样我们操作比较方便，而且便于移植，方便部署到其他

4.2 关键模块的实现

1 登录模块

模块描述: 该模块主要实现登录功能, 具体细节包括: 能够实现输入框的空值的判断, 能够与后台交互实现用户名和密码的判断, 如果错误则提示输入错误, 如果正确的话就跳转页面, 进入权限显示页面, 该页面提示用户所具有的权限, 并能进行注销。

实现技术: 通过 **Js** 获得输入框中的值, 进行空值判断。如果值不为空的话, 通过 **Jquery** 中的 **ajax** 技术把数据传给后台进行验证, 如果出错则弹出提示框, 正确的话跳入欢迎界面。

关键代码:

```
function add(){

    var flag=true;
    var password=this.password.value;
    var userName=this.userName.value;
    var str="";
    var flag1=false;

    if(password==""){
        str="请输入密码! ";
        flag1=true;
        document.getElementById("imply2").innerHTML="<a>请输入密码</a>";
        flag=false;
        this.password.focus();
    }
    else
        document.getElementById("imply2").innerHTML="";
}
```

```

if(userName==""){

    if(flag1)
        str="请填写用户名和密码! ";
    else
        str="请填写用户名! ";
    document.getElementById("imply1").innerHTML="<a>请填写用户名</a>";
    flag=false;
    this.userName.focus();
}
else
    document.getElementById("imply1").innerHTML="";

this.password.value="";

if(flag){

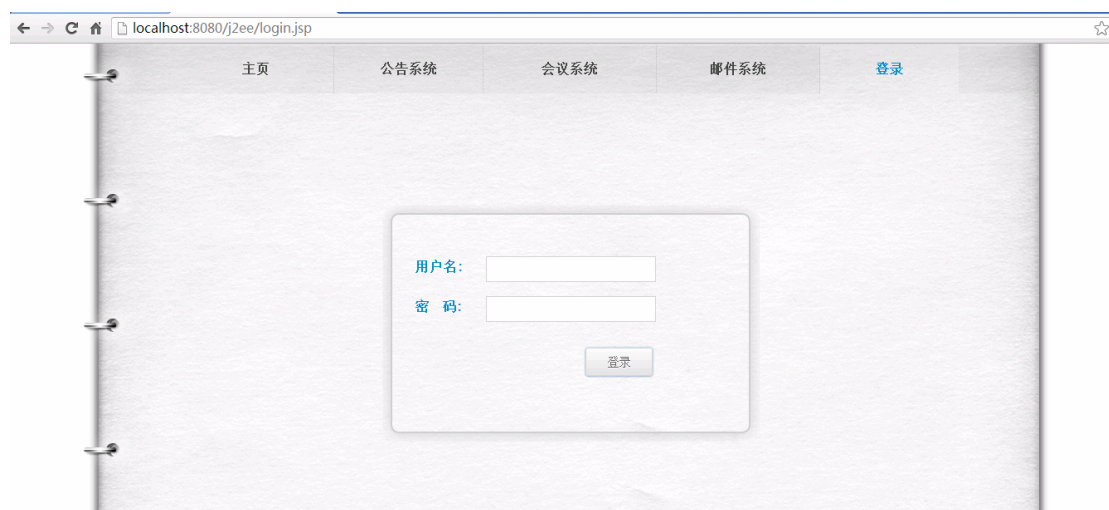
    $.ajax({
        url:"servlet/login",
        type:"post",
        async:false,
        data:"password="+password+"&userName="+userName,
        dataType:"text",
        success:function(data){
            if(data){
                show_loading();
                show_msg('登录成功咯! 正在为您跳转...', 'login.jsp');
            }else{
                show_loading();
                show_err_msg("登录失败, 请重新登录! ");
            }
        }
    });

}else{
    show_loading();
    show_err_msg(str);

}
}

```

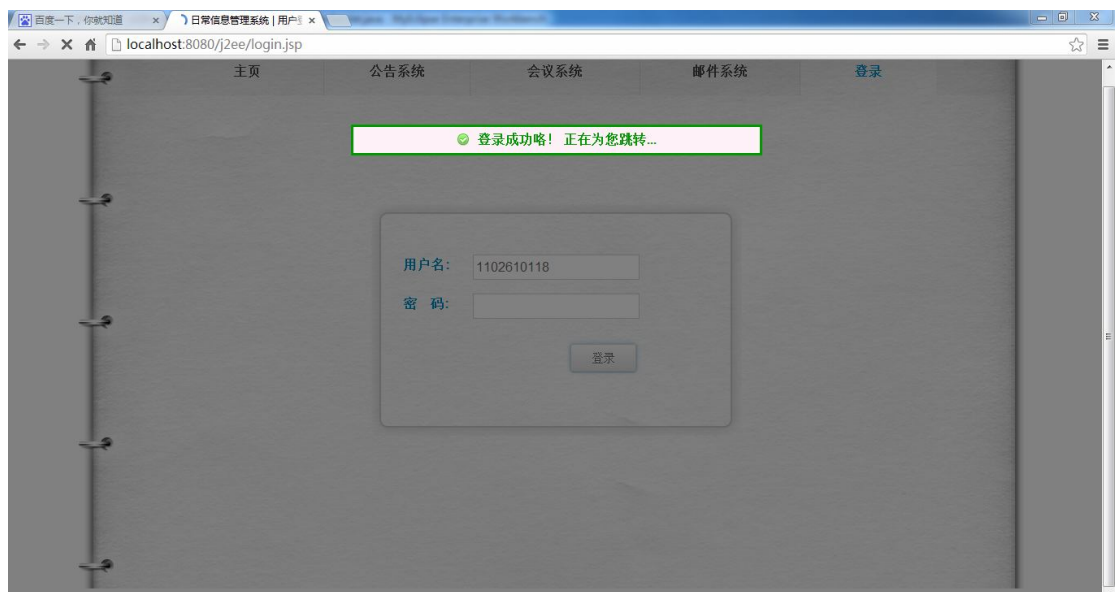
效果截图:



