目 录

[1 绪 论 1](#_Toc40027468)

[1.1研究的背景及意义 1](#_Toc40027469)

[1.1.1 选题的背景 1](#_Toc40027470)

[1.1.2 国内外研究现状 1](#_Toc40027471)

[1.1.3 研究的意义 2](#_Toc40027472)

[1.2 系统目标 2](#_Toc40027473)

[2 需求分析 6](#_Toc40027474)

[2.1业务需求 6](#_Toc40027475)

[2.1.1主要业务流程 6](#_Toc40027476)

[2.2 功能需求 14](#_Toc40027477)

[2.2.1角色分析 14](#_Toc40027478)

[2.2.2业务功能 15](#_Toc40027479)

[2.3 非功能需求 32](#_Toc40027480)

[2.3.1环境需求 32](#_Toc40027481)

[2.3.2性能需求 32](#_Toc40027482)

[2.3.3安全需求 32](#_Toc40027483)

[3 总体设计 34](#_Toc40027484)

[3.1 系统设计的原则 34](#_Toc40027485)

[3.2 系统体系结构设计 34](#_Toc40027486)

[3.3 系统功能结构设计 35](#_Toc40027487)

[4 数据库设计 37](#_Toc40027488)

[4.1 概念结构设计 37](#_Toc40027489)

[4.1.1 设计思路 37](#_Toc40027490)

[4.1.2 E-R图 37](#_Toc40027491)

[4.1.3 主要实体属性图 38](#_Toc40027492)

[4.2 逻辑结构设计 41](#_Toc40027493)

[4.2.1 设计思路 41](#_Toc40027494)

[4.2.2 逻辑模型 41](#_Toc40027495)

[4.3 物理结构设计 41](#_Toc40027496)

[4.3.1 存取方式 41](#_Toc40027497)

[4.3.2 存储结构 41](#_Toc40027498)

[5 界面设计 45](#_Toc40027499)

[5.1 界面关系图或工作流图 45](#_Toc40027500)

[5.2 界面设计成果 45](#_Toc40027501)

[5.2.1 主界面 45](#_Toc40027502)

[5.2.2 子界面 46](#_Toc40027503)

[6 详细设计 49](#_Toc40027504)

[6.1 系统主要功能模块介绍 49](#_Toc40027505)

[6.2个人信息管理模块设计 50](#_Toc40027506)

[6.2.1功能结构设计 50](#_Toc40027507)

[6.2.2类图设计 50](#_Toc40027508)

[6.2.3顺序图设计 51](#_Toc40027509)

[6.2.4 核心处理流程设计 52](#_Toc40027510)

[6.3信息类别管理模块设计 53](#_Toc40027511)

[6.3.1功能结构设计 53](#_Toc40027512)

[6.3.2类图设计 53](#_Toc40027513)

[6.3.3顺序图设计 54](#_Toc40027514)

[6.3.4 核心处理流程设计 55](#_Toc40027515)

[6.4在线办事管理模块设计 56](#_Toc40027516)

[6.4.1功能结构设计 56](#_Toc40027517)

[6.4.2类图设计 56](#_Toc40027518)

[6.4.3顺序图设计 57](#_Toc40027519)

[6.4.4 核心处理流程设计 58](#_Toc40027520)

[6.5评论管理模块设计 59](#_Toc40027521)

[6.5.1功能结构设计 59](#_Toc40027522)

[6.5.2类图设计 59](#_Toc40027523)

[6.5.3顺序图设计 60](#_Toc40027524)

[6.5.4 核心处理流程设计 61](#_Toc40027525)

[6.6留言管理模块设计 62](#_Toc40027526)

[6.6.1功能结构设计 62](#_Toc40027527)

[6.6.2类图设计 62](#_Toc40027528)

[6.6.3顺序图设计 63](#_Toc40027529)

[6.6.4核心处理流程设计 64](#_Toc40027530)

[6.7附件下载管理模块设计 65](#_Toc40027531)

[6.7.1功能结构设计 65](#_Toc40027532)

[6.7.2类图设计 65](#_Toc40027533)

[6.7.3顺序图设计 66](#_Toc40027534)

[6.7.4核心处理流程设计 67](#_Toc40027535)

[6.8网站资讯信息管理模块设计 68](#_Toc40027536)

[6.8.1功能结构设计 68](#_Toc40027537)

[6.8.2类图设计 68](#_Toc40027538)

[6.8.3顺序图设计 69](#_Toc40027539)

[6.8.4核心处理流程设计 70](#_Toc40027540)

[6.9图片素材管理模块设计 71](#_Toc40027541)

[6.9.1功能结构设计 71](#_Toc40027542)

[6.9.2类图设计 71](#_Toc40027543)

[6.9.3顺序图设计 72](#_Toc40027544)

[6.9.4核心处理流程设计 73](#_Toc40027545)

[7 编码 74](#_Toc40027546)

[7.1 代码实现与核心算法 74](#_Toc40027547)

[7.2 代码优化分析 76](#_Toc40027548)

[8 测试 77](#_Toc40027549)

[8.1 测试方案设计 77](#_Toc40027550)

[8.1.1 测试策略 77](#_Toc40027551)

[8.1.2 测试进度安排 77](#_Toc40027552)

[8.1.3 测试资源 77](#_Toc40027553)

[8.1.4 关键测试点 78](#_Toc40027554)

[8.2 测试用例构建 79](#_Toc40027555)

[8.2.1 测试用例编写约定 79](#_Toc40027556)

[8.2.2 测试用例设计 79](#_Toc40027557)

[8.2.3 关键测试用例 81](#_Toc40027558)

[8.2.4 测试用例维护 83](#_Toc40027559)

[9 总结与展望 84](#_Toc40027560)

[9.1 设计工作总结 84](#_Toc40027561)

[9.2 未来工作展望 84](#_Toc40027562)

[谢 辞 85](#_Toc40027563)

[参考文献 86](#_Toc40027564)

[附录A 外文翻译—原文部分 88](#_Toc40027565)

[附录B 外文翻译—译文部分 94](#_Toc40027566)

[附录C 软件使用说明书 99](#_Toc40027567)

[附录D 主要源代码 102](#_Toc40027568)

1 绪 论

## 1.1研究的背景及意义

政府门户网站是各级政府机构中最具活力的政府形象和最独具一格的旗帜。 同时，政府信息化是整个社会信息化的前提和基础，政府信息化必须发挥好领头羊的作用，进而促进整个社会的信息化。

现阶段出现了一些关系群众切身利益的问题，比如：相关的帮扶优惠政策；相关的办事程序等，部分群众不知道以至于没有得到相应的扶持或者是需要帮助时不知道该找谁。为了帮助人民群众更加方便的获取信息、提高办事效率。一个信息门户网站就显得很有必要了。

而且门户网站不仅可以有静态展示界面，还可以设置各种评论交流区，让不同用户方便实时地发表和反馈自己的意见。通过不同用户能够更全面更客观了解网站的服务与信息 ，用户之间的评论留言、 意见可供其他用户参考 。在21世纪，随着网络的普遍，各大交流平台竞相逐鹿，我们可以很好地借助这些平台，来与人民群众密切联系。人民群众能够通过网络途径对政府部门的各种行政行为进行监督，

而通过建设省电子政务及政府信息门户网站，不断强化地方政府网站的网上办事功能，竭尽所能地为当地人民群众提供便利的网上办事服务功能；加大政务信息整合和处理力度，严格信息的整合和管理标准，以软件固化流程和制度，对于政务信息资源的整理和利用提供了很大的便利，最终为政府决策和服务民众带来不可忽视的作用。

### 1.1.1 选题的背景

随着网络浪潮的兴起，信息技术的快速发展使得信息化已经成为当今时代的主体，在党的十八大曾经指出，信息化已经被确定了我国未来发展的重要方向，在经济和社会发展 的方方面面通过信息化的建设来有效的推动整个社会的快速发展。

而通过建设省电子政务及政府信息门户网站，不断强化地方政府网站的网上办事功能，尽可能的给广大人民群众提供便利的网上办事服务功能；加大政务信息整合和处理力度，严格信息的整合和管理标准，以软件固化流程和制度，对于政务信息资源的整理和利用提供了很大的便利，最终为政府决策和服务民众起到了不可忽视的作用。电子政务的建设可以有效的促进电子化政府的建设进程，而地方政府网站的建设，可以成为电子政务建设和电子政府建设有机结合的良好模型，信息化时代网络技术和数字技术的突飞猛进，人民生活水平日益提高，网民数量的逐年增长，以及网络舆论的重要影响作用，都促使我国电子政务建设进程的加速和必要。

### 1.1.2 国内外研究现状

发达国家将现代信息技术应用于各种政府业务，以实现政府业务流程的自动化，已有近50年的历史。 在过去的50年中，计算机技术在电子政务中的应用在技术上经历了三个阶段：大型机，微型计算机和局域网，互联网；在内容处理方面，逐渐从数据管理转化为知识管理；在覆盖领域方面，政府职能由内部管理逐渐转向外部服务。

而在国内，伴随着互联网技术的飞速发展和信息基础设施的进一步完善，虽然我国的电子政务也加速了建设进程，然而我国在电子政务建设方面还存在一些不可忽视的问题。如：电子政务建设的管理体制尚未理顺，形成了各种“信息孤岛”；业务信息系统建设中的“换汤不换药”问题；部分政府工作人员的观念和机能相对落后；电子信息相关法律、法规不健全；电子政务投资建设主体单一；政务公开不够充分、信息不够丰富、信息采集困难、应用水平偏低等情况，因此加快和推进政府信息门户网站建设的任务迫在眉睫。

### 1.1.3 研究的意义

建设省电子政务及政府信息门户网站不但可以为人民带来便捷的办事服务，提高人民群众生活幸福感，而且也为行政服务人员减少了管理方面的不便，进而提高了整个办事的效率。具体通过以下几个方面体现：利用先进的网络技术，实时发布和更新信息资讯，让广大人民能够第一时间了解政府最新的政策和信息；建立完善的政务公开和服务系统，充分体现了党和政府全心全意为人民服务的良好形象；整合各个部门和机构的数据，让他们各司其职，充分提高了工作效率。电子政务的建设可以有效的促进电子化政府的建设进程，而地方政府网站的建设，可以成为电子政务建设和电子政府建设有机结合的良好模型，信息化时代网络技术和数字技术的突飞猛进，人民生活水平日益提高，网民数量的逐年增长，以及网络舆论的重要影响作用，都促使我国电子政务建设进程的加速和必要。

## 1.2 系统目标

政府门户网站不仅是政府对外宣传的窗口，也是省人民，政治活动、经济活动和文化活动信息共享的网络平台。由此可见，一个清晰易懂的政府门户网站至关重要。而开发省电子政务及政府信息门户网站的目的就是为了不断提高政府行政、服务和管理方面的效率，充分利用网络为公众社会提供优质服务，推动和加速整个社会的信息化发展。该网站主要利用UML建模技术、SSM框架、HTML5、CSS3、JQuery、AJAX,黑盒测试等技术实现完善该系统。该系统主要可以分为三类用户：会员，普通管理员，超级管理员。开发该门户网站的目标主要表现在如下方面：

（1）利用先进的网络技术，实时发布和更新信息资讯，让广大人民第一时间能够了解政府最新的政策和信息；

（2）建立完善的政务公开和服务系统，充分体现了党和政府全心全意为人民服务的良好形象；

（3）整合各个部门和机构的数据，让他们各司其职，充分提高了工作效率。（4）整合信息资源，建立政府信息资源库；

省电子政务及政府信息门户网站主要分为网站前台和网站后台，要求如下：

**网站前台**

网站前台是对公众的互动展示平台，网站立足“信息公开、在线办事、互动交流、公共服务”四大功能定位，以“宣传，构架桥梁；政务服务，信息共享；解疑释惑，了解民意”为宗旨，以展示文化、宣传形象、服务人民为核心任务，力求体现“以人为本、服务公众”的理念。网站共开设“网站简介”、“市政信息”、“生育政策”、“就业政策”、“省内资讯”、“政务资讯”、“文件下载”、“工业园区”、“美丽”、“联系我们”、“留言板” 11大栏目，以感知群众情绪，读懂群众心声，服务群众生活。

1. 网站简介

这里主要是对省电子政务及政府信息门户网站的一个基本简介，这里主要包括网站的起源，定位，目标的介绍。

（2）政策动态

门户平台是一个对外公开的平台，提供政务信息的展示功能，公开的信息主要是省广大人民群众密切关心的相关政策，如省人民的“生育政策”、“就业政策”等。我们主要采用导航栏的方式在门户平台上方展现，浏览用户可以通过点击相关导航栏查看信息的详细内容。

（3）资讯动态

省门户网站是广大人民了解资讯信息的主要平台，所以资讯信息发布和更新必不可少。因此我们网站也有“省内资讯”、“政务资讯”两个模块，以便广大人民更好地了解最新资讯信息。

（4）公众互动

在门户平台中可提供除了将电子政务系统的建设情况和使用情况进行展示，还需提供与公众互动的功能。公众互动主要包括两个方面:公众评论和留言建议。网站的用户在进行登录之后，可以对网站发布的文章进行打分和评论。除此之外，如果用户有一些好的建议，用户可以在留言板模块进行专门的留言，后台工作人员可对评论和建议进行查看回复，公众可以查看自己的互动情况。

（5）互动统计

本系统通过使用公众互动功能中的反馈数据，在门户平台做互动基本情况的展示，比如市民留言方面，我们做了留言的类别管理，有市政建言，投诉举报，民意上报等方面，并且用饼状图的方式加以展示，可以更好地帮助管理员和相关人员来了解到自己工作中不足的方面，以后具体应该在哪方面加强。

（6）在线办事

众所周知，近年来，“互联网+政务服务”已成为推进政府治理现代化的重要途径。通过把网络平台和政务服务相结合，不仅可以让用户们做到足不出户就能办理事情，也为服务人员减少了与用户讲解交流的时间，提高了办事效率，节约了办事时间。

因此我们网站有在线办事这个功能，用户可以在前台提交办事申请，并且可以查看自己所有的办事信息。

（7）文件下载

在日常生活中，我们经常要下载大大小小的资源文件，省门户网站也支持这个功能，我们通过在后台上传一些政党信息文件，招标文件等等，然后用户可以在前 台页面进行查看下载。

（8）美丽

众所周知，自古就是一个人杰地灵的地方，在悠久的历史长河中，涌现出了一大批优秀的科学家、政治家和文学奖。如陶渊明、文天祥、詹天佑等。他们为祖国的繁荣发展贡献出了自己的力量，他们是历史的推动者，书写者，是广大人民学习和敬佩的榜样。而且的风景名声也是数不胜数，像滕王阁，庐山等，都是值得一去的好地方。因此我们有“美丽”这个模块来为大家介绍。

（9）工业园区

十九大曾经提出，发展是我们党执政兴国的第一要务，我们要想富起来，就不能停止发展生产。工业园区是地方政府经济发展的重要因素。在此，我们对的工业园区进行了一些简单的介绍。

**网站后台**

网站后台主要是管理员对网站进行信息类别管理，网站信息管理，公众互动管理，用户信息管理。

（1）信息类别管理

作为一个门户网站，需要发布各种不同类别的资讯，如就业，生育，政务，省内等各个方面。所以对资讯进行分类管理尤为重要，这里我们支持对不同类别的标题进行修改，然后按照标题来显示该版块下的文章。

（2）网站信息管理

确定了属于哪个版块之后，我们就可以对该板块下所有的文章信息进行修改和删除，同时在网站信息管理页面，我们也支持按照关键字和类别进行搜索。

（3）公众互动管理

公众互动管理主要包括评论管理和留言管理两个部分，管理员进行登录之后，可以对公众的留言和评论进行查看和回复等操作。

（4）个人信息管理

用户和管理员登录之后可以对自己的个人信息进行修改，当然，超级管理员拥有最高权限，所以也可以对用户的信息进行修改等操作。

# 2 需求分析

## 2.1业务需求

本系统根据三类使用者可划分为用户、普通管理员和超级管理员。对应的功能模块为：

（1）在线办事管理模块：用户登录之后在前台提交办事申请列表，管理员可以在后台查阅和审批。

（2）个人信息管理模块：未注册的游客首先进行注册；已注册的用户可以在修改个人信息页面完成修改。管理员登录后台之后，可以对自己和所有用户的信息进行修改等操作。

（3）信息类别管理模块：作为一个门户网站，需要发布各种不同类别的资讯，如就业，生育，政务，省内等各个方面。所以对资讯进行分类管理尤为重要，在该模块管理员可以发布和编辑不同类别的标题，然后按照标题来显示该版块下的文章。

（4）网站资讯信息管理模块：该模块管理员选择好类别之后，在该类别下可以发表和编辑文章信息。

（5）评论管理模块：该模块用户对文章资讯进行评论之后，管理员在后台可以看到所有的评论信息，并且可以进行删除评论等操作。

（6）留言管理模块：该模块用户在留言板提交留言之后，管理员在后台可以看到所有的留言信息，并且可以进行回复和删除等操作。

（7）图片素材管理模块：该模块管理员可以对前台首页的滚动图片素材进行增加图片，删除图片，修改图片等操作。

（8）附件下载管理模块：该模块管理员可以在后台上传相关文件，用户可以在前台页面进行下载。

### 2.1.1主要业务流程

（1）在线办事管理活动图

用户在线办事的活动图如下图2-1所示，用户在进行登入之后，点击在线办事，选择办事类型，填写好个人基本信息之后点击提交。同时也可以在该页面查看到自己以前办事的历史纪录，管理员则可以在后台查看和审批所有办事申请。

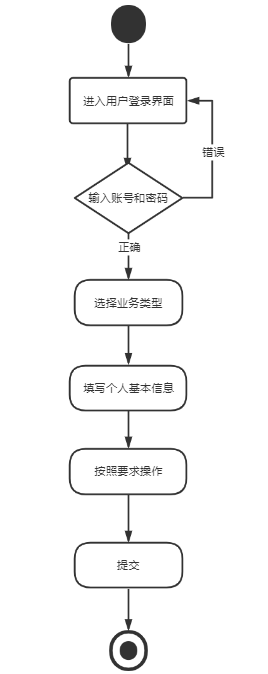


图2-1在线办事活动图

1. 网站资讯信息管理活动图

网站资讯信息管理的活动图如下图 2-2所示，网站包括“网站简介”、“市政信息”、“生育政策”、“就业政策”、“省内资讯”、“政务资讯”、“文件下载”、“工业园区”、“美丽”、“联系我们”、“留言板”11大分类，管理员需要经常对这些分类下的文章进行增加，删除，更新，查看。

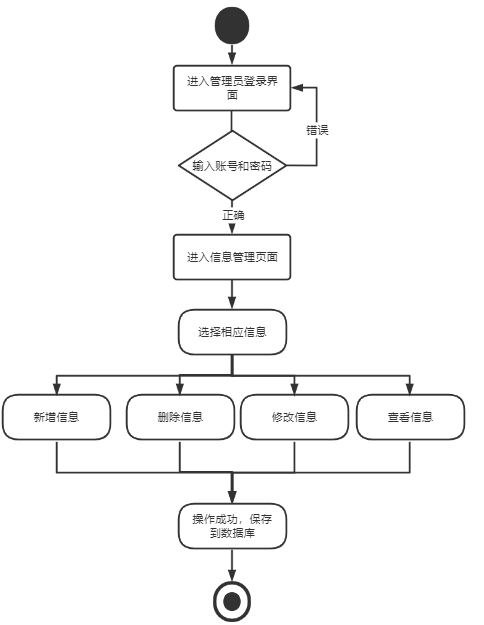


图2-2网站信息管理活动图

（3）用户评论管理活动图

网站评论管理的活动图如下图 2-3所示，用户登录之后，可以对网站内的资讯文章发表自己的评论看法。

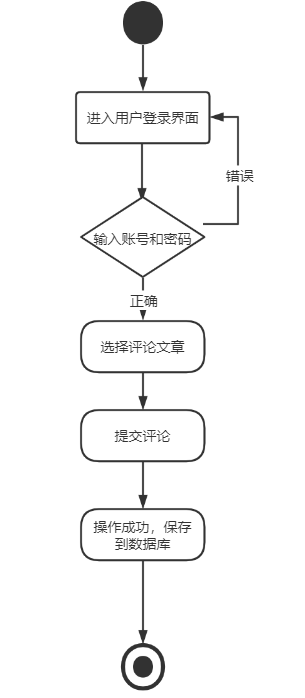


图2-3用户评论管理活动图

（4）用户留言管理活动图

用户留言管理的活动图如下图 2-4所示， 广大人民可能对政府管理者有什么更好的建议或对一些事情提出自己的意见看法。要求用户登录之后，选择留言的类别如投诉举报或者是市政建言等，然后提交自己的留言。

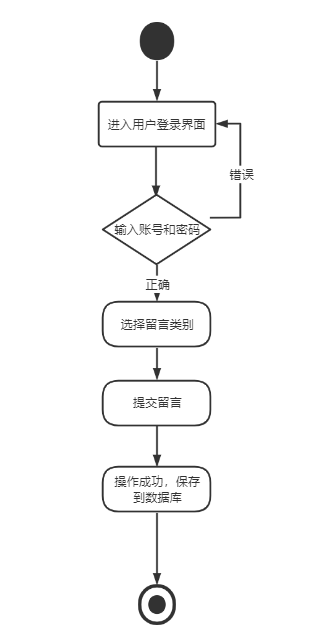


图2-4留言管理活动图

（5）个人信息管理活动图

个人信息管理的活动图如下图 2-5所示，要求用户或管理员登录之后，可以点击“修改个人信息”按钮，在该页面对个人基本信息进行重新编辑。

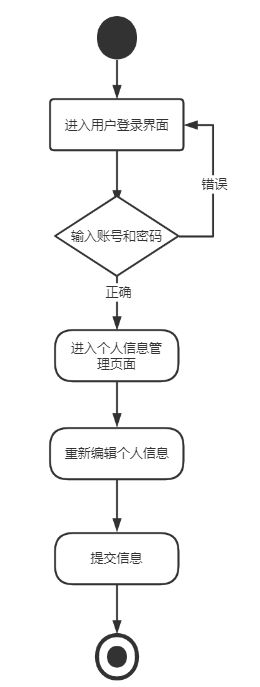


图2-5个人信息管理活动图

（6）信息类别管理活动图

信息类别管理的活动图如下图 2-6所示，网站包括“网站简介”、“市政信息”、“生育政策”、“就业政策”、“省内资讯”、“政务资讯”、“文件下载”、“工业园区”、“美丽”、“联系我们”、“留言板”11大分类，管理员可以对这些分类进行新增，修改，删除等操作。

