**上海电力学院**

**Java EE课程设计**



**题 目: 课程资源管理系统**

**学 号： 20131809**

**姓 名： 黄海琨**

**院 系： 计算机科学与技术学院**

**专业年级： 计算机科学与技术2013级**

**2016年6月6日**

**目录**

[1 设计题目 1](#_Toc453067729)

[1.1 实验题目 1](#_Toc453067730)

[1.2 问题描述 1](#_Toc453067731)

[1.3 基本要求 1](#_Toc453067732)

[1.3.1 系统主要功能： 1](#_Toc453067733)

[2 需求分析 1](#_Toc453067734)

[2.1 功能描述 1](#_Toc453067735)

[2.1.1 管理员 1](#_Toc453067736)

[2.1.2 教师 1](#_Toc453067737)

[2.1.3 普通用户 1](#_Toc453067738)

[2.2 组织结构 2](#_Toc453067739)

[2.3 功能详情 2](#_Toc453067740)

[2.3.1 管理员 2](#_Toc453067741)

[2.3.2 教师 4](#_Toc453067742)

[2.3.3 普通用户 6](#_Toc453067743)

[2.4 数据库设计 9](#_Toc453067744)

[2.4.1 数据库E-R图 9](#_Toc453067745)

[2.4.2 数据库描述 9](#_Toc453067746)

[2.4.3 关系模型 9](#_Toc453067747)

[2.4.4 数据库表 10](#_Toc453067748)

[3 详细设计 12](#_Toc453067749)

[3.1 文件上传 12](#_Toc453067750)

[3.2 批量导入 15](#_Toc453067751)

[4 关键处和创新点 17](#_Toc453067752)

[4.1 SSH（Spring+Struts2+Hibernate开源框架） 17](#_Toc453067753)

[4.2 搜索框实时查询 17](#_Toc453067754)

[4.3 用户登录/注册信息实时验证 19](#_Toc453067755)

[4.4 在线填写的HTML转换为PDF 20](#_Toc453067756)

[5 课程总结 21](#_Toc453067757)

[5.1 主要错误 21](#_Toc453067758)

[5.1.1 项目所需要的包未添加到Web/Web-INF/lib中 21](#_Toc453067759)

[5.1.2 Struts2和Spring整合错误 21](#_Toc453067760)

[5.1.3 Struts2传入空指针 21](#_Toc453067761)

[5.1.4 struts2默认上传文件大小限制 22](#_Toc453067762)

[5.1.5 Hibernate多表关系 22](#_Toc453067763)

[5.2 小结 22](#_Toc453067764)

# 设计题目

## 实验题目

课程资源管理系统

## 问题描述

如今课程资源的管理都是在云端实现的，把课程资源放置在云端服务器，可以让用户随时访问课程资源，也可以更好地管理资源，方便了教师和用户的使用。课程资源管理平台是对教学内容和课件、学生学习动态和学习能力的掌握平台。使用计算机使每一位学生、教师及教务管理部门进行整合管理，提高学生自主学习能力、发挥学习积极性也提高教学质量。

## 基本要求

### 系统主要功能：

1. 管理员系统管理和课程管理
2. 教师课程资源和教学资源管理
3. 用户访问系统在线资源浏览和下载

# 需求分析

## 功能描述

### 管理员

教师信息管理，教师信息批量导入，课程信息管理，课程信息批量导入，用户注册授权管理

### 教师

上传课程基本资料，教学资料，作品展示，在线填写课后作业

### 普通用户

查看课程，在线查看课程资料和下载，接受教师作业

## 组织结构

本课程资源管理系统内有教师管理、用户管理、课程管理、课程资料管理、课程作业管理、课程作品管理。

* 教师管理：教师登陆登出、教师资料批量导入
* 用户管理：用户登陆登出、用户权限审核
* 课程管理：课程批量导入、课程实时查询、参加课程
* 课程资源管理：上传课程资料、课程资料在线浏览
* 课程作业管理：作业在线填写、作业在线查看、作业提醒
* 课程作品管理：作品上传、作品在线浏览