图一 登录界面



图二 未输入密码



图三 输入正确用户名和密码



图四 成功登陆后页面

2 搜索模块

模块描述: 该模块主要实现按照关键字, 发布人, 类型这三种方式进行搜索, 如果搜索内容为空值的话就把所有的公告显示出来。

实现技术: 通过 URL 把相应的搜索值传递给 search.jsp, 在这张 JSP 中通过 request 获得传递过来的参数值, 根据这个值查询数据库, 进行显示。

关键代码:

```
$("#a#search").click(function(){
    var flag=0;
    if(document.getElementById("radio1").checked)
        flag=1;
    if(document.getElementById("radio2").checked)
        flag=2;
    if(document.getElementById("radio3").checked)
        flag=3;

    document.getElementById("ifrmname").src="search.jsp?data="+search1.value+"&radio="+flag;
});

. . . . .

String s=request.getParameter("data");
String flag=request.getParameter("radio");
String str=new String(s.getBytes("ISO-8859-1"),"UTF-8");
BaseDao dao=new BaseDaoImpl();
List list=null;
if(flag.equals("1")){
    list=dao.list("from Notice where title like '%" +str+"%' or body like '%" +str+"%' or publisher like '%" +str+"%' or type like '%" +str+"%' order by id desc");
}
else if(flag.equals("2")){
    list=dao.list("from Notice where publisher like '%" +str+"%' order by id desc");
}
else{
    list=dao.list("from Notice where type like '%" +str+"%' order by id desc");
}
```

效果截图:

表彰

搜索

☐

关键字

☐

发布人

☒

类型

图五 搜索表彰

公告栏		
公告类型	公告主题	公告时间
表彰	123	2013-11-29
表彰	11111111111111	2013-11-29
表彰	请121	2013-11-27
表彰	就打电话	2013-11-27
表彰	12	2013-11-27
表彰	函太	2013-11-27
表彰	3	2013-11-27
表彰	123	2013-11-27
表彰	12	2013-11-27
表彰	12	2013-11-27
1		

图六 搜索结果显示

3 分类模块

模块描述: 该某块主要对公告进行分类, 用户点击类型名可以直接查看此该类型的所有公告, 不用在用搜索框搜索。

实现技术: 通过响应时间调用 Js 中的响应函数, 该函数把类型放在 Url 中, 传递给 allnotice.jsp。该 Jsp 得到该参数, 然后查询数据库显示。

关键代码:

```
$("#a#a1").click(function(){
    document.getElementById("ifrmname").src="allnotice.jsp?data=1";
});

. . . . .

String s=request.getParameter("data");
String choice="";
if(s.equals("1"))
    choice="通知";
else if(s.equals("2"))
    choice="公告";
else if(s.equals("3"))
    choice="表彰";
else if(s.equals("4"))
    choice="批评";
else if(s.equals("5"))
    choice="会议通知";
else if(s.equals("6"))
    choice="紧急通知";
else
    choice="其它";
BaseDao dao=new BaseDaoImpl();

String sqlStr="from Notice where type='"+choice+"' order by date desc";
if(s.equals("0"))
    sqlStr="from Notice order by date desc";

List noticelist=dao.list(sqlStr);

. . . . .
```

效果截图:



图七 公告栏分类

公告栏		
公告类型	公告主题	发布时间
会议通知	123	2013-11-27
会议通知	12	2013-11-27
会议通知	12	2013-11-27

1

回顶部

图八 点击“会议通知”后的公告栏

4 管理模块

模块描述:该模块主要是对会议,公告,邮件等进行双击查看和删除等操作。还有一些细节:如没有删除对象时提醒用户先选着要删除的对象;计入按时间顺序列出,还要进行分页等操作。

实现技术:通过 JSP 中的 java 代码查询数据库,获得数据利用表格显示,点击按钮时同过响应事件去调用 JS 代码,在 JS 执行相应的操作。

关键代码:

```
<form method="POST" name="form1">
    <table frame="hsides" cellspacing="0px" >
        <a class="button glow button-rounded button-flat button-tiny"
            style="cursor:pointer;margin-left: 3px;margin-top: 3px;margin-bottom:
3px;" onclick="on_delete()" >删除会议</a>
        <a class="button glow button-rounded button-flat button-tiny"
            style="cursor:pointer;margin-left: 3px;margin-top: 3px;margin-bottom:
3px;" onclick="on_addfj()" >上传附件</a>    &nbsp;

        . . . . .

    <tr bgcolor="" style="height:30px" >
        <th align="left" width="60px">&nbsp; <b><%= "行号"%> </b></th>
        <th align="center" width="40px">&nbsp;&nbsp;&nbsp;<input
type="checkbox" name="allcheck" onclick="checkAll()" id="allcheck"/></th>
        <th align="center"
width="160px">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b><%= "会议主题"%> </b></th>
        <th align="center" width="140px">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b><%= "
会议类型"%> </b></th>
        <th align="center" width="100px">&nbsp;&nbsp;&nbsp;<b><%= "主持者
"%> </b></th>
        <th align="center"
width="220px">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b><%= "会议日
期"%> </b></th></tr>
    <%String stl;
    for( int i=(nowpage-1)*gs;i<nowpage*gs;++i){
        if(i==list.size())
            break;
        Meeting meeting=(Meeting)list.get(i);
        if(i%2==0)
            stl="one";
        else
```

```

        stl="three";
    %>
    <tr class="<%=stl%" style="cursor:pointer"
ondblclick="lookDetail(this.id)" id="<%=meeting.getId()%">
        <td align="center" ><%=i+1%></td>
        <td align="center" ">
            &nbsp;&nbsp;&nbsp;<input type="checkbox" name="mycheck"
value="<%=meeting.getId() %>" />
        </td>
        <td align="center" ><%=meeting.getTitle()%"></td>
        <td align="center" ><%=meeting.getType()%"></td>
        <td align="center" ><%=meeting.getHost()%"></td>
        <td align="center" ><%=meeting.getDate()%"></td>
    </tr>
    <%}%>
</table><br/>
<input type="hidden" value="<%=list.size()%" id="num" name="num" /></form>