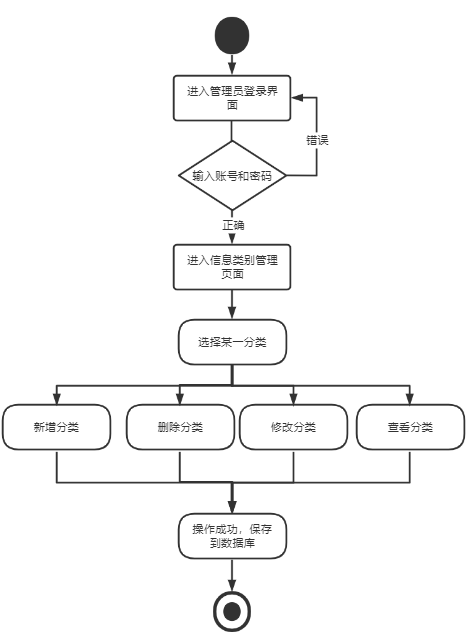


图2-6信息类别管理活动图

1. 图片素材管理活动图

图片素材管理的活动图如下图 2-7所示，管理员在后台点击“滚动图片上传”按钮，在该页面上传好图片之后，用户可以在网站首页看到这些图片组成的轮播图。

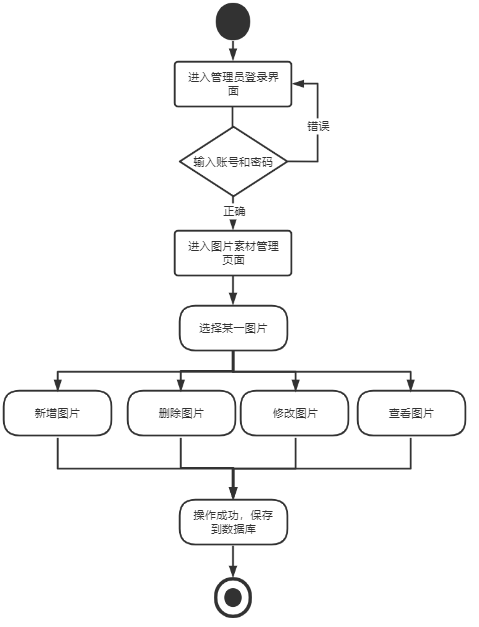


图2-7图片素材管理活动图

1. 附件下载管理活动图

附件下载管理的活动图如下图 2-8所示，管理员在后台上传好相关文件之后，用户可以点进前台详细页面，进行附件下载。

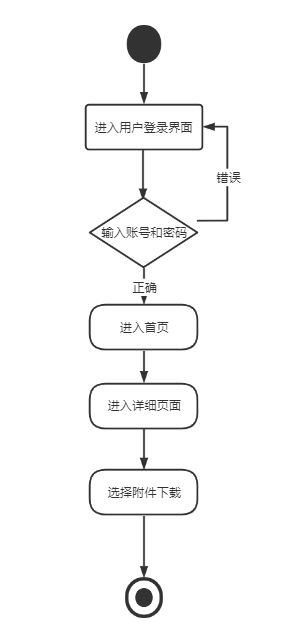


图2-8附件下载管理活动图

## 2.2 功能需求

### 2.2.1角色分析

根据省电子政务及政府信息门户网站的实际需求分析，系统涉及到角色如下表2-1所示：

表2-1 系统角色分析表

|  |  |
| --- | --- |
| 角色 | 职责或功能 |
| 用户 | 浏览网站信息，查看搜索新闻信息进行网上办事，进行公众互动。 |
| 普通管理员 | 拥有部分权限，只能对网站资讯信息，信息分类进行管理。 |
| 超级管理员 | 拥有最高权限，不仅可以对网站信息进行管理，还可以进行办事审批，留言管理，评论管理等。 |

### 2.2.2业务功能

通过对本系统进行的需求分析发现，省电子政务及政府信息门户网站可以拆分为以下八个子模块系统：在线办事管理模块、个人信息管理模块、信息类别管理模块、网站资讯信息管理模块、评论管理模块、留言管理模块、图片素材管理模块、附件下载管理模块。据此设计其功能需求总体用例图如下图2-9所示：

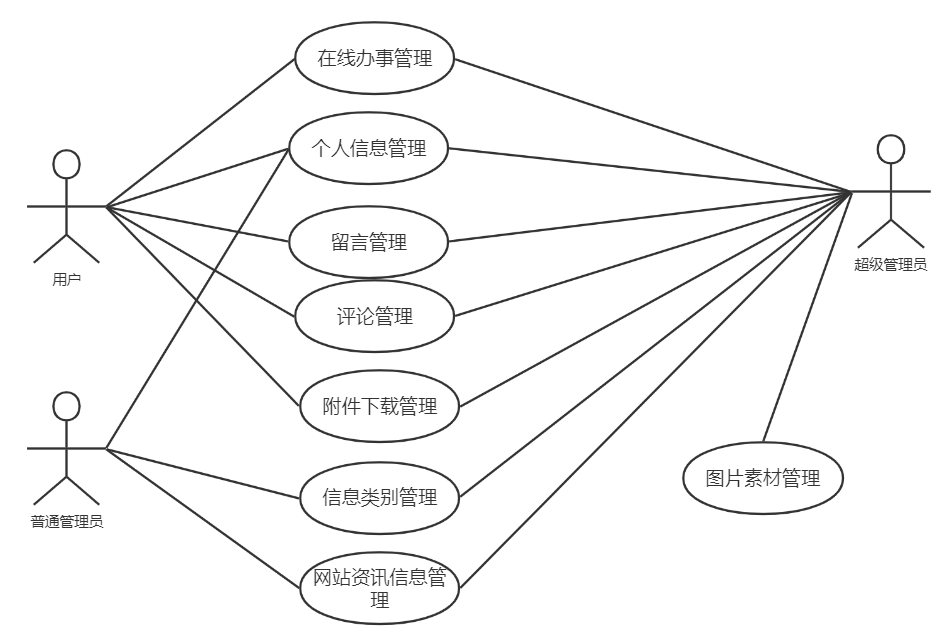


图2-9系统总体用例图

（1）个人信息管理

个人信息管理功能面向前台用户和后台管理员等一切用户，其中个人信息修改包括修改姓名、身份证号码、性别、电话号码、邮箱等信息，个人信息管理用例图如图2-10所示。

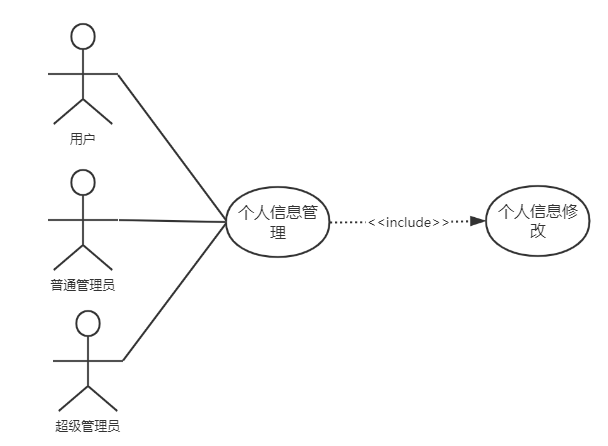


图2-10 个人信息管理用例图

在个人信息修改页面，用户修改邮箱、身份证号码、年龄、性别、地址等信息，个人信息修改用例描述如表2-2所示。

表2-2 个人信息修改用例描述

|  |  |
| --- | --- |
| 描述项 | 个人信息修改说明 |
| 用例名称 | 个人信息修改 |
| 标识符\* | YL01 |
| 用例描述 | 描述了使用本系统所有角色进行个人信息修改的过程 |
| 参与者表 | 所有角色 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 用户已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.用户在个人信息模块，进入个人信息页面；  2.点击修改，修改相应的信息；  3.修改后点击确定，修改成功； |
| 可选操作流 | 1．用户修改相应信息后，不想完成本次修改，点击取消按钮。 |

续表2-2

|  |  |
| --- | --- |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（2）在线办事管理

在线办事管理功能包括新增办事信息，修改办事信息，查询办事信息，删除办事信息，用户在进行登入之后，点击“在线办事”，选择好办事类别、标题、请求内容、相关文件后点击“提交信息”按钮。管理员则可以在后台查看和审批所有办事申请。在线办事管理用例图如图2-11所示。

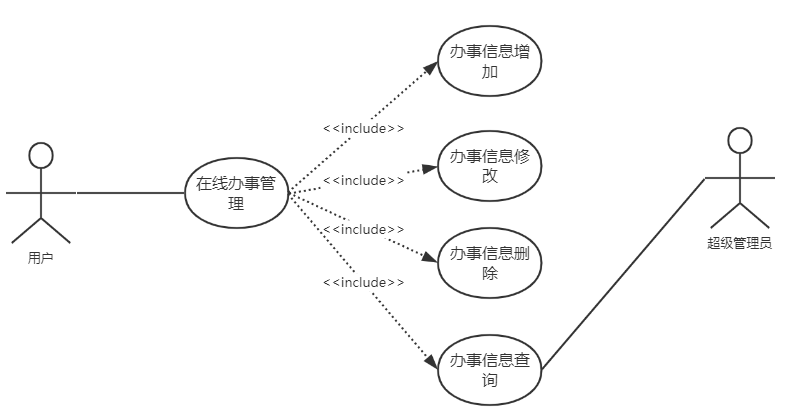


图2-11 在线办事管理用例图

在线办事管理页面，用户可以新增一条办事信息，办事信息增加用例描述如表2-3所示。

表2-3 办事信息增加用例描述

|  |  |
| --- | --- |
| 描述项 | 办事信息增加说明 |
| 用例名称 | 办事信息增加 |

续表2-3

|  |  |
| --- | --- |
| 标识符\* | YL02 |
| 用例描述 | 描述了用户使用本系统进行办事信息增加的过程 |
| 参与者表 | 用户 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 用户已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.用户在在线办事管理模块，进入在线办事信息管理页面；  2.点击新增，新增相应的信息；  3.新增后点击确定，刷新页面，新增成功； |
| 可选操作流 | 1．用户修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（3）信息类别管理

信息类别管理功能包括新增信息类别，修改信息类别，查询信息类别，删除信息类别，管理员在进行登入之后，点击“信息类别管理”，如果想要新增一条类别，点击“添加类别”，输入类别名称，然后提交即可。信息类别管理用例图如图2-12所示。

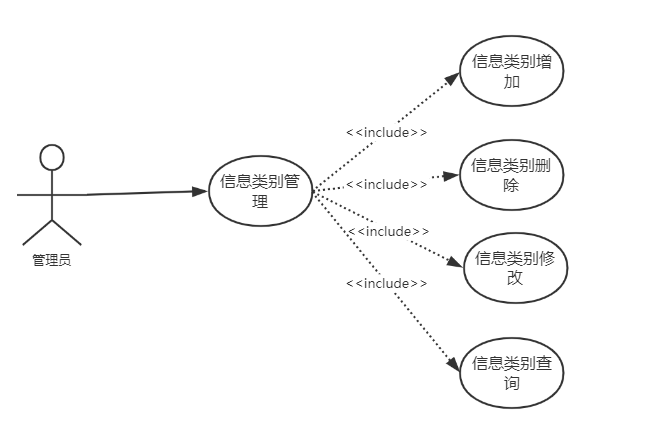


图2-12 信息类别管理用例图

信息类别管理页面，管理员可以删除某一条类别，信息类别删除用例描述如表2-4所示。

表2-4 信息类别删除用例描述

|  |  |
| --- | --- |
| 描述项 | 信息类别删除说明 |
| 用例名称 | 信息类别删除 |
| 标识符\* | YL03 |
| 用例描述 | 描述了管理员使用本系统进行信息类别删除的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在信息列表管理模块，进入信息类别管理页面；  2.点击删除，删除相应的信息；  3.删除后点击确定，删除成功； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |

续表2-4

|  |  |
| --- | --- |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（4）网站资讯信息管理

网站资讯信息管理功能包括新增资讯信息，修改资讯信息，查询资讯信息，删除资讯信息，管理员在进行登入之后，点击“网站在线信息管理”，如果想要新增一篇资讯，要求填写标题、所属类别、图片、附件、内容，然后提交即可。用户可以在前台页面看到最新的资讯信息。网站资讯信息管理用例图如图2-13所示。

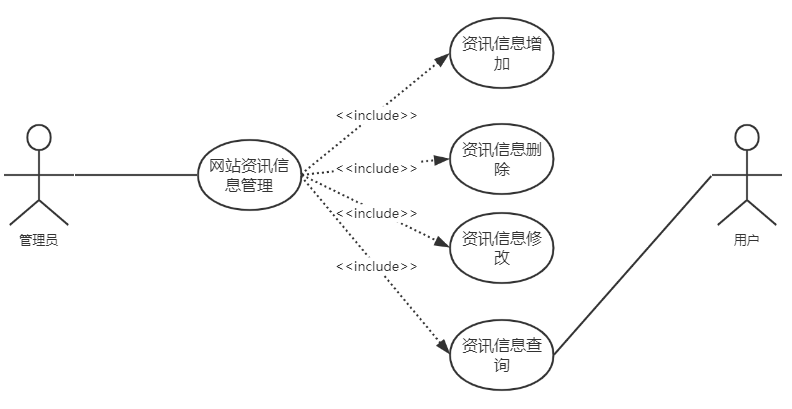


图2-13 网站资讯信息管理用例图

网站资讯信息管理页面，管理员可以修改资讯信息的标题、所属类别，图片、附件、内容等，网站资讯信息修改用例描述如表2-5所示。

表2-5 网站资讯信息修改用例描述

|  |  |
| --- | --- |
| 描述项 | 网站资讯信息修改说明 |
| 用例名称 | 网站资讯信息修改 |

续表2-5

|  |  |
| --- | --- |
| 标识符\* | YL04 |
| 用例描述 | 描述了管理员使用本系统进行网站资讯信息修改的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在网站资讯信息管理模块，进入网站资讯信息管理页面；  2.点击修改，修改某一条资讯的内容；  3.修改后点击确定，修改成功； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

网站资讯信息管理页面，管理员可以按照资讯信息的标题或者类别进行查询，网站资讯信息查询用例描述如表2-6所示。

表2-6网站资讯信息查询用例描述

|  |  |
| --- | --- |
| 描述项 | 网站资讯信息查询说明 |
| 用例名称 | 网站资讯信息查询 |
| 标识符\* | YL05 |
| 用例描述 | 描述了管理员使用本系统进行网站资讯信息查询的过程 |
| 参与者表 | 管理员 |

续表2-6

|  |  |
| --- | --- |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在网站资讯信息管理模块，进入网站资讯信息管理页面；  2.输入查询条件，点击查询，查询相应的信息；  3.查询成功，若查询结果为空，代表查询失败； |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（5）评论管理

评论管理功能包括用户在前台对某一篇资讯进行评论，管理员在后台对用户的评论进行审核管理，这既能加强网民之间的交流，还能够筛选过滤言论不当的评论，营造一个文明网络环境，评论管理用例图如图2-14所示。

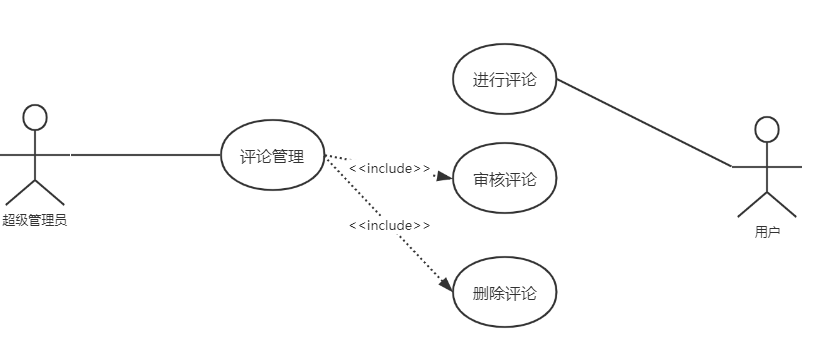


图2-14 评论管理用例图

评论管理页面，管理员为了营造一个科学上网的环境，必须对用户的评论进行审核，一旦发现那些涉及违法的评论必须进行删除，审核评论用例描述如表2-7所示。

表2-7审核评论用例描述

|  |  |
| --- | --- |
| 描述项 | 审核评论说明 |
| 用例名称 | 审核评论 |
| 标识符\* | YL06 |
| 用例描述 | 描述了管理员使用本系统进行审核评论的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 评论可显示或可被删除 |
| 基本操作流 | 1.管理员在评论管理模块，进入评论管理页面；  2.管理员审核评论； |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |

评论管理页面，管理员审核之后，发现那些涉及违法的评论对其进行删除，评论删除用例描述如表2-8所示。

表2-8评论删除用例描述

|  |  |
| --- | --- |
| 描述项 | 评论删除说明 |
| 用例名称 | 评论删除 |
| 标识符\* | YL07 |
| 用例描述 | 描述了管理员使用本系统进行评论删除的过程 |

续表2-8

|  |  |
| --- | --- |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 操作成功并返回当前页面 |
| 基本操作流 | 1.管理员在评论管理模块，进入评论管理页面；  2.选定要删除的评论，点击删除  3.删除后刷新页面，没有代表删除成功 |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

用户登录之后能在前台对新闻资讯信息进行实名评论，进行评论用例描述如表2-9所示。

表2-9进行评论用例描述

|  |  |
| --- | --- |
| 描述项 | 进行评论说明 |
| 用例名称 | 进行评论 |
| 标识符\* | YL08 |
| 用例描述 | 描述了用户使用本系统进行评论的过程 |
| 参与者表 | 用户 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 用户已登录系统 |

续表2-9

|  |  |
| --- | --- |
| 后置条件 | 评论信息具有展示在新闻详情界面下的可能 |
| 基本操作流 | 1.用户在一篇新闻的底部点击评论按钮；  2.输入评论内容，点击提交  3.刷新页面，没有代表评论失败 |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（6）留言管理

留言管理功能包括用户在前台留言板对政府或社会某一方面或现象进行留言，管理员在后台对用户的留言进行审核，回复，删除等。这既能加强政府和省人民之间的交流，还能够让老百姓对的建设出谋划策，留言管理用例图如图2-15所示。

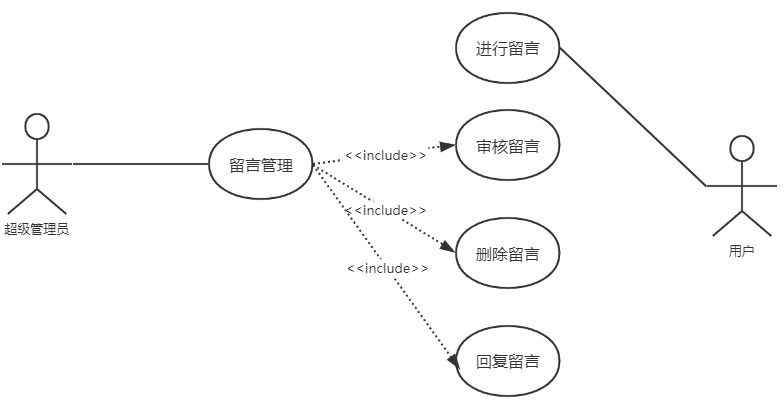


图2-15 留言管理用例图

该模块中，进行留言，审核留言，删除留言与之前评论管理模块类似，在此不再重复描述，下面重点讲解一下回复留言。

留言管理页面，管理员审核之后，对某些留言进行回复，留言回复用例描述如表2-10所示。

表2-10留言回复用例描述

|  |  |
| --- | --- |
| 描述项 | 留言回复说明 |
| 用例名称 | 留言回复 |
| 标识符\* | YL09 |
| 用例描述 | 描述了管理员使用本系统进行留言回复的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 操作成功并返回当前页面 |
| 基本操作流 | 1.管理员在留言管理模块，进入留言管理页面；  2.选定要回复的留言；  3.输入回复的内容； |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（7）图片素材管理

图片素材管理功能包括新增图片素材，修改图片素材，删除图片素材，管理员在进行登入之后，点击“滚动图片管理”，如果想要新增一个图片素材，要求填写标题、图片，然后提交即可。用户可以在前台页面看到最新的图片素材。图片素材管理用例图如图2-16所示

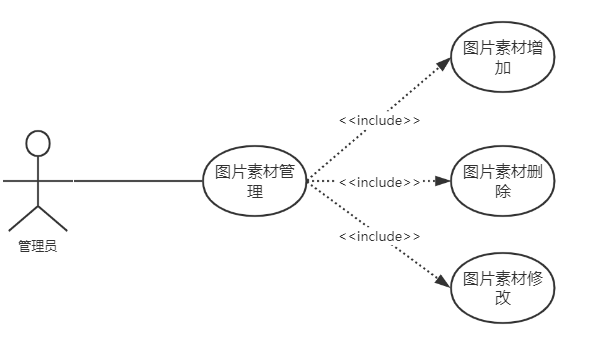


图2-16 图片素材管理用例图

图片素材管理页面，管理员可以修改图片素材的标题、图片，图片素材修改用例描述如表2-11所示。

表2-11 图片素材修改用例描述

|  |  |
| --- | --- |
| 描述项 | 图片素材修改说明 |
| 用例名称 | 图片素材修改 |
| 标识符\* | YL10 |
| 用例描述 | 描述了管理员使用本系统进行图片素材修改的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在图片素材管理模块，进入图片素材管理页面；  2.点击修改，修改某一条图片素材的标题和图片；  3.修改后点击确定，修改成功； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |

续表2-11

|  |  |
| --- | --- |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

图片素材管理页面，管理员可以新增一条或者多条图片素材，图片素材增加用例描述如表2-12所示。

表2-12 图片素材增加用例描述

|  |  |
| --- | --- |
| 描述项 | 图片素材增加说明 |
| 用例名称 | 图片素材增加 |
| 标识符\* | YL11 |
| 用例描述 | 描述了管理员使用本系统进行图片素材增加的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在图片素材管理模块，进入图片素材管理页面；  2.点击添加；  3.添加后点击提交，刷新当前页面，显示代表提交成功； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

