

# 《软件开发实践》课程报告

课题名称： RelaxBlog 个人博客系统

课题负责人名（学号）：

董建文(2015141463029)

同组成员名单（角色）：

黄伟(2015141463060, 开发人员)

指导教师： \_\_\_\_\_

评阅成绩： \_\_\_\_\_

评阅意见： \_\_\_\_\_

提交报告时间： 2017 年 6 月 10 日

# RelaxBlog 个人博客系统

软件工程 专业

学生   董建文   黄伟      指导老师   洪玫

**【摘要】** 随着自媒体时代的到来，博客与微博在网络中的影响力日益突出。而微博的篇幅限制和传播方式，导致技术类文章不适合以此形式发布，而博客在这方面则有不可替代的优势。旧版博客外观千篇一律，界面冗杂，不符合简洁清爽的流行风格。因此我们用 Spring MVC 开发了 web 应用 RelaxBlog 以满足那些期望拥有个人博客的用户的需求。最终项目通过验收测试，外观清爽，验错功能完备，可以作为通用型个人博客推广使用。

**关键词：**个人博客系统   RelaxBlog   SpringMVC   web 开发

## 1. 引言（项目背景）

如今自媒体在互联网领域流行开来，娱乐、技术、新闻等各式各样的信息通过博客、微博、直播等平台被广泛传播。微博的流行度虽高，但篇幅限制和实时性的特征使其无法支撑复杂知识、技术的分析和讲解，难以作为专业性文章的发布平台。直播则有着用户主播之间交互时间的统一性要求，且主题单一，不适合自学或检索。

在这种前提下，博客自始至终有着不可替代的地位。而大量的个人博客仍使用着陈旧的模板，缺乏清晰度，也不符合新的审美要求，如简洁化、清爽、动态化等。因此本项目将开发一款满足新界面、新元素要求的博客系统，以提供一个简洁易用、供网友交流沟通的网络平台。

本项目采用 SpringMVC+Hibernate 作为基本架构，采用 Java+JavaScript+Html+sql 作为主要开发语言，使用 IDEA、Mysql 和 chrome 作为主要开发工具，通过实体类和数据库的交互实现了文章、友情链接等的增删查改，通过动态加载页面实现了页面内容的变化。

项目最终通过了单元测试、功能测试和系统验收测试，达到了预期目的，可作为个人用户搭建博客系统的框架使用。

## 2. 相关工作（对现有的系统进行分析）<sup>[1]</sup>

目前国内外有着许多出名的博客系统，支撑着大量用户的访问和管理。

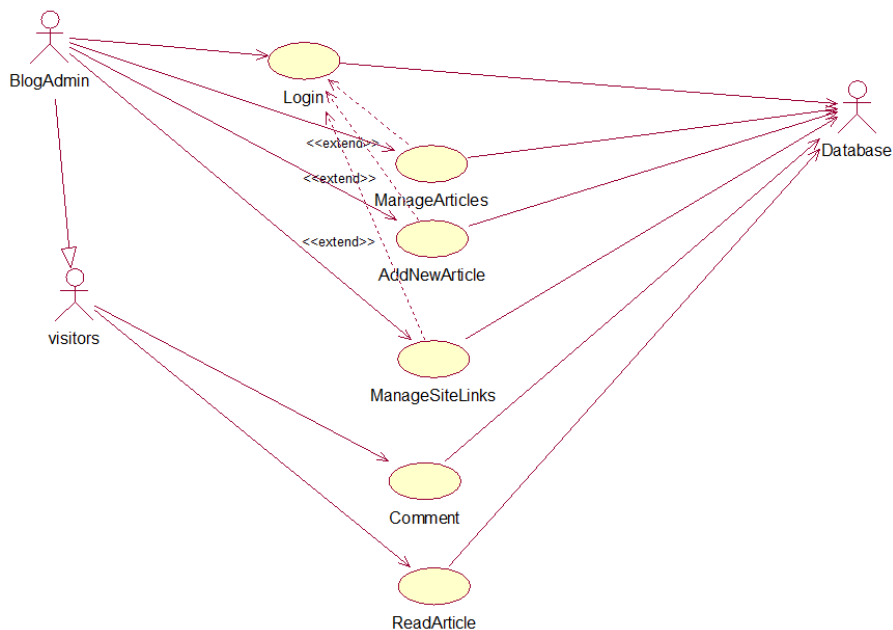
i.多用户大型博客系统，如新浪博客、CSDN，它们功能较全，性能强大，系统的可用性和稳定性相当高，但不符合个人博客的需求。

首先等级系统、首页等元素导致用户难以感受到自己博客主的存在感，而是以普通用户的身份投稿发表文章。其次，界面的内容复杂而拥挤，访问时浏览者会被各种推荐内容、数量众多的排行榜和新闻扰乱视线。

ii.开源的个人博客系统，以 DUZHI\_BLOG 为例,它提供了大量的附加功能,如”历史上的今天”、打赏等，但是文章界面不够友好，用户评论需要登录或注册账号，不符合我们的“自由交流”的初衷。

### 3. 问题描述（需求分析）

用例图：



软件功能：

网站登录	网站登录模块	输入检查管理员账号
		输入检查管理员密码
网站后台页面	后台主页面	显示文章总数量
		显示附件总数量
		显示友情链接数量
		显示最新文章
	发表文章	输入检查文章标题
		输入检查文章标签=文章编辑

		文章预览
		存为草稿
	文章管理	显示文章信息列表
		修改文章
		删除文章
		文章预览
	友情链接管理	添加新连接
		链接编辑
		连接删除
游客访问页面	访问主页	显示文章列表
		显示文章标签
		查看友情链接
		文章搜索
	浏览文章	显示文章内容
		显示最后编辑时间
		添加留言
		显示留言

### 3.1 网站登录模块

#### 输入检查管理员账号：

键盘输入字符串,长度不超过 30 字节,不为空。检查字符串长度,记录是否超长,是否为空,若不超长不空则记录名称。符合格式则不显示输出,不符合格式则确认时报错,显示相应错误

#### 输入检查管理员密码：

键盘输入字符串,长度不超过 10 字节,不为空,不以数字开头。检查字符串长度,记录是否超长,是否为空,若不超长不空则记录账户。符合格式则不显示输出,不符合格式则确认时报错,显示相应错误

### 3.2 后台主页面

#### 登录并访问主页面：

数据库计算总文章数量,显示在页面上, 格式“发表了 X 篇文章”

。

数据库计算友情链接总数量,显示在页面上, 格式“共有 X 个友情链接”。

数据库中文章标题按发表时间排序后,取前五条显示在页面上, 格式：标题-发布时期。

获取服务器时间并显示, 格式 XXXX/XX/XX。

---

### 3.3 发表文章

#### 输入文章标题：

键盘输入字符串,长度不超过 50 字节,不为空。检查字符串长度,记录是否超长,是否为空,若不超长不空则记录标题。符合格式则不显示输出,不符合格式则确认时报错,显示相应错误。

#### 输入文章标签：

键盘输入字符串,长度不超过 10 字节,可为空。检查字符串长度,记录是否超长,是否为空,若不超长不空则记录路径,若为空则默认为无标签。符合格式则不显示输出,不符合格式则确认时报错,显示相应错误。

#### 文章编辑

键盘输入字符串,通过相关编辑按钮修改字体格式等属性。最长不超过 10KB,可为空。检查文章长度,符合长度要求,点上传后记录入数据库。符合要求后上传至数据库,页面提示文章发表成功

#### 图片/表情插入

通过 html 编辑插入图片链接,切换至 html 格式编辑,使用格式引用

#### 文章预览

浏览文章发布后的情况,单击相应按钮,将把当前内容发布后的结果展示在页面上,显示发布后的文章阅读页面情况。

#### 存为草稿

保存当前文章内容,单击相应按钮,将目前文章编辑框内的内容上传至服务器,但在游客的文章界面中不显示。用户登录后可以在文章管理内继续编辑,将编辑框内内容上传至服务器,上传成功则提示“已保存至草稿箱”,因网络等原因上传失败则提示“保存失败”。

### 3.4 文章管理

#### 显示文章信息列表

将文章简要信息显示在页面上,登录并访问“文章管理”页面则自动显示,从数据库中读取文章信息并列表显示。显示表格,列信息为:文章标题-发布时间-标签-发布状态-操作。

#### 修改文章

修改文章正文,点击对应文章的“修改”按钮,点击按钮后进入正文编辑界面,从服务器读取文章内容,编辑过程与文章编辑功能相同,将服务器上的文章信息显示在编辑框中,等待编辑,完成后检查长度等格式,符合要求后更新到服务器上。

#### 删除文章

点击对应文章的“删除”按钮,删除指定的一篇文章。点击按钮后则将服务器端文章信息删除,页面上此文章信息消失。

---

## 文章阅读

阅读指定的一篇文章，点击按钮后从服务器端获取文章信息，以游客浏览页面显示文章内容。

### 3.5 友情链接管理

#### 添加新连接

登录并访问“友情链接管理”，点击“添加”按钮，输入新连接名称、连接地址，均为字符串，名称不超过 20 字节，地址不超过 128 字节，均不能为空将新的链接添加到友情链接栏中，更新到服务器上，更新完成后返回成功信息，接收到成功信息后页面显示“添加成功”。若接收到成功信息，页面显示“添加成功”，否则超时显示“添加失败，请刷新后重试”

#### 链接编辑

登录并访问“友情链接管理”，点击对应的友情链接后的“编辑”按钮，格式同上。输出内容同上。

#### 连接删除

登录并访问“友情链接管理”，点击对应的友情链接后的“删除”按钮。将已有的友情链接信息从服务器端删除，删除成功后返回成功信息，接收到成功信息后页面显示“删除成功”。

### 3.6 访问主页

#### 显示文章列表

访问博客主页，文章列表自动显示。

#### 显示文章摘要

访问博客主页，鼠标指向该文章

#### 显示文章标签

访问博客主页，文章标签自动显示

#### 查看友情链接

访问博客主页,点击"友情链接"按钮，从服务器获取友情链接信息,显示在页面上。

#### 文章搜索

通过关键字搜索文章，访问博客主页,在搜索栏输入内容,点击搜索。

从数据库中搜索相应文章,将搜索结果返回并显示在页面上，显示搜索到的相关文章,若没有搜索结果则显示空白。

### 3.7 浏览文章

#### 显示文章内容

访问博客主页,选定文章鼠标单击查看，从数据库中获取文章正文内容及页面，在浏览器中显示。

### 显示最后编辑时间

查看指定文章时,文章的最后编辑时间自动显示在文章标题下方。从数据库中获取文章最后编辑时间并在页面中显示。

### 添加留言

在指定文章后添加留言，输入邮箱地址，名字和留言内容，点击留言。

### 显示留言

浏览文章时可以看到当前文章的留言，查看指定文章时，若存在留言，则留言自动显示在文章下方。显示留言信息，若无留言则显示空白。

### 系统运行环境：

该系统为 B/S 三层结构，它的运行环境分客户端和数据库服务器端三部分。  
以下是系统的软件环境。

#### （1）客户端

操作系统：Windows8 或更新版本。

浏览器：IE6 以上，其它常见浏览器如 FireFox，Chrome。

#### （2）数据库服务器端

操作系统：Windows2000 Server 或更新版本。

数据库系统：Mysql。

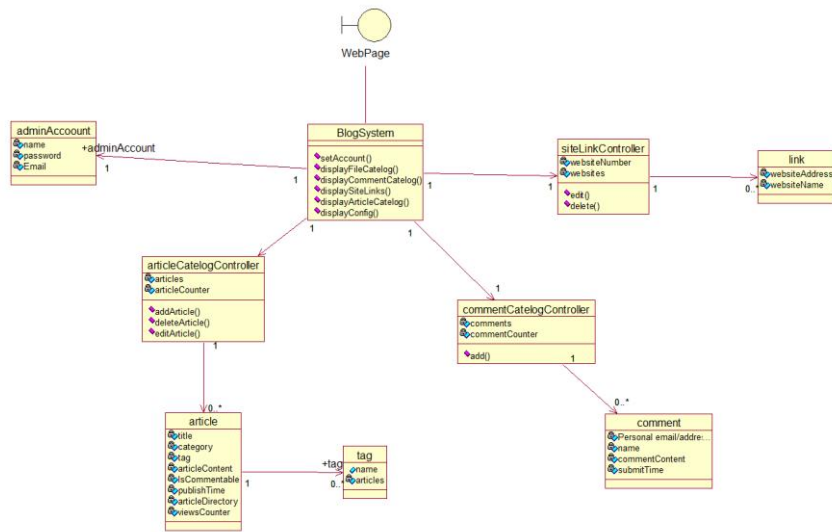
对系统进行需求分析和描述，包括系统的功能及其描述，系统处理的数据及其描述，系统的运行环境和运行方式（行为）描述。

## 4. 解决方案（概要设计）

系统开发时采用了瀑布模型进行开发，从需求、设计、编码到测试、部署。系统是基于 Web 的应用，采用了 B/S 架构。软件开发时采用了 SpringMVC 和 hibernate 框架。

设计类图：


















数据库设计如下：

articles 表

▼ **articles**

	art_id	INT(11) (auto increment)
	title	VARCHAR(255)
	created_date	DATE
	modified_date	DATE
	content	VARCHAR(10000)
	categories	VARCHAR(50)
	comments_num	INT(11)
	status	INT(11)
	PRIMARY	(art_id)

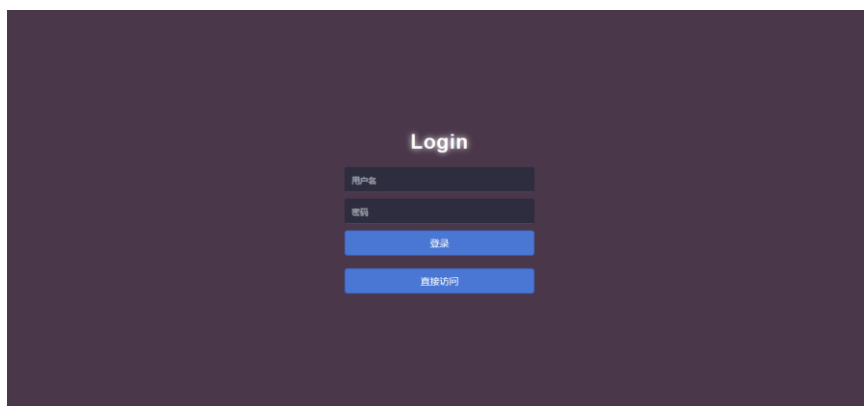
comments 表

▼		comments
		cmt_id INT(11) (auto increment)
		created_date DATE
		art_id INT(11)
		user_id INT(11)
		content VARCHAR(200)
		PRIMARY (cmt_id)
▼		links
		id INT(11) (auto increment)
		url VARCHAR(200)
		name VARCHAR(100)
		description VARCHAR(300)
		PRIMARY (id)

Links 表

**GUI 设计如下：**

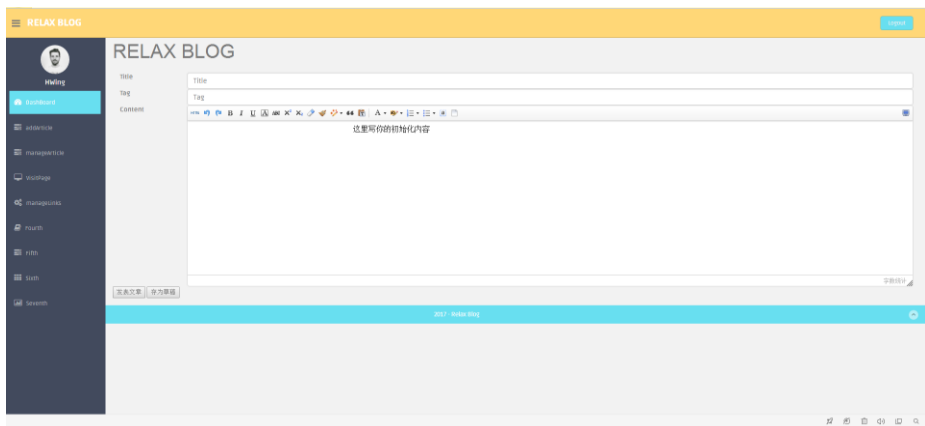
登录界面：



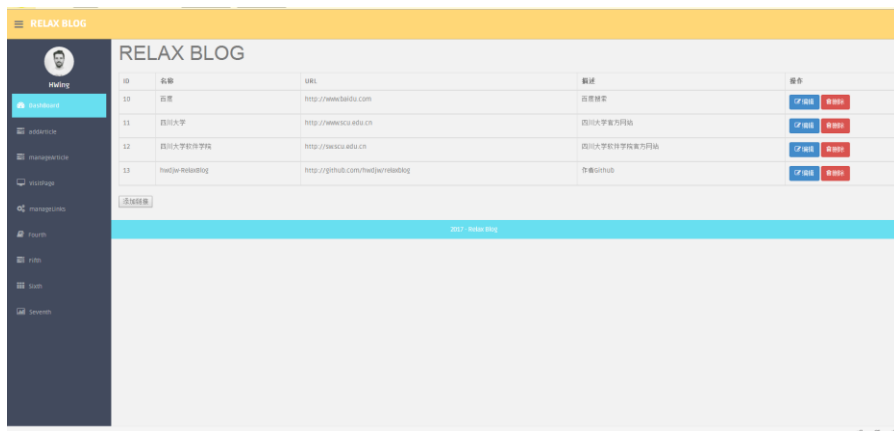
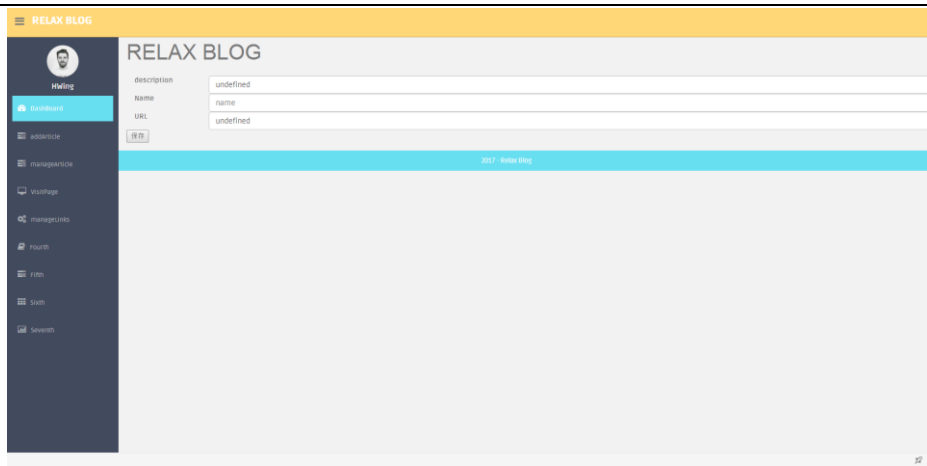
后台管理主页：