```

效果截图:



图九 会议管理模块显示

5 添加模块

模块描述: 添加模块能够添加相应的记录, 记录中的数据都要按照规定格式填写, 如果没有按规定格式填写, 则不能填写, 系统会给出提示。如果数据格式正确, 还要进行逻辑判断, 如果通过, 则添加成功, 添加成功后, 系统会告诉用户添加成功。

实现技术: 点击添加按钮, JS 脚本获得相应数据, 进行初步验证(是否为空, 格式验证), 如果成功通过 JQuery 中的 ajax 技术把部分数据传到后台, 进行逻辑上的验证, 如果成功, 则把全部数据传递到后台添加。

关键代码:

```
function addMeeting(){

    //var str="";
    //获得属性值
    var title=this.title.value;

    . . . . .

    //进行验证
    //空值验证
    var flag=true;

    if(meetThing==""){
        document.getElementById("imply10").innerHTML="<a>请填写会议内容
</a>";
        flag=false;
        this.meetThing.focus();
    }else
        document.getElementById("imply10").innerHTML="";

    . . . . .

    //逻辑验证,时间的问题,异步去读时间
    var t_end=endTimeh+endTimem;
    var t_start=startTimeh+startTimem;

    if(t_end<=t_start){

document.getElementById("imply5").innerHTML="<a>结束时间必须晚于开始时间
```



```

</a>";
        flag=false;
        this.endTime.focus();
    }else{
        document.getElementById("imply5").innerHTML="";
    }

    if(!flag)
    return;

    //参数传递

    .....
    var isOld=false;
    if(syear<nyear){
        isOld=true;
    }else{
        if(syear==nyear){
            if((smoonth*30+sdate)<=(nmonth*30+ndate))
                isOld=true;
        }
    }
    if(isOld){//不是今明两年
        document.getElementById("imply3").innerHTML="<a>请选择今天之后的
时间</a>";
        flag=false;
        this.date.focus();
    }else
        document.getElementById("imply3").innerHTML="";

    if(!flag)
    return;

    var startTime=syear+smoonth0+sdate0+startTimeh+startTimem;
    var endTime=syear+smoonth0+sdate0+endTimeh+endTimem;

    mainThing=mainThing.replace(/\n/g,'<br />');
    var daiban="有";
    if(document.getElementById("content1").checked)
        daiban="无";

    //时间冲突验证

```

```

Ext.Ajax.request({
    url: 'servlet/timeCheck',
    method: 'POST',
    async: false,
    params: {startTime: startTime, endTime: endTime, place: place},
    success: function(response){

        if(response.responseText){

            document.getElementById("imply4").innerHTML="";
            var stime=startTimeh+":"+startTimem;
            var etime=endTimeeh+":"+endTimem;

            //添加
            Ext.Ajax.request({
                url: 'servlet/addMeeting',
                method: 'POST',

                params: {daiban: daiban, title: title, type: type, date: date, startTime: startTi
me, endTime: endTime,

                place: place, host: host, groupId: groupId, mainThing: mainThing, meetThing: mee
tThing, stime: stime, etime: etime},
                success: function(response){
                    form1.action="meet_success.html";
                    form1.submit();
                },
                failure: function(response){

                }

            });

        }else{
            document.getElementById("imply4").innerHTML="<a>时间
冲突，请重选时间或地点! </a>";

            flag=false;
        }
    },
    failure: function(response){

    }

});
}

```

效果截图:

The image displays two screenshots of a web application interface for adding a meeting. The top screenshot shows the '添加会议' (Add Meeting) form. On the left, there is a sidebar with '会议系统' (Meeting System) and two sub-items: '管理会议' (Manage Meeting) and '发布会议' (Publish Meeting). The main form area contains the following fields:

- 会议主题: (0/50) [Text input]
- 会议类型: [Text input]
- 会议日期: [Text input]
- 开始时间: [Time picker] 点 [Time picker] 分
- 结束时间: [Time picker] 点 [Time picker] 分
- 地点: [Text input]
- 主持者: [Text input]
- 参会人员: [Dropdown menu]
- 待办事项: ☒ 无 ☐ 有
- 主要事项: [Large text area]

The bottom screenshot shows the '会议内容' (Meeting Content) section, which includes a large text area for entering the meeting details. At the bottom right of this section is a '提交' (Submit) button. Both screenshots include a '回顶部' (Back to Top) button in the bottom right corner.

图 十, 十一 添加会议模块

管理会议

发布会议

会议主题:

会议类型:

会议日期:

开始时间:

结束时间:

地点:

主持者:

参会人员:

待办事项:

主要事项:

(0/50) 请填写公告主题

请填写公告类型

请选择日期

请选择 点 请选择 分 请选择开始时间

请选择 点 请选择 分 请选择结束时间

请填写地点

请填写主持者

请选择参会人员

☒ 无 ☐ 有

1:

请填写主要事项

会议内容:

返回顶部

图 十二 添加会议空值判断

会议系统

管理会议

发布会议

添加会议

会议主题:

会议类型:

会议日期:

开始时间:

结束时间:

地点:

主持者:

参会人员:

待办事项:

主要事项:

12 (2/50)

12

2013-12-02

06 点 00 分 时间冲突, 请重选时间或地点!