图片素材管理页面，管理员可以删除一条或者多条图片素材，图片素材删除用例描述如表2-13所示。

表2-13 图片素材删除用例描述

|  |  |
| --- | --- |
| 描述项 | 图片素材删除说明 |
| 用例名称 | 图片素材删除 |
| 标识符\* | YL12 |
| 用例描述 | 描述了管理员使用本系统进行图片素材删除的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员在图片素材管理模块，进入图片素材管理页面；  2.点击删除，删除相应的信息；  3.删除后点击确定，删除成功； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

（8）附件下载管理

附件下载管理功能包括附件上传，附件下载，管理员在后台进行登入之后，在某篇新闻资讯里面上传相关附件。用户可以在前台的新闻资讯详细页面可以点击下载。附件下载管理用例图如图2-17所示

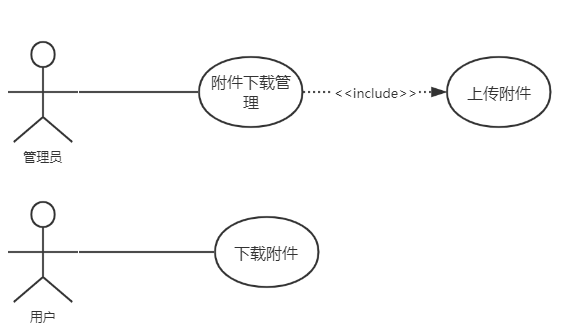


图2-17 附件下载管理用例图

在前台新闻资讯详情页面，用户可以对附件进行下载，所有用户对附件下载用例描述如表2-14所示。

表2-14附件下载用例描述

|  |  |
| --- | --- |
| 描述项 | 附件下载说明 |
| 用例名称 | 附件下载 |
| 标识符\* | YL13 |
| 用例描述 | 描述了使用本系统用户进行附件下载的过程 |
| 参与者表 | 用户 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 用户已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.用户进入新闻资讯详情页面；  2.点击下载，下载相应的附件；  3.下载后弹出提示，下载成功； |
| 可选操作流 | 无 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |

续表2-14

|  |  |
| --- | --- |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

在后台网站在线详细管理页面，管理员可以上传相关附件，管理员对附件上传用例描述如表2-15所示。

表2-15附件上传用例描述

|  |  |
| --- | --- |
| 描述项 | 附件上传说明 |
| 用例名称 | 附件上传 |
| 标识符\* | YL14 |
| 用例描述 | 描述了管理员进行附件上传的过程 |
| 参与者表 | 管理员 |
| 优先级 | 1 |
| 状态\* | 进行中 |
| 前置条件 | 管理员已登录系统 |
| 后置条件 | 系统给出操作成功提示 |
| 基本操作流 | 1.管理员进入网站新闻资讯管理页面；  2.上传相关附件；  3.点击提交，刷新页面； |
| 可选操作流 | 1．管理员修改相应信息后，不想完成本次修改，点击取消按钮。 |
| 被泛化用例表 | 该用例的特化用例列表 |
| 被包含用例表 | 无 |
| 被扩展用例表 | 无 |
| 修改历史记录\* | 暂无 |
| 问题\* | 暂无 |
| 决策\* | 暂无 |
| 频率\* | 暂无 |

## 2.3 非功能需求

### 2.3.1环境需求

服务器运行环境配置要求如表2-27所示。

表2-27 服务器端环境需求表

|  |  |
| --- | --- |
| 要求 | 内容 |
| 操作系统 | Red Hat Enterprise Linux 7.8 |
| 硬盘 | SCSI 硬盘144Gbytes及以上配置 |
| CPU | Intel Xeon 5500及以上配置 |

客户端运行环境需求如表2-28所示。

表2-28 客户端运行环境需求表

|  |  |  |
| --- | --- | --- |
| 要求 | | 内容 |
| CPU | Intel Core i5-4210M处理器及以上配置 | |
| 内存 | 4GB(2G\*2) DDR3L-1333及以上配置 | |
| 硬盘 | 256G SSD及以上配置 | |
| 显卡 | NVIDIA GT940M 2GB 及以上配置 | |

### 2.3.2性能需求

（1）作为政府的门户网站，网站的UI设计应该要稳重大气，突出重点，色彩色调应该要大方得体，不能太过浮夸。而且页面布局要合理，有利于用户快速浏览，能够快速获取自己想要的信息。

（2）网站能够快速响应。

（3）系统要设计友好且易于操作的人机界面。

（4）各个模块之间要做到高内聚，低耦合，方便扩展，以便于后期的维护管理。

（5）网站要考虑兼容性问题，支持不同的浏览器和操作系统。

（6）网页内信息内容准确显示，无错误乱码。

（7）代码方面要简洁明了，具备可移植性，方便后期的维护管理。

### 2.3.3安全需求

（1）容错性

系统在前台或者后台拥有表单验证功能，当用户输入错误或者不合法的数据时，系统会给出信息提示，提示用户修改或者重新输入数据。

（2）系统的外在环境安全

作为省门户网站，必须能够抵制一般的网络攻击，要有较完善的安全防护，要对机密数据采用成熟的加密技术进行加密，防止出现用户个人信息泄露等安全问题。任何时候都不能放松对安全的要求，要坚持把安全问题放在首位，只有在安全的基础上才能去考虑其他问题。

（3）系统内部安全

本系统进行了角色权限分配，除超级管理员之外，其他普通管理员只具有一部分权限，更好的保证了用户信息的安全。

3 总体设计

## 3.1 系统设计的原则

通过前期的需求分析，省电子政务及政府信息门户网站在设计过程中应当遵循以 下原则：

（1）高度可靠。因为该门户网站包含了省工作新闻信息发布，又包含了网络用户对相关监察案件的投诉、举报信件，所以在建设设计中必须要尽可能地保证系统的安全可靠性 ，要在建设平台中预留出足够的容错兼容，做好周全的保护措施，并随时监控系统运行情况。

（2）与时俱进。省门户网站的建设项目不是一时兴起，必须要做好科学长远的规划设 计，在系统的设计思想、结构框架、开发技术等方面从先进、 前瞻的思维思考分析，以发展的眼光看待问题，尤其要多考虑相当长的一段时期内省工作的发展趋势和新闻业务需求的特点。在开发技术的选择上， 尽量采用技术稳定的产 品，以确保网站系统在运行过程中的稳定性、安全性、易操作。

（3）易于维护。要严格按照相关标准规范开展系统设计开发，要满足构建化实现软件。

## 3.2 系统体系结构设计

通过对系统功能的详细分析，本系统采用B/S结构。网站前台主要采用JavaScript和Ajax来实现用户和浏览器进行交互，前后台数据传输，为用户提供更加流畅美观的浏览效果。后台采用的是SSM框架。Spring：是一个容器。类似与一个对象工厂，按要求标记的类都会被提前放入这个容器。SpringMVC：控制器和视图分发器。Mybaits：负责和数据库层打交道，处理与数据库的CRUD相关的操作。系统框架逻辑图如3-1图所示。



图 3-1 系统框架逻辑图

省电子政务及政府信息门户网站应该要前台界面设计美观、操作友好；后台页面简洁 利落、便于操作。应用 MVC 设计模式，结合省门户网站功能需求，系统结构体系图如下图3-2 所示。

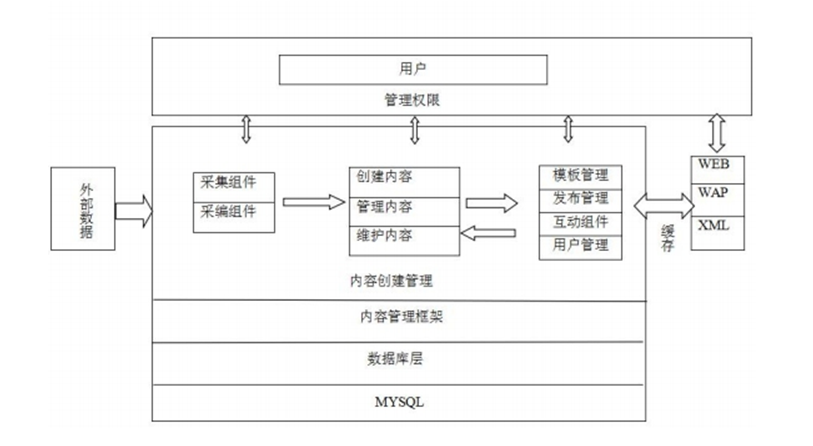


图 3-2 系统结构体系图

## 3.3 系统功能结构设计

本系统的系统总包图如图3-3所示。

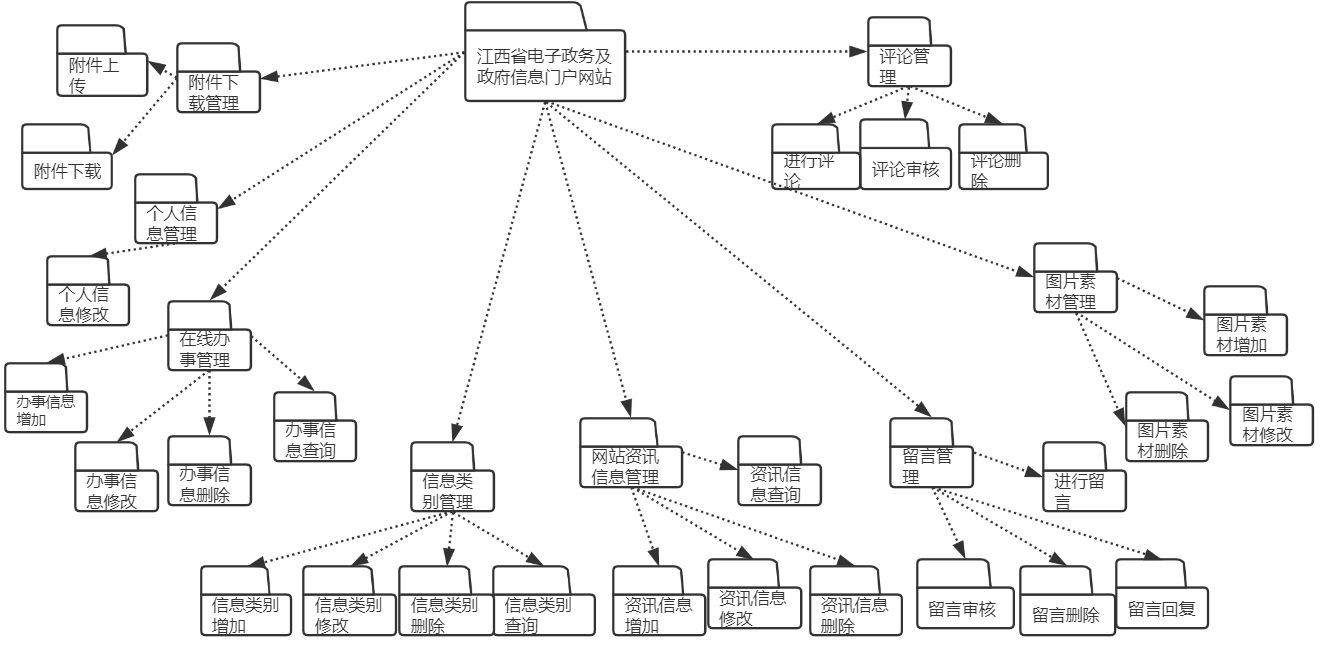


图3-3系统总包图

如图3-3所示，省电子政务及政府信息门户网站一共分为八个模块：个人信息管理模块、在线办事管理模块、信息类别管理模块、网站资讯信息管理模块、评论管理模块、留言管理模块、图片素材模块、附件下载管理模块。其中个人信息管理模块包含个人信息修改，在线办事管理模块分为办事信息增加、办事信息删除、办事信息修改、办事信息查询。信息类别管理模块分为信息类别增加、信息类别删除、信息类别修改、信息类别查询。网站资讯信息管理模块分为资讯信息增加、资讯信息删除、资讯信息修改、资讯信息查询。

评论管理模块分为进行评论、评论审核、评论删除。留言管理模块分为进行留言、留言审核、留言删除、留言回复。图片素材管理模块分为图片素材增加、图片素材删除、图片素材修改。附件下载管理模块分为附件上传、附件下载。

# 4 数据库设计

## 4.1 概念结构设计

### 4.1.1 设计思路

省电子政务及政府信息门户网站的实体包括用户，管理员，办事信息，评论，留言，文章，类别，图片，附件等。首先管理员在后台编辑好资讯类别，在该类别下发布文章或上传相关附件和图片，然后用户在前台可以浏览这些新闻资讯，并且可以进行评论，也可以下载相关附件。其次用户可以在前台提交办事申请，管理员可以在后台进行审批和处理。用户还可以在留言板模块进行留言，发布自己的建议看法，管理员在后台进行审核和回复。

由以上分析可知，在“发布”关系中，管理员实体与文章实体是1:N的关系。在“编辑”关系中，管理员实体与类别实体是1:N的关系。在“上传”关系中，管理员实体与附件实体是1:N的关系。在“上传”关系中，管理员实体与图片实体是1:N的关系。在“发表”关系中，用户实体与评论实体之间是1:N的关系。在“发表”关系中，用户实体与留言实体之间是1:N的关系。在“申请”关系中，用户实体与办事信息实体之间是1:N的关系。在“审核”关系中，管理员实体与留言实体是1:N的关系。在“审核”关系中，管理员实体与评论实体是1:N的关系。在“浏览”关系中，用户实体与新闻实体之间是M:N的关系。在“处理”关系中，管理员实体与办事信息实体之间是1:N的关系。

### 4.1.2 E-R图

省电子政务及政府信息门户网站的总体E-R图如下图4-1所示。

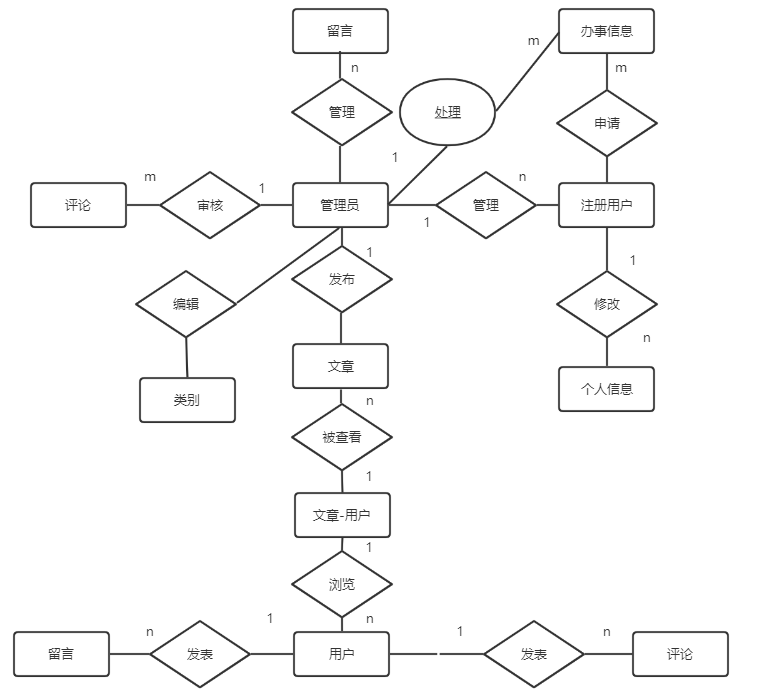


图4-1 系统总体E-R图

### 4.1.3 主要实体属性图

(1) 用户的实体属性主要包括用户名、用户密码、姓名、性别、头像、身份证、电话号码、邮箱、QQ，用户的实体图如图4-2所示。

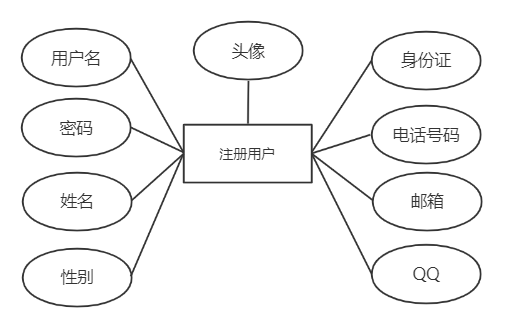


图4-2 用户实体属性图

(2) 文章的实体属性主要包括文章编号、文章标题、文章类别、添加人、添加时间、文章内容，文章的实体图如图4-3所示。

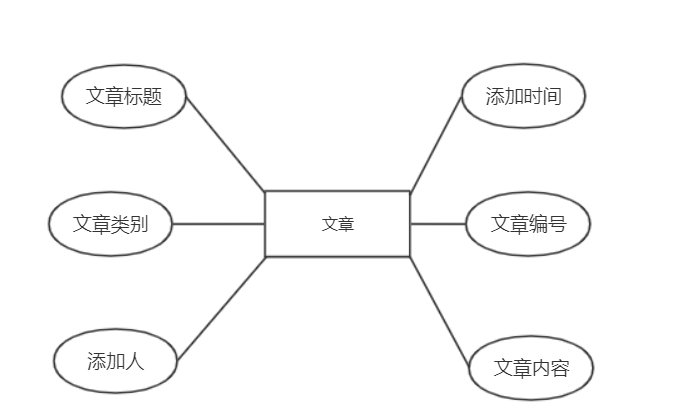


图4-3 文章实体属性图

(3) 办事信息的实体属性主要包括办事编号、办事类别、标题、请求内容、相关文件、受理状态，办事信息的实体图如图4-4所示。

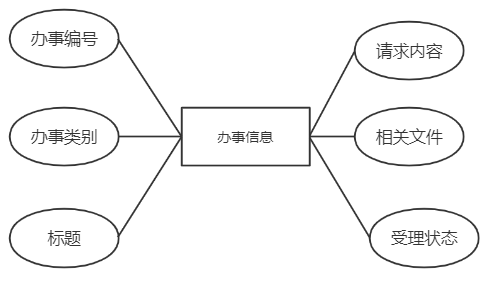


图4-4 办事实体属性图

(4) 评论的实体属性主要包括评论编号、评论文章、操作、评论内容、添加时间、评论人，评论的实体图如图4-5所示。

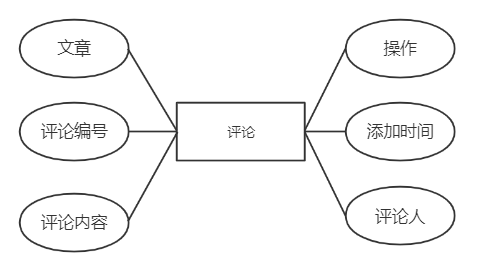


图4-5 评论实体属性图

(5) 留言的实体属性主要包括留言编号、留言名称、留言内容、留言人、管理员、添加时间，留言的实体图如图4-6所示。

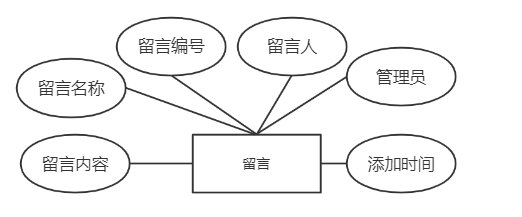


图4-6 留言实体属性图

(6) 图片的实体属性主要包括图片编号、图片描述、图片路径、上传时间，图片的实体图如图4-7所示。

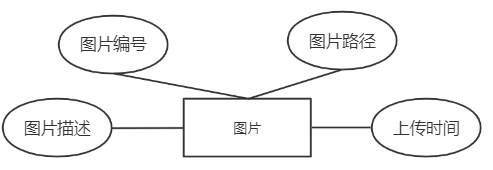


图4-7 图片实体属性图

(7) 附件的实体属性主要包括附件编号、附件标题、上传时间，附件的实体图如图4-8所示。

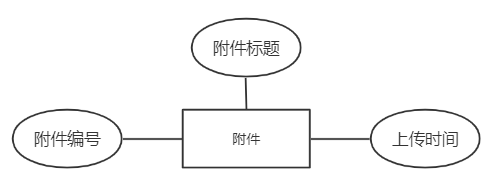


图4-8 附件实体属性图

(8) 类别的实体属性主要包括类别编号、类别名称，类别的实体图如图4-9所示。

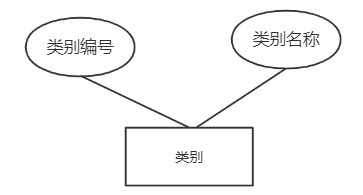


图4-9 类别实体属性图

## 4.2 逻辑结构设计

### 4.2.1 设计思路

设计逻辑结构时一般要分四步进行：

（1）将实体进行转换；

（2）将1:1联系进行转换；

（3）将1:m联系进行转换；

（4）将m:n联系进行转换；

### 4.2.2 逻辑模型

由实体属性E-R图得到的逻辑模型如下所示。

（1）用户基本信息：（用户名，用户密码，姓名，性别，头像，身份证，电话号码，邮箱，QQ）

（2）文章信息：（文章编号，文章标题，文章类别，添加人，添加时间，文章内容，）

（3）办事信息：（办事编号，办事类别，标题，请求内容，相关文件，受理状态）

（4）评论：（评论编号，评论文章，操作，评论内容，添加时间，评论人）

（5）留言：（留言编号，留言名称，留言内容，留言人，管理员，添加时间）

（6）图片：（图片编号，图片描述，图片路径，上传时间）

（7）附件：（附件编号，附件标题，上传时间）

（8）类别：（类别编号，类别名称）

备注:含下划线字段为该表的主键。

## 4.3 物理结构设计

### 4.3.1 存取方式

根据以上分析，本系统采用索引存取的方式对数据进行存取，通过在每个表中设置主键字段和经常需要查找的字段的B+树索引，提高数据库的查找效率。

### 4.3.2 存储结构

根据以上主要关系模型，数据总表如表4-1所示。

表4-1 表汇总

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| 表user | 存储注册用户信息 |
| 表admin | 存储管理员信息 |
| 表message | 存储留言板信息 |
| 表article | 存储文章信息 |
| 表service | 存储办事申请信息 |
| 表infor\_type | 存储信息分类 |
| 表picture | 存储图片信息 |
| 表attachment | 存储附件信息 |

1. 用户表

用户表共有9个字段，具体表结构如表4-2所示。

表4-2 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | User | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| U\_username | 字符串 | 非空 | 用户名 |  | 20 |
| U\_password | 字符串 | 非空 | 用户密码 |  | 20 |
| U\_sex | 字符串 | 非空 | 性别 |  | 4 |
| U\_name | 字符串 | 非空 | 姓名 |  | 20 |
| U\_province | 字符串 | 非空 | 身份证 |  | 50 |
| U\_mailbox | 字符串 | 非空 | 邮箱 |  | 30 |
| U\_QQ | 字符串 | 非空 | QQ |  | 20 |
| U\_hand | 字符串 | 非空 | 头像 |  | 100 |