图2-1 课程资源管理系统组织结构图

## 功能详情

### 管理员

本来系统中的权限最大的用户是管理员用户，管理员用户拥有教师管理，课程管理，系统维护，其他用户的初始密码和忘记密码的管理，管理员用户在系统的运行中起到非常重要的地位。在添加课程的时候，是管理员先添加课程和课程简介，然后再发送给教师，教师再添加更多的课程资料和讲义。在教师管理中，管理员可以使用Excel进行批量导入和批量赋予权限。



图2-2 管理员用况图

**【用况描述】**

用况名：登录退出

简述：管理员输入用户名和密码登录系统和退出

参与者：管理员

前置条件：管理员输入的用户名和密码通过系统验证

例外：登录失败。失败的原因有可能是因为网络连接出错、用户名或密码错误等。此时应重新检查网络连接情况，重新输入用户名、密码。

用况名：教师管理

简述：管理员对系统中的教师进行批量管理

参与者：管理员

前置条件：管理员必须成功登陆系统

例外：上传的文档上传失败。有可能是因为网络连接出现问题或文档信息不合法。此时应重新检查网络连接或重新上传文档。

用况名：课程信息管理

简述：管理员可以对课程进行开课时间和基本信息的介绍管理（包括：教学大纲、教学信息、教学进度表、实验进度表等）

参与者：管理员

前置条件：管理员必须成功登陆系统

例外：课程信息管理失败。有可能是因为网络连接出错。此时应重新检查网络连接。

用况名：系统维护

简述：管理员对系统进行日常网络维护管理和

参与者：管理员

前置条件：管理员必须成功登陆系统

例外：系统管理维护失败。有可能是因为网络连接出错或者系统出错。此时应检查网络连接或检查系统。

用况名：其他用户管理

简述：管理员在管理界面对其他用户（老师和用户）信息进行管理（包括：对用户信息进行增删改查以及密码重置）

参与者：管理员

前置条件：管理员必须成功登陆系统

例外：其他用户管理失败。有可能是因为网络连接出错或者输入信息不合法。此时应检查网络连接或者重新输入信息。

用况名：个人信息管理

简述：管理员对自己的个人信息和密码进行修改、维护

参与者：管理员

前置条件：管理员必须成功登陆系统

例外：个人信息管理失败。有可能是因为网络连接出现问题或输入信息不合法。此时应重新检查网络连接或重新输入信息。

### 教师

教师用户在本系统中可以对自己的信息进行管理维护，教学资源上传,教学信息发布,在线编辑和发布作业供留言回复的权限。在管理员用户添加好课程之后，教师用户可以上传课程资料，可以设置开课时间和回复用户留言的功能。



图2-3 教师用况图

**【用况描述】**

用况名：登录退出

简述：教师输入用户名和密码登录系统和退出

参与者：教师

前置条件：教师输入的用户名和密码通过系统验证

例外：登录失败。失败的原因有可能是因为网络连接出错、用户名或密码错误等。此时应重新检查网络连接情况，重新输入用户名、密码。

用况名：个人信息管理

简述：教师对自己的个人信息和密码进行修改、维护

参与者：教师

前置条件：教师必须成功登陆系统

例外：个人信息管理失败。有可能是因为网络连接出现问题或输入信息不合法。此时应重新检查网络连接或重新输入信息。

用况名：发布课程

简述：教师通过系统对某一课程发布基本信息（包括：教学大纲、教学信息、教学进度表、实验进度表等）

参与者：教师

前置条件：教师必须成功登陆系统

例外：发布课程失败。有可能是因为网络连接出错或者权限不够。此时应重新检查网络连接或者向管理员获取权限。

用况名：上传课程资料

简述：教师通过系统对某一课程上传课程资源（包括：课件PPT、实验指导书、课程讲义、模拟试卷等）

参与者：教师

前置条件：教师必须成功登陆系统

例外：上传教学资源失败。有可能是因为网络连接出错或者权限不够。此时应重新检查网络连接或者向管理员获取权限。

用况名：发布作业

简述：教师发布左右，系统推送作业到加入该课程的用户

参与者：教师

前置条件：教师必须成功登陆系统

例外：发布作业失败。有可能是因为网络连接出错或者作业内容不合法。此时应重新检查网络连接或者检查作业内容是否合法。如：内容是否为空。

用况名：上传作品

简述：教师上传学生制作好的优秀作品

参与者：教师

前置条件：教师必须成功登陆系统

例外：上传作业失败。有可能是因为网络连接出错，上传的格式不符合。此时应重新检查网络连接。

用况名：留言提醒和回复

简述：系统推送该课程的留言给教师，教师回复用户的提问

参与者：教师

前置条件：教师必须成功登陆系统

例外：回复留言失败。有可能是因为网络连接出错。此时应重新检查网络连接。

### 普通用户

普通用户可以在本系统中查看自己要加入的课程，在登录好之后就可以添加课程，查看课程资料，完成教师发布的课程作业，拥有留言提问的权限。用户可以管理自己的账户，和消息提醒功能。