06 点 10 分

12

郭一迪

科研A组; 闵大为; 郭一迪;

☒ 无 ☐ 有

1: 1

返回顶部

图 十三 添加会议逻辑判断



图 十四 会议添加成功提示

4.3 特殊问题及解决办法

1. 附件上传及与邮件对应

问题描述：要把附件传递到后台去，并且与相应的邮件对应，得考虑到用户上传附件后并不一定添加邮件。

解决方案：对于附件上传，我调用了别人的一个 `extjs` 组件，该组件能够很好地完成上传附件的任务。其次，邮件和附件对应的问题，我用时间作为标记，在添加邮件时，会去找那些时间对应的附件，绑定在一起。

关键代码:

```
function upLoad(){

    var addTime=this.addTime.value;
    //上传文件框
    var dialog = new Ext.ux.UploadDialog.Dialog({
        title: '上传附件',
        autoCreate: true,
        closable: true,
        collapsible: false,
        draggable: false,
        minWidth: 400,
        minHeight: 200,
        width: 400,
        height: 350,

        permitted_extensions:['JPG','jpg','jpeg','JPEG','GIF','gif','pdf','txt','jar',
        'rar','doc','wps'],
        proxyDrag: true,
        resizable: true,
        constrainttoviewport: true,
        url: 'servlet/emailFjUpLoad?addTime='+addTime,
        reset_on_hide: false,
        allow_close_on_upload: true
    });
    dialog.show('show-button');

    }

    . . . . .

    String receiverAdress=request.getParameter("receiverAdress");
    System.out.println(receiverAdress);
    String title=request.getParameter("title");
    System.out.println(new
String(title.getBytes("ISO-8859-1"),"UTF-8"));
    String body=request.getParameter("body");
```

```

System.out.println(body);
Date date = new Date();
String addTime=request.getParameter("addTime");

HttpSession session = request.getSession();
User user=(User)session.getAttribute("user");
System.out.println(user.getUserName());
System.out.println(user.getName());

Email email=new Email();
email.setReceiverAdress(new
String(receiverAdress.getBytes("ISO-8859-1"),"UTF-8"));
email.setSenderAdress(user.getUserName());
email.setTitle(new String(title.getBytes("ISO-8859-1"),"UTF-8"));
email.setBody(new String(body.getBytes("ISO-8859-1"),"UTF-8"));
email.setDate(new Timestamp(date.getTime()));
email.setSender(user.getName());
email.setIsReaded("0");
email.setRecDelete("0");
email.setSendDelete("0");

BaseDao dao=new BaseDaoImpl();

List emailfjs=dao.list("from Emailfj where addTime="+addTime);
email.setFjNum(new Integer(emailfjs.size()));
dao.save(email);

for(int i=0;i<emailfjs.size();++i){
    Emailfj emailfj=(Emailfj)emailfjs.get(i);
    emailfj.setEmailId(email.getId());
    dao.save(emailfj);
}

```


效果截图：



图 十五 添加附件

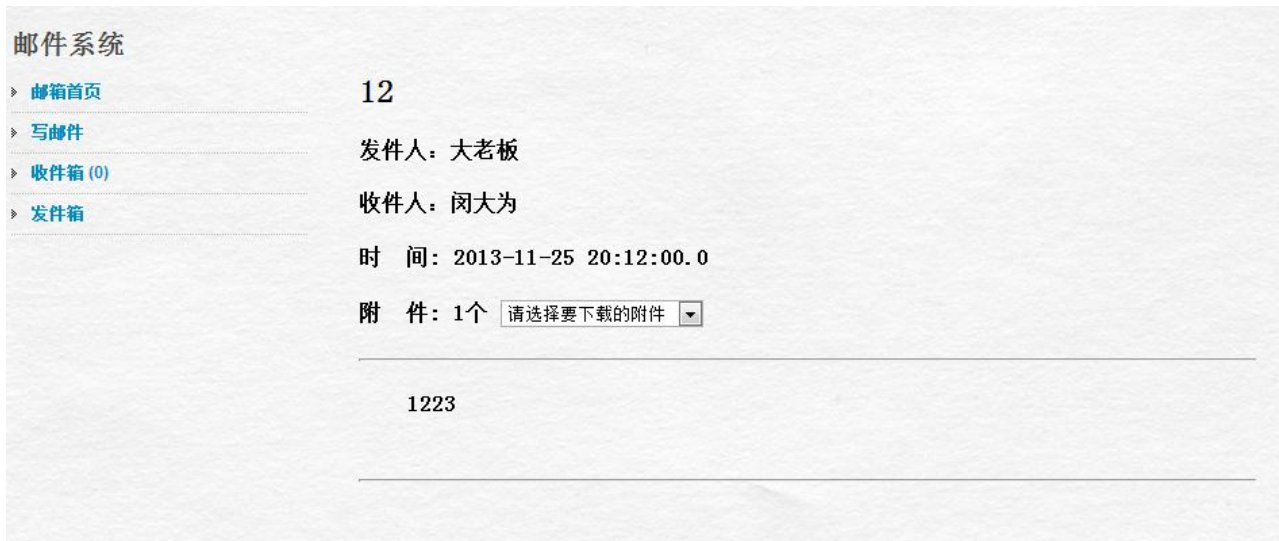


图 十六 下载附件

2. 会议提醒页

问题描述: 如果有人新建会议, 并且该人在这个会议中, 则他登录到系统后, 应该收到会议提醒。

解决方案: 在每个页面中嵌入 JS 代码, 设置定时器, 不断与后台进行异步数据交互, 查询有无新的会议提醒。

关键代码:

```
var handler = function(){
    $.ajax({
        url:"servlet/haveMeet",
        type:"get",
        async:false,
        dataType:"text",
        success:function(data){
            if(data){

document.getElementById('light').style.display='block';

document.getElementById('fade').style.display='block';
                $.ajax({
                    url:"servlet/haveMeet",
                    type:"post",
                    async:false,
                    dataType:"text",
                    success:function(data){}
                });
            }
        }
    });

};

var timer = setInterval( handler , 10000);
```