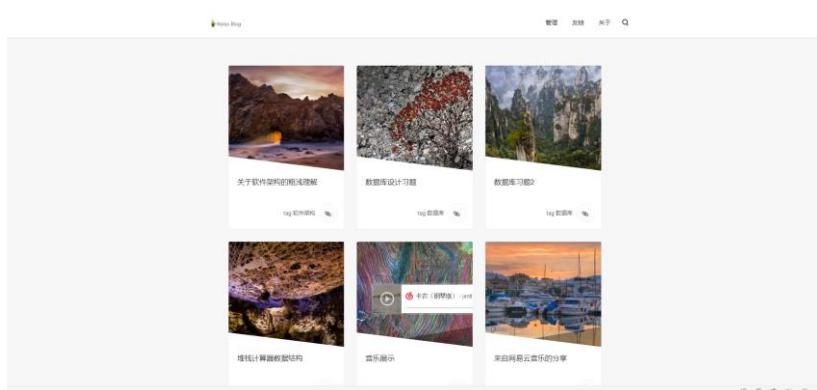
添加、管理文章：



添加、管理链接：



浏览主页：



浏览文章：

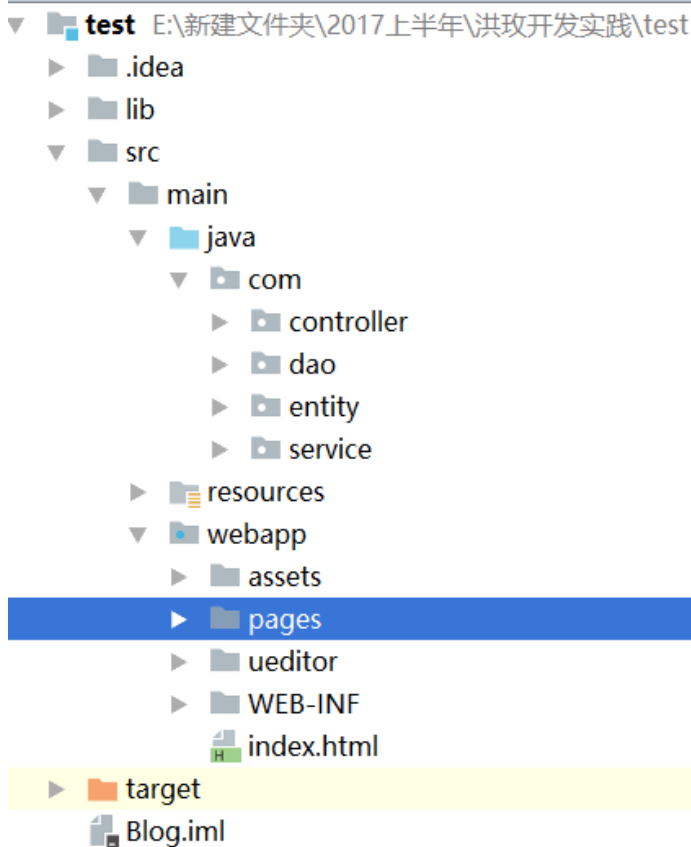


评论：



## 5. 实现细节（详细设计、实现）

框架：SpringMVC 和 hibernate，将前后段相互分离，以 xml 和 java 注解的形式进行映射及配置信息设定。



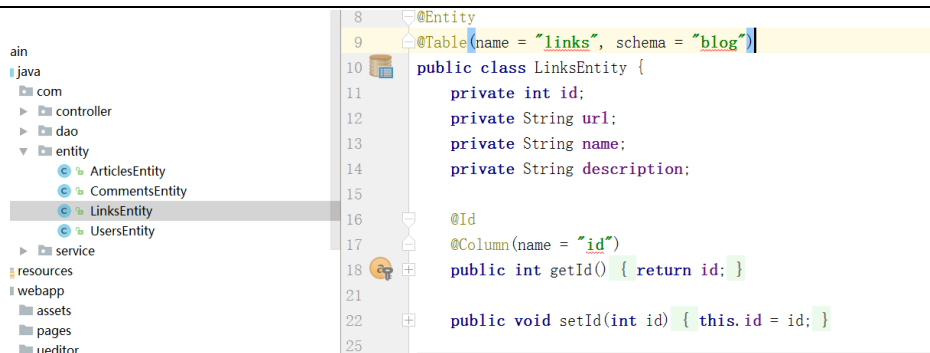
资源文件和代码文件的管理如上图，lib 文件夹内放置需要调用的库文件，java 文件夹内放有 controller（控制类），dao 层（实体信息获取），entity 实体类（对应数据库表项）和 service 类（用于支撑某项系统功能）。

各种页面文件（.html）、页面样式文件（.css）、脚本文件（.js）放在 webapp 下的 pages 和 assets 内，ueditor 放编辑器的实现文件，web-inf 放 web 配置文件。功能实现过程：

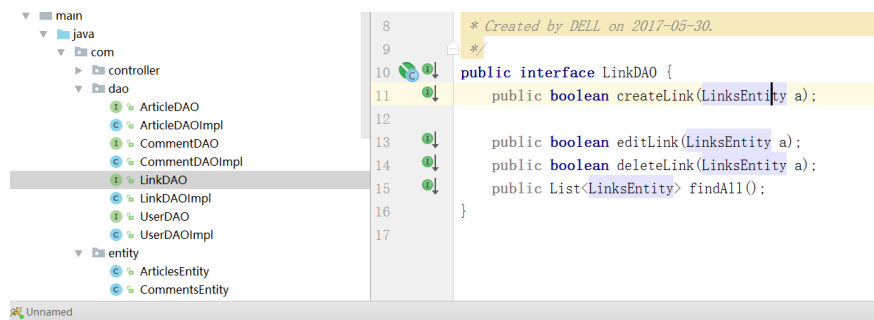
首先，在数据库建立对应的表来存放数据，如管理友情链接，则建立 links 表。代码如下：

```
CREATE TABLE `links` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `url` varchar(200) DEFAULT NULL,  
  `name` varchar(100) DEFAULT NULL,  
  `description` varchar(300) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=14 DEFAULT CHARSET=utf8;
```

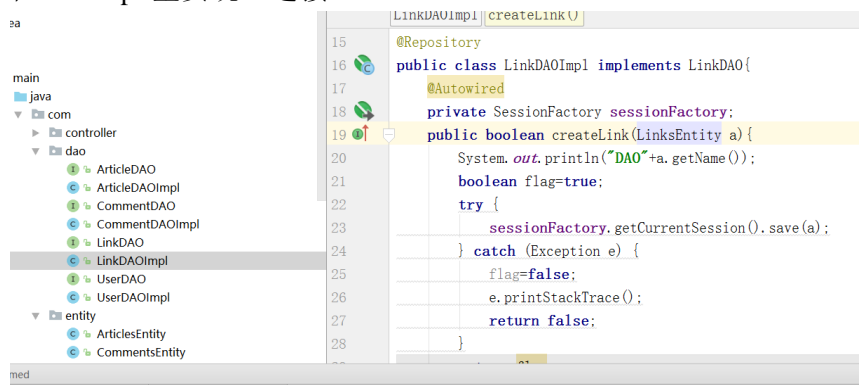
通过 links 表中的列，建立相应的 java 实体类，用于服务器信息的处理和传输。形式如下：



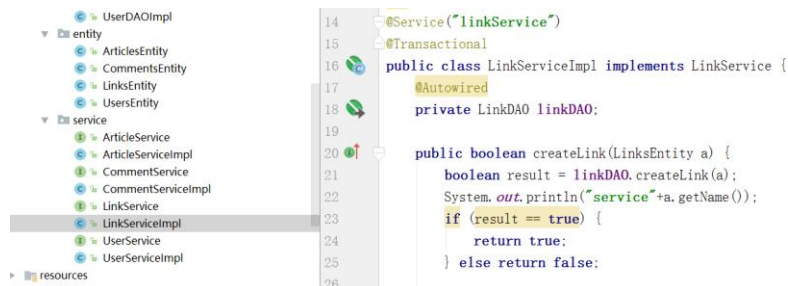
接下来，通过 dao 层建立数据库信息的管理（增删查改的接口）



在 daoimpl 里实现上述接口：



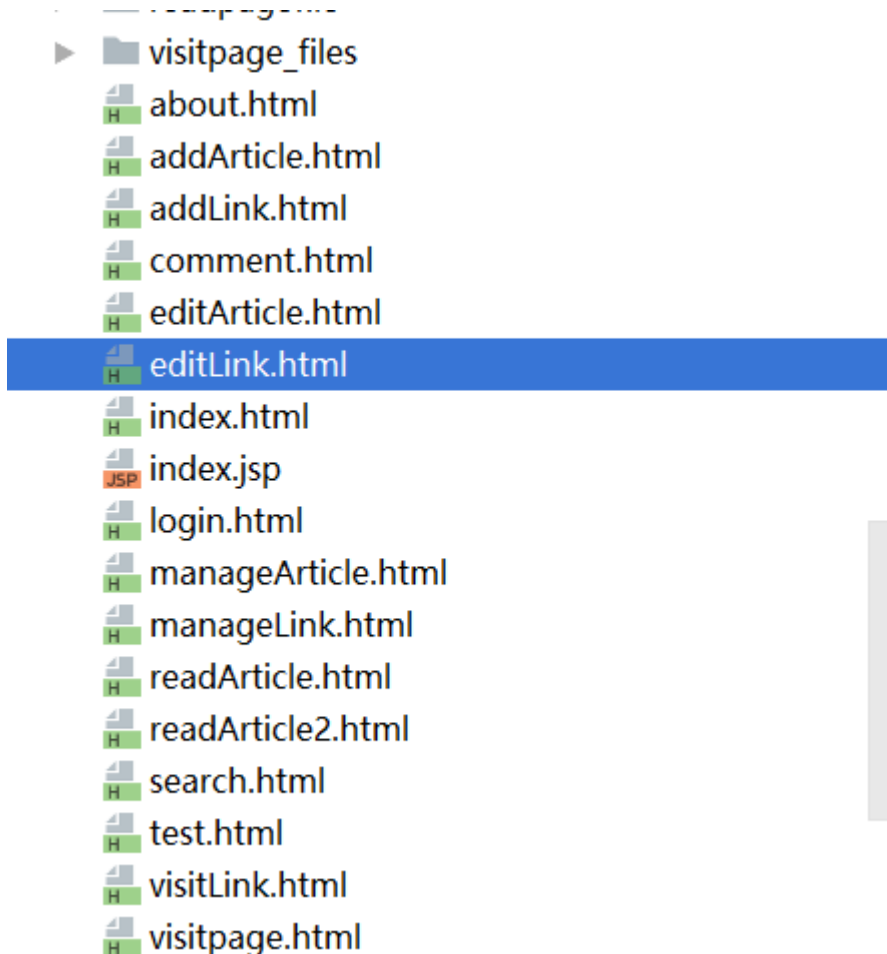
之后，在 service 层建立服务接口并实现



注意到@service 的注解,用于 web 应用中的前端函数调用的映射,前端只能

访问服务层,服务层才能访问 dao 层,而 dao 层拥有修改数据库实体信息的能力。如此分层后,后端功能可以独立实现,而不用将前后端杂糅在一起,提高了内聚性,降低了耦合性。

在实现前端时,需要利用设计阶段制作好的 GUI,仿照实现 html 和 css 文件,达到前段界面的完成。html 包含侧边栏,主内容,右上角按钮栏,搜索栏,下边栏几大元素。而 css 则负责调整颜色、布局以美化界面。



我们为不同的页面制作了不同的页面文件,具体如上。在页面中,功能的调用和展示通过 JavaScript 脚本文件进行动态获取和加载。

在主页获取文章列表时,使用了上述映射中的 all.action,调用获取文章服务,并将每篇文章的信息以 html 格式输出到页面,包装成一个前端“卡片”



```
result += '<div class=\\"post-list-item\\">'+
'<div class=\\"post-list-item-container\\">'+
'<div class=\\"item-thumb bg-deepgrey\\" style=\\"background-image:url(visitp'
'<a href=\\"'+ readArticle2.html?page='+data.artId+' \\"'+ onclick=\\"turnto(d'
'<div class=\\"item-desc\\">'+
'<p class=\\"text-overflow\\" style=\\"width:200px;\\">'+ data.content + '</p>'+
'</div> </a>'+
'<div class=\\"item-slant reverse-slant bg-deepgrey\\"></div>'+
'<div class=\\"item-slant\\"></div>'+
'<div class=\\"item-label\\">'+
'<div class=\\"item-title\\"><a href=\\"'+
'readArticle2.html?page='+
data.artId+' \\">'+
data.title + '</div>'+
'<div class=\\"item-meta clearfix\\">'+
'<div class=\\"item-meta-ico bg-ico-lock\\" style=\\"background: url(visitpag'
'<div class=\\"item-meta-cat\\"><a>'+ tag:' +data.categories+ '</a></div>'+
'</div> </div> </div> </div>'
```

效果如图



同理，文章管理页面也为 html 动态加载，实现代码如下：

```
        status='草稿';
    }

    result += "<tr><td>" + data.title + "</td>" +
        "<td>" + data.artId + "</td>" +
        "<td>" + data.createdDate + "</td>" +
        "<td>" + data.categories + "</td>" +
        "<td>" + status + "</td>" +
        "<td>" +
        "<a href='\"javascript:void(0)\"' onclick='\"editArticle(this)\"' <i class='\"fa fa-edit\"'></i> <span>编辑</span></a>" +
        "<a href='\"javascript:void(0)\"' onclick='\"deleteArticle(this)\"' <i class='\"fa fa-trash-o\"'></i> <span>删除</span></a>" +
        "<a href='\"javascript:void(0)\"' onclick='\"readArticle(this)\"' <i class='\"fa fa-rocket\"'></i> <span>预览</span></a>" +
        "</td>" +
        "</tr>"
    }
    result += "</tbody></table>";
    $('#mc')\html(result)
```

效果如下：

RELAX BLOG					
名称	ID	创建日期	标签	状态	
关于软件架构的浅度理解	43	2017-05-30	软件架构	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
关于软件架构的一个问题	44	2017-05-05	软件架构	草稿	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
数据库设计习题	45	2017-06-05	数据库	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
数据库习题2	46	2017-06-05	数据库	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
堆栈计算机数据机构	47	2017-06-05	数据结构与算法	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
关于计算机网络的思考	50	2017-05-05	计算机网络	草稿	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
音乐展示	52	2017-05-07	音乐	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>
来自网易云音乐的分享	51	2017-06-05	音乐	已发布	<a href="#">编辑</a> <a href="#">删除</a> <a href="#">预览</a>

对于所有的编辑框和提交按钮，我们都完成了格式校验，以保证数据在传输向后端时符合服务器的数据库表格式。除此之外，后台操作的成功与否也会返回给前端调用处，我们在 JavaScript 脚本中也会检验后台传来的信息，如果信息为“fail”，则会弹出提示，提醒用户相应操作失败。而“success”则会提示操作成功。

因此，我们的操作过程就有了双重保证，以增加系统的稳定性。  
检验范例代码如下（编辑文章前端）：

```
tcontent=ue.getContent();
var pageid=getQueryVariable("page");
tcate=$('#article-categories').val();

if(tttitle.length>50|| tttitle.length==0)
{
    alert("标题长度不合法");
    return;
}
if(tcontent.length>10000)
{
    alert("文章过长");
    return;
}
if(tcate.length>10)
{
    alert("标签过长");
    return;
}

function (data)
{
    //var a=data.equals('success')
    if(data=="success")
    {
        alert("保存成功");
        window.location.href="/blog/pages/manageArticle.html"
    }

    else{
        alert("保存失败! 格式不合法");
    }
}

);
```

系统开发过程使用 IDEA 和 mysql 作为开发工具，chrome 作为调试工具，Junit 作为测试工具。开发平台为 windows10 64 位。

## 6. 实验验证（测试、分析、评价）

进行系统测试时，使用的是等价类划分法设计测试用例。测试工具为 Junit 及人工测试，过程包括单元测试、系统功能测试、验收测试、回归测试。<sup>[2]</sup>

测试用例及结果如下表：

用例序号	功能序号	功能	设计输入	预计输出	实际输出	是否通过
1	1	登录功能	空白	提示用户名为空	提示用户名长度错误	通过
2			正确用户名 admin, 正确密码 123456	登录成功	登录成功	通过
3			合理长度用户名 admin, 错误密码 -11111	提示登录信息错误	提示登录信息错误	通过
4			超长长度用户名 a123456789101112	提示用户名长度错误	提示用户名长度错误	通过
5			正确用户名 admin, 超长密码 123456789012345678901	提示密码超长	提示密码长度错误	通过
6			正确用户名 admin, 空密码	提示密码为空, 重新输入	提示密码长度错误	通过
7	2	仪表盘显示文章总数	登录并访问管理主页	显示文章总数量	显示文章总数量	通过
8			未登陆	不显示文章总数量, 跳转登录	不显示文章总数量, 跳转登录	通过
9	3	仪表盘日期	登录并访问管理主页	显示日期	显示日期	通过
10			未登陆	不显示日期, 跳转登录	不显示日期, 跳转登录	通过
11	4	仪表盘显示友情链接总数	登录并访问管理主页	显示友情链接数量	显示友情链接数量	通过
12			未登陆	不显示友情链接数量, 跳转登录	不显示友情链接数量, 跳转登录	通过
13	5	仪表盘显示最新文章	登录并访问管理主页	显示最新文章	显示 5 篇文章概况	通过
14			未登陆	不显示最新文章, 跳转登录	不显示最新文章, 跳转登录	通过
15	6	创建文章	空白标题+合法分类及内容, 发表文章	提示标题长度错误	提示标题长度错误	通过

16			标题“软件架构”+标签“软件架构”+内容“abcd”发表文章	提示发表成功	提示发表成功	通过
17			标题“abcd”+超长标签+内容“abcd”发表文章	提示标签超长	提示标签超长	通过
18			超长标题+标签“1234”+内容“你好”发表文章	提示标题长度错误	提示标题长度错误	通过
19			标题“abcd”+标签“abcd”+超长内容，发表文章	提示内容超长	提示内容超长	通过
20			未登陆	跳转登录界面	跳转登录界面	通过
21			标题“abcd”+标签“abcd”+内容“abcd”，发表文章	提示发表成功	提示发表成功	通过
22			标题“abcd”+标签“abcd”+内容粗体，发表文章	提示发表成功	提示发表成功	通过
23			标题“abcd”+标签“abcd”+内容下划线，发表文章	提示发表成功	提示发表成功	通过
24			复制图片代码到内容，发表文章	提示发表成功	提示发表成功	通过
25			空白标题+合法分类及内容，存为草稿	提示标题长度错误	提示标题长度错误	通过
26			合法标题+标签“软件架构”+内容“abcd”存为草稿	提示发表成功	提示发表成功	通过
27			标题“abcd”+超长分类+内容“abcd”存为草稿	提示标签超长	提示标签超长	通过
28			超长标题+分类“1234”+内容“你好”存为草稿	提示标题长度错误	提示标题长度错误	通过
29			标题“abcd”+标签“abcd”+超长内	提示内容超长	提示内容超长	通过

			容， 存为草稿			
30			未登陆	跳转登录界面	跳转登录界面	未通过
31			标题“abcd”+标签“abcd”+内容“abcd”， 存为草稿	提示发表成功	提示发表成功	通过
32			标题“abcd”+标签“abcd”+内容粗体， 存为草稿	提示发表成功	提示发表成功	通过
33			标题“abcd”+标签“abcd”+内容下划线， 存为草稿	提示发表成功	提示发表成功	通过
34			复制图片代码到内容， 存为草稿	提示发表成功	提示发表成功	通过
35	8	管理页面显示文章信息	已登录， 2 篇文章	正确显示两篇文章信息	正确显示两篇文章信息	通过
36			已登录， 一篇文章一篇草稿	正确显示文章及草稿信息	正确显示文章及草稿信息	通过
37			未登陆	跳转登录界面	跳转登录界面	通过
38	9	修改文章	原 文 章 标 题“abcd”， 标签‘abcd’， 内容‘abcd’， 点击修改此文章	进入编辑页面， 正确显示文章标题、标签、内容并等待修改	进入编辑页面， 正确显示文章标题、标签、内容并等待修改	通过
39			原 文 章 标 题“abcd”， 标签为空， 内容‘abcd’， 点击修改此文章	进入编辑页面， 正确显示文章标题、标签、内容并等待修改	进入编辑页面， 正确显示文章标题、标签、内容并等待修改	通过
40			空白标题+合法分类及内容， 发表文章	提示标题长度错误	提示标题长度错误	通过
41			标题“软件架构”+标签“软件架构”+内容“abcd”发表文章	提示发表成功	提示发表成功	通过
42			标题“abcd”+超长标签+内容“abcd”发表文章	提示标签超长	提示标签超长	通过

43		超长标题+标签 “1234”+内容“你 好”发表文章	提示标题长度错误	提示标题长度错误	通过
44		标题“abcd”+标签 “abcd”+超长内 容，发表文章	提示内容超长	提示内容超长	通过
45		未登陆	跳转登录界面	跳转登录界面	通过
46		标题“abcd”+标签 “abcd”+内容 “abcd”，发表文 章	提示发表成功	提示发表成功	通过
47		标题“abcd”+标签 “abcd”+内容粗 体，发表文章	提示发表成功	提示发表成功	通过
48		标题“abcd”+标签 “abcd”+内容下划 线，发表文章	提示发表成功	提示发表成功	通过
49		复制图片代码到内 容，发表文章	提示发表成功	提示发表成功	通过
50		空白标题+合法分 类及内容，存为草 稿	提示标题长度错误	提示标题长度错误	通过
51		合法标题+标签“软 件架构”+内容 “abcd”存为草稿	提示发表成功	提示发表成功	通过
52		标题“abcd”+超长 分类+内容“abcd” 存为草稿	提示标签超长	提示标签超长	通过
53		超长标题+分类 “1234”+内容“你 好”存为草稿	提示标题长度错误	提示标题长度错误	通过
54		标题“abcd”+标签 “abcd”+超长内 容，存为草稿	提示内容超长	提示内容超长	通过
55		未登陆	跳转登录界面	跳转登录界面	通过
56		标题“abcd”+标签 “abcd”+内容 “abcd”，存为草 稿	提示发表成功	提示发表成功	通过