图2-4 普通用户用况图

**【用况描述】**

用况名：登录退出

简述：用户输入用户名和密码登录系统和退出

参与者：用户

前置条件：用户输入的用户名和密码通过系统验证

例外：登录失败。失败的原因有可能是因为网络连接出错、用户名或密码错误等。此时应重新检查网络连接情况，重新输入用户名、密码。

用况名：添加课程

简述：用户添加自己需要参加的课程

参与者：用户

前置条件：用户必须成功登陆系统

例外：添加课程失败。有可能是因为网络连接出现问题。此时应重新检查网络连接或重新添加课程。

用况名：个人信息管理

简述：用户对自己的个人信息和密码进行修改、维护

参与者：用户

前置条件：用户必须成功登陆系统

例外：个人信息管理失败。有可能是因为网络连接出现问题或输入信息不合法。此时应重新检查网络连接或重新输入信息。

用况名：下载课程资源

简述：用户通过系统对已经上传的课程资源进行下载

参与者：用户

前置条件：用户必须成功登陆系统

例外：下载课程资源失败。有可能是因为网络连接出错。此时应重新检查网络连接。

用况名：在线浏览资源

简述：用户通过系统在线浏览资源

参与者：用户

前置条件：用户必须成功登陆系统

例外：在线浏览失败。有可能是因为网络连接出错。此时应重新检查网络连接。

用况名：留言和回复

简述：用户可以在该课程下的留言专区向教师或其他用户提出问题

参与者：用户

前置条件：用户确定退出登录系统

例外：留言失败。有可能是因为网络连接出错或者输入的内容不合法。此时应重新检查网络连接。

## 数据库设计

### 数据库E-R图



图2-5 课程资源管理系统ER图

### 数据库描述

本系统采用MySQL数据库，MySQL数据库采用结构化的数据库查询语言，是一种开源的关系型数据库。

### 关系模型

* 用户（用户编号、用户名称、用户密码、用户邮箱、用户头像、审核状态、注册时间）
* 管理员（管理员编号、管理员名称、密码）
* 教师（教师编号、教师名称、密码、教师邮箱、教师状态）
* 课程（课程编号、开课时间、课程名称、课程介绍、授课教师、课程类型、课程图片）
* 课程资源（资源编号、课程编号、开课时间、资源名称、资源类型、资源描述、资源格式、资源序号、资源地址、上传者、上传时间）
* 课程详情（课程编号、开课时间、用户编号、加入时间）
* 作业详情（作业编号、课程编号、开课时间、用户编号、作业名称、作业地址、作业状态、创建时间）
* 消息留言（留言编号、课程编号、开课时间、发送者、接受者、留言内容、源留言编号、留言类型、留言状态、创建时间）
* 课程作品（作品编号、课程编号、作品名称、作品类型、作品格式、作品序号、作品地址、上传时间）