效果截图：



图 十七 会议提醒

3. 面局部刷新

问题描述： 当我们点击导航块按钮时，不是通过跳转页面实现页面的刷新，而是通过改页面的局部内容实现页面的改变。

解决方案： 在页面中使用 `iframe` 框架，点击按钮时，通过点击事件,调用 JS 函数，在 JS 中改变 `iframe` 的源头。

关键代码：

```
<ul class="categories">
    <li><span><a href="#" onclick="manageNotice()">管理公告
</a></span></li>
    <li><span><a href="#" onclick="addNotice()" >发布公告
</a></span></li>
```

```
</ul>
```

• • • • •

```
//管理公告
```

```
function manageNotice(){  
    //alert("管理公告");  
    document.getElementById("ifrmname").src="manageNotice.jsp";  
  
}
```

```
//添加公告
```

```
function addNotice(){  
    //alert("添加公告");  
    document.getElementById("ifrmname").src="addNotice.jsp";  
}
```

4. 日期时间选择

问题描述：要实现日期的选择。

解决方案：本来使用 Extjs 中的组件，但是后来为了使界面更加美观，决定减少使用 Extjs 组件，所以经过学习，下了一个 JS 组件。

关键代码：

```
<script type="text/javascript" src="js/date.js" charset="GBK"></script>
```

• • • • •

```
<div class="field">
```

```
    <label>会议日期: </label>
```

```
    <input id="date" name="date" type="text" style="border:1px  
solid #999;" onclick="fPopCalendar(event,this,this)" onfocus="this.select()"  
readonly="readonly" />
```

```
    <span id="imply3"></span>
```

```
</div>
```

效果截图：

添加会议

会议主题: (0/50)

会议类型:

会议日期:

开始时间:

结束时间:

地点:

主持人:

参会人员:

待办事项:

主要事项:

2013年 十二月

今天: 2013年12月3日

无 有

返回顶部

图 十八 日期选择

5. 文本框支持可选和可输入

问题描述：使得文本框既能够选择又能够输入。

解决方案：自己写 JS 代码，响应文本框的事件，实现文本框的输入选择要求。

关键代码：

```
<div class="field">
    <label>主持人: </label>
    <input id="host" name="host" type="text"
style="width:278px;height:25px;font-size:16;" onclick="show()"
```

```

onkeyup="change()">
    <div id="select" style="position: absolute; display: none">
        <%
            List users=dao.list("from User");
            for(int i=0;i<users.size();++i){
                User user=(User)users.get(i);
                int j=i+1;
                String unmae=user.getName();
            %>
            <div id="item<%=j%>" class='itemBg'
onmouseover="beMouseOver(<%=j%>)" onmouseout="beMouseOut(<%=j%>)"
onclick="beClick(<%=j%>)"><%=user.getName()%></div>
            <input type="hidden" id="value<%=j%>"
value="<%=user.getName()%>"
            <%} %>
            <div class="item_button"></div>
        </div>
        <input id="hostnum" type="hidden"
value="<%=users.size()+1%>"
        <span id="imply7"></span>
    </div>

//定位DIV显示
function positionDiv()
{
    var DivRef= document.getElementById("select");
    DivRef.style.position = "absolute";
    var t=document.getElementById("host");
    DivRef.style.top= getAbsolutePos(t).y+25;//相对文本框的TOP高度，
方法见下面
    DivRef.style.left= getAbsolutePos(t).x ;//相对文本框的LEFT宽度
    var hostNum=document.getElementById("hostnum");
    DivRef.style.height=hostNum * 25;//DIV的高度等于每行20个像素乘行数
    (也就是数组的长度，体现全局数组的作用，不然后要传入数组长度的参数)
}
//定位方法，不做解释
function getAbsolutePos(el)
{
    var SL = 0, ST = 0;
    var is_div = /^div$/i.test(el.tagName);
    if (is_div && el.scrollLeft) SL = el.scrollLeft;
    if (is_div && el.scrollTop) ST = el.scrollTop;
    var r = { x: el.offsetLeft - SL, y: el.offsetTop - ST};

```

```

        if (el.offsetParent)
        {
            var tmp = this.getAbsolutePos(el.offsetParent);
            r.x += tmp.x;
            r.y += tmp.y;
        }
        return r;
    }
}

function hid(){
    document.getElementById("select").style.display="none";
}

function show(){
    positionDiv();
    document.getElementById("select").style.display="inline";
}

function change(){
    if(document.getElementById("host").value.length>0)
        document.getElementById("select").style.display="none";
    else
        document.getElementById("select").style.display="inline";
}

function beMouseOverEFF(i)
{
    if(i>0)
        document.getElementById("item" + i).className="item_high";

}

//函数鼠标移开效果
function beMouseOutEFF(i)
{
    if(i>0){
        document.getElementById("item" + i).className="item_normal";
    }

}

//函数鼠标经过
function beMouseOver(i)
{
    document.getElementById("host").focus();
    beMouseOutEFF(z);
    z=i;
    beMouseOverEFF(z);
}

```

```

}

//函数鼠标移开
function beMouseOut(i)
{
    beMouseOutEFF(i);
}
//函数单击:
function beClick(i)
{
    var myid="value"+i;
    document.getElementById("host").value=document.getElementById(myid).value;
    document.getElementById("select").style.display="none";
    document.getElementById("host").focus();
}
}

```

效果截图:

图 十九 可输入文本框

6. 换行自动输出序号

问题描述：添加会议纪要时，当使用 **Enter** 键换行时，能够实现自动输入行号。

解决方案： 响应文本框的事件，通过 JS 代码控制文本框的内容。

关键代码：

```
<div class="field extra" style="margin-top: 10px">
  <label>主要事项:</label>
  <textarea id="mainThing" name="mainThing" cols="1" rows="1"
onkeyup="hh()" onfocus="hh0()"></textarea>
  <br/>
  <span style="margin-left: 125px" id="imply9"></span>
</div>

<input type="hidden" name="mainthing_num" id="mainthing_num" value="0">

function hh0(){
  if(document.getElementById("mainThing").value.length==0){
    document.getElementById("mainthing_num").value=1;
    document.getElementById("mainThing").value+="1: ";
  }

}

function hh(){
  if(event.keyCode == 13){
    var
value=parseInt(document.getElementById("mainthing_num").value);
    value+=1;
    document.getElementById("mainThing").value+=value+": ";
    document.getElementById("mainthing_num").value=value;
  }

}
```


效果截图：

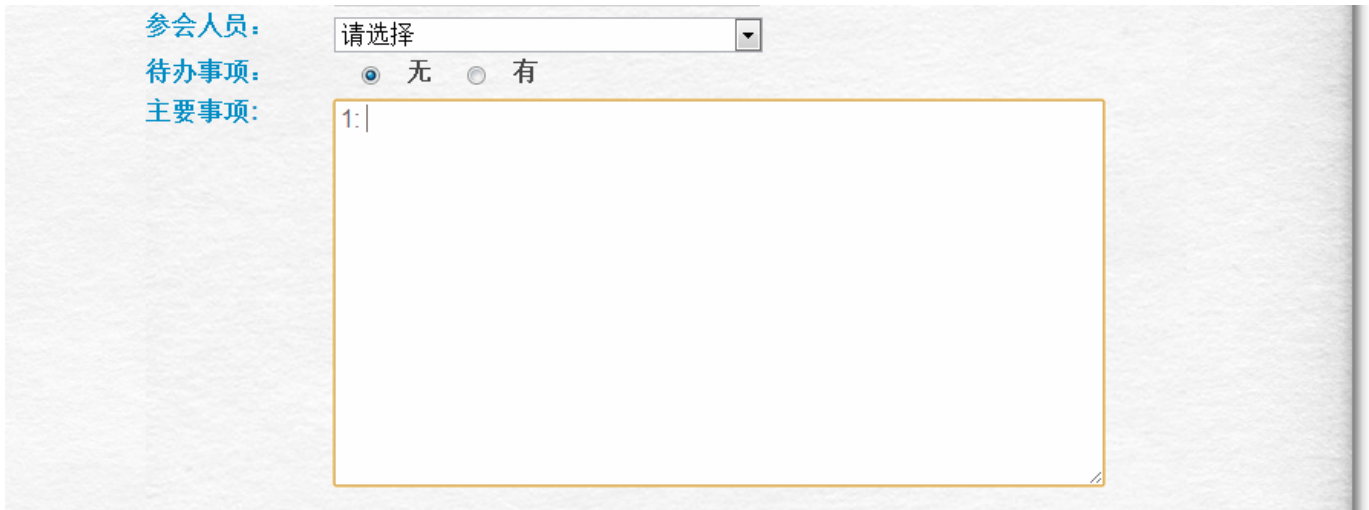


图 二十 换行自动标注

7. 分页问题

问题描述：当展示的数据过多的时候，能够实现分页功能。

解决方案：通过进入页面时传递过来的参数，动态显示数据。

关键代码：

```
<!-- 分页 -->
<%
    int gs=20;//行数
    int nowpage=1;//当前页
    if(request.getParameter("page")!=null)
        nowpage=Integer.parseInt(new
String(request.getParameter("page").getBytes("ISO-8859-1"),"UTF-8"));

    BaseDao dao=new BaseDaoImpl();
    List list=dao.list("from Notice order by id desc");

    int totalNum=list.size();
    int pageNum=totalNum/gs;//总共多少页
```

```

        if(totalNum%gs!=0)
            pageNum+=1;
        String str0="fy_hover";
        String str1="fy_normal";
        String strClass=str1;

        for(int i=0;i<pageNum;++i){//先无下一页
            if(i==(nowpage-1))
                strClass=str0;
            else
                strClass=str1;

            %>
            <span title="第<%=i+1%>页"    class="<%=strClass%>"
onclick="fy(this.id)" id="<%=i+1 %>">&nbsp;&nbsp;&nbsp;<%=i+1 %>&nbsp;&nbsp;&nbsp;</span>

            <%} %>

        . . . . .
    <%
String stl;
for( int i=(nowpage-1)*gs;i<nowpage*gs;++i){
    if(i==list.size())
        break;
    Notice notice=(Notice)list.get(i);
    if(i%2==0)
        stl="one";
    else
        stl="three";
    %>

    <tr class="<%=stl%>" style="cursor:pointer"
ondblclick="lookDetail(this.id)" id="<%=notice.getId()%>">
        <td align="center" ><%=i+1%></td>
        <td align="center" ">
            <input type="checkbox" name="mycheck"
value="<%=notice.getId() %>" />
        </td>
        <td align="center" ><%=notice.getTitle()%></td>
        <td align="center" ><%=notice.getType()%></td>
        <td align="center" ><%=notice.getPublisher()%></td>
        <td align="center"
><%=notice.getDate().getYear()+1900%>-<%=notice.getDate().getMonth()+1%>-<%=
notice.getDate().getDate()%></td>
    </tr>
    <%}%>

```

</table>

效果截图：

删除公告					
1 2					
行号	<input type="checkbox"/>	公告主题	公告类型	发布者	发布日期
1	<input type="checkbox"/>	123	表彰	大老板	2013-11-29
2	<input type="checkbox"/>	12	123	大老板	2013-11-29
3	<input type="checkbox"/>	111111111111111	表彰	大老板	2013-11-29
4	<input type="checkbox"/>	12	1212	大老板	2013-11-29
5	<input type="checkbox"/>	12	什么	大老板	2013-11-28
6	<input type="checkbox"/>	请121	表彰	大老板	2013-11-27
7	<input type="checkbox"/>	12	紧急通知	大老板	2013-11-27
8	<input type="checkbox"/>	就打电话	表彰	大老板	2013-11-27
9	<input type="checkbox"/>	123	会议通知	大老板	2013-11-27
10	<input type="checkbox"/>	12	表彰	大老板	2013-11-27
11	<input type="checkbox"/>	12	会议通知	大老板	2013-11-27
12	<input type="checkbox"/>	12	会议通知	大老板	2013-11-27