57			标题“abcd”+标签“abcd”+内容粗体，存为草稿	提示发表成功	提示发表成功	通过
58			标题“abcd”+标签“abcd”+内容下划线，存为草稿	提示发表成功	提示发表成功	通过
59			复制图片代码到内容，存为草稿	提示发表成功	提示发表成功	通过
60	10	删除文章	登录，删除第一篇文章	提示删除成功，更新页面	提示删除成功，更新页面	通过
61			登录，删除第三篇文章	提示删除成功，更新页面	提示删除成功，更新页面	通过
62			登录，删除第一草稿	提示删除成功，更新页面	提示删除成功，更新页面	通过
63			登录，删除第三草稿	提示删除成功，更新页面	提示删除成功，更新页面	通过
64			未登陆	跳转登录界面	跳转登录界面	通过
65	11	文章预览	登录，预览第一篇文章	打开预览页面，正确显示内容	打开预览页面，正确显示内容	通过
66			登录，删除第三篇文章	打开预览页面，正确显示内容	打开预览页面，正确显示内容	通过
67			登录，预览第一草稿	打开预览页面，正确显示内容	打开预览页面，正确显示内容	通过
68			登录，预览第三草稿	打开预览页面，正确显示内容	打开预览页面，正确显示内容	通过
69			未登陆	跳转登录界面	跳转登录界面	通过
70	12	添加链接	Name 为空	提示名称长度错误	提示名称长度错误	通过
71			Name= 'github' , URL 为空	提示 URL 长度错误	提示 URL 长度错误	通过
72			Name= 'github' , URL= 'github.com' ,描述为空	提交成功	未提示	通过
73			都为空	报告任意长度错误	报告任意长度错误	通过
74			Name 超长	提示名称超长	提示名称超长	通过
75			Name= 'github' , URL 超长,描述为空	提示 URL 超长	提示 URL 超长	通过



76			Name= 'github' , URL= 'github.com' ,描述超长	提示描述超长	提示描述超长	通过
77			未登陆	提示登录信息错误	未提示	通过
78			原 Name='github', URL='github.com', 描述为空, 进入编辑	正确显示原内容	未显示	通过
79			Name 为空	提示名称长度错误	提示名称长度错误	通过
80			Name= 'github' , URL 为空	提示 URL 长度错误	提示 URL 长度错误	通过
81	13	编辑链接	Name= 'github' , URL= 'github.com' ,描述为空	提交成功	未提示	通过
82			都为空	报告任意长度错误	报告任意长度错误	通过
83			Name 超长	提示名称超长	提示名称超长	通过
84			Name= 'github' , URL 超长,描述为空	提示 URL 超长	提示 URL 超长	通过
85			Name= 'github' , URL= 'github.com' ,描述超长	提示描述超长	提示描述超长	通过
86			未登陆	提示登录信息错误	未提示	通过
87	14	删除链接	删除第一个链接	提示删除成功	提示删除成功, 未更新页面	通过
88			删除第三个连接	提示删除成功	提示删除成功, 未更新页面	通过
89			未登陆	跳转登录界面	未跳转	通过
90	15	浏览主页时显示文章列表	登录	文章列表显示	文章列表显示	通过
91			未登陆	文章列表显示	文章列表显示	通过
92	16	浏览主页, 鼠标指向时显示文章摘要	登 录 , 文 章 为 'abcd'	摘要显示全篇文章	摘要显示全篇文章	通过
93			登录, 文章超过摘要大小	显示局部文章	显示局部文章	通过
94			未 登 录 , 文 章 为	摘要显示全篇文章	摘要显示全篇文章	通过

			‘abcd’		章	
95			未登录，文章超过摘要大小	显示局部文章	显示局部文章	通过
96	17	浏览主页时显示文章标签	未登陆	显示文章标签	显示文章标签	通过
97			登录	显示文章标签	显示文章标签	通过
98	18	浏览文章时正确显示内容	未登陆	显示文章内容	显示文章内容	通过
99			登录	显示文章内容	显示文章内容	通过
100	19	浏览文章时正确显示编辑时间	未登陆	显示文章编辑时间	显示文章编辑时间	通过
101			登录	显示文章编辑时间	显示文章编辑时间	通过
102	20	浏览文章时正确显示标题	未登陆	显示文章标题	显示文章标题	通过
103			登录	显示文章标题	显示文章标题	通过

经过 3 次系统功能测试（包括自测、分组测试和验收测试）及修复，所有测试用例已可顺利通过，满足功能需求的要求，完成 RelaxBlog 所有功能实现。

性能测试（本机环境）：

编辑文章<=100ms

添加文章<=100ms

删除文章<=10ms

浏览文章<=1s

编辑友情链接<=50ms

添加友情链接<=50ms

删除友情链接<=10s

登录<=100ms

页面跳转 $\leq 2s$

总体性能符合预期,无大范围抖动。

## 7. 结束语（结论、进一步的工作）

RelaxBlog 是一款由我们小组全程开发的 web 应用,从基础的架构搭建到功能的实现, 面临了诸多的困难与挑战。我们使用了 SpringMVC 和 hibernate 框架, 使前后端分离, 从可修改性、安全性、稳定性等方面提升了软件的质量属性。<sup>[3]</sup>

但从实用的角度来看, RelaxBlog 的功能还是比较单一。成熟的大型博客系统有着复杂的交互功能和储存功能, 不仅局限于几张数据表。同时这些系统还利用分布式架构提升了软件性能, 成熟的软件工程手段也使它们的错误率降到极低, 而我们的系统显然是远远不够成熟的。

但作为一款个人博客系统, 提供给少量用户浏览, 本系统还是能够胜任各项功能的。前端动态界面和清爽风格使人机交互变得友好, 没有冗杂的功能影响用户的观感, 新元素的融入也使网站紧随潮流, 不再呆板。此外, 简单的爬虫实现的文章爬取的附加功能也为用户提供了极大的便利。

随着软件开发实践项目进入尾声, RelaxBlog 并不会停止进步的脚步。我们将在未来对软件的架构进行优化, 对代码进行重构, 实现诸如大量文章导入、多系统多客户端、安全加密等功能, 提升软件的质量。

量和便捷度，最终使 RelaxBlog 一步步接近于真正的具有可用性的成熟系统。

## 参考文献

- [1] 中华人民共和国国家标准. 计算机软件产品开发文件编制指南. GB/T8567-1988
- [2] Roger S • Pressman. 软件工程——实践者的研究方法. 机械工业出版社, 2011
- [3] Len Bass, Paul Clements, Rick Kazman. 软件构架实践. 清华大学出版社, 2013

课程名称：软件开发实践  
2015141463029/2015141463060

学生姓名：董建文/黄伟      学      生      学      号      :

---

附录：

1. [项目开发计划](#)；
2. [软件需求规格说明书](#)；
3. [软件设计文档](#)；
4. [软件测试文档](#)；
5. [用户手册](#)；

# 附录 1

## 软件开发计划书

**项目名称：RelaxBlog**

**参与人员：董建文 黄伟**

## 目 录

<b>1 引言</b>	<b>33</b>
1.1 编写目的	33
1.2 背景	33
1.3 定义	34
1.4 参考资料	34
1.5 标准、条件和约定	35
<b>2 项目概述</b>	<b>36</b>
2.1 工作内容	37
2.2 主要参加人员	37
2.3 产品及成果	38
2.3.1 程序	38
2.3.2 文件	38
2.3.3 服务	-8-
2.3.4 非移交产品	38
2.4 验收标准	40
2.4.1 代码的验收	40
2.4.2 文档验收	40
2.4.3 服务验收	40
2.5 完成项目的最迟期限	41



2.6 本计划的审查者与批准者 .....	41
3 实施总计划 .....	42
3.1 开发过程 .....	42
3.1.1 需求分析 .....	42
3.1.2 系统设计 .....	42
3.1.3 编码及测试阶段 .....	43
3.1.4 文档、产品部署 .....	43
3.1.5 项目总结 .....	43
3.1.6 工作任务的分解 .....	44
3.2 接口人员 .....	45
3.3 进度 .....	45
3.4 预算 .....	47
3.5 关键问题 .....	47
4 支持条件 .....	48
4.1 计算机系统支持 .....	48
4.2 需要用户承担的工作 .....	50
4.3 需由外单位提供的条件 .....	50
5 专题计划要点 .....	51
5.1 开发人员培训计划 .....	51
5.2 测试计划 .....	51
5.3 质量保证计划 .....	52
5.4 人员配置计划 .....	52

## 1 引言

### 1.1 编写目的

为了保证项目团队按时保质地完成项目目标，便于项目团队成员更好地了解项目情况，使项目工作开展的各个过程合理有序，有必要以文件化的形式，把对于在项目生命周期内的工作任务范围、各项工作的任务分解、项目团队组织结构、各团队成员的工作责任、团队内外沟通协作方式、开发进度、经费预算、项目内外环境条件、风险对策等内容以书面的方式描述出来，作为项目团队成员以及项目干系人之间的共识与约定，项目生命周期内的所有项目活动的行动基础，项目团队开展和检查项目工作的依据。

本项目开发计划用于从总体上指导 **RelaxBlog** 项目顺利进行并最终得到通过评审的项目产品。本项目开发计划面向项目组全体成员。

### 1.2 背景

**RelaxBlog** 是我们小组开发的一个新型的博客，主要功能是实现一个由个人管理的，能够提供发布博客，更新博客，播放（云）音乐等功能，结合了文字、图像、其他博客或网站的链接及其它与主题相关的媒体，能够让读者以互动的方式留下意见为用户，旨在开发出简洁易用，界面清爽的个人博客系统，提供一个深度交流沟通的网络平台。项目周期为三个月。

### 1.3 定义

专门术语：

**Mysql:**系统服务器所使用的数据库关系系统（DBMS）。

**SQL:** 一种用于访问查询数据库的语言。

**Github:** 一个用于项目管理开发的网站。

**JAVA:** 一种面向对象编程语言。

**Dreamweaver:** 一种用于网页制作的软件。

**Microsoft Visio:** 微软的可视化图表工具。

**Rational Rose:** 用于 UML 设计的软件

**Microsoft Project:** 微软的项目管理软件

**Maven:** 一款高效的管理代码和资源的软件。

缩写：

**SQL: Structured Query Language**(结构化查询语言)。

**UML:** 统一建模语言、是一套用来设计软件蓝图的标准建模语言，是一种从软件分析、设计到编写程序规范的标准建模语言。

**GB:** 国家标准。

### 1.4 参考资料

文档格式要求按照我国 **GB/T8567-1988** 国家标准和 **IEEE/ANSI830-1993** 标准规范要求进行。包括以下文件：

- 软件工程项目开发文档范例
- 软件工程国家标准文档

书籍包括：

- 《软件项目管理》 朱少民，韩莹 编著，人民邮电出版社。
- 《软件项目管理》 **Rajeev T Shandilya** 编著 科学出版社。
- 《软件工程——实践者的研究方法》 **Roger S Pressman** 著

## 1.5 标准、条件和约定

本项目遵从以下标准：

**GB/T 13702-1992** 计算机软件分类与代码

**GB/T 20918-2007** 信息技术

**GB/T 19003-2008** 软件工程

**GB/T 8567-2006** 计算机软件文档编制规范

## 2 项目概述

随着人们知识水平层次的提高，博客成为日常生活中不可缺少的一部分。它可以用来：1、个人自由表达和出版； 2、知识过滤与积累； 3、深度交流沟通的网络新方式。博客，不等同于私人日记，博客的概念肯定要比日记大很多，它不仅仅要记录关于自己的点点滴滴，还注重它提供的内容能帮助到别人，也能让更多人知道和了解。

博客平台采用目前比较流行的 B/S 模式，**eclipse** 和 **intellij IDEA** 结合作为前端开发工具，后台数据库管理系统为 **Mysql**，对数据库访问的并发处理由 **Mysql** 控制。系统的运行平台为 **Windows**。本博客系统主要实现的功能如下：

安装系统需要用户名和密码才能进入系统。

用户可以根据需要进行博客文章的发布、管理。

用户可以设置页面界面和各种连接地址。

读者可以进行文章的评论、博客留言等操作。

平台支持流行表情和（云）音乐播放功能。

项目开发时间:90 天

开发工具: **Eclipse, Rational Rose, Microsoft Project, Maven intellij IDEA ,Mysql** 等

## 2.1 工作内容

- 1、 制作和修订项目开发计划；
- 2、 进行计划跟踪与监控；
- 3、 配合 SQA 的质量保证工作；
- 4、 工作产品及时进行受控管理；
- 5、 按计划提请阶段评审；
- 6、 提交测试部门评测开发产品；
- 7、 交付最终工作产品。
- 8、 项目实施总结
- 9、 项目验收

## 2.2 主要参加人员

团队总计两人，一名组长和一名队员，分工合作完成项目的开发工作

表 2.2 人员分工

姓名	角色	工作描述
董建文	组长	负责计划、分配任务、决策的实施，项目开发和文档撰写
黄伟	组员	负责项目的开发工作和文档撰写

## 2.3 产品及成果

### 2.3.1 程序

软件名称：**RelaxBlog 个人博客系统**

编程语言：**Java**

发布方式：**Github**

功能：提供博客文章的撰写、修改、删除，友情链接管理等功能，提供（云）音乐的播放，支持文章的爬取，评论，引用等。

### 2.3.2 文件

- 用户操作手册：本手册详细描述软件的功能、性能和用户界面，使用户对如何使用该软件得到具体的了解，为操作人员提供该软件各种运行情况的有关知识，特别是操作方法的具体细节。
- 软件维护手册：主要包括软件系统说明、程序模块说明、操作环境、支持软件的说明、维护过程的说明，便于软件的维护。

### 2.3.3 服务

- 提供用户使用系统的简单培训，讲解软件系统总体和各程序模块，使用方法，以及一些基本的问题解决方法

### 2.3.4 非移交产品

- 可行性分析报告：说明该软件开发项目的实现在技术上、经济上和社会因素上的可行性，评述为了合理地达到开发目标可供选择的各种可能实施方案，说明并论证所选定实施方案的理由。
- 项目开发计划：为软件项目实施方案制订出具体计划，应该包括各部分工作的负责人员、开发的进度、开发经费的预算、所需的硬件及软件资源等。
- 软件需求说明书（软件规格说明书）：对所开发软件的功能、性能、用户界面及运行环境等做出详细的说明。它是在用户与开发人员双方对软件需求取得共同理解并达成协议的条件下编写的，也是实施开发工作的基础。该说明书应给出数据逻辑和数据采集的各项要求，为生成和维护系统数据文件做好准备。
- 概要设计说明书：该说明书是概要实际阶段的工作成果，它应说明功能分配、模块划分、程序的总体结构、输入输出以及接口设计、运行设计、数据结构设计和出错处理设计等，为详细设计提供基础。
- 详细设计说明书：着重描述每一模块是怎样实现的，包括实现算法、逻辑流程等。
- 测试计划：为做好集成测试和验收测试，需为如何组织测试制订实施计划。计划应包括测试的内容、进度、条件、人员、测试用例的选取原则、测试结果允许的偏差范围等。
- 测试分析报告：测试工作完成以后，应提交测试计划执行情况的说明，对测试结果加以分析，并提出测试



的结论意见。

- 软件问题报告：指出软件问题的登记情况，如日期、发现人、状态、问题所属模块等，为软件修改提供准备文档。
- 软件修改报告：软件产品投入运行以后，发现了需对其进行修正、更改等问题，应将存在的问题、修改的考虑以及修改的影响做出详细的描述，提交审批。
- 源程序：软件开发过程中的全部代码以及注释。

## 2.4 验收标准

### 2.4.1 代码的验收

最后在交付之前进行小组内评审，与文档说明保持一致，代码书写风格统一，采用标准规范，没有下列错误：由于软件缺陷造成丢失数据，不符合设计要求，响应时间太长无法接受等问题。

### 2.4.2 文档验收

最后在交付之前进行小组内评审，文档格式符合预期，功能符合于设计要求，清晰易读，没有语病与歧义。

### 2.4.3 服务验收

服务软硬件达到文档说明的要求，完成项目报告，在项目结束后完成项目展示和答辩。

## 2.5 完成项目的最迟期限

### 交付日期

从**2017年3月8日**开始至**2017年5月31日**，完成对整个系统的可行性报告分析、需求分析说明书、开发计划说明说、系统设计书、项目测试、项目总结，对概念模型、存储模式、完整性控制、存取权限等进行了定义，对系统功能各模块进行了详细设计，定义了数据库总体结构、编码命名规范，并交付用户。交付日期为**2017年5月31日**，延期交付日为**6月14号**。

## 2.6 本计划的审查者与批准者

本系统由小组提出，自**2017年3月8号**正式批准，审查者为洪玫老师，批准人为洪玫老师。

## 3 实施总计划

### 3.1 开发过程

#### 3.1.1 需求分析

需求分析是整个设计中重要的一环，当可行性分析完成，项目立项，确定开发角色后，从3月15日开始至4月5日，有关的设计开发人员与管理人员共同对项目流程、管理方式进行分析，并进行资料的收集、整理。在完成了对有关数据信息的收集、归纳和分析整理后，确定了用户需求，对软件必须完成的功能进行了定义，在此基础上完成了数据定义，建立需求模型。

#### 3.1.2 系统设计

从4月6日开始至4月24日，完成对整个系统的分析设计，对概念模型、存储模式、完整性控制、网络传输等进行了定义，对系统功能各模块进行了详细设计、定义了数据库总体结构、编码命名规范。

### 3.1.3 编码阶段

从 4 月 25 日开始至 5 月 10 日，完成程序编码、数据库建立及网页的编制调试。为了避免错误积累，在编码的过程中，融入小规模测试和 BUG 修复。

### 3.1.4 测试阶段

从 5 月 11 日开始至 5 月 31 日，对每个模块都安排组员进行单独测试，系统联调及系统测试，对系统处理逻辑、异常处理能力、性能等进行详细地测试，对发现的问题进行彻底纠正。

### 3.1.5 项目总结

从 6 月 1 日开始至 6 月 14 日，对项目研发、部署等开发过程进行总结，准备项目成果的展示，完成最终项目报告和答辩。

## 3.1.6 工作任务的分解

表 3.1 工作情况

工作内容	负责人	参加人员
软件项目开发计划书	董建文	全体人员
项目开发进度报告（报告）	董建文	全体人员
需求分析（需求分析规格说明书）	黄伟	全体人员
系统分析（概要设计及说明书）	董建文	全体人员
详细设计（详细设计说明书）	董建文	全体人员
数据库建立	黄伟	黄伟
界面设计	董建文	董建文
测试计划（报告）	董建文	董建文
测试报告（报告）	黄伟	黄伟
项目开发总结报告（报告）	董建文	全体人员
用户操作手册（报告）	黄伟	黄伟
绘制数据流图、E-R 图等各种图表	黄伟	全体人员
软件安装、调试	董建文	全体人员

后期维护	黄伟	全体人员
------	----	------

### 3.2 接口人员

负责本项目同用户的接口人员为黄伟，负责用户操作手册的编写，以及软件运行并给用户演示，在后期中负责维护和更新。

### 3.3 进度

方法：采用迭代式开发

设置模块开发优先级：通过对博客系统的特点和信息流程等的分析，确定各模块开发的先后次序。

具体开发进度安排如下：

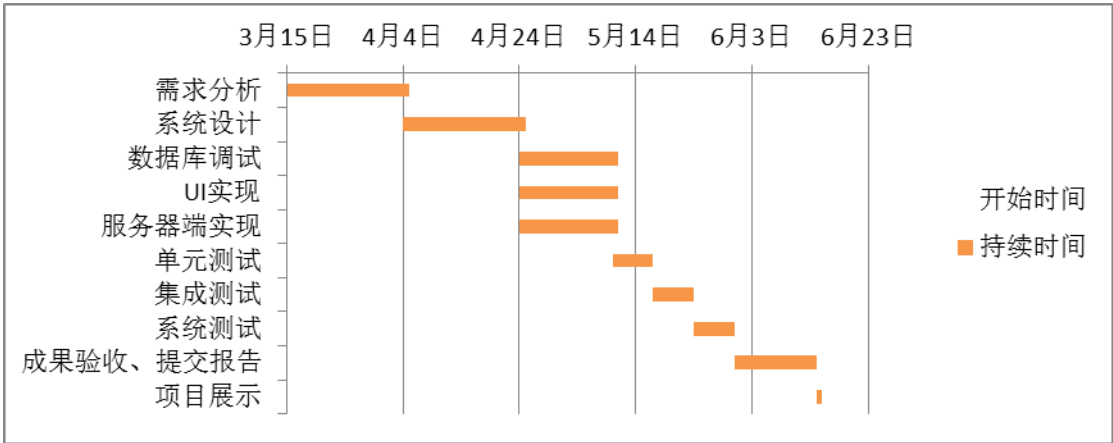
注：有四个里程碑，分别是需求完成时、详细设计完成时、系统编码完成时、整个项目工作完成时。

表 3.3 里程碑事件

里程碑名称	产品名称	提交日期	责任人
需求分析结束	需求说明书	2017.04.05	黄伟
系统设计结束	概要设计说明书	2017.04.24	董建文
系统编码完成	可运行系统	2017.05.10	黄伟
测试完成	测试报告	2017.05.31	黄伟

里程碑名称	产品名称	提交日期	责任人
成果展示	项目报告	2017.06.14	董建文

图 3.3 时间安排甘特图



3.4 预算

由于项目为课程实践项目，暂无预算。

3.5 关键问题

表 3.5 项目风险因素

风险排序	风险项名称	风险描述	风险缓解方案
1	专业知识和技术尚有欠缺	本次项目开发过程中涉及的知识较多，开发团队掌握的相关技术和知识背景有所欠缺	进行相应的学习和实践