### 数据库表

表2-1 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| User | userId | 用户编号 | int | N | Y |
| userName | 用户名称 | nvarchar(50) | Y | N |
| userPwd | 用户密码 | nvarchar(50) | Y | N |
| userEmail | 用户邮箱 | nvarchar(50) | Y | N |
| userImage | 用户头像 | nvaechar(255) | Y | N |
| userStatus | 审核状态 | int | Y | N |
| createTime | 注册时间 | timestamp | Y | N |
| 表2-2 管理员表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| Admin | adminId | 管理员编号 | int | N | Y |
| adminName | 管理员名称 | nvarchar(50) | Y | N |
| adminPwd | 密码 | nvarchar(50) | Y | N |
| 表2-3 教师表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| Teacher | teaId | 教师编号 | int | N | Y |
| teaName | 教师名称 | nvarchar(50) | Y | N |
| teaPwd | 密码 | nvarchar(50) | Y | N |
| teaEmail | 教师邮箱 | nvarchar(50) | Y | N |
| 表2-4 课程表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| Course | couId | 课程编号 | int | N | Y |
| startTime | 开课时间 | date | N | Y |
| couName | 课程名称 | nvarchar(50) | Y | N |
| couInfo | 课程介绍 | text | Y | N |
| couTea | 授课教师 | int | N | N |
| 表2-5 课程详情表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| CourseInfo | couId | 课程编号 | int | N | Y |
| startTime | 开课时间 | date | N | Y |
| userId | 用户编号 | int | N | Y |
| userName | 用户名称 | nvarchar(50) | Y | N |
| joinTime | 加入时间 | timestamp | Y | N |
|  | | | | | |
| 表2-6 资源表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| Source | srcId | 资源编号 | int | N | Y |
| couId | 课程编号 | int | N | N |
| startTime | 开课时间 | date | N | Y |
| couName | 课程名称 | nvarchar(50) | Y | N |
| srcName | 资源名称 | nvarchar(50) | Y | N |
| srcType | 资源类型 | nvarchar(50) | Y | N |
| srcInfo | 资源描述 | text | Y | N |
| srcFormat | 资源格式 | char(10) | Y | N |
| srcNo | 资源序号 | int | Y | N |
| srcUrl | 资源地址 | nvarchar(255) | Y | N |
| uploader | 上传者 | int | N | N |
| uploadTime | 上传时间 | timestamp | Y | N |
| 表2-7 作业详情表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| WorkInfo | WorkId | 作业编号 | int | N | Y |
| couId | 课程编号 | int | N | Y |
| startTime | 开课时间 | date | N | Y |
| usrId | 用户编号 | int | N | Y |
| workName | 作业名称 | nvarchar(50) | Y | N |
| workUrl | 作业内容 | nvarchar(255) | Y | N |
| workStatus | 作业状态 | int | Y | N |
| createTime | 创建时间 | timestamp | Y | N |
| 表2-8 留言表 | | | | | |
| 表名 | 字段名 | 描述 | 类型 | NULL | PK |
| Message | mesId | 留言编号 | int | N | Y |
| couId | 课程编号 | int | Y | N |
| startTime | 开课时间 | date | N | Y |
| fromer | 发送者 | int | Y | N |
| toer | 接受者 | int | Y | N |
| mesInfo | 留言内容 | text | Y | N |
| smesIf | 源留言编号 | int | Y | N |
| mesType | 留言类型 | int | Y | N |
| mesStatus | 留言状态 | int | Y | N |
| createTime | 创建时间 | timestamp | Y | N |

# 详细设计

## 文件上传

教师点击添加课程资料，然后选择上传文件，系统通过Struts2上传文件然后通过Copy函数复制一份临时文件到服务器的指定路径下，判断上传文件的格式类型，如果是doc、docx、xls、xlsx、ppt、pptx、txt格式文件，调用jacob启动OpenOffice宏服务，把文件转换为pdf格式文件，然后把pdf文件相对路径存入数据库中，删除上传时复制的临时文件；如果是其他格式包括pdf格式直接把文件相对路径保存到数据库中，返回消息告知用户添加成功。



图3-1 文件上传功能流程图

**【代码】**

**public boolean** addSource(Source source, String savePath, File uploadFile) {  
 **long** longtime = System.*currentTimeMillis*();  
 String filename = savePath+source.getCouId()+**"\_"**+source.getStartTime()+**"\_"**+source.getSrcName()+**"\_"**+longtime+**"."**+source.getSrcFormat();  
 String odtfilename = savePath+source.getCouId()+**"\_"**+source.getStartTime()+**"\_"**+source.getSrcName()+**"\_"**+longtime+**".odt"**;  
 String dstPath = ServletActionContext.*getServletContext*().getRealPath(filename);  
 File dstFile = **new** File(dstPath);  
 **fileOperate**.*copy*(uploadFile,dstFile,**null**);  
 String pdfFileName = savePath+source.getCouId()+**"\_"**+source.getStartTime()+**"\_"**+source.getSrcName()+**"\_"**+longtime+**".pdf"**;  
 String pdfFile = ServletActionContext.*getServletContext*().getRealPath(pdfFileName);  
 **switch** (source.getSrcFormat()){  
 **case "doc"**:  
 **if**(**fileOperate**.*office2PDF*(dstPath,pdfFile) == **false**){  
 **fileOperate**.*delFile*(filename);  
 **return false**;  
 }