1. 管理员表

管理员表共有5个字段，具体表结构如表4-3所示。

表4-3 管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Admin | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| A\_name | 字符串 | 非空 | 用户名 |  | 50 |
| A\_pwd | 字符串 | 非空 | 密码 |  | 20 |
| A\_xc | 字符串 | 非空 | 普通管理员 |  | 50 |

续表4-3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A\_addtime | 时间戳 | 非空 | 添加时间 |  |  |

1. 留言表

留言表共有6个字段，具体表结构如表4-4所示。

表4-4 留言表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Message | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| Title | 字符串 | 非空 | 留言名称 |  | 50 |
| Content | 字符串 | 非空 | 留言内容 |  | 20 |
| Name | 字符串 | 非空 | 留言人 |  | 50 |
| Manager | 字符串 | 非空 | 管理员 |  | 50 |
| Addtime | 时间戳 | 非空 | 添加时间 |  |  |

1. 文章表

文章表共有6个字段，具体表结构如表4-5所示。

表4-5 文章表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Article | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| Title | 字符串 | 非空 | 标题 |  | 50 |
| Type | 字符串 | 非空 | 类别 |  | 50 |
| Addperson | 字符串 | 非空 | 添加人 |  | 50 |
| Addtime | 时间戳 | 非空 | 添加时间 |  | 50 |
| Content | 字符串 | 非空 | 内容 |  | 50 |

1. 办事信息表

办事信息表共有10个字段，具体表结构如表4-6所示。

表4-6 办事信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Service | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 唯一标识 | 主键约束 | 16 |
| Number | 字符串 | 非空 | 编号 |  | 50 |
| Title | 字符串 | 非空 | 申请名 |  | 50 |
| Jb | 字符串 | 非空 | 办理项 |  | 50 |
| Province | 字符串 | 非空 | 身份证号 |  | 50 |
| Show | 布尔 | 非空 | 是否展示 |  | 16 |

续表4-6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Addtime | 时间戳 | 非空 | 添加时间 |  |  |
| Mark | 字符串 | 非空 | 备注 |  | 100 |
| Photo | 字符串 | 非空 | 个人照片 |  | 50 |
| Name | 字符串 | 非空 | 姓名 |  | 50 |

1. 文章类别表

文章类别表共有3个字段，具体表结构如表4-7所示。

表4-7 文章类别表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Infor\_type | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| Infor\_type | 字符串 | 非空 | 信息类型 |  | 50 |

1. 图片信息表

图片信息表共有5个字段，具体表结构如表4-8所示。

表4-8 图片信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Picture | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| Content | 字符串 | 非空 | 内容 |  | 50 |
| Address | 字符串 | 非空 | 路径 |  | 50 |
| Addperson | 字符串 | 非空 | 添加人 |  | 50 |
| Addtime | 时间戳 | 非空 | 添加时间 |  | 50 |
| Content | 字符串 | 非空 | 内容 |  | 50 |

1. 附件信息表

附件信息表共有3个字段，具体表结构如表4-9所示。

表4-9 附件信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Attachment | | | | |
| 列名 | 数据类型 | 空/非空 | 字段说明 | 约束条件 | 长度 |
| Id | 整型 | 非空 | 编号 | 主键约束 | 16 |
| Title | 字符串 | 非空 | 标题 |  | 50 |
| Addtime | 时间戳 | 非空 | 添加时间 |  | 50 |

5 界面设计

## 5.1 界面关系图或工作流图

省电子政务及政府信息门户网站从浏览器输入指定URL即可进入网站首页，在首页，用户通过点击切换导航栏，可以看到不同类别，点进去是该分类下的所有文章资讯，点击即可进入到文章详情页。用户登录之后可以在底部对文章进行评论，也可以下载相关附件。管理员通过登录进入网站后台，后台管理页面包括类别管理页、文章信息管理页、在线办事管理页、留言管理页、评论管理页、个人信息管理页。界面关系图如图5-1所示。

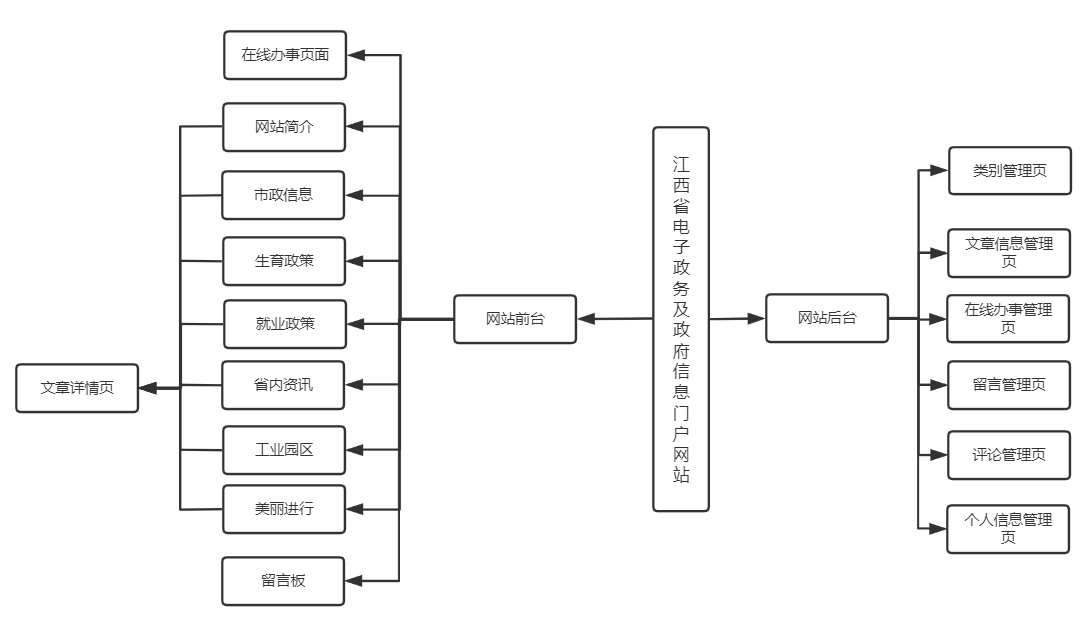


图5-1界面关系图

## 5.2 界面设计成果

### 5.2.1 主界面

（1）用户主界面

本系统运行即进入网站主界面，如图5-2所示：



图5-2网站主界面图

（2）管理员主界面

管理员登录后即可进入管理员主界面，如图5-3所示：



图5-3管理员主界面图

### 5.2.2 子界面

（1）用户登录注册页

系统游客/用户可选择进行系统登录/注册，在注册成功后登陆即可返回登录成功的用户主界面，如下图5-4所示：



图5-4 用户登录注册页面

（2）类别管理页

在该页面管理员登陆之后可以发布和编辑不同类别的标题，然后按照标题来显示该版块下的文章，如下图5-5所示：



图5-5 类别管理页面

1. 文章信息管理页

在该页面管理员登陆之后可以编辑文章信息，如下图5-6所示：

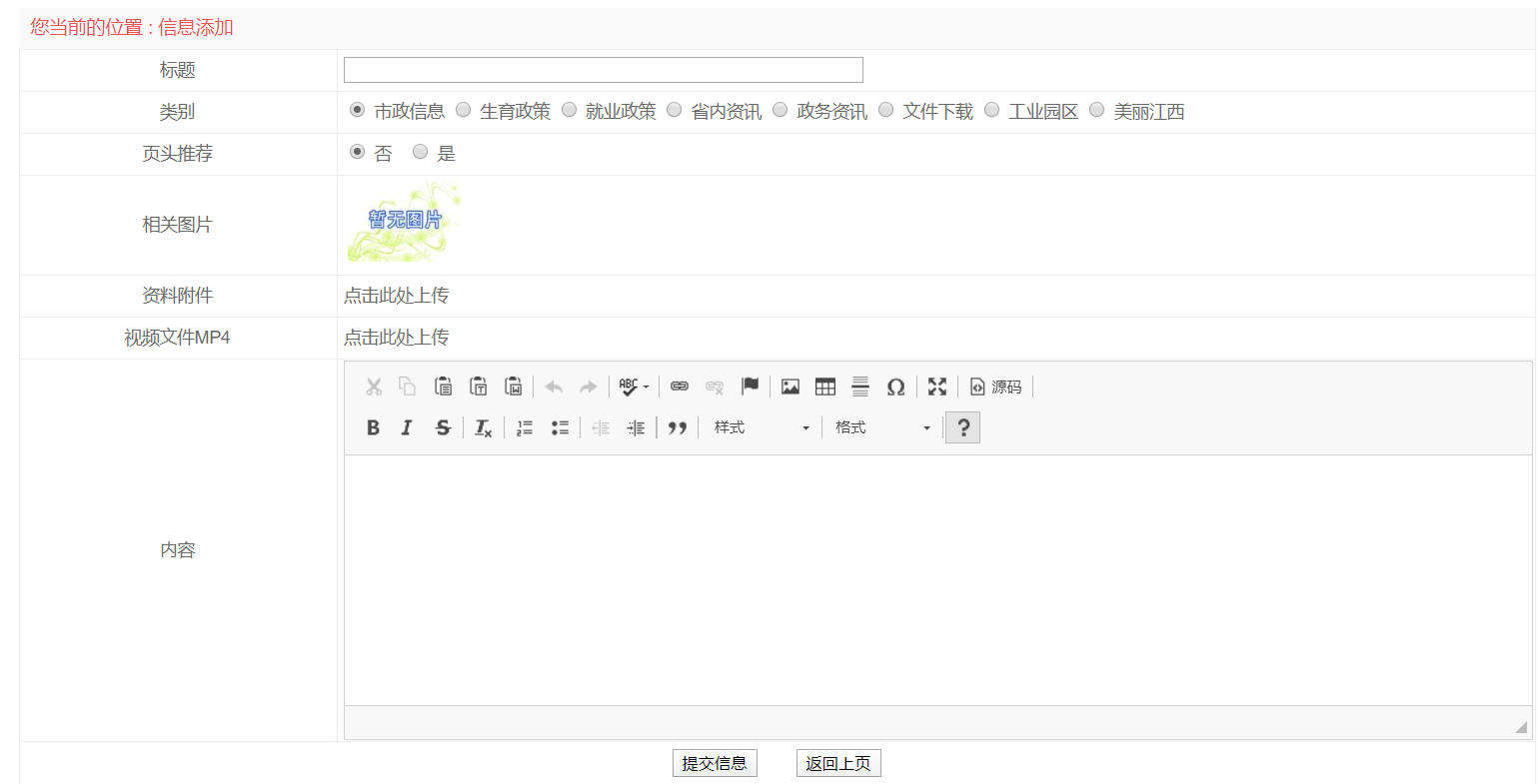


图5-6 文章信息管理页面

（4）在线办事管理页

在该页面管理员登陆之后可以审批用户提交的办事信息，如下图5-7所示：



图5-7 在线办事管理页面

（5）评论管理页

在该页面管理员登陆之后可以审核和删除用户的评论，如下图5-8所示：



图5-8 评论管理页面

1. 留言管理页

在该页面管理员登陆之后可以审核和删除以及回复用户的留言，如下图5-9所示：



图5-9 留言管理页面

6 详细设计

## 6.1 系统主要功能模块介绍

省电子政务及政府信息门户网站一共分为八个模块：个人信息管理模块、在线办事管理模块、信息类别管理模块、网站资讯信息管理模块、评论管理模块、留言管理模块、图片素材模块、附件下载管理模块。其中个人信息管理模块包含个人信息修改，在线办事管理模块分为办事信息增加、办事信息删除、办事信息修改、办事信息查询。信息类别管理模块分为信息类别增加、信息类别删除、信息类别修改、信息类别查询。网站资讯信息管理模块分为资讯信息增加、资讯信息删除、资讯信息修改、资讯信息查询。评论管理模块分为进行评论、评论审核、评论删除。留言管理模块分为进行留言、留言审核、留言删除、留言回复。图片素材管理模块分为图片素材增加、图片素材删除、图片素材修改。附件下载管理模块分为附件上传、附件下载。系统的模块结构图如图6-1所示。

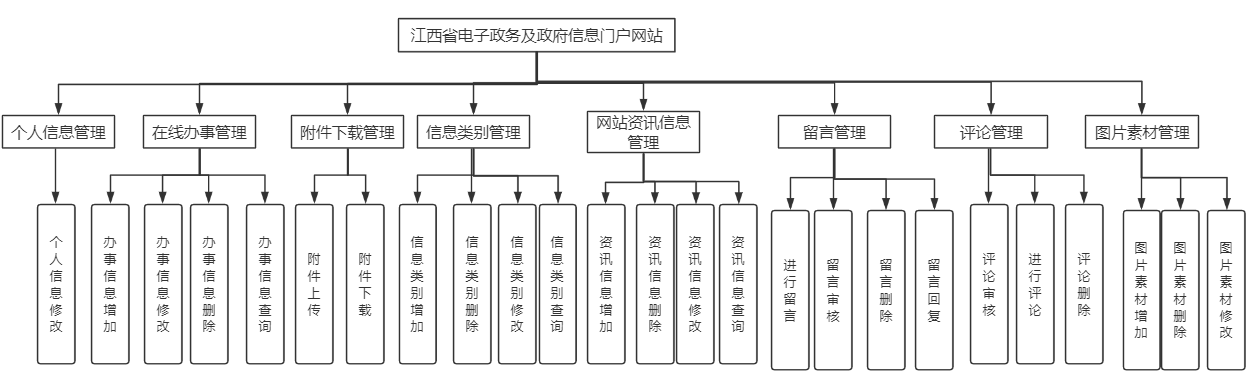


图6-1系统模块结构图

## 6.2个人信息管理模块设计

### 6.2.1功能结构设计

个人信息管理包图如图6-2所示。

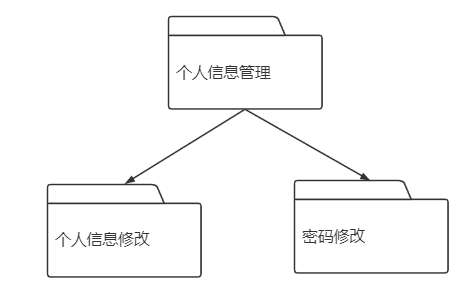


图6-2个人信息管理包图

个人信息管理功能包括个人信息修改以及密码修改，其中个人信息修改为所有用户修改姓名、年龄、性别、电话号码等信息，修改密码为所有用户修改个人登录密码。

### 6.2.2类图设计

个人信息管理类图如图6-3所示。

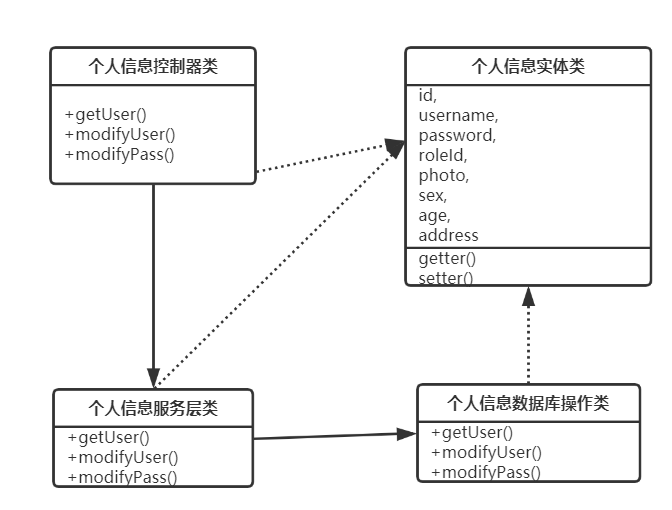


图6-3个人信息管理类图

在个人信息管理类图中，主要涉及到四个类的关联，个人信息控制器类关联个人信息服务层类，同时关联个人信息实体类。个人信息服务层类关联个人信息数据库操作类，同时关联个人信息实体类。个人信息管理涉及到的各个类的描述如下所示，其中个人信息控制器类描述如表6-1所示。

表6-1个人信息控制器类描述

|  |  |  |
| --- | --- | --- |
| 个人信息控制器类 | | |
| 类的描述：个人信息控制器类 | | |
| 操作描述 | GetUser () | 获取个人信息 |
| ModifyUser() | 修改个人信息 |
| ModifyPass () | 修改密码 |

个人信息服务层类描述如表6-2所示。

表6-2个人信息服务层类描述

|  |  |  |
| --- | --- | --- |
| 个人信息服务层类 | | |
| 类的描述：个人信息服务层类 | | |
| 操作描述 | GetUser () | 获取个人信息 |
| ModifyUser() | 修改个人信息 |
| ModifyPass () | 修改密码 |

个人信息数据库操作类描述如表6-3所示。

表6-3个人信息数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 个人信息数据库操作类 | | |
| 类的描述：个人信息数据库操作类 | | |
| 操作描述 | GetUser () | 获取个人信息 |
| ModifyUser() | 修改个人信息 |
| ModifyPass () | 修改密码 |

### 6.2.3顺序图设计

个人信息管理顺序图如图6-4所示。

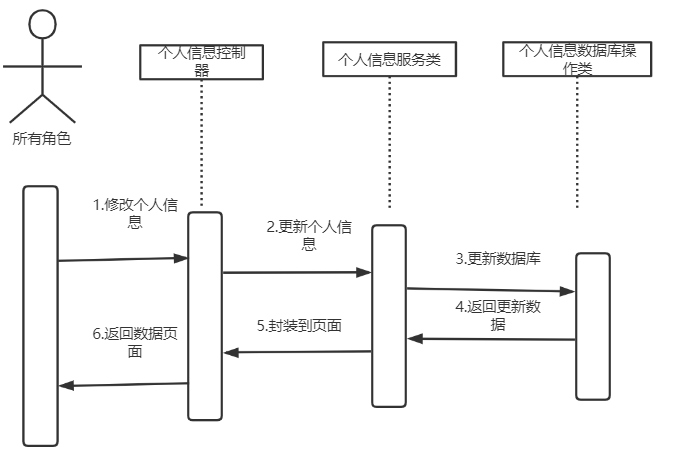


图6-4个人信息管理顺序图

在个人信息顺序图中，所有角色点击修改个人信息，此时前台请求通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回更新的数据，然后进行封装数据再由控制器层返回到前台的展示数据页面。

### 6.2.4 核心处理流程设计

个人信息管理活动图如图6-5所示。

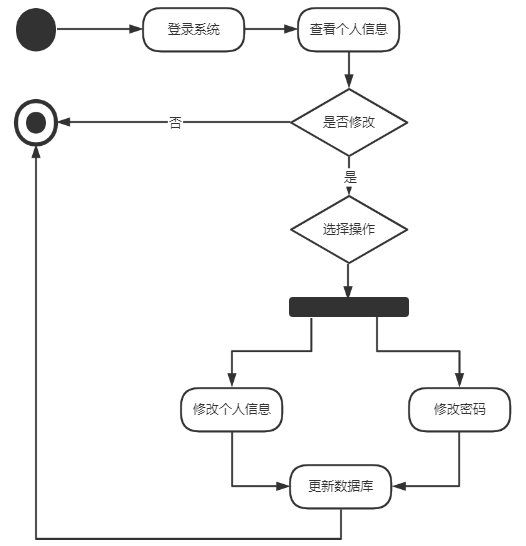


图6-5 个人信息管理活动图

## 6.3信息类别管理模块设计

### 6.3.1功能结构设计

信息类别管理包图如图6-6所示。

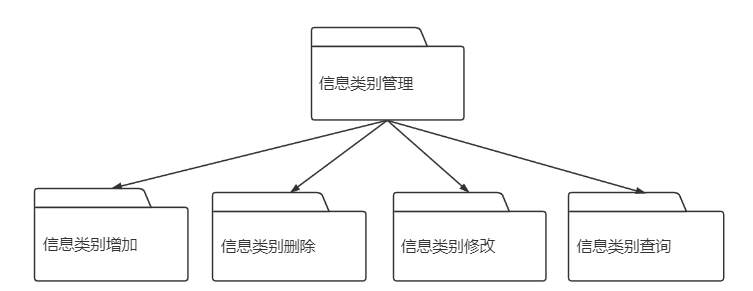


图6-6信息类别管理包图

### 6.3.2类图设计

信息类别管理类图如图6-7所示。

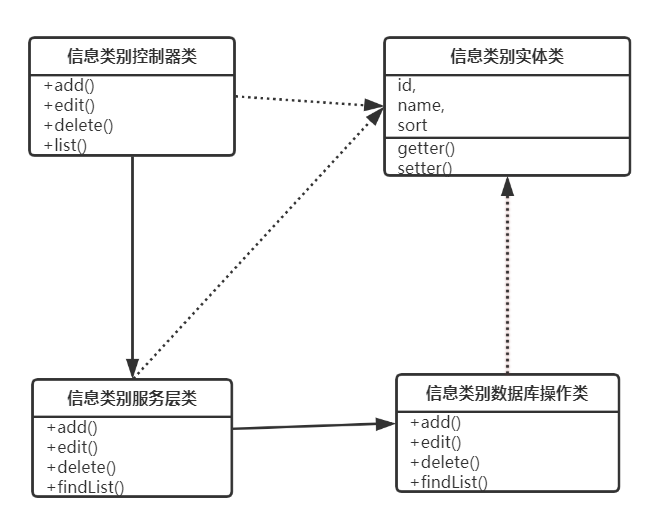


图6-7信息类别管理类图

在信息类别管理类图中，主要涉及到四个类的关联，信息类别控制器类关联信息类别服务层类，同时关联信息类别实体类。信息类别服务层类关联信息类别数据库操作类，同时关联信息类别实体类。信息类别管理涉及到的各个类的描述如下所示，其中信息类别控制器类描述如表6-4所示。

表6-4信息类别控制器类描述

|  |  |  |
| --- | --- | --- |
| 信息类别控制器类 | | |
| 类的描述：信息类别控制器类 | | |
| 操作描述 | add () | 增加类别 |
| edit() | 修改类别 |
| delete () | 删除类别 |
| List() | 分页获取所有类别 |

信息类别服务层类描述如表6-5所示。

表6-5信息类别服务层类描述

|  |  |  |
| --- | --- | --- |
| 信息类别服务层类 | | |
| 类的描述：信息类别服务层类 | | |
| 操作描述 | add () | 增加类别 |
| edit() | 修改类别 |
| delete () | 删除类别 |
| findList() | 分页获取所有类别 |