风险排序	风险项名称	风险描述	风险缓解方案
2	经验欠缺	成员开发经验不足，可能影响项目进度和软件质量	不断实践，加强测试，多抽出开发时间
3	对项目管理的理解尚浅	本次开发的小组成员对项目管理的细节理解不深	严格按照老师及相关工程要求进行文档的编撰和记录。

影响本计划完成的主要问题有：

- 没有经费和硬件设施有限
- 用户需求不清，存在误解及二义性
- 第一次进行完整开发流程，开发人员没有相应经验
- 开发时间有限

## 4 支持条件

### 4.1 计算机系统支持

开发时需要的支持条件：

- 硬件：

**CPU：Intel Core I5 以上，**

**内存：4G 以上；**

**硬盘：500G 以上；**

➤ **软件：**

**操作系统为 Window 7 -64 位**

**集成开发工具 Eclipse**

**数据库采用 Mysql，项目运行环境为 JDK6.0.**

**其他可能用到的开发工具包括：Dreamweaver,**

**Microsoft Visio, Rational Rose, Microsoft Project, Maven 等**

**运行时需要的支持条件：**

**一、服务器的要求**

**1. 服务器的中央处理部件（CPU）建议使用 2Ghz（以上） Xeon 处理器芯片。**

**2. 服务器内存必须使用服务器专用 ECC 内存**

**3. 为了保证数据存储的绝对可靠，建议硬盘使用磁盘冗余阵列（RAID 01）**

**4. 其他性能在经济条件允许的情况下，应该尽量使用高速稳定的配件。**

**二、服务器上应该配备的软件**

**1. 操作系统：Microsoft Windows 2000 server 或者 Microsoft Windows 2000 Advanced server**

**2. 数据库：Mysql**

**3. 服务器必须使用专业的防火墙和反病毒软件。**

**4. 除了为了运行必须配备的程序以外，服务器上建议尽量不要安装其他无关程序，以减少程序的混乱或者程序的意外冲突。**

**5. 各系的操作系统尽量统一。（Windows 9x 系列或者 Windows 2000 系列）。这样可以避免管理软件因为操作系统版本不一致造成的过多的开销。**

**6. 各系的机器必须也安装反病毒软件和防火墙。以防止网络上的蠕虫病毒在整个网络范围内的蔓延。**

#### 4.2 需要用户承担的工作

安装博客系统前自主或在开发小组指导下租用服务器，配备相应硬件系统，或直接使用云服务器，在服务器上按用户手册进行平台软件安装和配置。

#### 4.3 需由外单位提供的条件

本系统为独立开发，不需要外单位提供条件。

## 5 专题计划要点

### 5.1 开发人员集训计划

从项目开始起，小组成员定期针对项目中将使用到的软件或技术进行学习和练习，并相互沟通，以便提高技术水平和项目质量，确保项目进行顺利。

表 5.1 小组集训内容

培训内容	时间	参与者及负责人
<b>MS Project 使用</b>	<b>2017-3-18 到 2017-3-19</b>	小组所有成员，由董建文负责
<b>Rose 使用</b>	<b>2017-3-25 到 2017-3-26</b>	小组所有成员，由黄伟负责
<b>Mysql 使用</b>	<b>2017-4-4 到 2017-4-7</b>	小组所有成员，由黄伟负责
<b>Github 并行开发</b>	<b>2017-4-8</b>	小组所有成员，由董建文负责

### 5.2 测试计划

2017 年 5 月 11 日—2017 年 5 月 31 日对软件系统进行各项测试工作；

### 5.3 质量保证计划

严格按照项目开发过程中的各项步骤，从项目立项，可行性研究报告、需求分析报告、项目开发计划等，进行具体实施。

### 5.4 人员配置计划

该项目开发小组共 2 人：

组长：董建文

组员：黄伟

## 附录 2

# 软件需求规格说明书

项目名称：RelaxBlog

参与人员：董建文 黄伟

## 修订历史记录

日期	版本	说明	作者
2017/3/30	1.0	未评审的初稿	董建文、黄伟
2017/4/24	2.0	已评审	董建文、黄伟

## 目录

1 引言.....	56
1.1 目的 .....	56
1.2 文档格式 .....	56
1.3 预期的读者和阅读建议 .....	56
1.4 范围.....	57
1.5 术语.....	58
1.6 参考文献 .....	58
2 系统概述.....	58
2.1 概述.....	58
2.2 功能.....	59
2.3 运行环境 .....	59
2.4 假设与依赖.....	60
3 系统特性.....	60
3.1 系统角色 .....	60
3.2 博客管理 .....	61
3.2.1 增加文章.....	61
3.2.2 修改博文.....	61
3.2.3 删除文章.....	62
3.2.4 文章阅读.....	62
3.2.5 添加文章.....	62
3.2.6 添加友情链接.....	63
3.2.7 编辑友情链接.....	63
3.2.8 删除友情链接.....	63

---

3.3 游客访问	63
3.3.1 显示文章列表	63
3.3.2 显示文章摘要	64
3.3.3 显示文章标签	64
3.3.4 显示指定页的文章列表	64
3.3.5 显示最新文章	65
3.3.6 显示友情链接	65
3.3.7 通过关键字搜索文章	65
3.3.8 查看文章	65
3.3.9 增加评论留言	66
3.3.10 查看留言	66
3.4 后台界面	67
3.4.1 显示文章总数	67
3.4.2 显示时间	67
3.5.3 显示友情链接总数	67
3.5.4 显示最新文章	67
4 非功能性需求	68
4.1 性能需求	68
4.2 安全性需求	68
4.3 可用性需求	68
4.4 用户文档	69
4.5 其它需求	69
5 外部接口需求	69
5.1 用户接口	69
5.2 硬件接口	69
5.3 软件接口	69
5.4 通信接口	69



## 1 引言

### 1.1 目的

此文档从整体到局部，从各个角度对系统进行描述。首先从整体进行概述，展示系统的全貌和功能分块，然后又对功能需求、性能需求和其它非功能性需求进行了详细的描述。其中对功能需求的描述从数据和联系进行阐述，配合了 UML 的用例图、类图、序列图、状态图等，主要描述了每一用例的各种基本事件流。这些文字和图形都为了本文档能详细准确地描述用户的需求，同时也为用户更容易地理解这些需求的描述创造了条件。

此文档详尽说明了这一软件产品的需求和规格，这些规格说明是进行设计的基础，也是编写测试用例和进行系统测试的主要依据。同时，该文档也是用户确定软件功能需求的主要依据。

### 1.2 文档格式

本文档按以下要求和约定进行书写：

- (1) 页面的左边距为 2.5cm，右边距为 2.0cm，装订线靠左，行距为最小值 20 磅。
- (2) 标题最多分三级，分别为黑体小三、黑体四号、黑体小四，标题均加粗。
- (3) 正文字体为宋体小四号，无特殊情况下，字体颜色均采用黑色。
- (4) 出现序号的段落不采用自动编号功能而采用人工编号，各级别的序号依次为 (1)、1)、a) 等，特殊情况另作规定。

### 1.3 预期的读者和阅读建议

本文档的主要内容共分 4 部分：综合描述、系统特性、和非功能性需求和外

部接口描述。综合描述部分主要对系统的整体结构进行了大致的介绍；系统特性部分对系统的功能需求进行了详细描述，是本文的主要部分；非功能性需求部分对非功能需求进行了详细的描述；外部接口需求部分对用户界面、软件接口、硬件接口和通讯接口等进行了描述。

本文档面向多种读者对象：

（1）项目管理者：项目管理者可以根据该文档了解预期产品的功能，并据此进行统筹安排、项目管理。

（2）设计员：对需求进行分析，并设计出系统，包括数据库的设计。

（3）程序员：配合《设计报告》，根据这些文档编写代码，以及《用户手册》。

（4）测试员：根据本文档编写测试用例，并对软件产品进行功能性测试和非功能性测试。

（5）销售人员：了解预期产品的功能和性能。

（6）用户：了解预期产品的功能和性能，并与分析人员一起对整个需求进行讨论和协商。

（7）其他人员：如部门领导、公司领导等可以据此了解产品的功能和性能。

在阅读本文档时，首先要了解产品的功能概貌，然后可以根据自身的需要对每一功能进行适当的了解。

## 1.4 范围

RelaxBlog 是一款用于发表文章、与游客留言互动的博客平台系统，此系统展示给用户的主要功能有以下几条：

1. 设定管理员账户信息并提供登录验证接口
2. 发表并管理已发表的文章
3. 管理文章附件
4. 管理网站部分页面链接信息
5. 供游客阅读文章内容

以下的涉及到平台运行的条件不在本系统的功能范围之内

1. 服务器的搭建
2. 游客的注册和登录
3. 链接网站/音乐的维护
4. 检测或纠正文章\评论内容

## 1.5 术语

**数据库 (database):** 数据库是长期储存在计算机内、有组织的、可共享的数据集合。数据库中的数据指的是以一定的数据模型组织、描述和储存在一起、具有尽可能小的冗余度、较高的数据独立性和易扩展性的特点并可在一定范围内为多个用户共享。

**用例图 (use-case diagram):** 是指由参与者 (Actor)、用例 (Use Case) 以及它们之间的关系构成的用于描述系统功能的视图, 呈现了一些参与者和一些用例, 以及它们之间的关系, 主要用于对系统、子系统或类的功能行为进行建模。

**接口 (interface):** 接口是计算机系统中两个独立的部件进行信息交换的共享边界。

**时序图 (Sequence Diagram):** 又称序列图、顺序图, 指通过描述对象之间发送消息的时间顺序显示多个对象之间的动态协作的视图。

**RB:** 本项目名称 RelaxBlog 缩写

## 1.6 参考文献

- [1] 《UML 说明》, 长江软件有限公司
- [2] 《需求规格报告格式标准》, 长江软件有限公司过程化组织
- [3] GB/T8567——2006 《计算机软件文档编制规范》;
- [4] 《软件工程导论(第五版)》 张海藩编著 清华大学出版社
- [5] 《软件工程》 李浪、朱雅莉、熊江主编 华中科技大学出版社;
- [6] 《软件文档写作教程》 马平、黄冬梅编著 电子工业出版社;

## 2 系统概述

### 2.1 概述

本项目旨在开发一个新型博客, 主要功能是实现一个由个人管理的, 能够提供发布博客, 更新博客, 播放 (云) 音乐等功能, 结合了文字、图像、其他博客

或网站的链接及其它与主题相关的媒体，能够让读者以互动的方式留下意见为用户，旨在开发出简洁易用，界面清爽的个人博客系统，提供一个深度交流沟通的网络平台。

## 2.2 功能

网站登录	网站登录模块	输入检查管理员账号
		输入检查管理员密码
网站后台页面	后台主页面	显示文章总数量
		显示附件总数量
		显示友情链接数量
		显示最新文章
	发表文章	输入检查文章标题
		输入检查文章标签=
		文章编辑
		文章预览
		存为草稿
	文章管理	显示文章信息列表
		修改文章
		删除文章
		文章预览
	友情链接管理	添加新连接
		链接编辑
		连接删除
游客访问页面	访问主页	显示文章列表
		显示文章标签
		查看友情链接
		文章搜索
	浏览文章	显示文章内容
		显示最后编辑时间
		添加留言
		显示留言

## 2.3 运行环境

该系统为 B/S 三层结构，它的运行环境分客户端、应用服务器端和数据库服

务器端三部分。

以下是系统的软件环境。

(1) 客户端

操作系统：Windows8 或更新版本。

浏览器：IE6 以上，其它常见浏览器如 FireFox，Chrome。

(2) 应用服务器端

操作系统：Windows2000 Server 或更新版本。

应用服务器：Tomcat 5.5 或更新版本。

数据库访问：JDBC。

(3) 数据库服务器端

操作系统：Windows2000 Server 或更新版本。

数据库系统：Mysql 或更新版本。

## 2.4 假设与依赖

本项目是否能够成功实施，主要取决于以下的条件：

- 1) 用户愿意使用我们的博客系统。
- 2) 博客系统没有明显缺陷
- 3) 操作流畅
- 4) 用户自己有个人服务器支持我们的博客系统的安装

## 3 系统特性

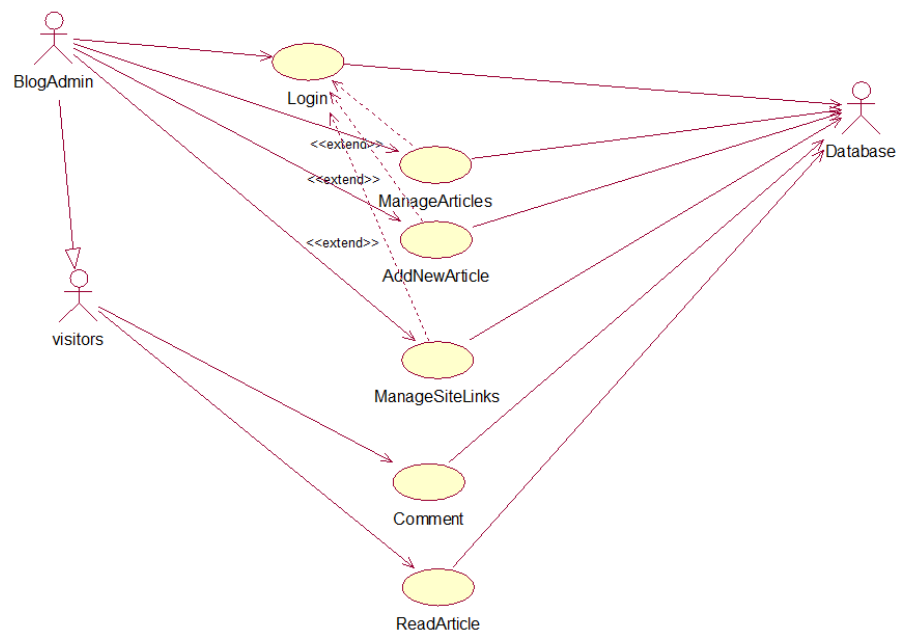
### 3.1 系统角色

本系统主要用于学校的以下几类人员：

(1) 系统管理员，完成系统管理与维护，例如，维护文章，管理用户评论等。

(2) 游客，查看博主所写的各项文章。

(3) 博主，博主和管理员是同一个人，主要完成博文的撰写等。



用例图：

## 3.2 博客管理

### 3.2.1 增加文章

使用者：博主。

目的：增加新的博文。

基本事件流：

- 1、用户进入管理博文界面，本用例开始；
- 2、系统显示学生信息输入界面，用户输入标题，正文等。
- 3、用户确认文章信息，保存文章到数据库。
- 4、本用例结束

### 3.2.2 修改博文

使用者：博主。

目的：修改已有的博文。

基本事件流：

- 1、用户进入管理博文界面，本用例开始；

- 2、系统显示学生信息输入界面，用户输入标题，正文，以及表情图片等。
- 3、用户确认文章信息，保存文章到数据库。
- 4、设置是否允许评论。
- 5、本用例结束

### **3.2.3 删除文章**

**使用者：博主。**

**目的：删除已有的博文。**

**基本事件流：**

- 1、用户进入管理博文界面，本用例开始；
- 2、博主选择所要删除的文章等。
- 3、用户确认，从数据库中删除数据。
- 4、本用例结束

### **3.2.4 文章阅读**

**使用者：博主。**

**目的：阅读已有的博文。**

**基本事件流：**

- 1、用户进入博文界面，本用例开始；
- 2、博主选择所要阅读的文章等。
- 3、用户确认，从数据库中查找数据，显示到界面。
- 4、本用例结束

### **3.2.5 添加文章**

**使用者：博主。**

**目的：添加博文。**

**基本事件流：**

- 1、用户进入管理博文界面，本用例开始；
- 2、点击添加文章。
- 3、用户确认输入内容，向数据库添加信息。
- 4、本用例结束

### 3.2.6 添加友情链接

使用者：博主。

目的：增加一些与博主关系密切的一些友情地址。

基本事件流：

- 1、用户进入友情链接界面，本用例开始；
- 2、博主输入将要连接的地址。
- 3、用户确认，添加到数据库，并在友情链接中增加细节。
- 4、本用例结束

### 3.2.7 编辑友情链接

使用者：博主。

目的：编辑一些与博主关系密切的一些友情地址。

基本事件流：

- 1、用户进入友情链接界面，本用例开始；
- 2、博主输入将要重新连接的地址。
- 3、用户确认，添加到数据库，并在友情链接中改变原有的数据。
- 4、本用例结束

### 3.2.8 删除友情链接

使用者：博主。

目的：编辑一些不能继续访问的一些友情链接地址。

基本事件流：

- 1、用户进入友情链接界面，本用例开始；
- 2、博主输入将要删除连接的地址。
- 3、用户确认，查找数据库，并在友情链接中删除原有的数据。
- 4、本用例结束

## 3.3 游客访问

### 3.3.1 显示文章列表

使用者：游客。



**目的：查看文章列表。**

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、访问博客主页，文章列表自动显示。
- 3、本用例结束。

### **3.3.2 显示文章摘要**

**使用者：游客。**

**目的：查看文章摘要。**

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、鼠标移动到每篇文章的卡票上，自动显示文章内容。
- 3、本用例结束。

### **3.3.3 显示文章标签**

**使用者：游客。**

**目的：查看文章所属标签。**

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、访问特定文章主页，文章分类自动显示。
- 3、本用例结束。

### **3.3.4 显示指定页的文章列表**

**使用者：游客。**

**目的：查看指定页文章列表。**

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、访问博客主页，点击页面下方的“下一页”“上一页”“第 X 页”对应按钮进行跳转。
- 3、本用例结束。

### 3.3.5 显示最新文章

使用者：游客。

目的：查看最新文章。

基本事件流：

- 1、用户进入主界面，本用例开始；
- 2、在页面下方自动显示最新发表的博文
- 3、本用例结束。

### 3.3.6 显示友情链接

使用者：游客。

目的：博主友链以及相似博主的文章地址。

基本事件流：

- 1、用户进入主界面，本用例开始；
- 2、访问博客主页,点击"友情链接"按钮
- 3、跳转到友情链接地址。
- 4、本用例结束

### 3.3.7 通过关键字搜索文章

使用者：游客。

目的：搜索特定文章。

基本事件流：

- 1、用户进入主界面，本用例开始；
- 2、访问博客主页,点击"友情链接"按钮
- 3、跳转到友情链接地址。
- 4、本用例结束

### 3.3.8 查看文章

使用者：游客。

目的：查看文章。

基本事件流：

- 1、用户进入主界面，本用例开始；

- 2、点击想要查看的文章
- 3、查找数据库是否有相似的文章，并进行结果的返回，显示到屏幕上
- 5、本用例结束

### 3.3.9 增加评论留言

**使用者：**游客。

**目的：**增加评论留言，促进交流。

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、访问文章界面,滑动到文章下方
- 3、查看指定文章时,在文章最后的留言框中输入留言者名称,联系方式,留言内容,点确定提交。
- 4、将信息保存到数据库
- 5、本用例结束

### 3.3.10 查看留言

**使用者：**游客。

**目的：**搜索特定文章。

**基本事件流：**

- 1、用户进入主界面，本用例开始；
- 2、访问文章界面,滑动到文章下方
- 3、查看指定文章时,查看不同用户的留言评论，增加交流
- 4、本用例结束

## 3.4 后台界面

### 3.4.1 显示文章总数

**使用者：**管理员

**目的：**帮助管理员查看所有。

**基本事件流：**

- 1、用户进入后台界面，本用例开始；
- 2、在 DASHBOARD 界面有文章基本数量信息。
- 3、本用例结束

### 3.4.2 显示时间

**使用者：**管理员

**目的：**帮助管理员查看所有。

**基本事件流：**

- 1、用户进入后台界面，本用例开始；
- 2、在 DASHBOARD 界面有时间信息。
- 3、本用例结束

### 3.5.3 显示友情链接总数

**使用者：**管理员

**目的：**帮助管理员查看所有。

**基本事件流：**

- 1、用户进入后台界面，本用例开始；
- 2、在 DASHBOARD 界面有友情链接数量信息。
- 3、本用例结束

### 3.5.4 显示最新文章

**使用者：**管理员

**目的：**帮助管理员查看所有。

**基本事件流：**

- 1、用户进入后台界面，本用例开始；
- 2、在 DASHBOARD 界面有最新文章信息。

### 3、本用例结束

## 4 非功能性需求

### 4.1 性能需求

- (1) 客户端一般响应时间（除报表统计、数据导入）不超过 1 秒。
- (2) 报表统计时间不超过 30 秒。
- (3) 支持 100 名用户并发访问，并保证性能不受影响