**...**

}  
 source.setSrcUrl(pdfFileName);  
 **if**(**sourceDao**.addSource(source)){  
 **return true**;  
 }**else** {  
 **return false**;  
 }  
}



图3-2 文件上传功能实现

## 批量导入

管理员点击批量上传，然后选择上传文件，系统通过Struts2上传文件然后通过Copy函数复制一份临时文件到服务器的指定路径下，判断文件类型是否为xls、xlsx格式，如果不是提醒用户重新上传，如果是，通过poi组件打开文件，一行行读取记录，把每行记录存为对象保存到数据库中，保存完毕返回值告诉用户执行完毕。



图3-3 批量导入功能流程图

**【代码】**

**public boolean** loadTeacherInfo(String uploadFileFileName) {String targetDirectory = ServletActionContext.*getServletContext*().getRealPath(uploadFileFileName);  
 File target = **new** File(targetDirectory);Workbook wb = **null**;  
 **try**{  
 FileInputStream fi = **new** FileInputStream(target);  
 **if** (uploadFileFileName.toLowerCase().endsWith(**".xls"**)) {  
 wb = **new** HSSFWorkbook(fi);  
 }**else if**(uploadFileFileName.toLowerCase().endsWith(**".xlsx"**)) {  
 wb = **new** XSSFWorkbook(fi);  
 }  
 Sheet sheet = wb.getSheetAt(0);  
 **int** rowNum = sheet.getLastRowNum()+1;  
 List teacherList = **new** ArrayList<Teacher>();  
 *//i 从1开始表示第一行为标题 不包含在数据中* **for**(**int** i=1;i<rowNum;i++){  
 Teacher teacher = **new** Teacher();  
 Row row = sheet.getRow(i);  
 **int** cellNum = row.getLastCellNum();  
 **for**(**int** j=0;j<cellNum;j++){  
 Cell cell = row.getCell(j);  
 String cellValue = **null**;  
 **switch**(cell.getCellType()){ *//判断excel单元格内容的格式，并对其进行转换，以便插入数据库* **case** 0 : cellValue = String.*valueOf*((**int**)cell.getNumericCellValue()); **break**;  
 **case** 1 : cellValue = cell.getStringCellValue(); **break**;  
 **case** 2 : cellValue = String.*valueOf*(cell.getDateCellValue()); **break**;  
 **case** 3 : cellValue = **""**; **break**;  
 **case** 4 : cellValue = String.*valueOf*(cell.getBooleanCellValue()); **break**;  
 **case** 5 : cellValue = String.*valueOf*(cell.getErrorCellValue()); **break**;  
 }  
 **switch**(j){**case** 0 : teacher.setTeaName(cellValue);**break**;  
 **case** 1 : teacher.setTeaEmail(cellValue);**break**;  
 **case** 2 : teacher.setTeaPwd(cellValue);**break**;  
 }  
 }  
 teacher.setTeaStatus(1);  
 teacherList.add(teacher);  
 }  
 teacherIntoDB(teacherList);  
 **fileOperate**.*delFile*(uploadFileFileName);  
 **return true**;  
 }**catch**(IOException e) {  
 e.printStackTrace();  
 **return false**;  
 }  
 }



图3-4 批量导入功能实现

# 关键处和创新点

## SSH（Spring+Struts2+Hibernate开源框架）

本次课程设计采用了开源框架进行编程，是自己在课后自主学习然后编写，使用开源框架更好的体现了MVC三层架构。

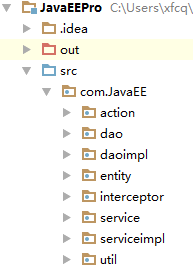
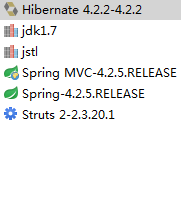
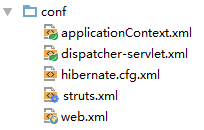
  