信息类别数据库操作类描述如表6-6所示。

表6-6信息类别数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 信息类别数据库操作类 | | |
| 类的描述：信息类别数据库操作类 | | |
| 操作描述 | add () | 增加类别 |
| edit() | 修改类别 |
| delete () | 删除类别 |
| findList() | 分页获取所有类别 |

### 6.3.3顺序图设计

信息类别管理顺序图如图6-8所示。

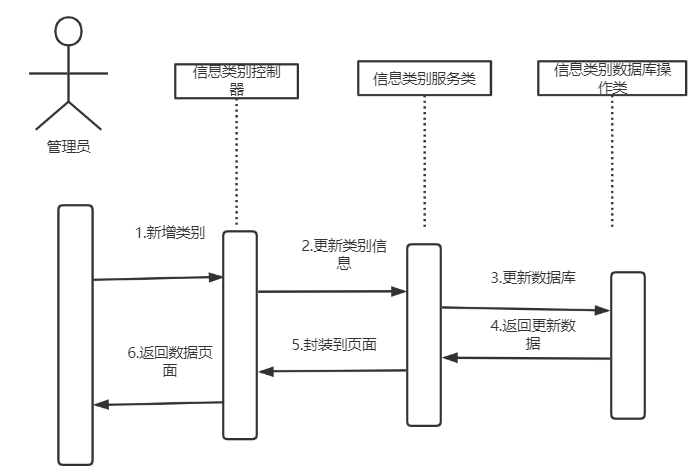


图6-8信息类别管理顺序图

在信息类别顺序图中，管理员点击新增类别信息，此时前台请求通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回更新后的数据，然后进行封装数据再由控制器层返回到前台的展示数据页面。

### 6.3.4 核心处理流程设计

信息类别管理活动图如图6-9所示。

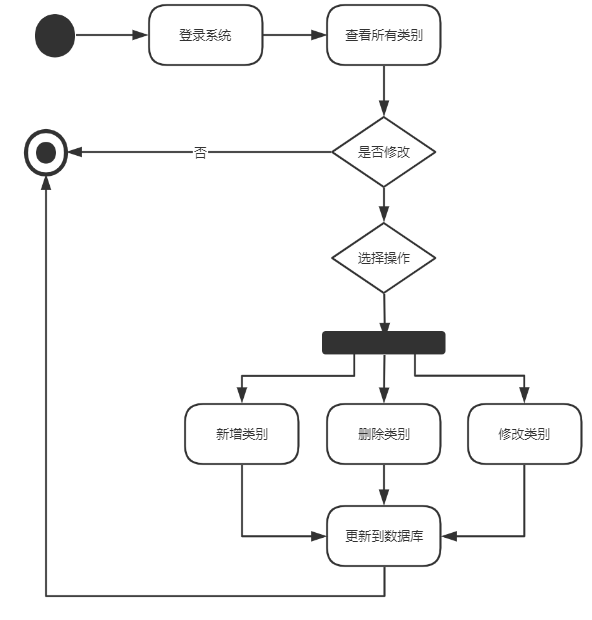


图6-9 信息类别管理活动图

## 6.4在线办事管理模块设计

### 6.4.1功能结构设计

在线办事管理包图如图6-10所示。

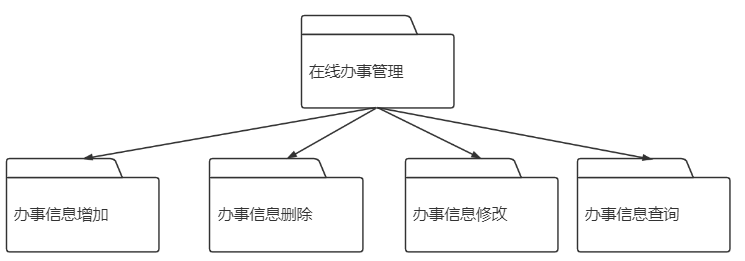


图6-10在线办事管理包图

### 6.4.2类图设计

在线办事管理类图如图6-11所示。

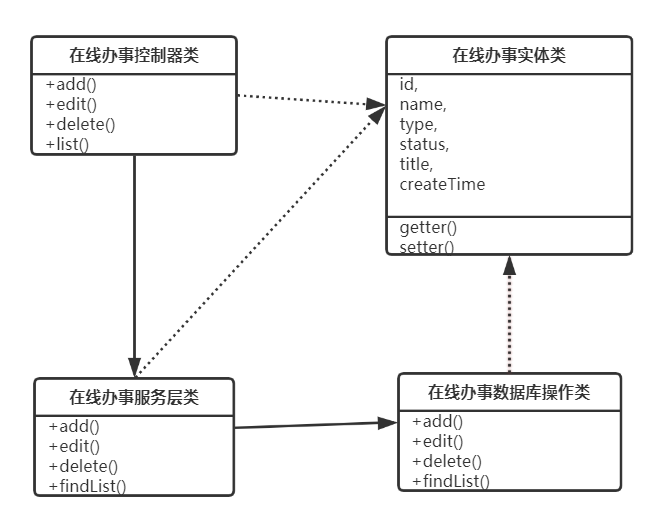


图6-11在线办事管理类图

在线办事管理类图中，主要涉及到四个类的关联，在线办事控制器类关联在线办事服务层类，同时关联在线办事实体类。在线办事服务层类关联在线办事数据库操作类，同时关联在线办事实体类。在线办事管理涉及到的各个类的描述如下所示，其中在线办事控制器类描述如表6-7所示。

表6-7在线办事控制器类描述

|  |  |  |
| --- | --- | --- |
| 在线办事控制器类 | | |
| 类的描述：在线办事控制器类 | | |
| 操作描述 | add () | 增加办事信息 |
| edit() | 修改办事信息 |
| delete () | 删除办事信息 |
| List() | 分页获取所有办事信息列表 |

在线办事服务层类描述如表6-8所示。

表6-8在线办事服务层类描述

|  |  |  |
| --- | --- | --- |
| 在线办事服务层类 | | |
| 类的描述：在线办事服务层类 | | |
| 操作描述 | add () | 增加办事信息 |
| edit() | 修改办事信息 |
| delete () | 删除办事信息 |
| findList() | 分页获取所有办事信息列表 |

在线办事数据库操作类描述如表6-9所示。

表6-9在线办事数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 在线办事数据库操作类 | | |
| 类的描述：在线办事数据库操作类 | | |
| 操作描述 | add () | 增加办事信息 |
| edit() | 修改办事信息 |
| delete () | 删除办事信息 |
| findList() | 分页获取所有办事信息列表 |

### 6.4.3顺序图设计

在线办事管理顺序图如图6-12所示。

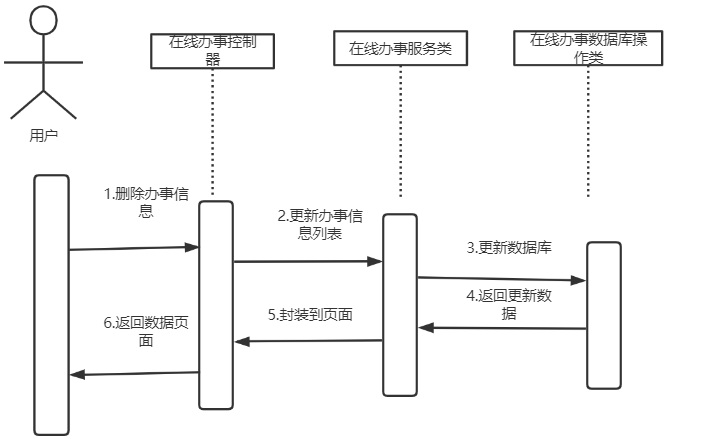


图6-12在线办事管理顺序图

在在线办事管理顺序图中，用户点击新增办事信息，此时前台请求通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回更新的数据，然后进行封装数据再由控制器层返回到前台的展示数据页面。

### 6.4.4 核心处理流程设计

在线办事管理活动图如图6-13所示。

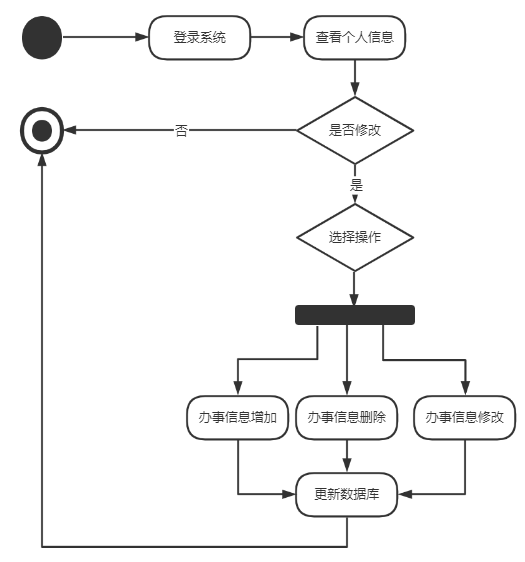


图6-13 在线办事管理活动图

## 6.5评论管理模块设计

### 6.5.1功能结构设计

评论管理包图如图6-14所示。

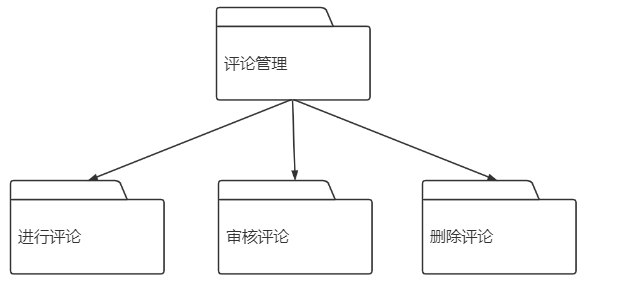


图6-14评论管理包图

### 6.5.2类图设计

评论管理类图如图6-15所示。

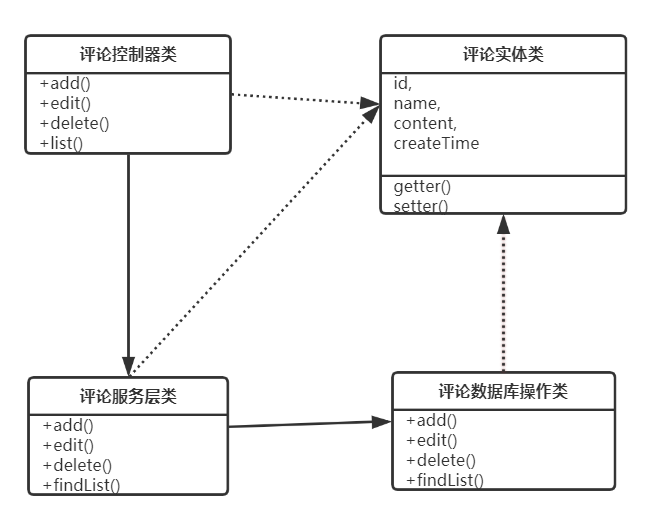


图6-15评论管理类图

评论管理类图中，主要涉及到四个类的关联，评论控制器类关联评论服务层类，同时关联评论实体类。评论服务层类关联评论数据库操作类，同时关联评论实体类。评论管理涉及到的各个类的描述如下所示，其中评论控制器类描述如表6-10所示。

表6-10评论控制器类描述

|  |  |  |
| --- | --- | --- |
| 评论控制器类 | | |
| 类的描述：评论控制器类 | | |
| 操作描述 | add () | 增加评论信息 |
| edit() | 修改评论信息 |
| delete () | 删除评论信息 |
| List() | 分页获取所有评论列表 |

评论服务层类描述如表6-11所示。

表6-11评论服务层类描述

|  |  |  |
| --- | --- | --- |
| 评论服务层类 | | |
| 类的描述：评论服务层类 | | |
| 操作描述 | add () | 增加评论信息 |
| edit() | 修改评论信息 |
| delete () | 删除评论信息 |
| findList() | 分页获取所有评论信息列表 |

评论数据库操作类描述如表6-12所示。

表6-12评论数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 评论数据库操作类 | | |
| 类的描述：评论数据库操作类 | | |
| 操作描述 | add () | 增加评论信息 |
| edit() | 修改评论信息 |
| delete () | 删除评论信息 |
| findList() | 分页获取所有评论信息列表 |

### 6.5.3顺序图设计

评论管理顺序图如图6-16所示。

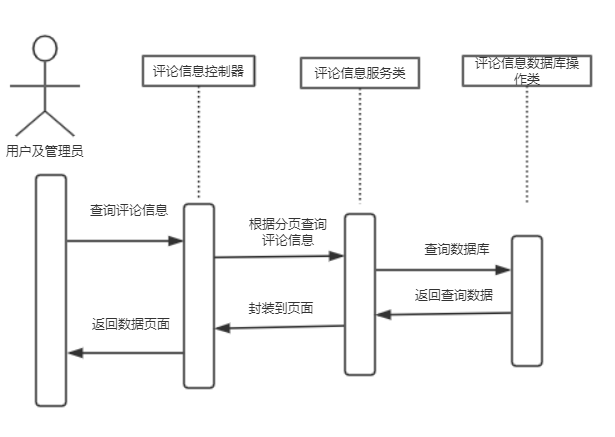


图6-16评论管理顺序图

在评论管理顺序图中，用户和管理员查看所有评论信息，此时前台请求通过控制器层进行处理，然后修改的请求进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回更新后的数据，然后进行封装数据再由控制器层返回到前台的展示数据页面。

### 6.5.4 核心处理流程设计

评论管理活动图如图6-17所示。

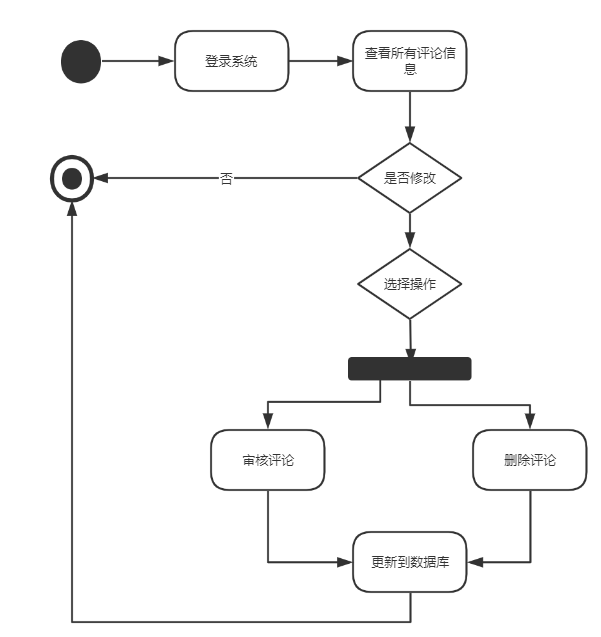


图6-17 评论管理活动图

## 6.6留言管理模块设计

### 6.6.1功能结构设计

留言管理包图如图6-18所示。

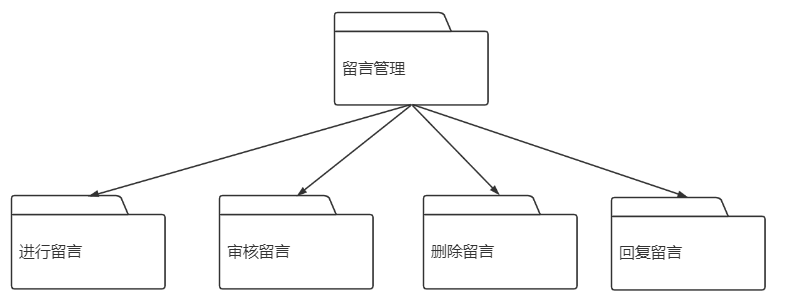


图6-18留言管理包图

### 6.6.2类图设计

留言管理类图如图6-19所示。

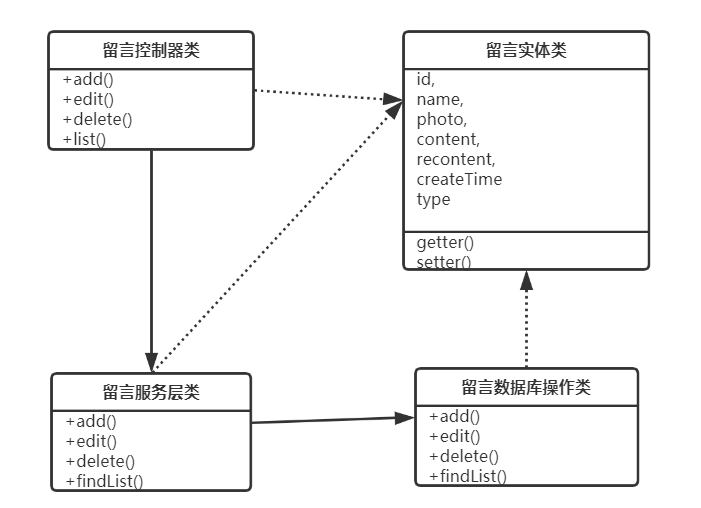


图6-19留言管理类图

留言管理类图中，主要涉及到四个类的关联，留言控制器类关联评论服务层类，同时关联留言实体类。留言服务层类关联留言数据库操作类，同时关联留言实体类。留言管理涉及到的各个类的描述如下所示，其中留言控制器类描述如表6-13所示。

表6-13留言控制器类描述

|  |  |  |
| --- | --- | --- |
| 留言控制器类 | | |
| 类的描述：留言控制器类 | | |
| 操作描述 | add () | 增加留言信息 |
| edit() | 修改留言信息 |
| delete () | 删除留言信息 |
| List() | 分页获取所有留言列表 |

留言服务层类描述如表6-14所示。

表6-14留言服务层类描述

|  |  |  |
| --- | --- | --- |
| 留言服务层类 | | |
| 类的描述：留言服务层类 | | |
| 操作描述 | add () | 增加留言信息 |
| edit() | 修改留言信息 |
| delete () | 删除留言信息 |
| findList() | 分页获取所有留言信息列表 |

留言数据库操作类描述如表6-15所示。

表6-15留言数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 留言数据库操作类 | | |
| 类的描述：留言数据库操作类 | | |
| 操作描述 | add () | 增加留言信息 |
| edit() | 修改留言信息 |
| delete () | 删除留言信息 |
| findList() | 分页获取所有留言信息列表 |

### 6.6.3顺序图设计

留言管理顺序图如图6-20所示。

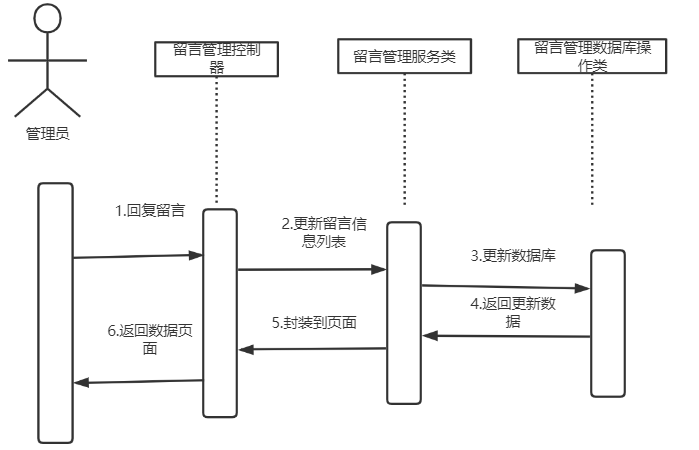


图6-20留言管理顺序图

在留言顺序图中，管理员回复某条留言，此时后台请求通过控制器层进行处理，然后修改的请求进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回页面。

### 6.6.4核心处理流程设计

留言管理活动图如图6-21所示。



图6-21 留言管理活动图

## 6.7附件下载管理模块设计

### 6.7.1功能结构设计

附件下载管理包图如图6-22所示

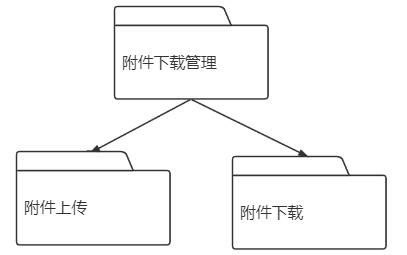


图6-22附件下载管理包图

### 6.7.2类图设计

附件管理类图如图6-23所示。

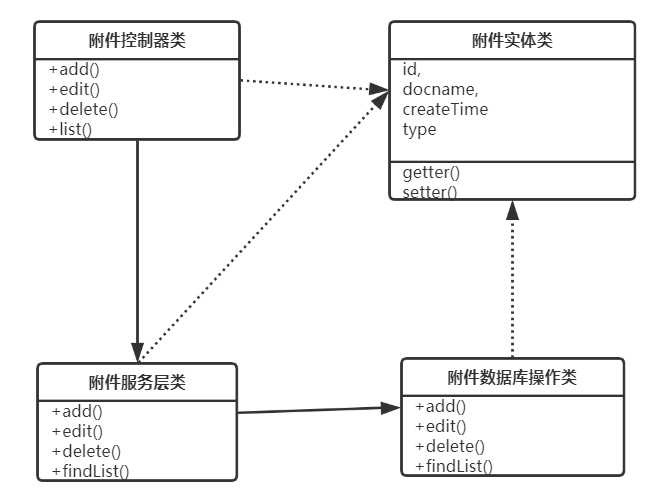


图6-23附件管理类图

附件下载管理类图中，主要涉及到四个类的关联，附件下载控制器类关联附件下载服务层类，同时关联附件下载实体类。附件下载服务层类关联附件下载数据库操作类，同时关联附件下载实体类。附件下载管理涉及到的各个类的描述如下所示，其中附件下载控制器类描述如表6-16所示。

表6-16附件下载控制器类描述

|  |  |  |
| --- | --- | --- |
| 附件下载控制器类 | | |
| 类的描述：附件下载控制器类 | | |
| 操作描述 | add () | 增加附件信息 |
| edit() | 修改附件信息 |
| delete () | 删除附件信息 |
| List() | 分页获取所有附件列表 |

附件下载服务层类描述如表6-17所示。

表6-17附件下载服务层类描述

|  |  |  |
| --- | --- | --- |
| 附件下载服务层类 | | |
| 类的描述：附件下载服务层类 | | |
| 操作描述 | add () | 增加附件信息 |
| edit() | 修改附件信息 |
| delete () | 删除附件信息 |
| findList() | 分页获取所有附件列表 |

附件下载数据库操作类描述如表6-18所示。

表6-18附件下载数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 附件下载数据库操作类 | | |
| 类的描述：附件下载数据库操作类 | | |
| 操作描述 | add () | 增加附件信息 |
| edit() | 修改附件信息 |
| delete () | 删除附件信息 |
| findList() | 分页获取所有附件列表 |