### 4.2 安全性需求

- (1) 权限控制

根据不同用户角色，设置相应权限，用户的重要操作都做相应的日志记录以备查看，没有权限的用户禁止使用系统。

- (2) 重要数据加密

本系统对一些重要的数据按一定的算法进行加密，如用户口令、重要参数等。

### 4.3 可用性需求

- (1) 方便操作，操作流程合理

尽量从用户角度出发，以方便使用本产品。如：新增学生信息时，敲入回车键光标的自动跳转、输入法的自动转换，信息检索时输入汉语简拼快速检索到结果等。可以通过快速键方便用户录入信息，所有操作可仅通过键盘完成。

- (2) 支持没有计算机使用经验、计算机使用经验较少及有较多计算机使用经验的用户均能方便地使用本系统。

- (3) 控制必录入项

本系统能够对必须录入的项目进行控制，使用户能够确保信息录入的完整。同时对必录入项进行有效的统一的提示。

- (4) 容错能力

系统具有一定的容错和抗干扰能力，在非硬件故障或非通讯故障时，系统能够保证正常运行，并有足够的提示信息帮助用户有效正确地完成任务。

- (5) 操作完成时有统一规范的提示信息

例如删除操作时，系统可提示警示框“您确认删除记录吗？操作不可恢复！”，

用户点击确认后，系统才执行删除操作，删除后可直接返回相关页面。

## 4.4 用户文档

同本软件一起发行的用户文档包括：

- (1) 安装手册：Word 格式文件/PDF 文档。
- (2) 用户手册：Word 格式文件/PDF 文档。
- (3) 在线帮助：HTML Help 格式文件，联机式。

## 4.5 其它需求

- (1) 支持多浏览器。
- (2) 系统安装方便，易于维护。

# 5 外部接口需求

## 5.1 用户接口

本系统采用 B/S 架构，所有界面使用 WEB 风格，用户界面的具体细节将在概要设计文档中描述。

## 5.2 硬件接口

无特殊要求

## 5.3 软件接口

无特殊需求。

## 5.4 通信接口

无特殊需求。

## 附录 3

# 软件设计文档

项目名称：RelaxBlog

参与人员：董建文 黄伟

# 目录

1. 引言 .....	72
1.1 编写目的.....	72
1.2 项目背景.....	72
1.3 定义 .....	72
1.4 参考资料.....	72
2. 任务概述.....	73
2.1 目标 .....	73
2.2 运行环境.....	73
2.3 需求概述.....	73
2.4 条件与限制.....	73
3. 总体设计.....	75
3.1 处理流程.....	75
3.1.1 添加文章.....	75
3.1.2 评论.....	76
3.1.3 登录.....	78
3.1.4 文章管理.....	79
3.1.5 链接管理.....	81
3.2 总体结构.....	84
4. 接口设计.....	85
4.1 外部接口.....	85
4.2 内部接口.....	85
5. 数据结构设计.....	85
5.1 逻辑结构设计.....	85
5.2 物理结构设计.....	86
5.3 数据结构与程序的关系.....	88
6. 运行设计.....	88
6.1 运行模块的组合.....	88
6.2 运行控制.....	88
6.3 运行时间.....	89
7. 出错处理设计.....	89
7.1 出错输出信息.....	89
7.2 出错处理对策.....	89
8. 安全保密设计.....	89
9. 维护设计.....	90



# 1. 引言

## 1.1 编写目的

本阶段完成系统的大致设计并明确系统的数据结构与软件结构。本概要设计说明书的目的就是进一步细化软件设计阶段得出的软件概貌，把它加工成在程序细节上非常接近与源程序开发的软件表示。

预期读者：软件测试员、程序开发员、软件分析员

## 1.2 项目背景

- a. 项目的委托单位：无
- b. 开发团队：SUNDAY 小组
- c. 主管部门：SUNDAY 小组
- d. 该软件系统与其他系统的关系：系统相对独立

## 1.3 定义

- a. Mysql:系统服务器所使用的数据库关系系统（DBMS）。
- b. SQL：一种用于访问查询数据库的语言
- c. 事务流：数据进入模块后可能有多种路径进行处理。
- d. 主键：数据库表中的关键域。值互不相同。
- e. 外部主键：数据库表中与其他表主键关联的域。
- f. ROLLBACK:数据库的错误恢复机制。
- g. 缩写：
- h. SQL：Structured Query Language(结构化查询语言)。
- i. ATM：Asynchronous Transfer Mode (异步传输模式)。
- j. UML：统一建模语言、是一套用来设计软件蓝图的标准建模语言，是一种从软件分析、设计到编写程序规范的标准化建模语言。

## 1.4 参考资料

- a. 项目开发计划；
- b. RelaxBlog 需求规格说明书；
- c. RelaxBlog 需求清单
- d. 《软件工程概论》 李存珠编著 南京大学计算机系出版 2001 年 8 月
- e. 《软件工程——实践者的研究方法》 Roger S Pressman 著

## 2. 任务概述

### 2.1 目标

- a. 由于本系统用于个人博客系统，使用频繁，因此可靠性要较高、安全性较高。
- b. 系统的运行速度要快

### 2.2 运行环境

该系统为 B/S 三层结构，它的运行环境分客户端、应用服务器端和数据库服务器端三部分。以下是系统的软件环境。

#### （1）客户端

操作系统：Windows8 或更新版本。

浏览器：IE9 以上，其它常见浏览器如 FireFox，Chrome。

#### （2）应用服务器端

操作系统：Windows7 或更新版本。

应用服务器：Tomcat 5.5 或更新版本。

数据库访问：JDBC。

#### （3）数据库服务器端

操作系统：Windows7 或更新版本。

数据库系统：MYSQL5.7 或更新版本。

### 2.3 需求概述

现今很多技术开发人员都有撰写个人博客的习惯，一方面进行技术的交流学习，另一方面也可以进行自己开发心得以及经验的记录。

本项目旨在开发一个新型博客，主要功能是实现一个由个人管理的，能够提供发布博客，更新博客，播放（云）音乐等功能，结合了文字、图像、其他博客或网站的链接及其它与主题相关的媒体，能够让读者以互动的方式留下意见为用户，旨在开发出简洁易用，界面清爽的个人博客系统，提供一个深度交流沟通的网络平台。

### 2.4 条件与限制

开发时需要的支持条件：

硬件：

服务器：Pentium III 500 以上或更高，

内存：512M 以上；

硬盘：至少 80G 以上；

CD-ROM：32 倍速以上；  
网络适配器：10MB/100MB 自适应；  
打印机一台  
UPS(选配)  
工作站：Pentium 4 以上微机；  
内存：512MB  
硬盘：至少 80 以上；  
CD-ROM：32 倍速以上；  
网络适配器：10MB/100MB 自适应  
网络：至少一台服务器  
至少一台工作站  
使用 TCP/IP 协议的局域网

**软件：**

操作系统为 Window XP，使用集成开发工具 Eclipse5.5.1,数据库采用 SQL Server2000，项目运行环境为 JDK6.0。  
其他开发工具包括：Dreamweaver, Microsoft Visio, Rational Rose, Power Designer Trial 11, TomCat6.0 CVSNT2.5.03

**运行时需要的支持条件：****服务器的要求**

1. 服务器的中央处理部件（CPU）建议使用 PIII 1G（以上） Xeon 处理器芯片。
2. 服务器内存必须使用服务器专用 ECC 内存
3. 为了保证数据存储的绝对可靠，硬盘应使用磁盘冗余阵列（RAID 01）
4. 为了防止服务器不可预测的故障，或者服务器的定期维护对公司整个业务造成的影响，所有建议使用两台服务器。两台服务器应构成双机热备份。中间使用 Watchdog 电路。这样的结构可以保证整个系统的长时间不间断工作，即使在服务器定期维护的时候也可以使用后备另一台服务器工作。
5. 服务器应支持热插拔电源
6. 服务器必须配备 UPS（不间断电源）。
7. 服务器应该放在学校内部。不然无法进行程序调试。
8. 服务器应该必须有固定 IP 地址。
9. 其他性能在经济条件允许的情况下，应该尽量使用高速稳定的配件。

**服务器上应该配备的软件**

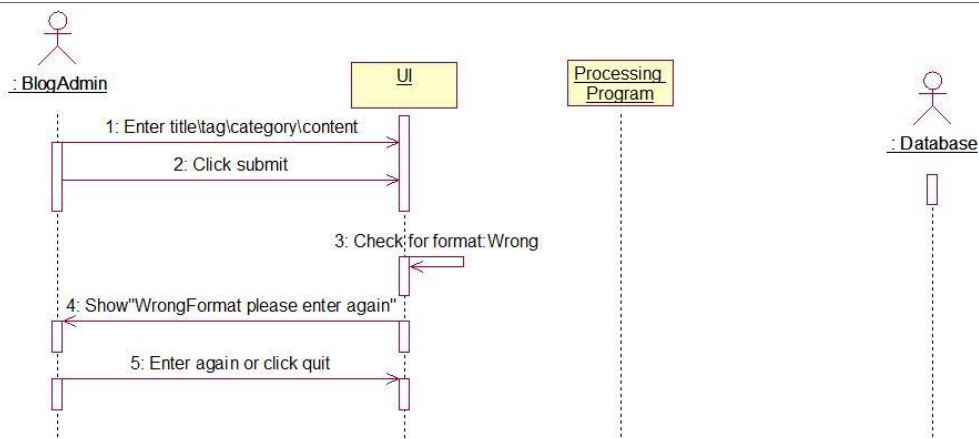
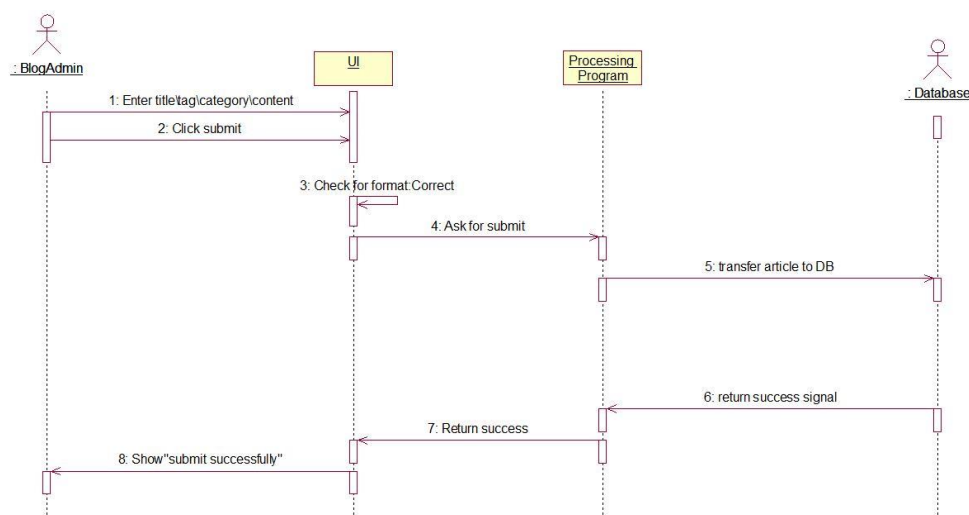
1. 操作系统：Microsoft Windows 2000 server 或者 Microsoft Windows 2000 Advanced server
2. 数据库：Microsoft SQL Server 2000 (简体中文版)
3. 服务器必须使用专业的防火墙和反病毒软件。
4. 除了为了运行必须配备的程序以外，服务器上建议尽量不要安装其他无关程序，以减少程序的混乱或者程序的意外冲突。
5. 各系的操作系统尽量统一。（Windows 9x 系列或者 Windows 2000 系列）。这样可以避免管理软件因为操作系统版本不一致造成的过多的开销。
6. 各系的机器必须也安装反病毒软件和防火墙。以防止网络上的蠕虫病毒在整个网络范围内的蔓延。
7. 如果要打印涉及字段较多的报表，应该配备针式打印机。