图 二十一 分页

8 CSS 中的类冲突问题

问题描述：一些类样式冲突。

解决方案：新建 **CSS** 类文件并在页面代码中进行连接，在相应的内容前定义类型

解决结果：解决类冲突而引起的效果冲突消失问题，设置成理想样式。

9 一些样式问题

问题描述：个别标签或按钮样式不能修改且不符和整体风格。

解决方案：用做好的图片代替

解决结果：由于制作技术问题导致和背景不太协调，但总体效果还好。

10 CSS 中的类冲突问题

问题描述：部分好的标签文字特效不能直接应用。

解决方案：找到其相应的 CSS 中的类加入自己新建的 CSS 文件中并关联使用。

解决结果：能够正确应用并达到预期效果。

5 个人体会

5.1 闵大为的个人小结

这次课设刚开始时我和小组成员一起仔细研读了老师给的项目要求，经过漫长和激烈的讨论，我们最终确定了我们项目的总体样式，我们画了界面的草图，明确了不同用户的权限，确定了页面之间的跳转关系，随后大家一起进行了数据库设计。在完成了这些前期工作后，我们分配了任务，我主要编写邮件系统和会议系统这两个模块。

在刚学期刚开始时，我主要采用的是 Extjs 来编写前台，在刚开始编写时，我对 JS 还不是很熟悉，在敲代码时，常常多敲一个逗号或者敲错一个字母，这些错误常常会使的程序显示出现错误，但在检查错误时却很难检查出这些细微的错误，这导致刚开始时一个逗号或者一个字母，常常让我的代码编写陷入停滞状态，短则耽误几分钟，长则耽误几个小时，有时这些错误让我几乎抓狂。由于刚开始时，我对 Extjs 中的异步交互不太熟悉，与后台的交互代码也写得异常辛苦。但是后来随着解决的错误增多，我对 Extjs 的掌握逐渐熟练，编程速度也渐渐加快，在编程的过程中，我感受到了 Extjs 强大的功能，很多界面元素 Extjs 都帮我们做好了，我们只需要调用一下就可以了。但是 Extjs 让我们头疼的一点是它的界面比较朴素，如果我要修改它的样式就要修改它的 Css 文件，这是一项十分庞大修改任务，而且修改出来的样式不一定好看，于是我们小组讨论后，果断抛弃 Extjs，转战不用 Extjs 的 JSP。

虽然刚开始的版本让我们积累了一定的编程经验，但是离开 Extjs，要实现 Extjs 提供的功能，这对我们对 JS 脚本掌握程度的要求大大提高了，而且要自己搭配一些元素，使得界面变得好看。这是一个很痛苦的过程，但是在小组成员共同努力和探讨下，我们渐渐步入正轨了，我们在新的版本中还实现了一些原来没有实现的功能，比如邮件提醒，登录后注销，输入字符长度提示等等。虽然这些都是一些小的细节，但是能够很好地增加用户体验。虽然我们后来的版本，跟一些大的门户网站没法比，但是经过我们的努力，我们对我们的项目还是十分满意的。

这次课设，让我收获了很多。首先，在完成项目的过程中，我对 Extjs 有了进一步的了解，在学习 Extjs 的过程中，我知道了 JS 可以做很多事情。我对 Hibernate 也有了一定的了解，在移植项目时我感到了 Hibernate 的方便，在使用数据库时我更加感到了 Hibernate 把数据库中的表映射成一个 JavaBean 的好处，它使得我们操作数据库更加的方便。其次，我的 Servlet 和 JSP 的编写能力也有了进一步提升，在实战中发现，如果巧妙的使用一些代码，便能很轻松的实现一些功能。接着，更重要的是，在这次项目中，我们遇到了很多困难，但是我们小组成员一起讨论，一起解决，一个人的力量是有限的，但大家一起团结起来，我们的力量就变得非常足了，而且大家在这次项目中做到了很好的沟通，遇到问题困难时也能相互理解，相互鼓劲，相互帮助的，我还是很感动的。最后，我又完成了一个作品，对于此次项目，我还是比较满意的。

在此，十分感谢小组成员的辛勤付出，正是大家的辛勤劳动，才使我们能够很好地完成项目。十分感谢徐老师的悉心指导，正是徐老师 J2EE 课上的详细讲解，才使得我们在编程过程中走了很少弯路，实现了很多功能。

5.2 郭一迪的个人小结

经过团队一起的努力，我们终于把 J2EE 课设完成了，回过头来想想在写课设过程中的一些经历，颇有感触。

开学初，当确定课设题目之后，我们小组成员就一起开始讨论这个项目的大致内容，在经历了大家的讨论之后，我们最终确定了这个项目的总体设计和数据

库方面的设计，为后续的工作打下了坚实的基础。在这个过程中给我感触颇深的就是，以前写课设的时候基本上是一个人，所以项目总体设计以及数据库设计都按照个人意愿来的，现在由于是团队合作，每个人都有自己的想法，所以在讨论的过程中免不了有一些分歧，于是，我们需要更好的把自己的观点表达和阐述给团队其他成员听，这就需要我们有很好的沟通能力和团队适应能力，这样才能使团队合作更好。