图4-1 SSH框架项目结构

## 搜索框实时查询

用户在输入框输入内容时会触发JavaScript的autocomplete事件，然后通过Ajax异步传输技术把输入的内容传到action中，action模糊查询数据库中的数据，使用json数据类型返回数据，JavaScript解析json数据，把内容放入下拉框中。



图4-2搜索框实时查询功能流程图

**【代码】**

**$**( **"#tags"** ).autocomplete({  
 source: **function**( request, response ) {  
 **$**.ajax({  
 **type**: **"post"**,  
 **url**: **"indexSearch.action"**,  
 **data**: {**searchInfo**: **$**(**"#tags"**).val()},  
 **dataType**: **"json"**,  
 success: **function**( data ) {response( **$**.map( data, **function**( item ) {  
 **return** {  
 *// data: item,* **label**: item.couName,  
 **value**: item.couName  
 };  
 }));  
 }  
 });  
 },  
 **minLength**: 1});



图4-3 搜索框实时查询功能实现

## 用户登录/注册信息实时验证

在用户注册的时候，当文本框出现失焦的时候，系统就触发ajax查询数据库中的信息，然后把信息结果返回给用户。登录和注册使用的技术是一样的，这里就不一一列举了。



图4-4 用户注册ajax验证

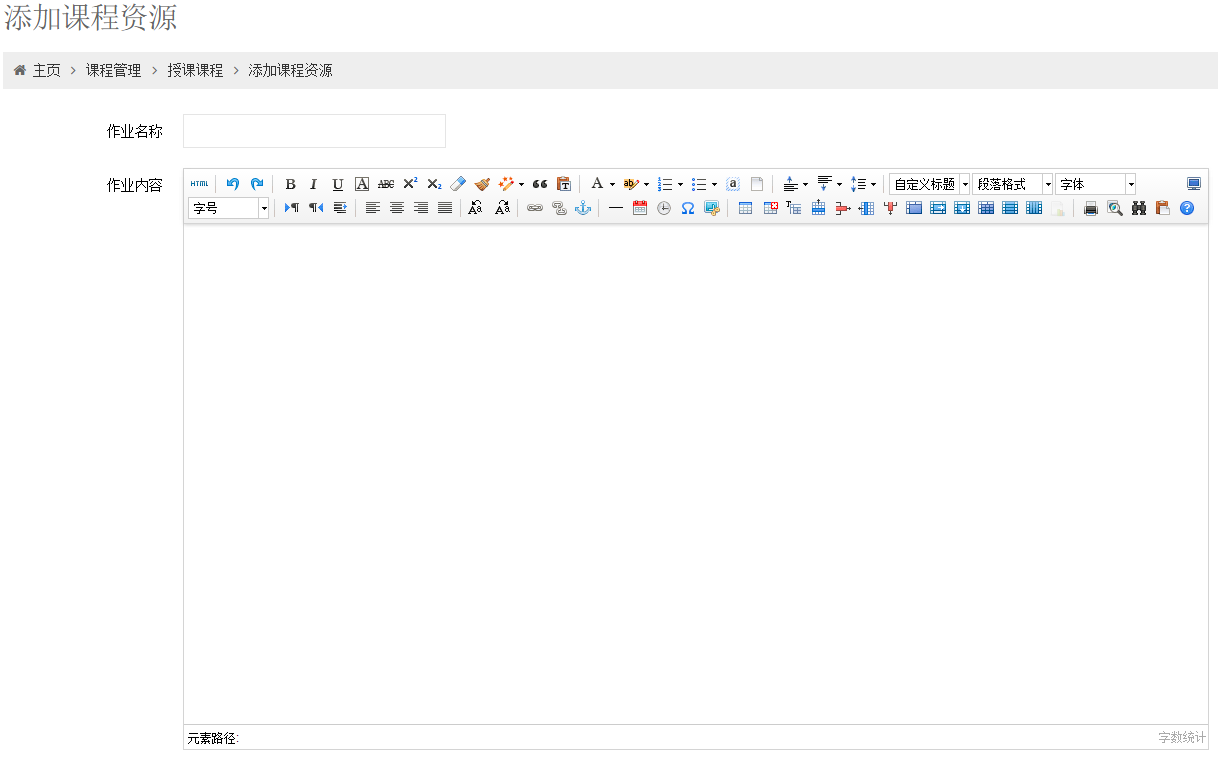
**【代码】**

**$**(**"#registerEmail"**).**blur**(**function** () {  
 **$**.ajax({  
 **type**: **"post"**,  
 **url**: **"emailAvailable.action"**,  
 **data**: {**userEmail**: **$**(**"#registerEmail"**).val()},  
 **dataType**: **"text"**,