## 3. 总体设计

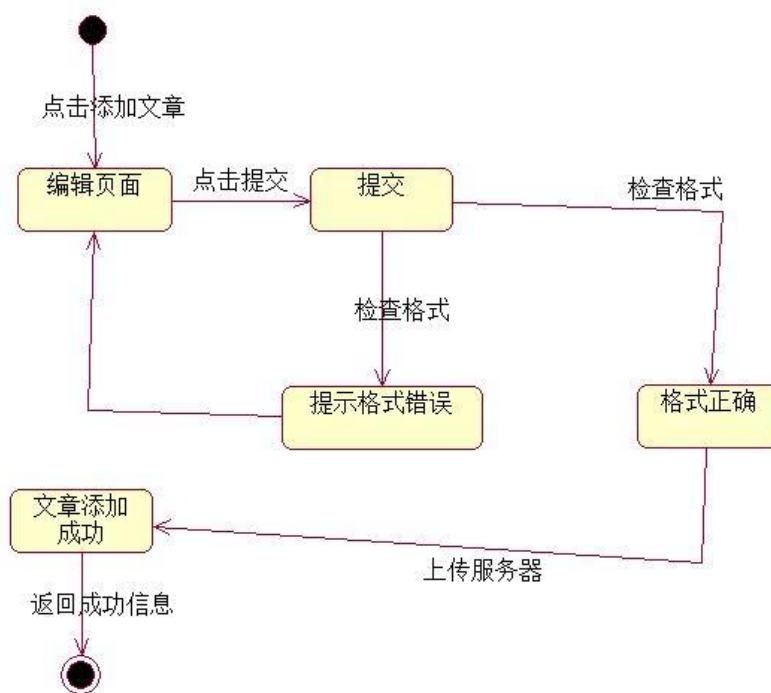
### 3.1 处理流程

#### 3.1.1 添加文章

时序图：

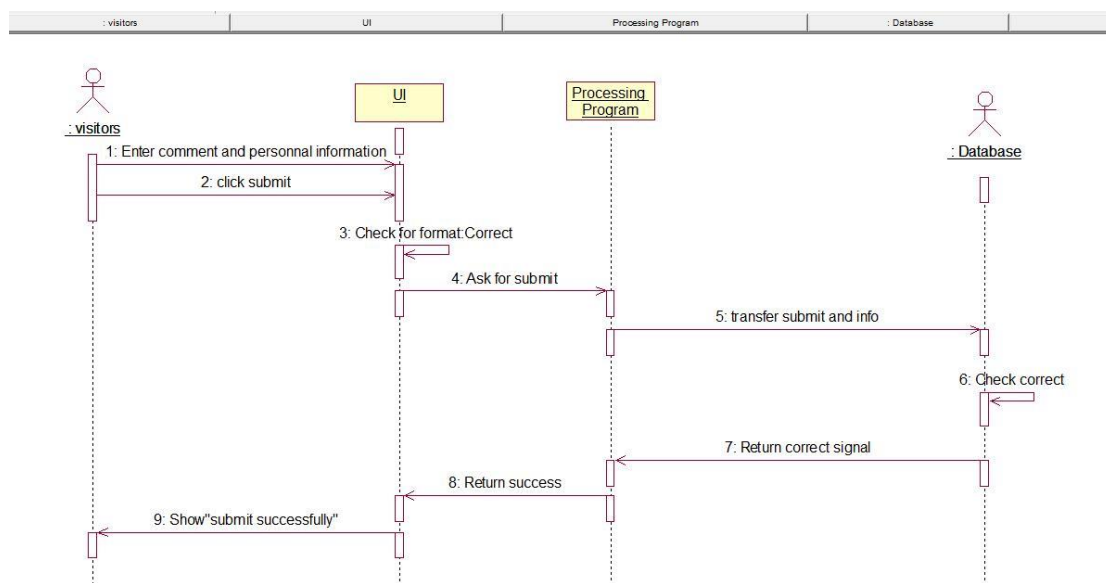


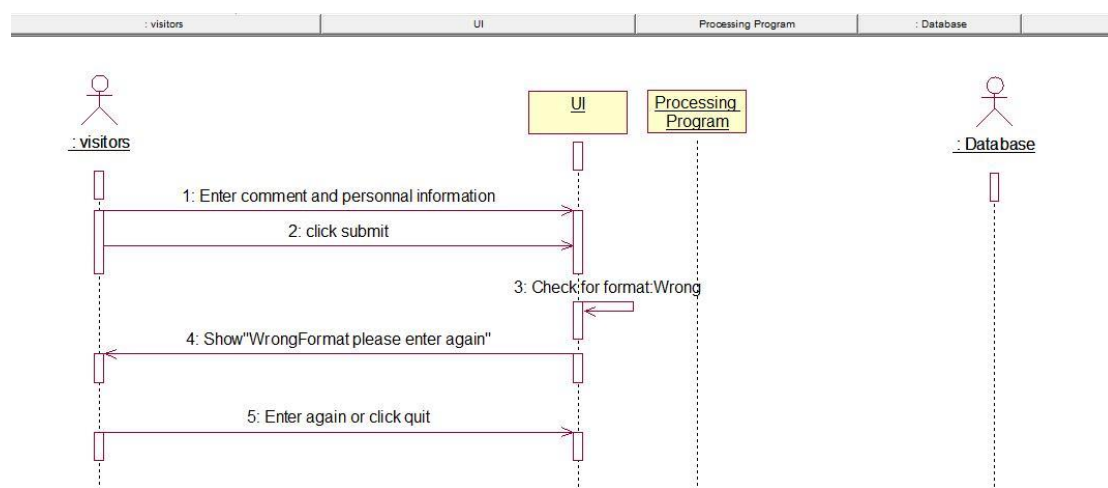
状态图：



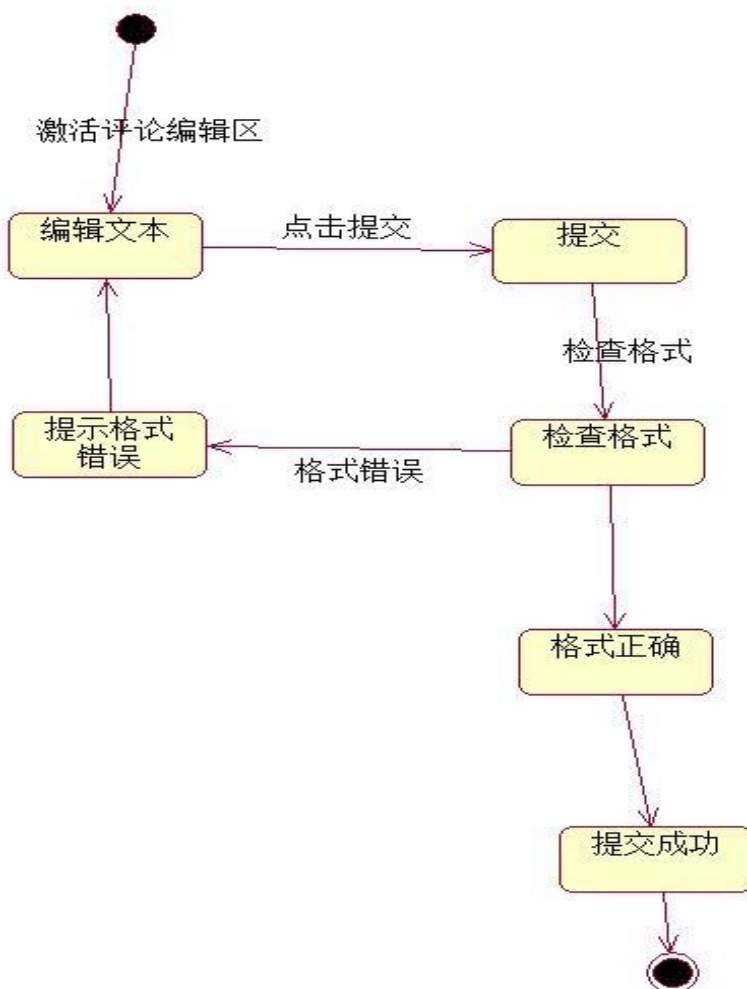
### 3.1.2 评论

时序图：



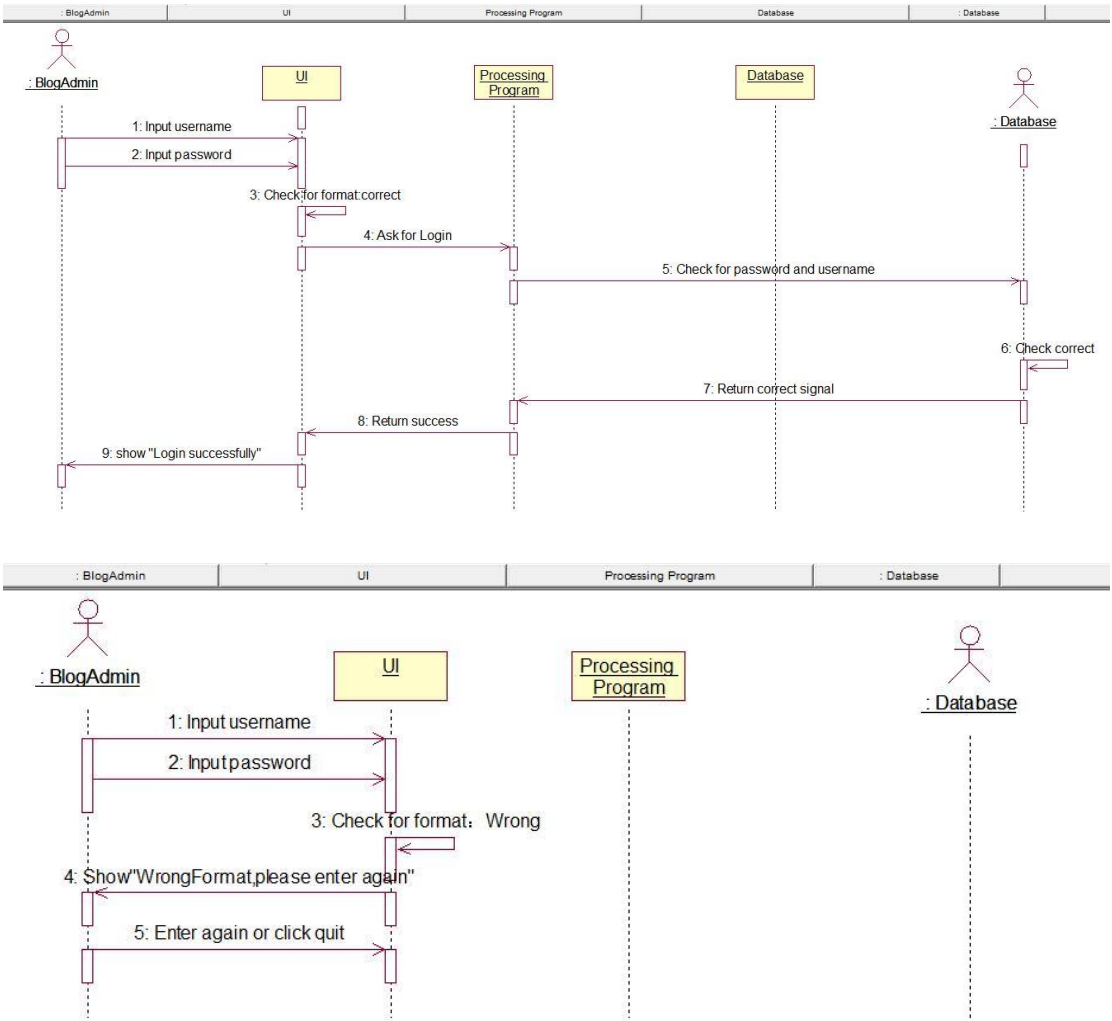


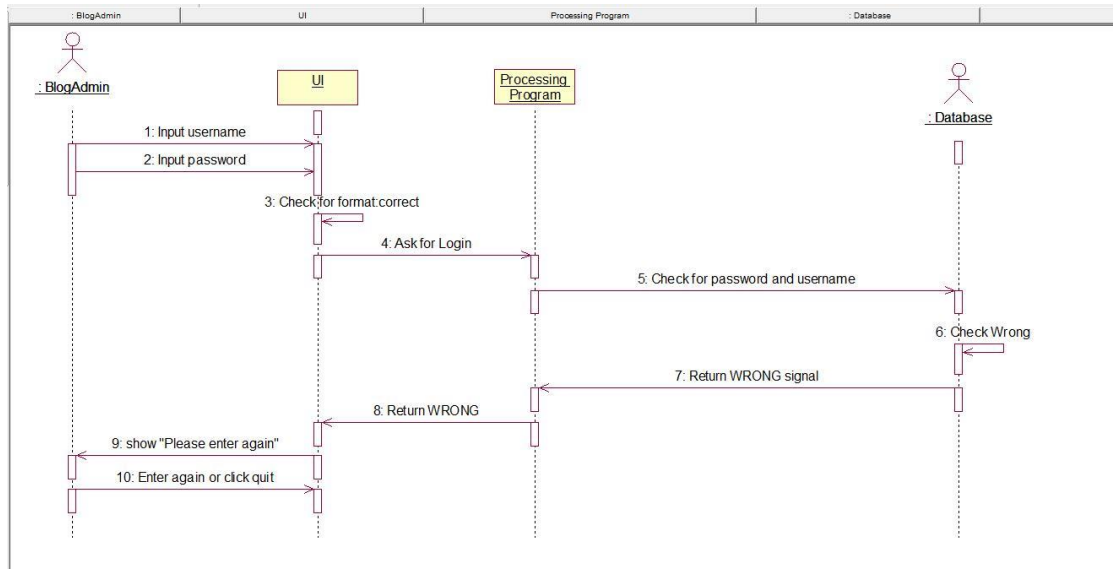
状态图:



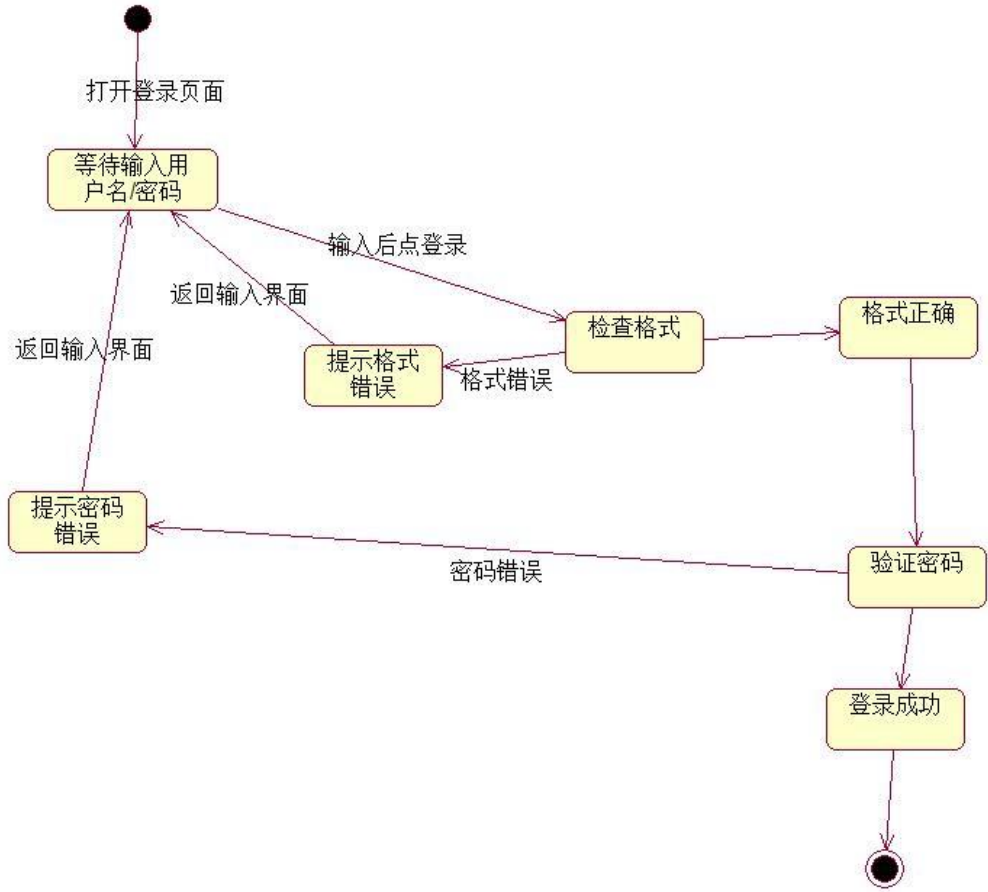
3.1.3 登录

时序图：





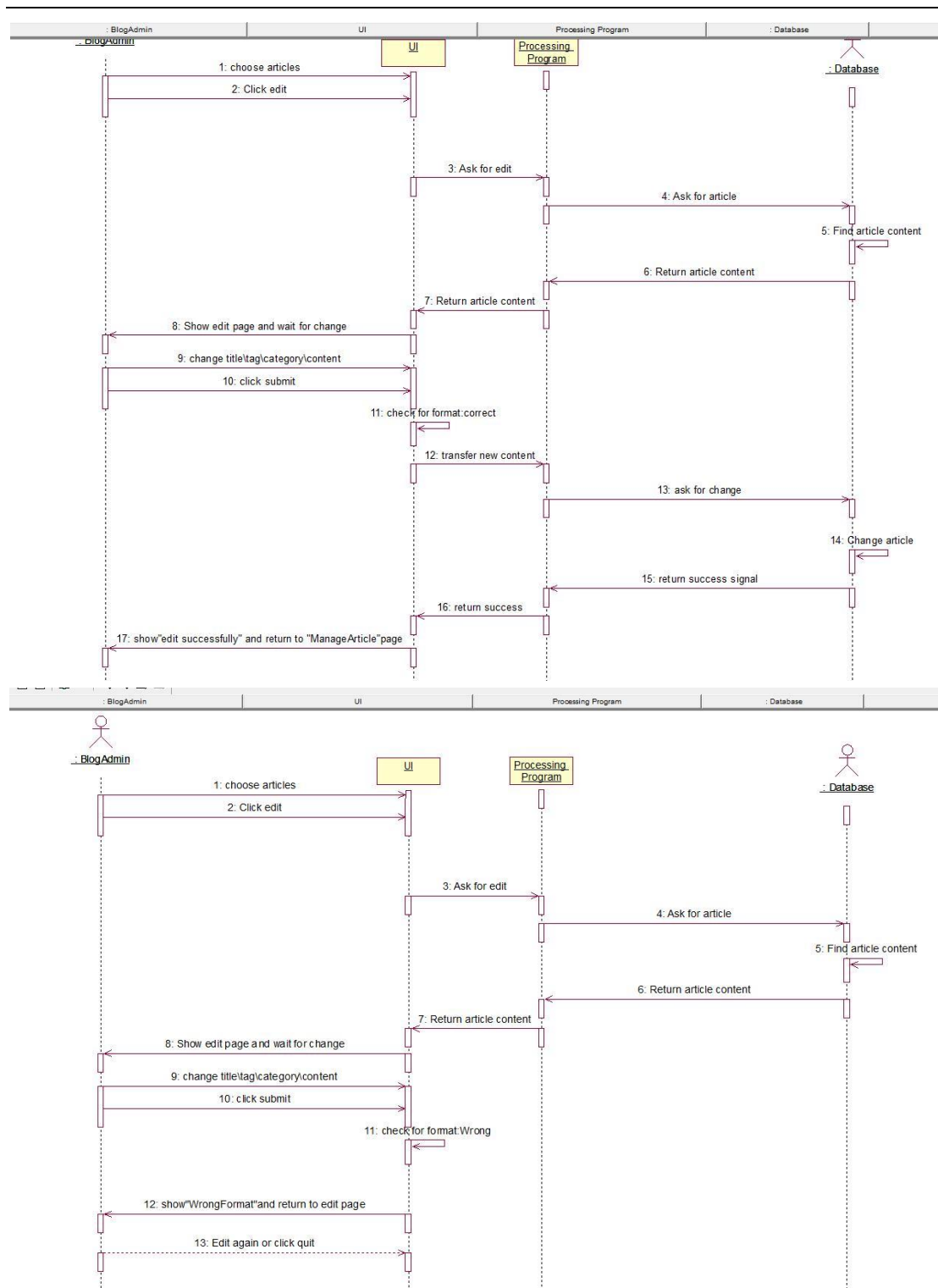
状态图：



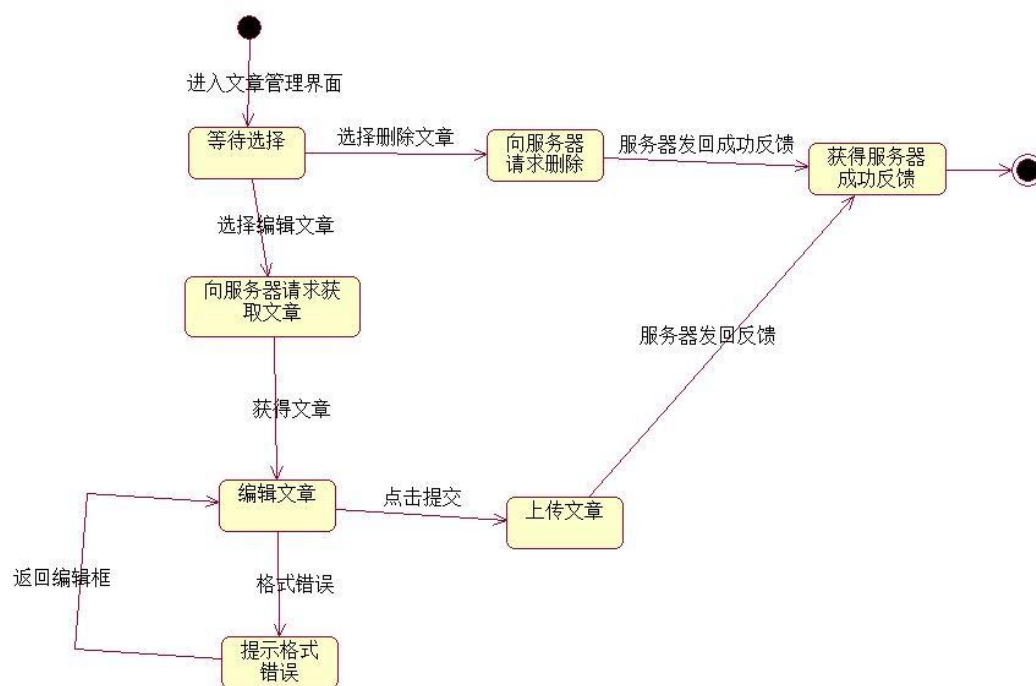
3.1.4 文章管理

时序图：



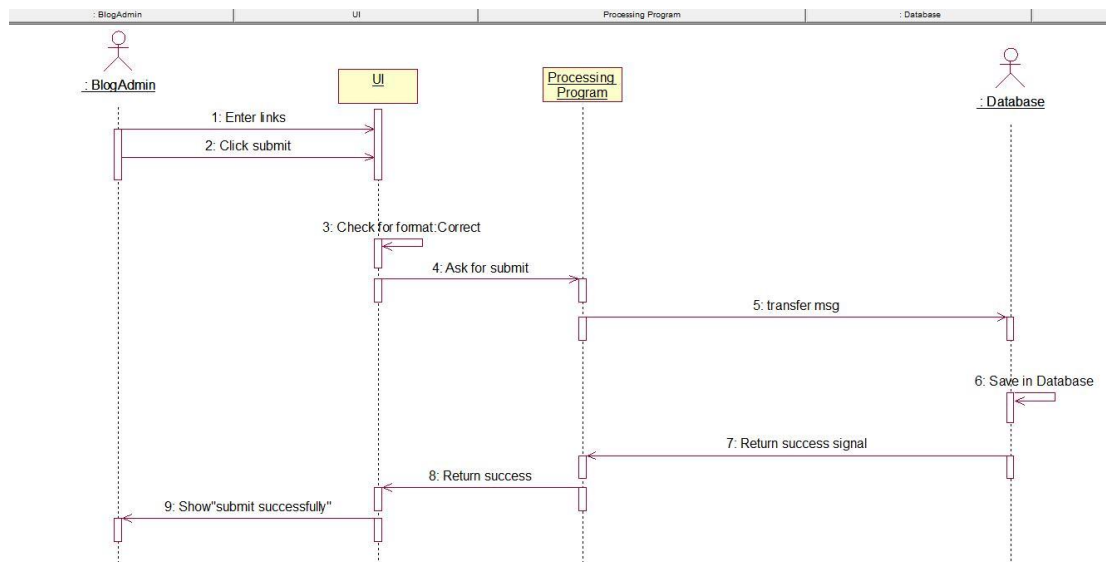


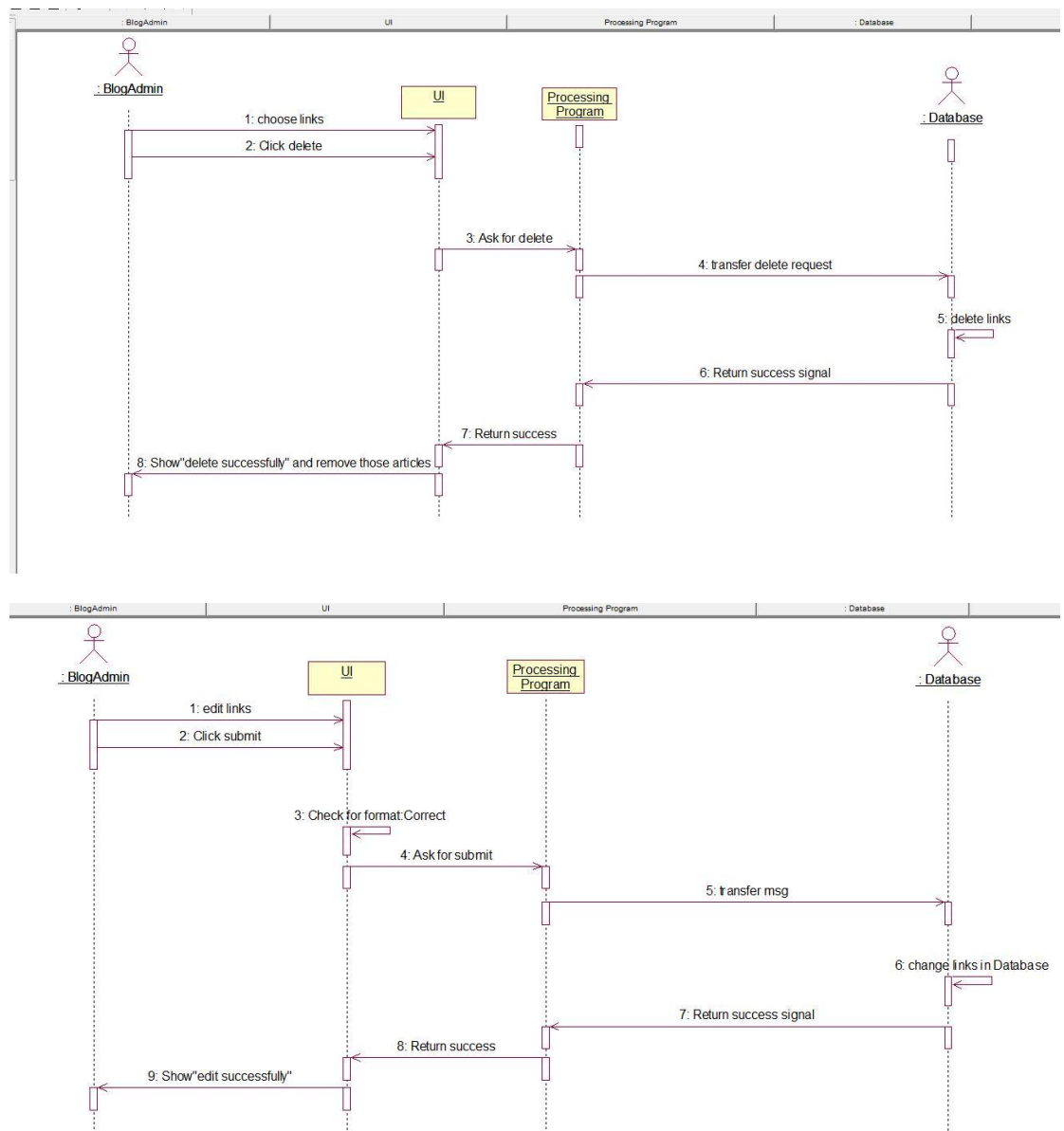
状态图：



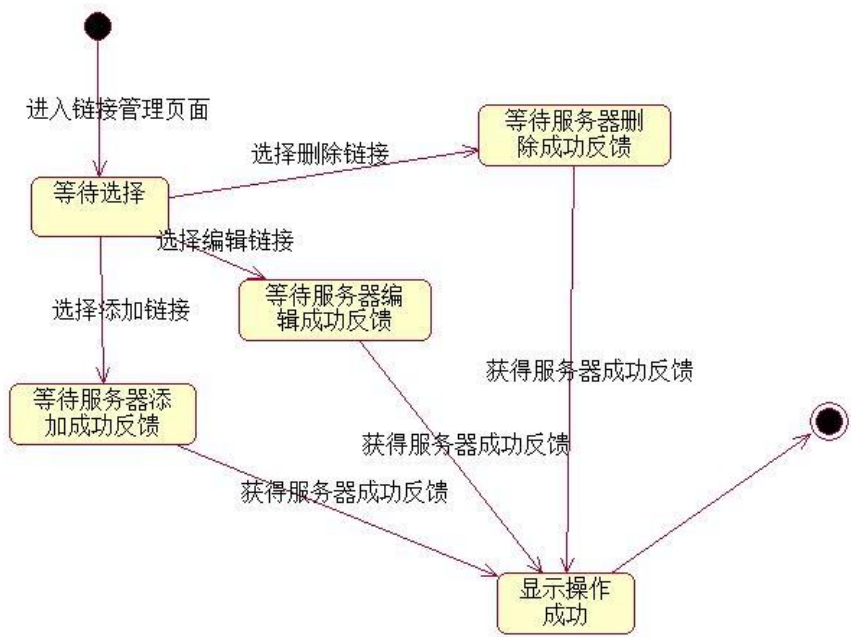
### 3.1.5 链接管理

时序图：



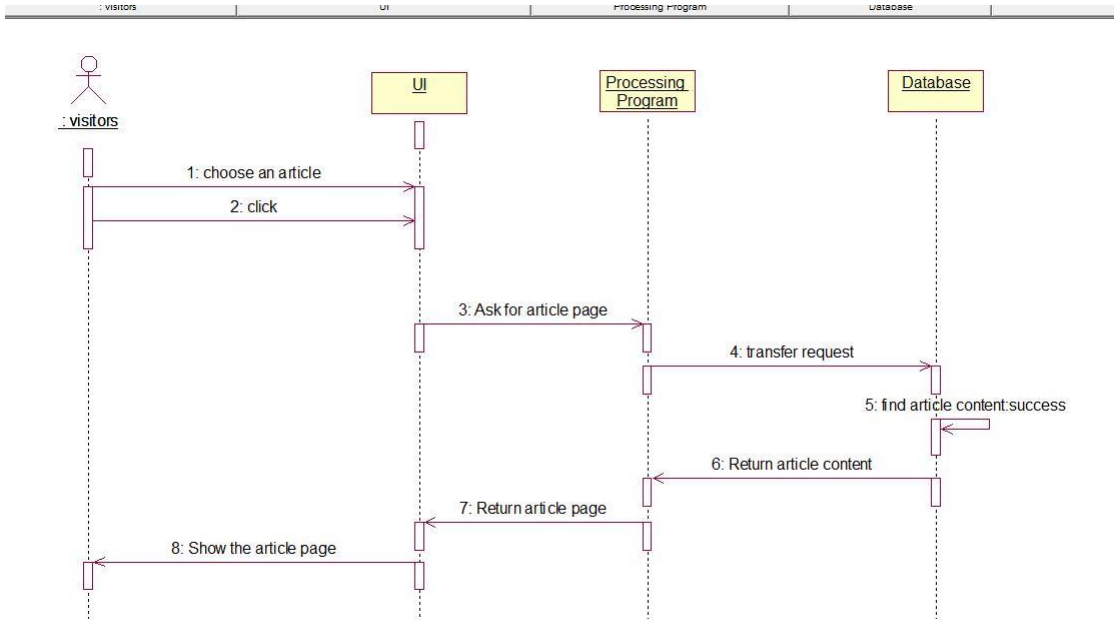


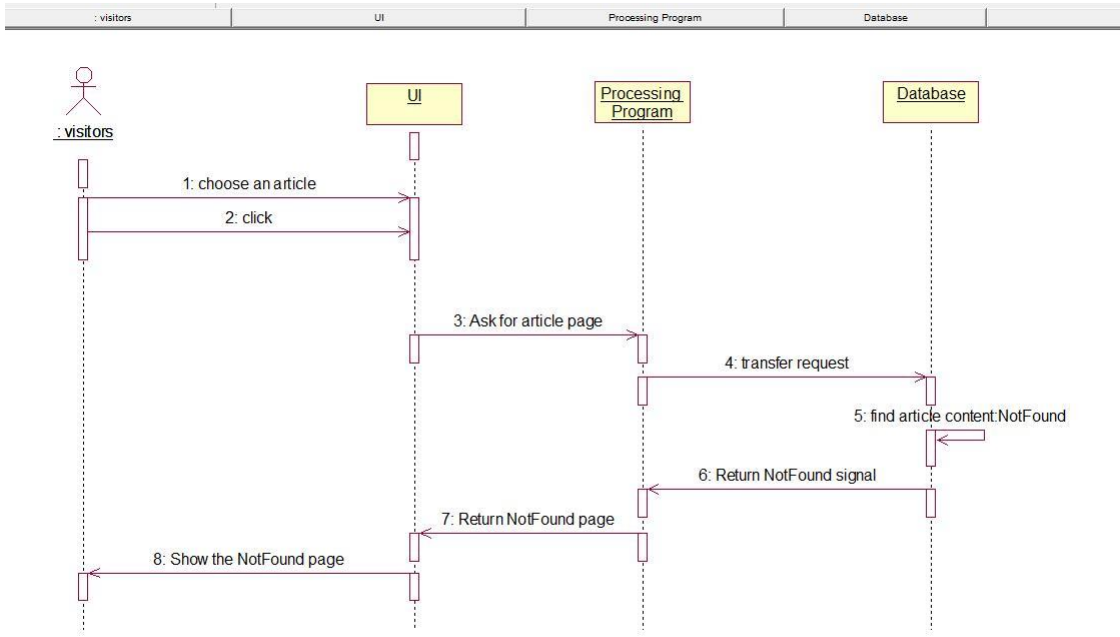
状态图：



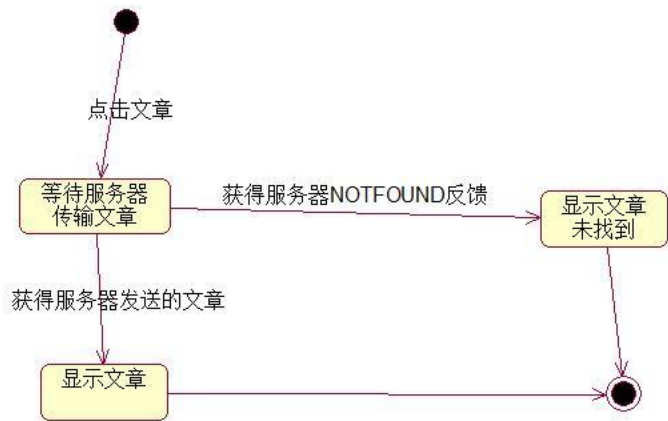
3.1.6 阅读文章

时序图



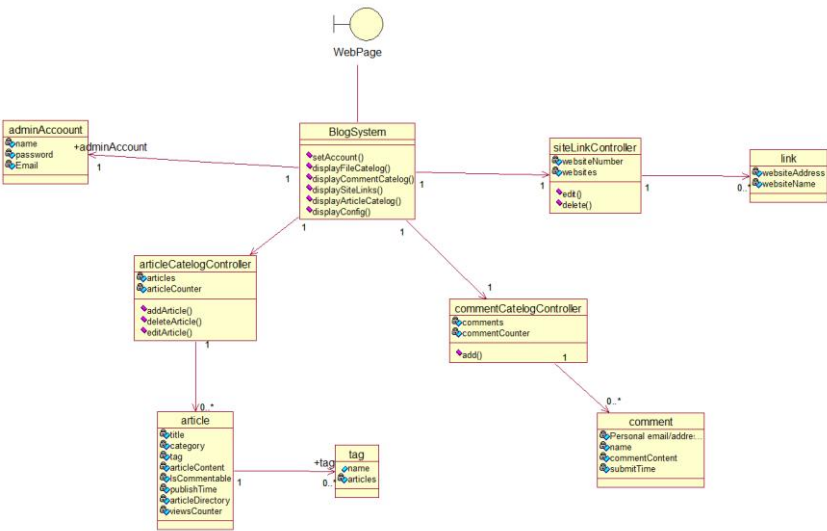


状态图：



3.2 总体结构

总体设计类图



## 4. 接口设计

### 4.1 外部接口

1. USB 接口，连接鼠标等设备。
2. DVI 接口，DVI(Digital Visual Interface，数字视频接口)，它是 D-SUB 接口的继承者，用来传输数字信号。
3. SCSI 接口，SCSI(Small Computer System Interface，支持包括磁盘驱动器、磁带机、光驱、扫描仪在内的多种设备。

### 4.2 内部接口

通过面向对象语言设计类，在 public 类中实现调用；类间实现严格封装

## 5. 数据结构设计

### 5.1 逻辑结构设计

文章字典数据类型

属性名	存储代码	类型	备注
编号	Id	Int	数据库设置为自增
标题	Ttitle	Varchar	非空

创建日期	Create_date	Date	存储创建日期
修改日期	Modified_date	Date	储存最后修改日期
内容	Content	File	文章较大以文件形式存储
标签	Catefories	Varchar	储存文件的分类
评论数	Comments_num	Int	

评论表字典数据类型

属性名	存储代码	类型	备注
编号	Id	Int	自增
评论日期	Create_date	Date	非空
评论者	Author	Varchar	非空
评论者编号	Author_id	Int	
评论内容	Content	Text	

用户字典数据类型

属性名	存储代码	类型	备注
用户编号	User_id	Int	
用户名	Username	Varchar	
登录密码	Password	Varchar	
用户邮箱	Email	Varchar	
用户主页地址	Home_url	varchar	

## 5.2 物理结构设计

文章字典数据类型

```

public String getId() { return id; }
public void setId(String id) { this.id = id; }
public String getTitle() { return title; }
public void setTitle(String title) {...}

public Date getCreateDate() { return createDate; }
public void setCreateDate(Date createDate) { this.createDate = createDate; }
public Date getModifiedDate() { return modifiedDate; }
public void setModifiedDate(Date modifiedDate) { this.modifiedDate = modifiedDate; }
public File getContent() { return content; }
public void setContent(File content) { this.content = content; }
public String getAuthorId() { return authorId; }
public void setAuthorId(String authorId) { this.authorId = authorId; }
public String getCatefories() { return Catefories; }
public void setCatefories(String catefories) { Catefories = catefories; }
public int getCommentsNum() { return commentsNum; }
public void setCommentsNum(int commentsNum) { this.commentsNum = commentsNum; }

```

评论字典数据类型

```
public int getId() { return id; }

public void setId(int id) { this.id = id; }

public Date getCreateDate() { return createDate; }

public void setCreateDate(Date createDate) { this.createDate = createDate; }

public String getAuthor() { return author; }

public void setAuthor(String author) { this.author = author; }

public int getAuthorId() { return authorId; }

public void setAuthorId(int authorId) { this.authorId = authorId; }

public int getOwnerId() { return ownerId; }

public void setOwnerId(int ownerId) { this.ownerId = ownerId; }

public Text getContent() { return content; }

public void setContent(Text content) { this.content = content; }
```

关系字典数据结构

```
public String usrId;
public String relId;

public String getUsrId() {
    return usrId;
}

public void setUsrId(String usrId) {
    this.usrId = usrId;
}

public String getRelId() {
    return relId;
}

public void setRelId(String relId) {
    this.relId = relId;
}
```

用户字典数据结构



```
public String userId;
public String username;
public String password;
public URL emailURL;
public URL homeURL;

public String getUserId() { return userId; }

public void setUserId(String userId) { this.userId = userId; }

public String getUsername() { return username; }

public void setUsername(String username) { this.username = username; }

public String getPassword() { return password; }

public void setPassword(String password) { this.password = password; }

public URL getEmailURL() { return emailURL; }

public void setEmailURL(URL emailURL) { this.emailURL = emailURL; }

public URL getHomeURL() { return homeURL; }

public void setHomeURL(URL homeURL) { this.homeURL = homeURL; }
```

## 5.3 数据结构与程序的关系

数据结构与程序是软件的重要组成部分，程序的正确执行依赖于合理的数据结构

# 6. 运行设计

## 6.1 运行模块的组合

本程序主要是以一个窗口为模块，一般一个窗口完成一个特定的功能，主窗口通过打开另一个子窗口来实现个模块之间不同功能的连接和组合。各模块之间相对独立，程序的可移植性好。各模块之间主要以传递数据项的引用来实现模块之间的合作和数据共享，不同的窗口之间实现使用 session 以及 cookie 来进行数据的共享与传递。

## 6.2 运行控制

运行控制将严格按照各模块间的函数调用关系来实现。

在网络传输方面，客户机再发送数据后，将等待服务器的确认到信号，收到后，在此等待服务器发送数据，然后对数据进行确认，服务器再接收到数据后发送确认信号，在对数据处理、访问数据库后，将返回信息送回客户机，并等待确认。采用 http 协议进行数据的传输，后台采用 jdbc 对数据库进行使用

## 6.3 运行时间

各模块运行时间不定，这也跟用户的操作以及数据的大小有关

# 7. 出错处理设计

## 7.1 出错输出信息

错误类型	原因	解决办法
数据库连接错误	数据库设置不正确或数据库异常	取消操作，并重新尝试
数据库查询错误	没有按照要求进行操作	取消操作，并重新尝试
输入错误	输入格式不规范	重新按照标准输入
不可预知错误	未知异常	暂无
其他操作错误	用户的不当操作使程序发生错误	终止操作，并提醒用户正确的操作方式

## 7.2 出错处理对策

我们对于本程序的几种可能的错误进行了分析，分别进行了不同的处理。主要的错误可能有：

**数据库连接错误：**这类错误主要是数据库设置不正确，或 SQL Server 异常引起的，我们只要取消本次操作，提醒用户检查数据库问题就可。

**数据库查询错误：**这类错误主要是用户没有按照，参数进行输入，我们中断该操作，并提示用户按照标准进行操作即可。

**输入错误：**这主要是用户输入不规范造成的，我们在尽量减少用户出错的条件的情况下，主要也是通过对话框，提醒用户，然后再次操作。

**其他操作错误：**对于用户的不正当操作，有可能使程序发生错误。我们主要是中止操作，并提醒用户中止的原因和操作的规范。

**其他不可预知的错误：**程序也会有一些我们无法预知或没考虑完全的错误，我们对此不可能作出安全的异常处理，这时我们主要要保证数据的安全，所以要经常的进行数据库备份，并能及时的和我们联系，以逐步的完善我们的程序

# 8. 安全保密设计

1. 数据库只为特定的用户开放，其他用户无法访问
2. 权限控制，根据不同用户角色，设置相应权限，用户的重要操作都做相应的日志记

录以备查看，没有权限的用户禁止使用系统。

3. 重要数据加密，本系统对一些重要的数据按一定的算法进行加密，如用户口令、重要参数等。
4. 由于个人博客系统是属于个人的一片领域因此该博客系统的安全性要达到一定的标准，服务器性能良好
5. 允许博主设置是否允许评论等操作。

## 9.维护设计

1. 采用 `github` 进行版本控制，可以进行版本的更新还原；
2. 利用好 `github` 平台进行，项目的管理开发，以及维护

## 附录 4

# 测试计划书

项目名称：RelaxBlog

参与人员：董建文 黄伟

# 1 引言

## 1.1 编写目的

本测试计划的编写是为软件的单元测试、系统测试等测试环节提供指导和约束，以便于软件的功能的完备和完整性。此文档面向的使用者为项目指导老师、项目负责人、项目组员及其他测试相关人员。

## 1.2 背景

说明：

本测试计划面向的软件系统为 RelaxBlog 个人博客系统，

RelaxBlog 是我们小组开发的一个新型的博客，主要功能是实现一个由个人管理的，能够提供发布博客，更新博客，播放（云）音乐等功能，结合了文字、标签、代码块及其他博客或网站的链接及其它与主题相关的媒体，能够让读者以互动的方式留下意见为用户，旨在开发出简洁易用，界面清爽的个人博客系统，提供一个深度交流沟通的网络平台。项目周期为三个月。

## 1.4 参考资料

- 测试计划书国家标准（GB8567——88）
- 软件学院软件开发事件课程要求
- 需求规格说明书
- 软件项目开发计划书
- 设计说明书

# 2 计划

## 2.1 软件说明

功能编号、软件功能、功能描述、输入数据描述、输出结果描述等信息如下表

编号（标识符）	功能	功能描述	需要的输入数据描述	输出结果描述
---------	----	------	-----------	--------

1	输入用户账号	输入登录所用用户 账户	键盘输入字符串,长度不超过 10 字节	符合格式则不显示输出,不符合格式则确认时报错,显示相应错误
2	输入用户密码	输入登录所用用户 密码	键盘输入字符串,长度不超过 20 字节,不少于 6 字节,只允许英文\数字\标点符号	符合格式则不显示输出,不符合格式则确认时报错,显示相应错误
3	显示文章总数量	将文章总数量显示 在页面上	登录并访问主页面	发表了 X 篇文章
4	显示当前日期时间	将文章总数量显示 当前日期时间	登录并访问主页面	日期为 XXXX/X/X
5	显示友情链接数量	将友情链接总数量 显示在页面上	登录并访问主页面	友链了 X 个网站
6	显示最新文章	将最新发布的五个 文章标题显示在页面上	登录并访问主页面	最新文章:各文章,每文章一行
7	输入文章标题	输入文章的标题	键盘输入字符串,长度不超过 50 字节,不为空	符合格式则不显示输出,不符合格式则确认时报错,显示相应错误
8	输入文章标签	输入文章类别	标签:键盘输入字符串,长度不超过 10 字节,可为空	符合格式则不显示输出,不符合格式则确认时报错,显示相应错误
9	文章编辑	输入文章正文内容	键盘输入字符串,通过相关编辑按钮修改字体格式等属性。最长不超过 10KB,可为空	符合要求后上传至数据库,页面提示保存成功,否则提示相应错误
10	存为草稿	保存当前文章内容	单击相应按钮	将编辑框内内容上传至服务器,上传成功则提示“已保存”
11	显示文章信息列表	将文章简要信息显示 在页面上	登录并访问“文章管理”页面则自动显示	显示表格,列信息为:文章标题-ID-发布时间-标签-状态-操作
12	修改文章	修改文章正文	点击对应文章的“修改”按钮	将服务器上的文章信息显示在编辑框中,等待编辑,完成后检查长度等格式,符合要求后更新到服务器上

13	删除文章	删除指定的一篇文章	点击对应文章的“删除”按钮	若确认则显示“删除成功”，同时删除服务器端文章内容。取消则无显示
14	文章预览	阅读指定的一篇文章	点击对应文章的“阅读”按钮	显示文章内容
15	添加新连接	添加新的友情链接	登录并访问“友情链接管理”，点击“添加”按钮，输入新连接名称、连接地址，均为字符串，名称不超过100字节，地址不超过200字节，描述300字节，前两者不能为空	若接收到成功信息，页面显示“添加成功”，否则显示“添加失败，请刷新后重试”
16	链接编辑	编辑已存在的友情链接	登录并访问“友情链接管理”，点击“编辑”按钮，输入新连接名称、连接地址，均为字符串，名称不超过100字节，地址不超过200字节，描述300字节，前两者不能为空	若接收到成功信息，页面显示“编辑成功”，否则超时时显示“编辑失败，请刷新后重试”
17	链接删除	删除已存在的友情链接	登录并访问“友情链接管理”，点击对应的友情链接后的“删除”按钮。	若接收到成功信息，页面显示“删除成功”，否则超时时显示“删除失败，请刷新后重试”
18	显示文章列表	显示博客主发布的文章列表	访问博客主页，文章列表自动显示	在页面上显示各文章的标题，发布时间，标签信息
19	显示文章摘要	显示文章摘要内容（第一段）	访问博客主页，鼠标指向该文章	显示指定文章的前20字。
20	显示文章标签	显示文章标签信息	访问博客主页，文章标签自动显示	显示文章的标签信息
21	查看友情链接	显示友情链接信息	访问博客主页，点击“友情链接”按钮	显示友情链接信息，包括名称和链接，若无友情链接则显示“暂无友情链接”
22	显示文章内容	查看指定的文章	访问博客主页，选定文章鼠标单击查看	在浏览器中显示文章内容
23	显示最后编辑时间	显示指定文章的最后编辑时间	查看指定文章时，文章的最后编辑时间自动显示在文章标题下方	在浏览器中显示文章最后编辑时间

24	添加留言	在指定文章后添加留言	查看指定文章时,在文章最后的留言框中输入留言者名称,联系方式,留言内容,点确定提交。名称不超过20 字节,不能为空;联系方式不超过30 字节,可以为空;留言内容不超过300 字节,不能为空;均以字符串形式输入	接收到服务器成功信息则显示“发表成功”,超时未接收到信息则提示“发表失败,请刷新重试”
25	显示留言	浏览文章时可以看到当前文章的留言	查看指定文章时,若存在留言,则留言自动显示在文章下方	显示留言信息,若无留言则显示空白

## 3 测试设计说明

### 3.1 测试 1（标识符）

考虑输入域的分割，输入合法范围的左侧、中间和右侧。

#### 3.1.1 控制

人工测试，直接在登录框内输入，记入 excel 表格

#### 3.1.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	空白	输入域分割
2	合理长度，正确用户名	输入域分割
3	合理长度，错误用户名	输入域分割
4	超长长度	输入域分割