### 6.7.3顺序图设计

附件下载管理顺序图如图6-24所示。



图6-24附件下载管理顺序图

在附件下载管理顺序图中，管理员点击上传附件，此时后台通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果返回更新后的数据，然后进行封装数据再由控制器层返回到前台的展示数据页面。

### 6.7.4核心处理流程设计

附件下载管理活动图如图6-25所示。

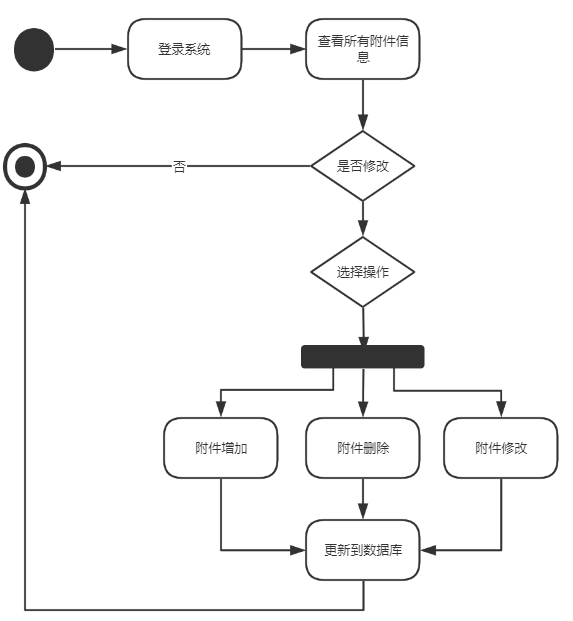


图6-25 附件下载管理活动图

## 6.8网站资讯信息管理模块设计

### 6.8.1功能结构设计

网站资讯信息管理包图如图6-26所示。

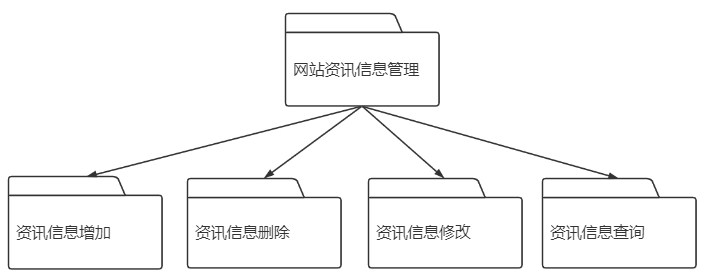


图6-26网站资讯信息管理包图

### 6.8.2类图设计

网站资讯信息管理类图如图6-27所示。

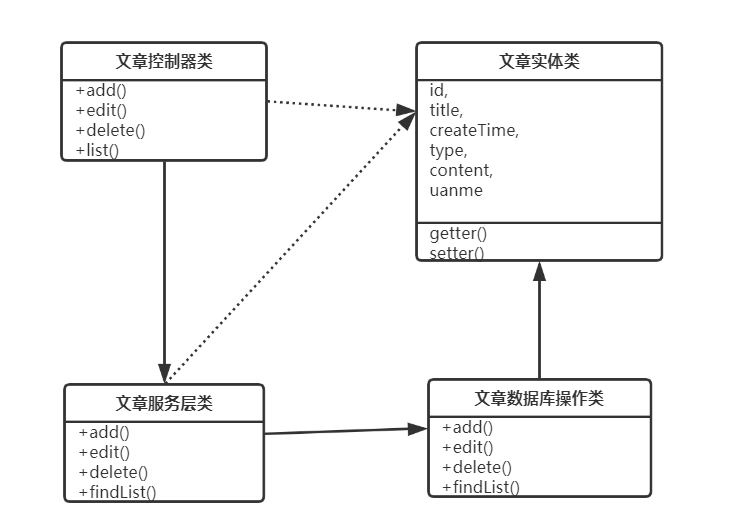


图6-27网站资讯信息管理类图

网站资讯信息管理类图中，主要涉及到四个类的关联，网站资讯信息控制器类关联网站资讯信息服务层类，同时关联网站资讯信息实体类。网站资讯信息服务层类关联网站资讯信息数据库操作类，同时关联网站资讯信息实体类。网站资讯信息管理涉及到的各个类的描述如下所示，其中网站资讯信息控制器类描述如表6-19所示。

表6-19网站资讯信息控制器类描述

|  |  |  |
| --- | --- | --- |
| 网站资讯信息控制器类 | | |
| 类的描述：网站资讯信息控制器类 | | |
| 操作描述 | add () | 增加网站资讯信息 |
| edit() | 修改网站资讯信息 |
| delete () | 删除网站资讯信息 |
| List() | 分页获取所有网站资讯信息列表 |

网站资讯信息服务层类描述如表6-20所示。

表6-20网站资讯信息服务层类描述

|  |  |  |
| --- | --- | --- |
| 网站资讯信息服务层类 | | |
| 类的描述：网站资讯信息服务层类 | | |
| 操作描述 | add () | 增加网站资讯信息 |
| edit() | 修改网站资讯信息 |
| delete () | 删除网站资讯信息 |
| findList() | 分页获取所有网站资讯信息列表 |

网站资讯信息数据库操作类描述如表6-21所示。

表6-21网站资讯信息数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 网站资讯信息数据库操作类 | | |
| 类的描述：网站资讯信息数据库操作类 | | |
| 操作描述 | add () | 增加网站资讯信息 |
| edit() | 修改网站资讯信息 |
| delete () | 删除网站资讯信息 |
| findList() | 分页获取所有网站资讯信息列表 |

### 6.8.3顺序图设计

网站资讯信息管理顺序图如图6-28所示。

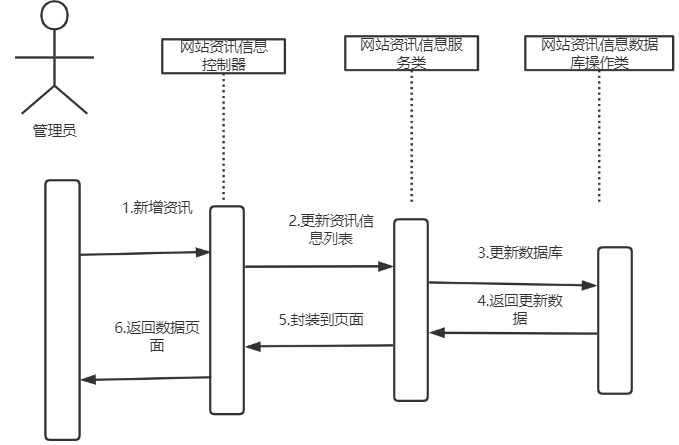


图6-28网站资讯信息管理顺序图

在网站资讯信息顺序图中，管理员点击新增资讯信息，此时前台请求通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果就可以返回前台页面。

### 6.8.4核心处理流程设计

网站资讯信息管理活动图如图6-29所示。

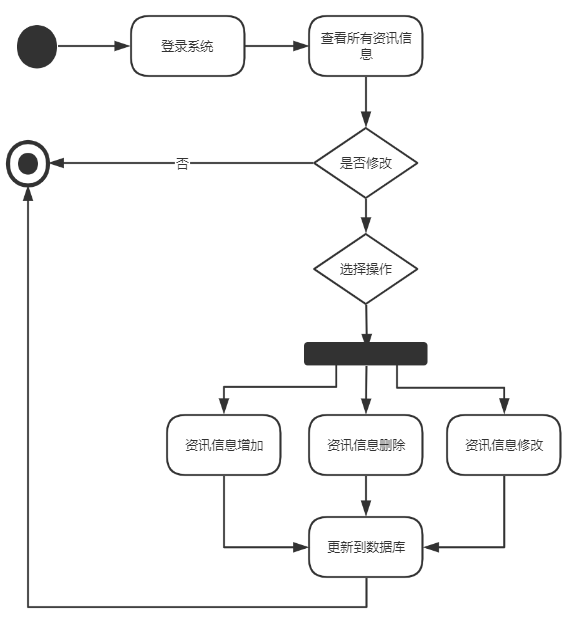


图6-29 网站资讯信息管理活动图

## 6.9图片素材管理模块设计

### 6.9.1功能结构设计

图片素材管理包图如图6-30所示。

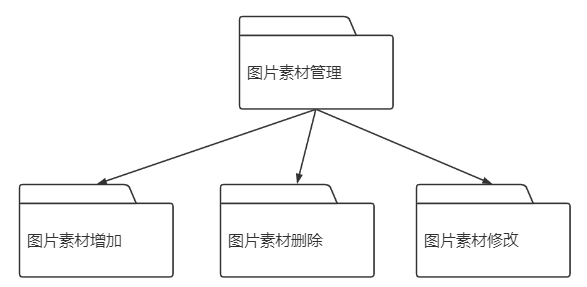


图6-30图片素材管理包图

### 6.9.2类图设计

图片素材管理类图如图6-31所示。

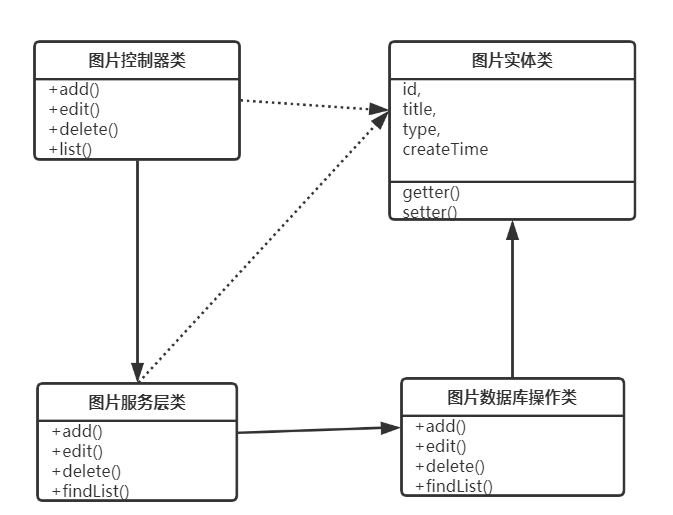


图6-31图片素材管理类图

图片素材管理类图中，主要涉及到四个类的关联，图片素材控制器类关联图片素材服务层类，同时关联图片素材实体类。图片素材服务层类关联图片素材数据库操作类，同时关联图片素材实体类。图片素材管理涉及到的各个类的描述如下所示，其中图片素材控制器类描述如表6-22所示。

表6-22图片素材控制器类描述

|  |  |  |
| --- | --- | --- |
| 图片素材控制器类 | | |
| 类的描述：图片素材控制器类 | | |
| 操作描述 | add () | 增加图片素材 |
| edit() | 修改图片素材 |
| delete () | 删除图片素材 |
| List() | 分页获取所有图片素材列表 |

图片素材服务层类描述如表6-23所示。

表6-23图片素材服务层类描述

|  |  |  |
| --- | --- | --- |
| 图片素材服务层类 | | |
| 类的描述：图片素材服务层类 | | |
| 操作描述 | add () | 增加图片素材 |
| edit() | 修改图片素材 |
| delete () | 删除图片素材 |
| findList() | 分页获取所有图片素材列表 |

图片素材数据库操作类描述如表6-24所示。

表6-24图片素材数据库操作类描述

|  |  |  |
| --- | --- | --- |
| 图片素材数据库操作类 | | |
| 类的描述：图片素材数据库操作类 | | |
| 操作描述 | add () | 增加图片素材 |
| edit() | 修改图片素材 |
| delete () | 删除图片素材 |
| findList() | 分页获取所有图片素材列表 |

### 6.9.3顺序图设计

图片素材管理顺序图如图6-32所示。

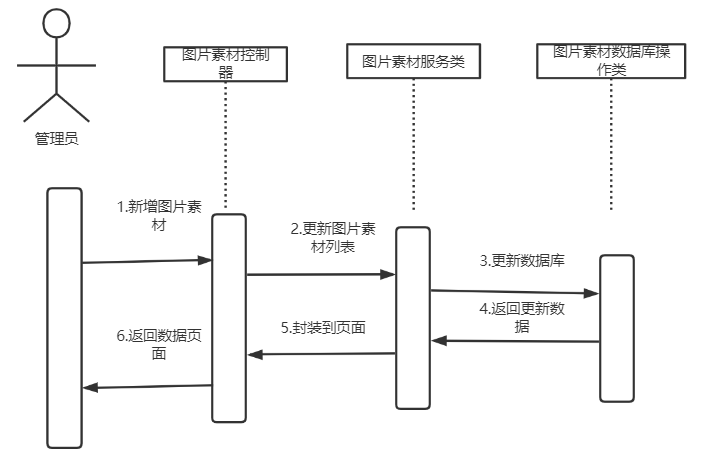


图6-32图片素材管理顺序图

在图片素材顺序图中，管理员点击修改新增图片素材，此时前台请求通过控制器层进行处理，然后修改的请求就可以进入服务层，服务层再调用数据库层，数据库层拿到数据库更新结果就可以进行封装返回前台页面

### 6.9.4核心处理流程设计

图片素材管理活动图如图6-33所示。

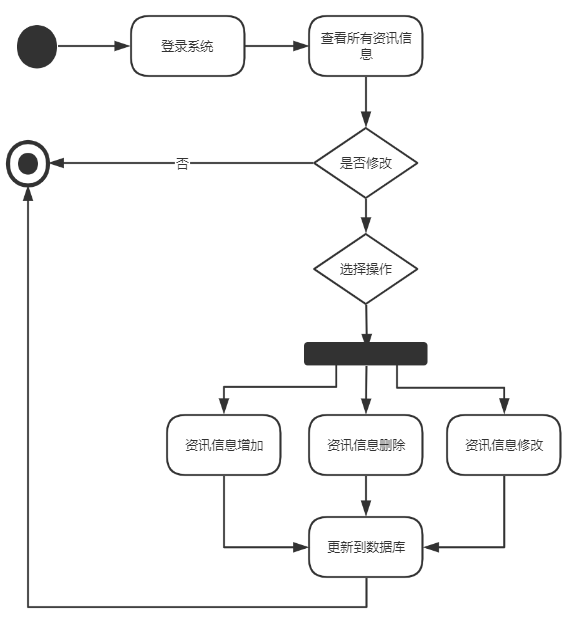


图6-33 图片素材管理活动图

7 编码

## 7.1 代码实现与核心算法

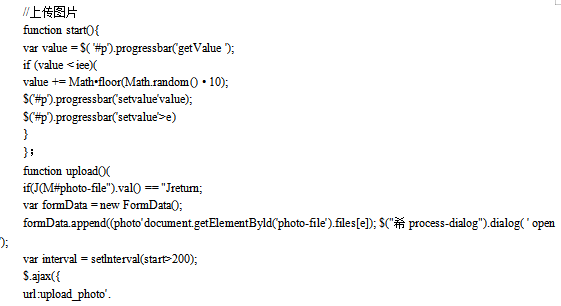
(1) 创建验证码核心代码，本段代码主要是验证码图片创建。



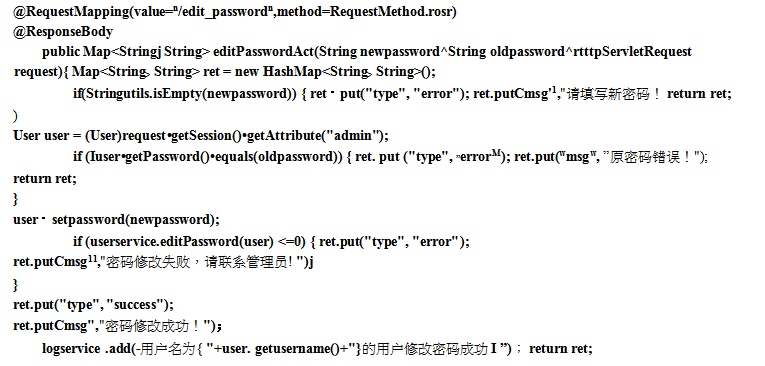
（2）后台登录拦截器核心代码，本段代码主要是跳转重定向。



（3）Ajax异步上传图片核心代码，本段代码主要是上传图片。

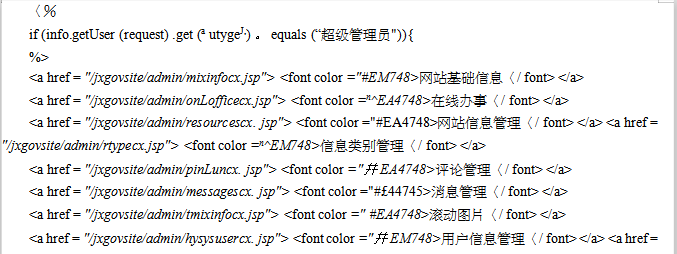


（4）修改密码核心代码，本段代码主要是修改密码。



## 7.2 代码优化分析

针对系统游客/用户的查询浏览操作，系统将及时返回给用户对应的新闻页面。其中顶部导航栏样式和搜索框功能会始终显示在上部，以及根据不同的资讯类别返回所有新闻信息时的页面，也采用了复用形式：及返回至写好的同一页面，采用<c:forEach>循环和MyBatis的分页插件PageHelper将搜索到的信息以列表形式插入进同一页面。避免了同时创建了多个冗余的页面。



8 测试

## 8.1 测试方案设计

### 8.1.1 测试策略

在省电子政务及政府信息门户网站的开发这一漫长阶段中，会面临各种各样的难题，网络技术瞬息万变，我们技术人员要想不被时代淘汰，就必须让自己与时俱进。同时我们要一边开发一边进行测试，因为我们如果等到开发完成再进行测试，可能那时发现了错误再想补救就已经晚了，或者要付出不可承受的代价。因此我们在开发的过程中要对系统的每一个环节进行完备的测试。

在省电子政务及政府信息门户网站的测试方法为白盒测试和黑盒测试。测试的步骤为先进行单元测试，然后对每一个子系统进行测试，接着进入系统测试,再进行验收测试，最后是平行运行。

### 8.1.2 测试进度安排

在省电子政务及政府信息门户网站的测试进度安排如下表8-1所示。

表8-1 测试进度安排表

|  |  |  |
| --- | --- | --- |
| 测试阶段 | 测试任务 | 测试日期及用时 |
| 单元测试 | 将白盒测试应用于模块接口、重要通路、出错通路、数据结构等。 | 第8学期4~7周 |
| 黑盒测试  （前台部分） | 将黑盒测试应用于系统前台所有功能，收集测试结果和问题。 | 第8学期第7周 |
| 黑盒测试  （后台部分） | 将黑盒测试应用于系统后台所有功能，收集测试结果和问题。 | 第8学期第6周 |
| 白盒测试 | 对系统重要功能点进行白盒测试，确保程序主要执行通路可以达到预期结果。 | 第8学期第7周 |
| 系统集成测试 | 对系统进行测试和组装，检查是否有与接口相关的问题。 | 第8学期第7周 |

### 8.1.3 测试资源

在省电子政务及政府信息门户网站作为一套B/S架构的项目，一旦我们在线上发布，毫无疑问，我们的用户会通过不同设备进行访问，故测试资源中必须包含不同浏览器，不同设备以测试在不同浏览器，不同的手机设备环境下是否能够正常运行，测试资源表(硬件环境)如表8-2所示。

表8-2 测试资源表

|  |  |  |
| --- | --- | --- |
| 硬件环境 | | |
| 资源类型 | 资源描述 | 数量 |
| 内存 | 8G DDR3L | 1 |
| 硬盘 | 256G SATA3接口 | 1 |
| 处理器 | Intel I5-4210M | 1 |
| 操作系统 | Windows 10 1903企业版 | 1 |
| JDK版本 | JDK 9.0.1 | 1 |
| 浏览器1 | Chrome 浏览器 | 1 |
| 浏览器2 | FireFox 浏览器 | 1 |
| 浏览器3 | 360安全浏览器 | 1 |
| 浏览器4 | IE浏览器 | 1 |
| 手机1 | Moto G4 | 1 |
| 手机2 | Galaxy S5 | 1 |
| 手机3 | iPhone X | 1 |
| 手机4 | iPhone 8plus | 1 |
| Ipad1 | Ipad | 1 |
| Ipad2 | Ipad Pro | 1 |

测试资源表(软件环境)如表8-3所示。

表8-3 测试资源表(软件环境)

|  |  |
| --- | --- |
| 软件环境 | |
| 资源类型 | 软件 |
| 应用数据库 | Mysql 5.7 |
| 应用服务器 | Tomcat 8.5.16 |

### 8.1.4 关键测试点

（1）用户能否进行在线办事提交。

（2）用户能否对文章进行评论。

（3）用户能否在留言板进行留言。

（4）管理员能否编辑信息类别。

（5）管理员能否编辑网站文章信息。

（6）用户能否下载附件。

（7）超级管理员能否对留言进行回复。

（8）超级管理员能否进行办事审批。

## 8.2 测试用例构建

### 8.2.1 测试用例编写约定

（1）用例编号：测试用例的编号，具有唯一性。

（2）用例名称：测试用例的命名。

（3）前提条件：该测试用例执行前需要满足的条件。

（4）操作步骤：测试用例如何执行。

（5）预期结果：理论上测试用例应该得到的结果。

（6）实际结果：实际上测试用例应该得到的结果。

（7）问题描述：测试用例执行后的错误表现。

（8）BUG编号：出错的编号。

（9）测试结果与结论：选择通过或者不通过。

（10）测试执行者：测试用例执行的人员。

### 8.2.2 测试用例设计

本系统测试用例设计大致如表8-4所示：

表 8-4 测试用例设计

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 测试项 | 描述/输入/操作 | 期望结果 |
| 01 | 游客注册 | 注册信息与已注册用户一致 | 提示该信息已存在 |
| 02 | 用户搜索 | 输入文章关键字进行搜索 | 跳转至对相应文章页面 |
| 03 | 进行评论 | 进行评论 | 评论内容同步显示 |
| 04 | 修改文章 | 后台修改文章相关信息 | 前台文章信息同步 |
| 05 | 新增类别 | 后台新增资讯类别 | 前台同步资讯类别 |
| 06 | 进行留言 | 进行留言 | 留言内容同步显示 |
| 07 | 在线办事 | 在线申请一条办事记录 | 办事列表更新记录 |
| 08 | 附件下载 | 下载文章里面的附件 | 附件下载成功 |

(1) 游客注册测试用例表如表8-5所示。

表8-5 游客注册测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL01 |
| 用例名称 | 游客注册 |
| 前提条件 | 无 |
| 操作步骤 | 游客点击注册，输入一个系统以及存在的用户名。 |
| 预期结果 | 提示该用户名已存在 |