success: **function** (data) {  
 **json** = eval(**"("** + data + **")"**);  
 **if** (**json**.**result** == **"true"**) {  
 **$**(**"#div\_show"**).text(**"该邮箱已注册"**);  
 }  
 }  
 });  
});

## 在线填写的HTML转换为PDF

教师在线填写的作业，当教师选择提交的时候，系统使用iText组件将HTML的内容转换为PDF文件，在用户查看作业的时候，查看的是PDF文件在线浏览，在线填写使用百度的UEditor组件。



**图4-5 HTML2PDF功能**

**【代码】**

**public void** htmlCode2pdf(String htmlCode, String pdfFile) **throws** Exception {  
 InputStream htmlFileStream = **new** ByteArrayInputStream(htmlCode.getBytes(**"UTF-8"**));  
 BaseFont bfCN = BaseFont.*createFont*(**"STSongStd-Light"**, **"UniGB-UCS2-H"**, **false**);  
 *// 中文字体定义* Font chFont = **new** Font(bfCN, 12, Font.***NORMAL***, BaseColor.***BLUE***);  
 Font secFont = **new** Font(bfCN, 12, Font.***NORMAL***, **new** BaseColor(0, 204, 255));  
 Document document = **new** Document();  
 PdfWriter pdfwriter = PdfWriter.*getInstance*(document, **new** FileOutputStream(pdfFile));  
 pdfwriter.setViewerPreferences(PdfWriter.***HideToolbar***);  
 document.open();InputStreamReader isr = **new** InputStreamReader(htmlFileStream, **"UTF-8"**);XMLWorkerHelper.*getInstance*().parseXHtml(pdfwriter, document, isr);  
document.close();  
 }