在确定完项目的总体设计之后，我们便分工开始了各自的工作。在这次 J2EE 课设中，我负责其中的主页模块设计和公告模块的设计，以及整个项目中一些小的功能方面的优化。在工作过程中，我也经常遇到一些问题和技術瓶颈，这就促使我不断的去学习。在编写过程中，jquery 和 js 是必须的，所以我就花了一部分时间先学习了一下 jquery 和 js 方面的内容，在解决登陆以及其他一些方面运用异步调用的技术，我又去学习了一下 ajax 方面的内容。又比如在 select 下拉框的运用中，在公告和会议中要用到手动输入和下拉选择 2 中功能，开始我先用 select 做出了手动输入功能，但是事后发现不能输入汉字，于是我只能用 div 块重新做这方面的内容，而这其中少不了 js 的运用。总之在编写程序的过程中，也是一个不断学习的过程，它敦促着我们不断的去获取知识。当然，在整个项目的编写进程中，我们也会或多或少的遇到一些设计问题，这时，我们一般会采取小组讨论的方式，从而决定出最终的解决方案。三个人的力量毕竟要比一个人的力量强很多。

作为项目的编程人员，各自完成自己的模块部分并不是一个大问题，但是后期的项目整合是一个大问题。由于以前团队合作的经验不多，再加上没有系统学过团队合作过程中的一些规范和注意事项，每个人编写代码的习惯也不一样，于是，在整合的过程中就问题百出，一些页面在自己的电脑上能正常运行，没有问题，但是整合到别人电脑上却运行不了了。在这种情况下，我们便不得不仔细的检查代码，寻找一些出错的地方，有些很细小的问题便是 2 个人写的 CSS 中有冲突，导致最终的显示出现了问题。不得不说，在整合方面的问题是一个很麻烦的问题，就比如可能因为一个小小的地方的大小写不对应，就导致程序出现问题，而这些方面的检查需要清晰的头脑和仔细的态度才能发现。所幸，最后在大家的集体努力下，我们最终把项目很好的正好在一起，这离不开大家好的配合和认真的态度。

这次课设最终圆满的完成，而我也从这次课设中学习和体会到了很多东西。首先，我们要更好的适应团队合作。团队合作是我们今后走进计算机领域所必须面对的，一个人的力量再强大，那也是有限的，要完成一个大的项目，必须通过团队的力量，我们要相信团队成员，更好的和他们配合。因此，如何更好的和他人沟通，如何把自己的程序写的更规范，是我今后所要学习的内容。其次，我们要不断的提高自己。计算机领域的知识瞬息万变，要想在这方面做的很好，我们只能不断的学习，不断的汲取新知识，充实自己，这样你才不会被这个时代淘汰。最后，我觉得作为一个编程人员，我们应该更注重自己的创意，编写代码人人都会，但是并不是所有人都是有想法有创意的。社会中，那些成功人士无不都是有创意，能抓准时机的人。因此在培养自己编程能力的同时，我们要不断提高自己的思维能力，培养自己的创新能力，只有这样，我们才能做的更好。

最后，非常感谢徐建老师这一年来的悉心教导！

5.3 于新哲的个人小结

在此次的课设开发过程中，让我对 **WEB** 开发又有了新的认识，通过前期的 **EXT JS** 以及后期的 **HTML CSS** 不同的开发过程中，让我对这两种框架都有了深入的了解，尤其在用 **HTML CSS** 制作页面的过程中，让我发现了和以前制作视频一样的乐趣，有了 **CSS** 的支持，可以让页面变得炫目多彩。当然了，现实中的编码也不是说说这么简单的，也涉及到很多东西。这次课设和以往的一样，又让我学到了很多知识，这些东西有很多不是能通过上课和读书能得到的，是真正一行行代码敲出来，一行行复制粘贴调整调试出来的经验和知识，让我能真正掌握的知识，能在以后的编程中用到的知识。虽然目前深层的算法和代码有的没有用到有的只是运用了很少，但是页面效果相关的代码还是掌握了很多的。这也让我在开发的过程中多 **CSS** 和 **HTML** 越来越感兴趣。

其次，除了页面代码的学习，关于页面的设计及排版，我也吸取了很多经验，对于页面的布局，相应颜色的搭配，字体的大小效果等等都要做考虑，尽量使得一个页面的整体效果能融合一体。通过借鉴网上的例子和模板，再加上自己的理解与改动，尽量让我们的页面有自己的特色和风格。我们小组最初的设想是根据日常信息管理模块本身是用来办公的，所以整体风格要尽量简洁清晰又不失特点，所以整体的背景选取了笔记簿的图片作为背景，符合办公的环境。页面的分页也清晰的用简单大方的默认字体显示在每个页面最显眼的位置，方便用户能够随意选择想要的功能区。在每个页面中我们也尽量显得大方得体，没有过多的用特别炫目的特效进行处理，而是简单的运用文字的颜色搭配和布局来达到简约的效果。并在个别按钮中运用了一定的效果，使得整体都能给人一种简洁方便大气的效果！

最后，用过这次的课设开发，让我对界面设计，数据库设计，**HTML CSS** 语言的运用等都有了新的认识，在之前相关知识的模糊感现在渐渐变得清晰了。在前期的共同讨论研究中让我对一个程序或者网页项目的开发中所涉及的设计方面的知识也增加了不少，包括数据库的设计，页面布局的设计，通过小组三个人的努力把方方面面的尽量想的周全尽善尽美一些。事实证明前期的设计也确实能够在很大程度上减轻我们在后期编程具体功能实现时候的压力，能够尽量按照预计的目标一步一步完成。同时也让我对团队协作充满了信心，因为有闵大为和郭一迪所以课设的难度减轻了不少，虽然我在编程方面做得不如他们多，但通过前期的共同学习也些许补足了一下这方面的缺口，对于后期的代码整合等起了积极的作用。从大一到现在的每一次课设都能让我的编程能力和其他能力有所提升，也逐渐让我对自己的专业一次次的加深理解，让我对日后的方向有了越来越明确的认识。相信每一次的课设的经验都会是我日后最宝贵的财富。