续表8-5

|  |  |
| --- | --- |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(2) 用户搜索测试用例表如表8-6所示。

表8-6 用户搜索测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL02 |
| 用例名称 | 用户搜索 |
| 前提条件 | 用户成功登录系统 |
| 操作步骤 | 1. 用户输入相应搜索内容的关键字，点击搜索按钮。 |
| 预期结果 | 跳转至对相应文章页面 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(3) 用户评论测试用例表如表8-7所示。

表8-7 用户评论测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL03 |
| 用例名称 | 用户评论 |
| 前提条件 | 用户成功登录系统 |
| 操作步骤 | 1. 用户点进一篇文章 2. 输入评论内容 |
| 预期结果 | 评论内容同步显示。 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |

续表8-7

|  |  |
| --- | --- |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(4) 用户留言测试用例表如表8-8所示。

表8-8 用户留言测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL04 |
| 用例名称 | 用户留言 |
| 前提条件 | 用户成功登录系统 |
| 操作步骤 | 1. 用户点进留言板  2. 输入留言内容。 |
| 预期结果 | 留言内容同步显示。 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

### 8.2.3 关键测试用例

(1) 修改文章信息测试用例表如表8-9所示。

表8-9 修改文章信息测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL05 |
| 用例名称 | 修改文章信息 |
| 前提条件 | 管理员成功登录系统 |
| 操作步骤 | 1. 管理员选择一篇文章，点击修改按钮。 2. 修改文章相关信息后点击确定。 |
| 预期结果 | 系统提示修改成功并刷新页面。 |
| 实际结果 | 与预期结果相符 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(2) 新增类别测试用例表如表8-10所示。

表8-10 新增类别测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL06 |
| 用例名称 | 新增类别 |
| 前提条件 | 管理员成功登录系统，并进入类别管理页 |
| 操作步骤 | 1．管理员点击新增信息  2．输入类别的标题 |
| 预期结果 | 前台同步资讯类别 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(3) 在线办事测试用例表如表8-11所示。

表8-11 在线办事测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL07 |
| 用例名称 | 在线办事 |
| 前提条件 | 用户成功登录系统 |
| 操作步骤 | 1. 用户点击添加信息 2. 输入在线办事的内容。 |
| 预期结果 | 办事列表更新一条记录 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

(4) 附件下载测试用例表如表8-12所示。

表8-12 附件下载测试用例表

|  |  |
| --- | --- |
| 用例编号 | CSYL08 |
| 用例名称 | 附件下载 |
| 前提条件 | 用户成功登录系统 |
| 操作步骤 | 1用户点进文章详情页面  2选中附件点击下载 |
| 预期结果 | 提示附件下载成功 |
| 实际结果 | 与预期结果相符 |
| 问题描述 | 暂无 |
| BUG编号 | 暂无 |
| 测试结果与结论 | 测试通过 |
| 测试执行者 |  |

### 8.2.4 测试用例维护

（1）及时添加新的测试用例。在系统开发周期过程中，一些需求可能会被添加进来，而已有的测试用例可能覆盖不到新的需求，此时需要增加测试用例来覆盖到新的需求中。

（2）及时删除多余的测试用例。在测试用例设计时，由于考虑不周可能导致多组测试用例经过相同的执行通路，此时需要剔除多余的测试用例保持一组即可。

（3）改进测试用例。在系统开发周期过程中，可能在初期一组测试用例覆盖多条测试路径，但由于后期系统改进导致此组测试用例覆盖度不如之前，但也覆盖到多数，这是需要改进测试用例。

（4）及时删除无效的测试用例。在系统开发周期过程中，可能由于功能的删除而导致此组测试用例无效，这时就需要删除这组无效的测试用例。

9 总结与展望

## 9.1 设计工作总结

经过此次政府门户网站的设计与实现，我从中学习到了不少知识，当然这并不足以让我满足，我深知自己还要学习的知识很多，就单纯在网站设计方面，需要学习的东西也是不计其数的，可以说我所掌握的只不过是一些比较基础、比较简单的，要想更好的进行网站开发，进一步的学习是必不可少的。

通过政府门户网站的设计与实现，理解了Ajax技术的基本原理，并初步了解了基于Ajax和java技术开发Web应用的流程，对于如：JDK、My eclipse、tomcat、Mysql等各类开发工具的安装配置以及开发环境的搭建有了初步的认识。

但是由于技术水平有限，系统的实现还存在不足之处，下面就作简要介绍：

首先：本政府门户网站比较简单，并不像现有政府门户网站那样功能强大，毕竟还处于学习阶段，很难做到尽善尽美，但是总的来说，其基本功能已经实现，而且，我也确实从此次开发中了解了基于Ajax和java技术开发Web应用的流程，我想这比什么都重要。

其次：页面的想过给人的感觉不是那么美观，因为我在开发过程当中，没有把美观作为开发工作的重点，因此没有加入背景图片，也没有对页面进行更高水平的CSS样式规范，因此整个页面给人比较单调的感觉。但是我相信，基本功能我能够实现了，美化工作我也能够做得很好，这或许不会在这次毕业设计项目中得到体现，但在以后的学习工作中，在实际的项目开发中，美化也是必须的一步，当然这完全可以交给专门的美化工作组去完成，但是，我们还是应该学习更多的东西，多不代表浅，对于网站开发的各方面的知识，我们都应该有更深更广的学习。

然后：对于一些业务流程考虑不够全面，没有在现实生活中做到细致的调研，导致做出来过于简单，不符合日常生活中的办事流程，距离成熟的省电子政府及政府信息门户网站相差甚远。但是既然知道了自己的不足之处，我在以后学习中就会多加注意和继续完善。

## 9.2 未来工作展望

学海无涯，通过这次毕业设计我学到了很多新的东西，虽然系统已经完成，但是同时发现自己非常多不足的地方，例如，网站的在线办事功能不太贴合用户日常的办事流程。自己以后工作时应该多多从用户的角度去考虑问题，很多问题不能光凭自己的想法，只有换位思考，才能达到共赢。互联网技术日新月异，自己在未来的日子里面应该更加勤勉，去不断学习新的东西，不断增加自己知识的深度和广度，希望自己以后能够越做越好。

谢 辞

同时，我也要感谢我们整个项目开发小组对我的支持和帮助，在他们的帮助下，我顺利地解决了很多开发过程中遇到的问题和困难，并得到他们良好的建议。

最后，向所有曾经帮助，关心过我的家人、老师、同学和朋友们致以真诚的谢意！

参考文献

[1]李咏.Web开发中MVC设计模式的研究与应用[J]. 高校技术开发月刊, 2016(9):55-56.

[2]方斌, 刘耀钦. 基于B/S模式使用JSP+MySQL进行Web开发[J]. 郧阳师范高等专科高校学报, 2018, 34(3):15-16.

[3]李张永, 陈和平, 顾进广. 跨平台移动Web开发框架与数据交互方法[J]. 计算机工程与设计, 2017, 35(5):1827-1832.

[4]孙奇.基于MVC模式的Web开发框架Ruby on Rails的研究[D]. 北京交通大学, 2016.

[5]马晓波.用 J2EE 构造网站技术的研究[D]内蒙古:内蒙古大学，2015.

[6]凤平.Web 网站安全技术分析[J]计算机系统应用，2017 年 12 月

[7]池超明.葫芦岛港集团门户网站设计与实现[D]吉林: 吉林大学，2016

[8]李红升.基于 ASP 的网站开发[J]计算机光盘软件与应用，2016（08）:

[9]刘中兵.Java Web 核心框架[M].电子工业出版社，2016

[10]张玉.贵阳市政府门户网站的设计与实现[D].云南大学.2018.

[11]Rasmus Lerdorf ,Peter Mac Intyre.Kevin Tahoe Programming PHP ， Edition[M] O·Reilly， 2016.4.

[12]张海籓，牟永敏编 .软件工程导论（第6版）[M]. 北京：清华大学出版社，2018

[13]杨洁. 基于ASP.NET Ajax的Web开发研究及应用[D]. 武汉理工大学, 2015.

[14]贾广宇. MVC设计模式下Web开发框架的研究与应用[D]. 大连海事大学, 2016.

[15]李浩.基于J2EE的WEB开发框架的研究与应用[D]. 汕头大学, 2015.

[16]Kelley L A, Al. E. The Phyre2 web portal for protein modeling, prediction and analysis.[J]. Nature Protocol, 2016, 10(6):845-58.

[17]Ferrel P J, Kawai K, Everhart D, et al. Distributing web applications across a pre-existing web: US, US8943035[P]. 2018.

[18]Christophe Mazzia,Yvan Capowiez,Gaëlle Marliac,Didier Josselin,Alain Pasquet. Spinosad application in an apple orchard affects both the abundance of the spider Araneus diadematus and its web construction behaviour[J]. Ecotoxicology,2020,29(4).

[19]Tiago Diehl de Souza,Enedir Ghisi. Harvesting rainwater from scaffolding platforms and walls to reduce potable water consumption at buildings construction sites[J]. Journal of Cleaner Production,2020,258.

[20]Ling Chen,Yuhong Wang,Ming-Fung Francis Siu. Detecting semantic regions of construction site images by transfer learning and saliency computation[J]. Automation in Construction,2020,114.

[21]Mingyuan Zhang,Rui Shi,Zhen Yang. A critical review of vision-based occupational health and safety monitoring of construction site workers[J]. Safety Science,2020,126.

[22]Rasmus Lerdorf ,Peter Mac Intyre.Kevin Tahoe Programming PHP ， Edition[M] O·Reilly， 2016.4

[23]Deere Steven,Hui Xie,Galea Edwin R.,Cooney David. An evacuation model validation data-set for high-rise construction sites[J]. Fire Safety Journal,2020(prepublish).

[24]Husqvarna AB; Patent Issued for Identifying And Locating A Substitute Battery For A Construction Job Site Power Tool (USPTO 10,628,898)[J]. Computer Technology Journal,2020.

[25] Engineering - Construction Engineering; Study Findings from Shanghai Jiao Tong University Provide New Insights into Construction Engineering (Hazardous Area Risk-based Evacuation Simulation and Analysis of Building Construction Sites)[J]. Journal of Engineering,2020.

[26] Wix.com Ltd.; Researchers Submit Patent Application, "System And Method For The Generation Of An Adaptive User Interface In A Website Building System", for Approval (USPTO 20200117431)[J]. Internet Weekly News,2020.

[27]李艳, 方娟, 董帅. 基于先进可视化引擎的态势展示系统研究与实现[J]. 无线互联科技, 2019(8).

[28]魏曦, 孙艳芝. 基于自动建模的室内实景三维展示系统设计与实现[J]. 科技视界, 2019(12).

[29]杨丰宇, 于本成. Design and Research of Enterprise Promotion Platform Based on Internet%基于互联网+的企业推广平台设计与研究[J]. 电脑知识与技术, 2018, 014(006):61-62.

[30]赵春刚, 才智, 高志军, et al. UML使用基础教程[M]. 北京大学出版社, 2018.

附录A 外文翻译—原文部分

THE TECHNIQUE DEVELOPMENT HISTORY OF JSP

**By:**Kathy Sierra and Bert Bates

**Source:** Servlet&JSP

The Java Server Pages( JSP) is a kind of according to web of the script plait distance technique, similar carries the script language of Java in the server of the Netscape company of server- side JavaScript( SSJS) and the Active Server Pages(ASP) of the Microsoft. JSP compares the SSJS and ASP to have better can expand sex, and it is no more exclusive than any factory or some one particular server of Web. Though the norm of JSP is to be draw up by the Sun company of, any factory can carry out the JSP on own system.

The After Sun release the JSP( the Java Server Pages) formally, the this kind of new Web application development technique very quickly caused the people's concern. JSP provided a special development environment for the Web application that establishes the high dynamic state. According to the Sun parlance, the JSP can adapt to include the Apache WebServer, IIS4.0 on the market at inside of 85% server product.

This chapter will introduce the related knowledge of JSP and Databases, and JavaBean related contents, is all certainly rougher introduction among them basic contents, say perhaps to is a Guide only, if the reader needs the more detailed information, pleasing the book of consult the homologous JSP.

**1.1 GENERALIZE**

The JSP(Java Server Pages) is from the company of Sun Microsystems initiate, the many companies the participate to the build up the together of the a kind the of dynamic the state web the page technique standard, the it have the it in the construction the of the dynamic state the web page the strong but the do not the especially of the function. JSP and the technique of ASP of the Microsoft is very alike. Both all provide the ability that mixes with a certain procedure code and is explain by the language engine to carry out the procedure code in the code of HTML. Underneath we are simple of carry on the introduction to it.

JSP pages are translated into servlets. So, fundamentally, any task JSP pages can perform could also be accomplished by servlets. However, this underlying equivalence does not mean that servlets and JSP pages are equally appropriate in all scenarios. The issue is not the power of the technology, it is the convenience, productivity, and maintainability of one or the other. After all, anything you can do on a particular computer platform in the Java programming language you could also do in assembly language. But it still matters which you choose.

JSP provides the following benefits over servlets alone:

• It is easier to write and maintain the HTML. Your static code is ordinary HTML: no extra backslashes, no double quotes, and no lurking Java syntax.

• You can use standard Web-site development tools. Even HTML tools that know nothing about JSP can be used because they simply ignore the JSP tags.

• You can divide up your development team. The Java programmers can work on the dynamic code. The Web developers can concentrate on the presentation layer. On large projects, this division is very important. Depending on the size of your team and the complexity of your project, you can enforce a weaker or stronger separation between the static HTML and the dynamic content.

Now, this discussion is not to say that you should stop using servlets and use only JSP instead. By no means. Almost all projects will use both. For some requests in your project, you will use servlets. For others, you will use JSP. For still others, you will combine them with the MVC architecture . You want the appropriate tool for the job, and servlets, by themselves, do not complete your toolkit.

**1.2 SOURCE OF JSP**

The technique of JSP of the company of Sun, making the page of Web develop the personnel can use the HTML perhaps marking of XML to design to turn the end page with format. Use the perhaps small script future life of marking of JSP becomes the dynamic state on the page contents.( the contents changes according to the claim of)

The Java Servlet is a technical foundation of JSP, and the large Web applies the development of the procedure to need the Java Servlet to match with with the JSP and then can complete, this name of Servlet comes from the Applet, the local translation method of now is a lot of, this book in order not to misconstruction, decide the direct adoption Servlet but don't do any translation, if reader would like to, can call it as" small service procedure". The Servlet is similar to traditional CGI, ISAPI, NSAPI etc. Web procedure development the function of the tool in fact, at use the Java Servlet hereafter, the customer need not use again the lowly method of CGI of efficiency, also need not use only the ability come to born page of Web of dynamic state in the method of API that a certain fixed Web server terrace circulate. Many servers of Web all support the Servlet, even not support the Servlet server of Web directly and can also pass the additional applied server and the mold pieces to support the Servlet. Receive benefit in the characteristic of the Java cross-platform, the Servlet is also a terrace irrelevant, actually, as long as match the norm of Java Servlet, the Servlet is complete to have nothing to do with terrace and is to have nothing to do with server of Web. Because the Java Servlet is internal to provide the service by the line distance, need not start a progress to the each claimses, and make use of the multi-threading mechanism can at the same time for several claim service, therefore the efficiency of Java Servlet is very high.

But the Java Servlet also is not to has no weakness, similar to traditional CGI, ISAPI, the NSAPI method, the Java Servlet is to make use of to output the HTML language sentence to carry out the dynamic state web page of, if develop the whole website with the Java Servlet, the integration process of the dynamic state part and the static state page is an evil-foreboding dream simply. For solving this kind of weakness of the Java Servlet, the SUN released the JSP.

A number of years ago, Marty was invited to attend a small 20-person industry roundtable discussion on software technology. Sitting in the seat next to Marty was James Gosling, inventor of the Java programming language. Sitting several seats away was a high-level manager from a very large software company in Redmond, Washington. During the discussion, the moderator brought up the subject of Jini, which at that time was a new Java technology. The moderator asked the manager what he thought of it, and the manager responded that it was too early to tell, but that it seemed to be an excellent idea. He went on to say that they would keep an eye on it, and if it seemed to be catching on, they would follow his company's usual "embrace and extend" strategy. At this point, Gosling lightheartedly interjected "You mean disgrace and distend."

Now, the grievance that Gosling was airing was that he felt that this company would take technology from other companies and suborn it for their own purposes. But guess what? The shoe is on the other foot here. The Java community did not invent the idea of designing pages as a mixture of static HTML and dynamic code marked with special tags. For example, Cold Fusion did it years earlier. Even ASP (a product from the very software company of the aforementioned manager) popularized this approach before JSP came along and decided to jump on the bandwagon. In fact, JSP not only adopted the general idea, it even used many of the same special tags as ASP did.

The JSP is an establishment at the model of Java servlets on of the expression layer technique, it makes the plait write the HTML to become more simple.Be like the SSJS, it also allows you carry the static state HTML contents and servers the script mix to put together the born dynamic state exportation. JSP the script language that the Java is the tacit approval, however, be like the ASP and can use other languages( such as JavaScript and VBScript), the norm of JSP also allows to use other languages.

**1.3JSP CHARACTERISTICS**

Is a service according to the script language in some one language of the statures system this kind of discuss, the JSP should be see make is a kind of script language. However, be a kind of script language, the JSP seemed to be too strong again, almost can use all Javas in the JSP.

Be a kind of according to text originally of, take manifestation as the central development technique, the JSP provided all advantages of the Java Servlet, and, when combine with a JavaBeans together, providing a kind of make contents and manifestation that simple way that logic separate. Separate the contents and advantage of logical manifestations is, the personnel who renews the page external appearance need not know the code of Java, and renew the JavaBeans personnel also need not be design the web page of expert in hand, can use to take the page of JavaBeans JSP to define the template of Web, to build up a from have the alike external appearance of the website that page constitute. JavaBeans completes the data to provide, having no code of Java in the template thus, this means that these templates can be written the personnel by a HTML plait to support. Certainly, can also make use of the Java Servlet to control the logic of the website, adjust through the Java Servlet to use the way of the document of JSP to separate website of logic and contents.

Generally speaking, in actual engine of JSP, the page of JSP is the edit and translate type while carry out, not explain the type of. Explain the dynamic state web page development tool of the type, such as ASP, PHP3 etc., because speed etc. reason, have already can't satisfy current the large electronic commerce needs appliedly, traditional development techniques are all at to edit and translate the executive way change, such as the ASP → ASP+;PHP3 → PHP4.

In the JSP norm book, did not request the procedure in the JSP code part( be called the Scriptlet) and must write with the Java definitely. Actually, have some engines of JSP are adoptive other script languages such as the EMAC- Script, etc., but actually this a few script languages also are to set up on the Java, edit and translate for the Servlet to carry out of. Write according to the norm of JSP, have no Scriptlet of relation with Java also is can of, however, mainly lie in the ability and JavaBeans, the Enterprise JavaBeanses because of the JSP strong function to work together, so even is the Scriptlet part not to use the Java, edit and translate of performance code also should is related with Java.

**1.4JSP MECHANISM**

To comprehend the JSP how unite the technical advantage that above various speak of, come to carry out various result easily, the customer must understand the differentiation of" the module develops for the web page of the center" and" the page develops for the web page of the center" first.

The SSJS and ASP are all in several year ago to release, the network of that time is still very young, no one knows to still have in addition to making all business, datas and the expression logic enter the original web page entirely heap what better solve the method. This kind of model that take page as the center studies and gets the very fast development easily. However, along with change of time, the people know that this kind of method is unwell in set up large, the Web that can upgrade applies the procedure. The expression logic write in the script environment was lock in the page, only passing to shear to slice and glue to stick then can drive heavy use. Express the logic to usually mix together with business and the data logics, when this makes be the procedure member to try to change an external appearance that applies the procedure but do not want to break with its llied business logic, apply the procedure of maintenance be like to walk the similar difficulty on the eggshell. In fact in the business enterprise, heavy use the application of the module already through very mature, no one would like to rewrite those logics for their applied procedure.HTML and sketch the designer handed over to the implement work of their design the Web plait the one who write, make they have to double work- Usually is the handicraft plait to write, because have no fit tool and can carry the script and the HTML contents knot to the server to put together. Chien but speech, apply the complexity of the procedure along with the Web to promote continuously, the development method that take page as the center limits sex to become to get up obviously.

At the same time, the people always at look for the better method of build up the Web application procedure, the module spreads in customer's machine/ server the realm. JavaBeans and ActiveX were published the company to expand to apply the procedure developer for Java and Windows to use to come to develop the complicated procedure quickly by" the fast application procedure development"( RAD) tool. These techniques make the expert in the some realm be able to write the module for the perpendicular application plait in the skill area, but the developer can go fetch the usage directly but need not control the expertise of this realm.

Be a kind of take module as the central development terrace, the JSP appeared. It with the JavaBeans and Enterprise JavaBeans( EJB) module includes the model of the business and the data logic for foundation, provide a great deal of label and a script terraces to use to come to show in the HTML page from the contents of JavaBeans creation or send a present in return. Because of the property that regards the module as the center of the JSP, it can drive Java and not the developer of Java uses equally. Not the developer of Java can pass the JSP label( Tags) to use the JavaBeans that the deluxe developer of Java establish. The developer of Java not only can establish and use the JavaBeans, but also can use the language of Java to come to control more accurately in the JSP page according to the expression logic of the first floor JavaBeans.

See now how JSP is handle claim of HTTP. In basic claim model, a claim directly was send to JSP page in. The code of JSP controls to carry on hour of the logic processing and module of JavaBeanses' hand over with each other, and the manifestation result in dynamic state bornly, mixing with the HTML page of the static state HTML code. The Beans can be JavaBeans or module of EJBs. Moreover, the more complicated claim model can see make from is request other JSP pages of the page call sign or Java Servlets.

The engine of JSP wants to chase the code of Java that the label of JSP, code of Java in the JSP page even all converts into the big piece together with the static state HTML contents actually. These codes piece was organized the Java Servlet that customer can not see to go to by the engine of JSP, then the Servlet edits and translate them automatically byte code of Java.

Thus, the visitant that is the website requests a JSP page, under the condition of it is not knowing, an already born, the Servlet actual full general that prepared to edit and translate completes all works, very concealment but again and efficiently. The Servlet is to edit and translate of, so the code of JSP in the web page does not need when the every time requests that page is explain. The engine of JSP need to be edit and translate after Servlet the code end is modify only once, then this Servlet that editted and translate can be carry out. The in view of the fact JSP engine auto is born to edit and translate the Servlet also, need not procedure member begins to edit and translate the code, so the JSP can bring vivid sex that function and fast developments need that you are efficiently.