# 课程总结

## 主要错误

### 项目所需要的包未添加到Web/Web-INF/lib中

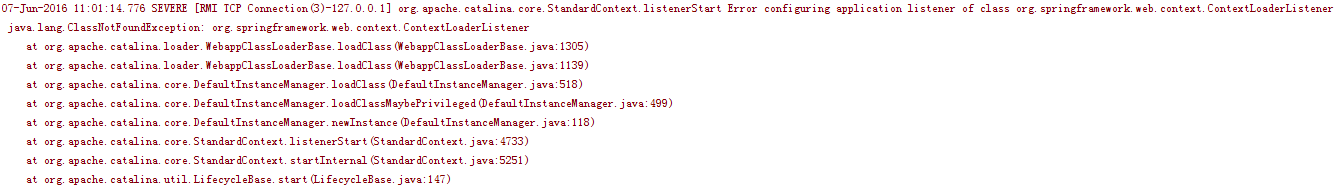


图5-1 未添加包错误

**【解决方案】**

把项目需要的包导入好后要添加到lib文件夹中，因为那是Tomcat启动项目的项目文件lib文件夹。

### Struts2和Spring整合错误

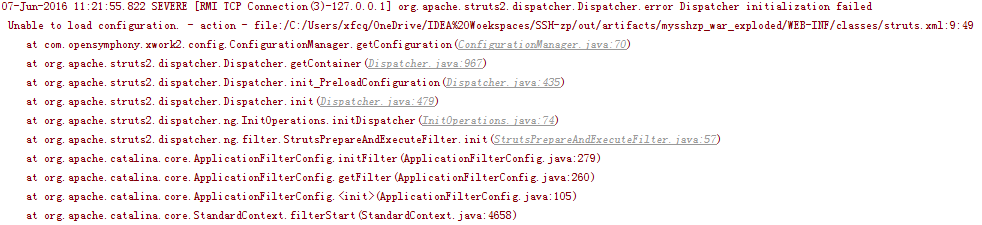


图5-2 Struts2和Spring整合错误

**【解决方案】**

添加struts2-spring-plugin.jar的连接包



### Struts2传入空指针

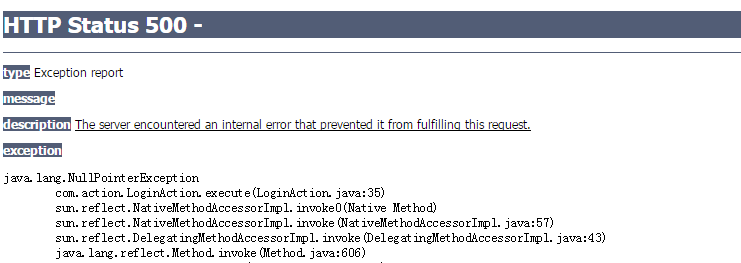


图5-3 struts2传入空指针错误

**【解决方案】**

查看struts2的对应action是否正确，action中的传入的参数是否有set、get方法。

### struts2默认上传文件大小限制

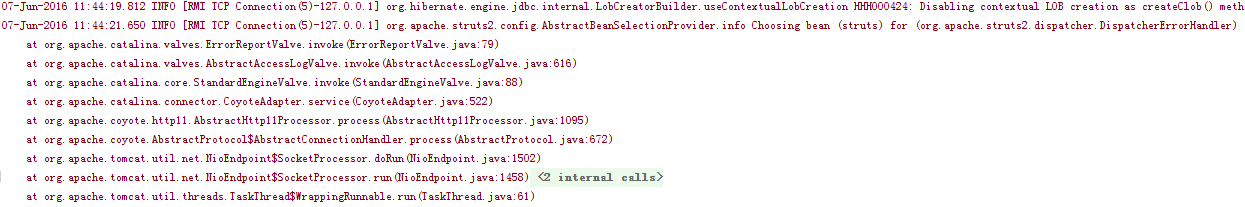


图5-4 struts2默认上传文件大小错误

**【解决方案】**

设置struts2上传文件大小，这里设置最大为500MB。

<constant name="struts.multipart.maxSize" value="524288000" />

### Hibernate多表关系

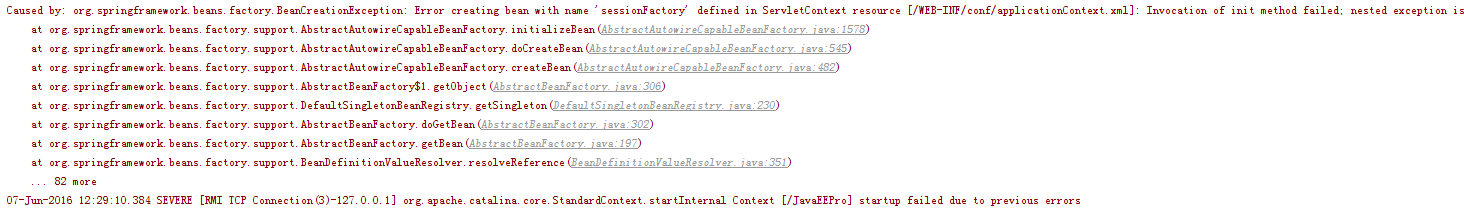


图5-5 Hibernate多表错误

**【解决方案】**

多表关系中不能在两张表都拥有对同一字段的写入

@ManyToOne  
@JoinColumn(name = **"userId"**, referencedColumnName = **"userId"**, nullable = **false**, insertable=**false**, updatable=**false**)

## 小结

在本次课程设计中，我采用了SSH的开源框架编程，在第一次接触开源框架的时候，B/S的框架只是学习了一点，所以自己就在课下主动学习了SSH。

使用SSH的感觉是MVC的三层架构非常清晰，由Struts2的Action完成业务层，由Hibernate完成持久化存储，由Spring完成框架的整合和注入，使得系统在并发量的控制上合理分配。

我本次使用的数据库是自己的云端服务器上的MySQL数据库，但是在项目启动连接一段时间后，Hibernate会丢失连接，为了解决这个问题我是用了<property name="hibernate.c3p0.testConnectionOnCheckout">true</property>这个设置让系统自动重联，虽然解决了不要重启项目的问题，但是重联的等待时间不是很友好，这个问题一直困扰着我。