Compared with the traditional CGI, the JSP has the equal advantage. First, on the speed, the traditional procedure of CGI needs to use the standard importation of the system to output the equipments to carry out the dynamic state web page born, but the JSP is direct is mutually the connection with server. And say for the CGI, each interview needs to add to add a progress to handle, the progress build up and destroy by burning constantly and will be a not small burden for calculator of be the server of Web. The next in order, the JSP is specialized to develop but design for the Web of, its purpose is for building up according to the Web applied procedure, included the norm and the tool of a the whole set. Use the technique of JSP can combine a lot of JSP pages to become a Web application procedure very expediently.

附录B 外文翻译—译文部分

**JSP的技术发展历史**

**作者：** Kathy Sierra and Bert Bates

**来源：** Servlet&JSP

Java Server Pages(JSP)是一种基于web的脚本编程技术，类似于网景公司的服务器端Java脚本语言—— server-side JavaScript(SSJS)和微软的Active Server Pages(ASP)。与SSJS和ASP相比，JSP具有更好的可扩展性，并且它不专属于任何一家厂商或某一特定的Web服务器。尽管JSP规范是由Sun公司制定的，但任何厂商都可以在自己的系统上实现JSP。

在Sun正式发布JSP(Java Server Pages)之后，这种新的Web应用开发技术很快引起了人们的关注。JSP为创建高度动态的Web应用提供了一个独特的开发环境。按照Sun的说法，JSP能够适应市场上包括Apache WebServer、IIS4.0在内的85%的服务器产品。

本文将介绍JSP相关的知识，以及JavaBean的相关内容，当然都是比较粗略的介绍其中的基本内容，仅仅起到抛砖引玉的作用，如果读者需要更详细的信息，请参考相应的JSP的书籍。

**1.1 概述**

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，其在动态网页的建设中有其强大而特别的功能。JSP与Microsoft的ASP技术非常相似。两者都提供在HTML代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。下面我们简单的对它进行介绍。

JSP页面最终会转换成servlet。因而，从根本上，JSP页面能够执行的任何任务都可以用servlet来完成。然而，这种底层的等同性并不意味着servlet和JSP页面对于所有的情况都等同适用。问题不在于技术的能力，而是二者在便利性、生产率和可维护性上的不同。毕竟，在特定平台上能够用Java编程语言完成的事情，同样可以用汇编语言来完成，但是选择哪种语言依旧十分重要。

和单独使用servlet相比，JSP提供下述好处：

JSP中HTML的编写与维护更为简单。JSP中可以使用常规的HTML：没有额外的反斜杠，没有额外的双引号，也没有暗含的Java语法。

能够使用标准的网站开发工具。即使是那些对JSP一无所知的HTML工具，我们也可以使用，因为它们会忽略JSP标签（JSP tags）。

可以对开发团队进行划分。Java程序员可以致力于动态代码。Web开发人员可以将经理集中在表示层（presentation layer）上。对于大型的项目，这种划分极为重要。依据开发团队的大小，及项目的复杂程度，可以对静态HTML和动态内容进行弱分离（weaker separation）和强分离（stronger separation）。

此处的讨论并不是说人们应该放弃使用servlet而仅仅使用JSP。事实上，几乎所有的项目都会同时用到这两种技术。在某些项目中，更适宜选用servlet，而针对项目中的某些请求，我们可能会在MVC构架下组合使用这两项技术。我们总是希望用适当的工具完成相对应的工作，仅仅是servlet并不一定能够胜任所有工作。

**1.2 JSP的由来**

Sun公司的JSP技术，使Web页面开发人员可以使用HTML或者XML标识来设计和格式化最终页面。使用JSP标识或者小脚本来生成页面上的动态内容(内容是根据请求来变化的)。

Java Servlet是JSP技术的基础，而且大型的Web应用程序的开发需要Java Servlet和JSP配合才能完成，Servlet这个名称源于Applet，现在国内的翻译方式很多，本书为了避免误会，决定直接采用Servlet而不做任何翻译，读者如果愿意，可以称之为“小服务程序”。Servlet其实和传统的CGI、ISAPI、NSAPI等Web程序开发工具的作用是相似的，在使用Java Servlet以后，用户不必再使用效率低下的CGI方式，也不必使用只能在某个固定Web服务器平台运行的API方式来动态生成Web页面。许多Web服务器都支持Servlet，即使不直接支持Servlet的Web服务器也可以通过附加的应用服务器和模块来支持Servlet。得益于Java的跨平台的特性，Servlet也是平台无关的，实际上，只要符合Java Servlet规范，Servlet是完全与平台无关且是与Web服务器无关的。由于Java Servlet内部是以线程方式提供服务，不必对于每个请求都启动一个进程，并且利用多线程机制可以同时为多个请求服务，因此Java Servlet效率非常高。

但Java Servlet也不是没有缺点，和传统的CGI、ISAPI、NSAPI方式相同，Java Servlet是利用输出HTML语句来实现动态网页的，如果用Java Servlet来开发整个网站，动态部分和静态页面的整合过程会非常难以实现。为了解决Java Servlet的这种缺点，SUN推出了JSP。

许多年前，Marty受到邀请,参加一个有关软件技术的小型研讨会.坐在Marty旁边的人是James Gosling--- Java编程语言的发明者。隔几个位置,是来自华盛顿一家大型软件公司的高级经理。在讨论过程中,研讨会的主席提出了Jini的议题,这在当时是一项新的Java技术。主席向该经理询问他的想法.他回答说,虽然现在言之过早，但这看起来会是非常有前途的一项技术。他们会持续关注这项技术,如果这项技术变得流行起来,他们会遵循公司的“接受并扩充(embrace and extend)”的策略.此时, Gosling随意地插话说“你的意思其实就是不接受且不扩充(disgrace and distend)。”

在此, Gosling的抱怨显示出，他感到这个公司会从其他公司那里拿走技术,用于他们自己的目的.出人意料的是，形势已经完全不同。Java团队并没有发明这一思想----将页面设计成由静态HTML和用特殊标签标记的动态代码混合组成.。ColdFusion多年前就已经这样做了。甚至ASP(来自于前述经理所在公司的一项产品)都在JSP出现之前推广了这种方式。实际上,JSP不只采用了这种通用概念,它甚至使用许多和ASP相同的特殊标签。

JSP是建立在Java servlets模型之上的表达层技术，它使编写HTML变得更简单。像SSJS一样，它也允许你将静态HTML内容与服务器端脚本混合起来生成动态输出。JSP把Java作为默认的脚本语言，然而，就像ASP可以使用其他语言(如JavaScript和VBScript)一样，JSP规范也允许使用其他语言。

**1.3 JSP的特点**

按照脚本语言是服务于某一个子系统的语言这种论述，JSP应当被看作是一种脚本语言。然而，作为一种脚本语言，JSP又显得过于强大了，在JSP中几乎可以使用全部的Java类。

作为一种基于文本的、以显示为中心的开发技术，JSP提供了Java Servlet的所有好处，并且，当与一个JavaBeans类结合在一起时，JSP提供了一种使内容和显示逻辑分开的简单方式。分开内容和显示逻辑的好处是，更新页面外观的人员不必懂得Java代码，而更新JavaBeans类的人员也不必是设计网页的行家里手，就可以用带JavaBeans类的JSP页面来定义Web模板，以建立一个由具有相似的外观的页面组成的网站。JavaBeans类完成数据提供，这样在模板中就没有Java代码，这意味着这些模板可以由一个HTML编写人员来维护。当然，也可以利用Java Servlet来控制网站的逻辑，通过Java Servlet调用JSP文件的方式来将网站的逻辑和内容分离。

一般来说，在实际的JSP引擎中，JSP页面在执行时是编译式，而不是解释式的。解释式的动态网页开发工具如ASP、PHP3等由于速度等原因已经满足不了当前大型电子商务应用的需要了，传统的开发技术都在向编译执行的方式改变，如ASP→ASP+；PHP3→PHP4。

在JSP规范书中，并没有明确要求JSP中的程序代码部分(称为Scriptlet)一定要用Java来写。实际上，有一些JSP引擎就是采用的其他脚本语言，如EMAC-Script、WebL等，但实际上这几种脚本语言也是构建在Java上面，编译为Servlet来实现的。按照JSP规范书写，和Java没有任何关系的Scriptlet也是可以的，不过，由于JSP的强大功能主要在于能和JavaBeans、Enterprise JavaBeans共同运转，所以即使是Scriptlet部分不使用Java，编译成的执行代码也应该是与Java相关的。

**1.4 JSP的机制**

要理解JSP怎样联合以上各种所提到的技术的优点，从而轻而易举地实现各种效果，用户必须首先了解“组件为中心的网页开发”和“页面为中心的网页开发”的区别。

SSJS和ASP都是在几年前推出的，那时网络还很年轻，没有人知道除了把所有的商务、数据和表达逻辑统统堆进原始网页中之外还有什么更好的解决方法。这种以页面为中心的模型容易学习并且得到相当快速的发展。然而，随着时间的推移，人们认识到这种方法不适于构建大型的、可升级的Web应用程序。在脚本环境中书写的表达逻辑被锁在页面内，只有通过剪切和粘贴才能被重用。表达逻辑通常和商务及数据逻辑混在一起，这使得当程序员试图改变一个应用程序的外观而不想破坏与之紧密结合的商务逻辑时，应用程序的维护就变得十分艰难。其事实上，企业中可重用组件的应用早已经很成熟，没有人愿意为它们的应用程序重写那些逻辑。HTML和图形设计师把它们的设计的实施工作交给了Web编写者，使他们不得不加倍工作—— 常常是手工编写，因为没有合适的工具可以把服务器端脚本与HTML内容结合起来。简而言之，随着Web应用程序的复杂性不断提升，以页面为中心的开发方式的局限性变得明显起来。

与此同时，人们一直在寻找建立Web应用程序的更好方法，组件在客户机/服务器领域流行起来。JavaBeans和ActiveX被“快速应用程序开发”(RAD)工具发行商推广给Java和Windows应用程序开发者用来快速开发复杂的程序。这些技术使某领域内的专家可以为本领域内的垂直应用编写组件，而开发者可以直接拿来使用而不必掌握这一领域的专门技术。

作为一种以组件为中心的开发平台，JSP出现了。它以JavaBeans和Enterprise JavaBeans(EJB)组件包含商务和数据逻辑的模型为基础，提供大量标签和一个脚本平台用来在HTML页中显示由JavaBeans产生或回送的内容。由于JSP的以组件为中心的性质，它可以被Java和非Java开发者同样使用。非Java开发者可以通过JSP的标签(Tags)来使用高级Java开发者创建的JavaBeans。Java开发者不仅可以创建和使用JavaBeans，还能在JSP页中使用Java语言来更精密地控制基于底层JavaBeans的表达逻辑。

现在来看看JSP是如何处理HTTP请求的。在基本请求模型中，一个请求直接被送到JSP页中。JSP代码控制着进行逻辑处理时与JavaBeans组件的交互，并在动态生成的、混合了静态HTML代码的HTML页中显示结果。Beans 可以是JavaBeans 或EJB组件。另外，更加复杂的请求模型可看作从被请求页呼叫其他JSP页或Java Servlets。

JSP引擎实际上要把JSP标签、JSP页中的Java代码甚至连同静态HTML内容都转换为大块的Java代码。这些代码块被JSP引擎组织到用户看不到的Java Servlet中去，然后Servlet自动把它们编译成Java字节码。这样，当网站的访问者请求一个JSP页时，在它不知道的情况下，一个已经生成的、预编译过的Servlet实际上将完成所有的工作，非常隐蔽而又高效。因为Servlet是编译过的，所以网页中的JSP代码不需要在每次请求该页时被解释一遍。JSP引擎只需在Servlet代码最后被修改后编译一次，然后这个编译过的Servlet就可以被执行了。由于是JSP引擎自动生成并编译Servlet，不用程序员动手编译代码，所以JSP能带给你高效的性能和快速开发所需的灵活性。

和传统的CGI相比较，JSP有相当的优势。首先，在速度上，传统的CGI程序需要使用系统的标准输入输出设备来实现动态网页的生成，而JSP是直接和服务器相关联的。而且对于CGI来说，每一个访问就需要新增加一个进程来处理，进程不断地建立和销毁对于作为Web服务器的计算机将是不小的负担。其次，JSP是专门为Web开发而设计的，其目的是为了建立基于Web的应用程序，其中包含了一整套的规范和工具。使用JSP技术可以很方便地将一大堆JSP页面组合成为一个Web应用程序。

附录C 软件使用说明书

本系统通过指定URL即可进入网站主界面，如下图所示：



管理员登录后即可进入管理员主界面，如下图所示：



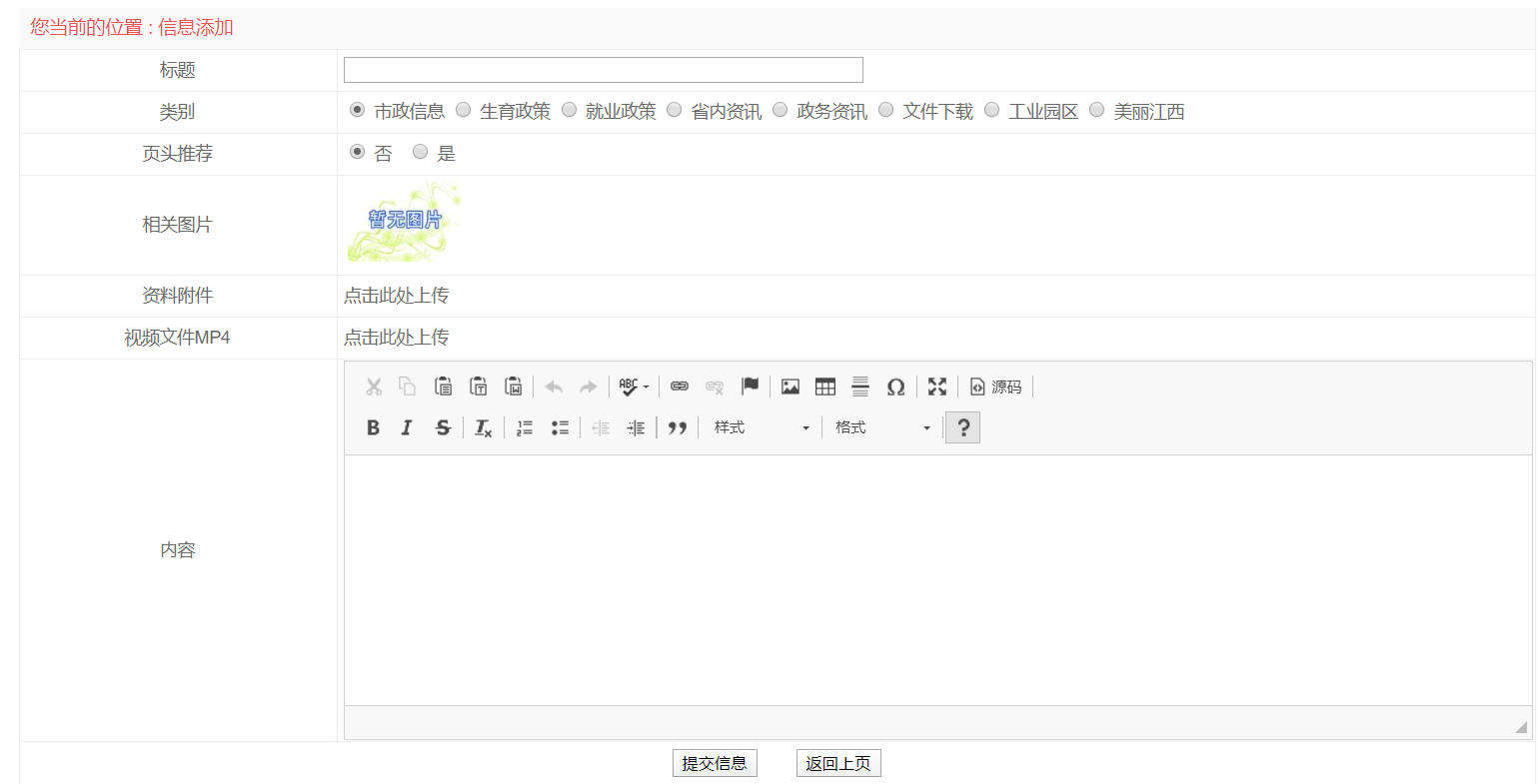
系统游客/用户可选择进行系统登录/注册，在注册成功后登陆即可返回登录成功的用户主界面，如下图所示：



在该页面管理员登陆之后可以发布和编辑不同类别的标题，然后按照标题来显示该版块下的文章，如下图所示：



在该页面管理员登陆之后可以编辑文章信息，如下图所示：



在该页面管理员登陆之后可以审批用户提交的办事信息，如下图所示：



在该页面管理员登陆之后可以审核和删除用户的评论，如下图所示：



在该页面管理员登陆之后可以审核和删除以及回复用户的留言，如下图所示：



附录D 主要源代码

<!-- 数据源配置 -->

<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource"

destroy-method="close">

<property name="driverClassName" value="${jdbc.driver}" />

<property name="url" value="${jdbc.url}" />

<property name="username" value="${jdbc.username}" />

<property name="password" value="${jdbc.password}" />

<property name="maxActive" value="10" />

<property name="maxIdle" value="5" />

</bean>

<!-- 配置 sqlSessionFactory -->

<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">

<property name="dataSource" ref="dataSource"></property>

<!-- 加载 sqlMapConfig.xml -->

<property name="configLocation" value="classpath:sqlMapConfig.xml"></property>

</bean>

<!-- 系统开始 -->

<!-- 配置 AdminDao -->

<bean id="adminDao" class="com.dao.AdminDao">

<property name="sqlSessionFactory" ref="sqlSessionFactory"></property>

</bean>

@RequestMapping("/mimaAdmin")

public void mimaAdmin(HttpServletRequest request, HttpServletResponse response)

throws Exception {

try {

JSONObject result = new JSONObject();

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

String adminPassword1 = (String) request.getParameter("adminPassword1");

Admin admin = new Admin();

admin.setAdminName("admin");

admin.setAdminPassword(adminPassword);

if (adminService.queryAdmins(admin, 0, 0).size()==1) {

admin = (Admin)(adminService.queryAdmins(admin, 0, 0)).get(0);

admin.setAdminPassword(adminPassword1);

adminService.modifyAdmin(admin);

request.setAttribute("error", "密码修改成功！");

request.getRequestDispatcher("adminmima.jsp").forward(request,

response);

}else{

result.put("success", "true");

request.setAttribute("error", "原密码错误，请重新输入！");

request.getRequestDispatcher("adminmima.jsp").forward(request,

response);

}

} catch (Exception e) {

e.printStackTrace();

}

}

@Repository

public class AdminDao extends SqlSessionDaoSupport{

@Autowired

private AdminMapper adminMapper;

public List getAdminList(Admin record,int page,int rows) {

List<Admin> list = adminMapper.selectAll(record,page,rows);

return list;

}

public Admin getAdminById(int id){

Admin admin = adminMapper.selectByPrimaryKey(id);

return admin;

}

public void update(Admin admin) {

adminMapper.updateByPrimaryKey(admin);

}

public void delete(Integer id) {

adminMapper.deleteByPrimaryKey(id);

}

public void add(Admin admin) {

adminMapper.insert(admin);

}

}

public interface AdminMapper {

/\*\*

\* This method was generated by MyBatis Generator. This method corresponds to the database table t\_admin

\* @mbggenerated

\*/

int deleteByPrimaryKey(Integer adminId);

/\*\*

\* This method was generated by MyBatis Generator. This method corresponds to the database table t\_admin

\* @mbggenerated

\*/

int insert(Admin record);

/\*\*

\* This method was generated by MyBatis Generator. This method corresponds to the database table t\_admin

\* @mbggenerated

\*/

Admin selectByPrimaryKey(Integer adminId);

/\*\*

\* This method was generated by MyBatis Generator. This method corresponds to the database table t\_admin

\* @mbggenerated

\*/

List<Admin> selectAll(@Param("admin")Admin record,@Param("page")int page,@Param("rows")int rows);

/\*\*

\* This method was generated by MyBatis Generator. This method corresponds to the database table t\_admin

\* @mbggenerated

\*/

int updateByPrimaryKey(Admin record);

}

@Service

public class AdminService {

@Autowired

private AdminDao adminDao;

public List queryAdmins(Admin record,int page,int rows) {

// TODO Auto-generated method stub

return adminDao.getAdminList(record,page,rows);

}

public Admin getAdmin(int parseInt) {

// TODO Auto-generated method stub

return adminDao.getAdminById(parseInt);

}

public void modifyAdmin(Admin admin) {

// TODO Auto-generated method stub

adminDao.update(admin);

}

public void deleteAdmin(Integer id) {

// TODO Auto-generated method stub

adminDao.delete(id);

}

public void save(Admin admin) {

// TODO Auto-generated method stub

adminDao.add(admin);

}

}

<script type="text/javascript">

function check() {

var adminPassword = document.form1.adminPassword.value;

var adminPassword1 = document.form1.adminPassword1.value;

var adminPassword2 = document.form1.adminPassword2.value;

var reg = /^[a-zA-Z0-9\_]{6,}$/;

if (document.form1.adminPassword1.value == "") {

alert("请输入确认密码");

document.form1.adminPassword1.focus();

return false;

}

if (document.form1.adminPassword1.value != document.form1.adminPassword2.value) {

alert("两次输入密码不一致");

document.form1.adminPassword2.focus();

return false;

}

}

</script>

<body>

<div id="login">

<div class="logo">修改密码</div>

<div class="main">

<form class="formname" action="<%=basePath%>mimaAdmin" name="form1" method="post" onSubmit="return check()">

<ul>

<li>原密码： <input class="wa" type="password" name="adminPassword" id="adminPassword"/></li>

<li>新密码： <input class="wa" type="password" name="adminPassword1" id="adminPassword1"/></li>

<li>再输入： <input class="wa" type="password" name="adminPassword2" id="adminPassword2"/></li>

<li><input type="submit" value="修 改" class="bnt" style="width:48%" />&nbsp;<input type="reset" value="重 置" class="bnt" style="width:48%" /></li>

<li><font color="red">${error }</font></li>

</ul>

</form>

</div>

</div>