#### 3.1.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	提示长度错误
2	检测密码正确与否
3	提示登录信息错误
4	提示长度错误



### 3.1.4 过程

打开登录页面（通过访问管理页面或输入域名进入），输入用户名密码，点击登录，观察执行结果。

## 3.2 测试 2（标识符）

考虑输入域的分割，输入合法范围的左侧、中间和右侧。

### 3.2.1 控制

人工测试，直接在登录框内输入，记入 excel 表格

### 3.2.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	空白	输入域分割
2	合理长度，正确密码	输入域分割
3	合理长度，错误密码	输入域分割
4	超长长度	输入域分割

### 3.2.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	提示长度错误
2	检测用户名正确与否
3	提示登录信息错误
4	提示长度错误

### 3.2.4 过程

打开登录页面（通过访问管理页面或输入域名进入），输入用户名密码，点击登录，观察执行结果。

### 3.3 测试 3-6（标识符）

系统可选状态较少，输出由算法和数据库决定。因此测试较为简单。

#### 3.3.1 控制

人工测试，直接打开主页面，观察结果，记入 excel 表格

#### 3.3.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	已登录	简单分类
2	未登陆	简单分类

#### 3.3.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	正确显示信息
2	跳转登录页面

#### 3.3.4 过程

直接打开主页面，观察执行结果是否符合预期。

### 3.4 测试 7-10（标识符）

考虑输入域的分割，输入合法范围的左侧、中间和右侧。

#### 3.4.1 控制

人工测试，访问添加文章页面，输入相应用例，观察结果，记入 excel 表格

#### 3.4.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
----	----	----

1	空白标题+合法标签及内容，发表文章	输入域分割
2	标题“软件架构”+标签“软件架构”+内容“abcd”发表文章	输入域分割
3	标题“abcd”+超长标签+内容“abcd”发表文章	输入域分割
4	超长标题+标签“1234”+内容“你好”发表文章	输入域分割
5	标题“abcd”+标签“abcd”+超长内容，发表文章	输入域分割
6	未登陆	输入域分割
7	标题“abcd”+标签“abcd”+内容“abcd”，发表文章	输入域分割
8	标题“abcd”+标签“abcd”+内容粗体，发表文章	输入域分割
9	标题“abcd”+标签“abcd”+内容下划线，发表文章	输入域分割
10	复制图片到内容，发表文章	输入域分割
11	空白标题+合法标签及内容，存为草稿	输入域分割
12	合法标题+标签“软件架构”+内容“abcd”存为草稿	输入域分割
13	标题“abcd”+超长标签+内容“abcd”存为草稿	输入域分割
14	超长标题+标签“1234”+内容“你好”存为草稿	输入域分割
15	标题“abcd”+标签“abcd”+超长内容，存为草稿	输入域分割
16	未登陆	输入域分割
17	标题“abcd”+标签“abcd”+内容“abcd”，存为草稿	输入域分割
18	标题“abcd”+标签“abcd”+内容粗体，存为草稿	输入域分割
19	标题“abcd”+标签“abcd”+内容下划线，存为草稿	输入域分割
20	复制图片到内容，存为草稿	输入域分割

### 3.4.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
----	------

1	提示标题长度错误
2	提示发表成功
3	提示标签超长
4	提示标题长度错误
5	提示内容超长
6	跳转登录界面
7	提示发表成功
8	提示发表成功
9	提示发表成功
10	提示发表成功
11	提示标题长度错误
12	提示发表成功
13	提示标签超长
14	提示标题长度错误
15	提示内容超长
16	跳转登录界面
17	提示发表成功
18	提示发表成功
19	提示发表成功
20	提示发表成功

### 3.4.4 过程

打开添加文章界面，输入文章标题、文章标签、文章内容，修改、整理格式，点发表文章或存为草稿

## 3.5 测试 11（标识符）

系统可选状态较少，输出由算法和数据库决定。因此测试较为简单。

### 3.5.1 控制

人工测试，直接打开主页面，观察结果，记入 excel 表格

### 3.5.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	已登录	简单分类

2	未登陆	简单分类
---	-----	------

### 3.5.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	正确显示信息
2	跳转登录页面

### 3.5.4 过程

直接打开主页面，观察执行结果是否符合预期。

## 3.6 测试 7-10（标识符）

考虑输入域的分割，输入合法范围的左侧、中间和右侧。

### 3.6.1 控制

人工测试，访问修改文章页面，输入相应用例，观察结果，记入 excel 表格

### 3.6.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	空白标题+合法标签及内容，发表文章	输入域分割
2	标题“软件架构”+标签“软件架构”+内容“abcd”发表文章	输入域分割
3	标题“abcd”+超长标签+内容“abcd”发表文章	输入域分割
4	超长标题+标签“1234”+内容“你好”发表文章	输入域分割
5	标题“abcd”+标签“abcd”+超长内容，发表文章	输入域分割
6	未登陆	输入域分割
7	标题“abcd”+标签“abcd”+内容“abcd”，发表文章	输入域分割

8	标题“abcd”+标签“abcd”+内容粗体，发表文章	输入域分割
9	标题“abcd”+标签“abcd”+内容下划线，发表文章	输入域分割
10	复制图片到内容，发表文章	输入域分割
11	空白标题+合法标签及内容，存为草稿	输入域分割
12	合法标题+标签“软件架构”+内容“abcd”存为草稿	输入域分割
13	标题“abcd”+超长标签+内容“abcd”存为草稿	输入域分割
14	超长标题+标签“1234”+内容“你好”存为草稿	输入域分割
15	标题“abcd”+标签“abcd”+超长内容，存为草稿	输入域分割
16	未登陆	输入域分割
17	标题“abcd”+标签“abcd”+内容“abcd”，存为草稿	输入域分割
18	标题“abcd”+标签“abcd”+内容粗体，存为草稿	输入域分割
19	标题“abcd”+标签“abcd”+内容下划线，存为草稿	输入域分割
20	复制图片到内容，存为草稿	输入域分割

### 3.6.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	提示标题长度错误
2	提示发表成功
3	提示标签超长
4	提示标题长度错误
5	提示内容超长
6	跳转登录界面
7	提示发表成功
8	提示发表成功
9	提示发表成功
10	提示发表成功
11	提示标题长度错误
12	提示发表成功
13	提示标签超长
14	提示标题长度错误

15	提示内容超长
16	跳转登录界面
17	提示发表成功
18	提示发表成功
19	提示发表成功
20	提示发表成功

## 6.4 过程

打开修改文章界面，输入文章标题、文章标签、文章内容，修改、整理格式，点发表文章或存为草稿

## 3.7 测试 13 及 17（标识符）

系统可选状态较少，输出由算法和数据库决定。因此测试较为简单。

### 3.7.1 控制

人工测试，直接点击页面上的产出，观察结果，记入 excel 表格

### 3.7.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	已登录	简单分类
2	未登陆	简单分类
3	已登录但数据库关闭	简单分类

### 3.7.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	正确删除
2	跳转登录页面
3	提示删除失败

### 3.7.4 过程

直接点击删除，观察执行结果是否符合预期。

## 3.8 测试 14,18,19,20,21,22,23（标识符）

系统可选状态较少，输出由算法和数据库决定。因此测试较为简单。

### 3.8.1 控制

人工测试，直接打开相应页面，观察结果，记入 excel 表格

### 3.8.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	已登录	简单分类
2	未登陆	简单分类

### 3.8.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	正确显示信息
2	跳转登录页面

### 3.8.4 过程

直接打开相应页面，观察执行结果是否符合预期。

## 3.9 测试 15 和 16（标识符）

考虑输入域的分割，输入合法范围的左侧、中间和右侧。

### 3.9.1 控制

人工测试，直接在编辑框内输入，点击提交，观察结果记入 excel 表格



### 3.9.2 输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

序号	输入	策略
1	Name 为空	输入域分割
2	Name= 'github', URL 为空	输入域分割
3	Name= ' github ', URL= 'github.com', 描述为空	输入域分割
4	都为空	输入域分割
5	Name 超长	输入域分割
6	Name= 'github', URL 超长, 描述为空	输入域分割
7	Name= ' github ', URL= 'github.com', 描述超长	输入域分割
8	未登陆	输入域分割

### 3.9.3 输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

序号	预期结果
1	提示名称长度错误
2	提示 URL 长度错误
3	提交成功
4	报告任意长度错误
5	提示名称超长
6	提示 URL 超长
7	提示描述超长
8	提示登录信息错误

### 3.9.4 过程

打开添加或编辑页面（通过访问管理页面或输入域名进入），输入信息，点击提交，观察执行结果。

## 4 评价准则

### 4.1 范围

测试用例涵盖了最新版本需求及具体实现的功能,尽可能地包括了输入的各种情况以及预期输出.

### 4.2 数据整理

对于单元测试,尽可能地使用测试工具获取输出的值域是否在预期范围之内,通过自动化测试统计测试结果。对于功能测试和系统测试,较多地采用人工执行测试并用 excel 统计测试结果和输入输出细节信息。

### 4.3 尺度

评价测试是否通过,是以输出结果的类型、输入内容来决定,符合预期输出则为通过。

## 附录 5

# 用户手册

项目名称：RelaxBlog

参与人员：董建文 黄伟

---

1 引言 .....	0
1.1 编写目的.....	0
1.2 背景 .....	0
1.3 定义 .....	0
1.4 参考资料.....	1
2 用途 .....	1
2.1 功能 .....	1
2.2 性能 .....	4
2.2.1 精度.....	4
2.2.2 时间特性.....	4
2.2.3 灵活性.....	4
2.3 安全保密.....	5
2.3.1 后台管理对登录身份的验证.....	5
2.3.2 数据库信息的保护.....	5
3 运行环境.....	6
3.1 硬设备.....	6
3.3 数据结构.....	6
4 使用过程.....	6
4.1 安装与初始化.....	7
4.2 输入 .....	7
4.2.1 输入数据的现实背景.....	7
4.2.2 输入格式.....	8
4.2.3 输入举例.....	9
4.3 输出对每项输出作出说明.....	12
4.3.1 输出数据的现实背景.....	12
4.3.2 输出格式.....	12
4.3.3 输出举例.....	14
4.4 文卷查询.....	16
4.5 出错处理和恢复.....	17

# 用户手册

## 1 引言

### 1.1 编写目的

本用户手册的编写面向所有 RelaxBlog 的用户和测试人员，旨在帮助使用或测试 RelaxBlog 的人员正常地配置、运行、使用本系统，并对功能、运行环境、输入输出格式和错误处理恢复做出详细解释。

### 1.2 背景

本测试计划面向的软件系统为 RelaxBlog 个人博客系统，

RelaxBlog 是我们小组开发的一个新型的博客，主要功能是实现一个由个人管理的，能够提供发布博客，更新博客，播放（云）音乐等功能，结合了文字、标签、代码块及其他博客或网站的链接及其它与主题相关的媒体，能够让读者以互动的方式留下意见为用户，旨在开发出简洁易用，界面清爽的个人博客系统，提供一个深度交流沟通的网络平台。项目周期为三个月。

小组组长：董建文 组员：黄伟

### 1.3 定义

**Hibernate:**是一个开放源代码的对象关系映射框架，它对 JDBC 进行了非常轻量级的对象封装，它将 POJO 与数据库表建立映射关系，是一个全自动的 orm 框架，hibernate 可以自动生成 SQL 语句，自动执行。

**JDBC:**（Java DataBase Connectivity,java 数据库连接）是一种用于执行 SQL 语句的 Java API，可以为多种关系数据库提供统一访问，它由一组用 Java 语言编写的类和接口组成。JDBC 提供了一种基准，据此可以构建更高级的工具和接口，使数据库开发人员能够编写数据库应用程序。

**Spring MVC:** Spring MVC 属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。

**MySQL:** MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL 是最好的 RDBMS (Relational Database Management System, 关系数据库管理系统) 应用软件。

---

**Tomcat:** Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器，属于轻量级应用服务器。

## 1.4 参考资料

- 测试计划书国家标准（GB8567——88）
- 软件学院软件开发事件课程要求
- 需求规格说明书
- 软件项目开发计划书
- 设计说明书

# 2 用途

## 2.1 功能

编号（标识符）	功能	功能描述
1	输入用户账号	输入登录所用用户账户
2	输入用户密码	输入登录所用用户密码
3	显示文章总数量	将文章总数量显示在页面上
4	显示当前日期时间	将文章总数量显示当前日期时间
5	显示友情链接数量	将友情链接总数量显示在页面上
6	显示最新文章	将最新发布的五个文章标题显示在页面上

7	输入文章标题	输入文章的标题
8	输入文章标签	输入文章类别
9	文章编辑	输入文章正文内容
10	存为草稿	保存当前文章内容
11	显示文章信息列表	将文章简要信息显示在页面上
12	修改文章	修改文章正文
13	删除文章	删除指定的一篇文章
14	文章预览	阅读指定的一篇文章
15	添加新连接	添加新的友情链接

16	链接编辑	编辑已存在的友情链接
17	链接删除	删除已存在的友情链接
18	显示文章列表	显示博客主发布的文章列表
19	显示文章摘要	显示文章摘要内容（第一段）
20	显示文章标签	显示文章标签信息
21	查看友情链接	显示友情链接信息
22	显示文章内容	查看指定的文章
23	显示最后编辑时间	显示指定文章的最后编辑时间
24	添加留言	在指定文章后添加留言
25	显示留言	浏览文章时可以看到当前文章的留言

服务器的搭建、数据库的配置等过程不在本系统的功能域内。



## 2.2 性能

### 2.2.1 精度

本系统涉及的数据传输基于 HTML 协议，对精度的要求基于此协议。通常情况下，传输精确至位，以保证数据正确。

对于输入精度和输出的数据，均以字节为基本精度，保证字符串的数据正确。

字符串编码集为 UTF-8

### 2.2.2 时间特性

测试机环境：

I7 6700hq

主存 8G

显卡 GTX960M

客户机与服务器：LOCALHOST

测试用户连接数:1

测试情况：

按键响应时间 $\leq 10\text{ms}$

更新处理时间 $\leq 100\text{ms}$

数据传输时间 $\leq 100\text{ms}$

计算时间 $\leq 10\text{MS}$

### 2.2.3 灵活性

说明本软件所具有的灵活性，即当用户需求（如对操作方式、运行环境、结果精度、时间特性等的要求）有某些变化时，本软件的适应能力。

1. 编辑文章时,可以通过富文本格式编辑，可以通过 word 或 html 文档复制粘贴，也可以通过编辑框的切换功能切换至 html 格式编辑，修改标签类型、颜色，以及添加外部工具如云音乐播放器或视频。
2. 页面比例及大小可以自适应调节，浏览器的窗口大小不影响页面的正常浏览，因此移动端的浏览不需单独制作页面。
3. 管理页面等不同页面可以通过网址直接访问，如保存至收藏夹，而不必从主页点击相应按钮访问。用这种方法访问管理页面的前提是管理员身份登录过，否则会跳转至登录页面。

## 2.3 安全保密

### 2.3.1 后台管理对登录身份的验证

2.3.1.1 访问后台主页（index.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

2.3.1.2 访问后台文章管理（manageArticle.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

2.3.1.3 访问添加文章页面（addArticle.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

2.3.1.4 访问友情链接管理主页（manageLink.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

2.3.1.5 访问编辑文章页面（editArticle.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

2.3.1.6 访问编辑友情链接页面（editLink.html）时，系统会检测登录状态，如果用户未登录，将自动跳转至登录页面。且跳转前，用户将无法得到后台数据。

### 2.3.2 数据库信息的保护

2.3.2.1 使用 hibernate 封装了 JDBC 中的实体类“article”（文章），使查询过程不需要 sql 语句字符串，从而在很大程度上避免了 sql 注入。

2.3.2.2 使用 hibernate 封装了 JDBC 中的实体类“link”（友情链接），使查询过程不需要 sql 语句字符串，从而在很大程度上避免了 sql 注入。

## 3 运行环境

### 3.1 硬设备

列出为运行本软件所要求的硬设备的最小配置，如：

- a. CPU 与内存：  
服务器：INTEL 2.0Ghz 双核，内存 4G，硬盘空闲空间 500G（建议。  
CPU 和内存要求随访问量增加而适当增加，硬盘视数据库数据量而定）  
客户机：INTEL 2.0Ghz 单核，内存 1G，硬盘空闲空间 128M。（建议。  
能运行最新 chrome、火狐等浏览器即可）
- b. 所要求的带宽：  
服务器：运行最低要求 2M，支持数位访客访问。带宽建议随访问量增加而增加，建议 50M 以上。
- c. 外部系统访问页面：需要联机；内部访问页面：可脱机。3.2 支持软件说明为运行本软件所需要的支持软件，如：
  - a. 操作系统：  
WINDOWS 7 -32 位或 64 位；  
Windows8-32 位或 64 位；  
Windows10-32 位或 64 位；（服务器及客户机）
  - b. JDK-1.8.0 或以上版本（服务器）。
  - c. 数据库 mysql5.7 或以上（服务器）。
  - d. Tomcat7.0 或以上版本（服务器）。
  - e. 浏览器：chrome，firefox，360chrome。（客户机）

### 3.3 数据结构

列出为支持本软件的运行所需要的数据库或数据文卷。

## 4 使用过程

在本章，首先用图表的形式说明软件的功能同系统的输入源机构、输出接

收机构之间的关系。

## 4.1 安装与初始化

一步一步地说明为使用本软件而需进行的安装与初始化过程，包括程序的存储形式、安装与初始化过程中的全部操作命令、系统对这些命令的反应与答复。表征安装工作完成的测试实例等。如果有的话，还应说明安装过程中所需用到的专用软件。

特别说明：本软件需要服务器已经安装了 tomcat。

- a. 将 blog.war 包复制到 tomcat/webapp 目录下
- b. 修改 tomcat/conf/server.xml 文件，在 host 节点中增加如下五行

`<Context path="/" docBase="relaxblog" debug="0" privileged="true"/>`

path 是虚拟路径；

docBase 是应用程序的物理路径；

workDir 是这个应用的工作目录，存放运行时生成的与这个应用相关的文件；

debug 则是设定 debug level, 0 表示提供最少的信息, 9 表示提供最多的信息

- c. 进入 tomcat/bin, 点击 startup.bat 启动 tomcat 服务器
- d. 打开浏览器，输入 127.0.0.1:8000/blog/index.html 或者 localhost:8000/blog 访问
- e. 点击界面上方管理按钮可以进行后台的登陆功能。

## 4.2 输入

### 4.2.1 输入数据的现实背景

说明输入数据的现实背景，主要是

- a. 情况——发表文章，编辑文章，添加管理友情链接；
- b. 情况出现的频度——周期性的；
- c. 情况来源——一个人博客主；
- d. 输入媒体——鼠标键盘；
- e. 限制——出于安全、保密考虑而对访问这些输入数据所加的限制，要求只有管理员可以进行数据库的相应操作；
- f. 质量管理——对输入数据合理性的检验以及当输入数据有错误时应采取的措施，如建立出错情况的记录等，对每次输入数据的格式都进行了检验；
- g. 支配——发表文章时可以根据用户选择，采取保存草稿或者发表文章。

## 4.2.2 输入格式

编号	功能	需要的输入数据描述
1	输入用户账号	键盘输入字符串,长度不超过 10 字节
2	输入用户密码	键盘输入字符串,长度不超过 20 字节,不少于 6 字节,只允许英文\数字\标点符号
3	显示文章总数量	登录并访问主页面
4	显示当前日期时间	登录并访问主页面
5	显示友情链接数量	登录并访问主页面
6	显示最新文章	登录并访问主页面
7	输入文章标题	键盘输入字符串,长度不超过 50 字节,不为空
8	输入文章标签	标签:键盘输入字符串,长度不超过 10 字节,可为空
9	文章编辑	键盘输入字符串,通过相关编辑按钮修改字体格式等属性。最长不超过 10KB, 可为空
10	存为草稿	单击相应按钮
11	显示文章信息列表	登录并访问“文章管理”页面则自动显示
12	修改文章	点击对应文章的“修改“按钮
13	删除文章	点击对应文章的“删除“按钮
14	文章预览	点击对应文章的“阅读”按钮
15	添加新连接	登录并访问“友情链接管理”, 点击“添加”按钮,输入新连接名称、连接地址,均为字符串,名称不超过 100 字节,地址不超过 200 字节,,描述 300 字节,前两者不能为空

16	链接编辑	登录并访问“友情链接管理”，点击“编辑”按钮，输入新连接名称、连接地址，均为字符串，名称不超过 100 字节，地址不超过 200 字节，，描述 300 字节，前两者不能为空
17	链接删除	登录并访问“友情链接管理”，点击对应的友情链接后的“删除”按钮。
18	显示文章列表	访问博客主页，文章列表自动显示
19	显示文章摘要	访问博客主页，鼠标指向该文章
20	显示文章标签	访问博客主页，文章标签自动显示
21	查看友情链接	访问博客主页,点击"友情链接"按钮
22	显示文章内容	访问博客主页,选定文章鼠标单击查看
23	显示最后编辑时间	查看指定文章时,文章的最后编辑时间自动显示在文章标题下方
24	添加留言	查看指定文章时,在文章最后的留言框中输入留言者名称,联系方式,留言内容,点确定提交。名称不超过 20 字节，不能为空；联系方式不超过 30 字节，可以为空；留言内容不超过 300 字节，不能为空；均以字符串形式输入
25	显示留言	查看指定文章时，若存在留言，则留言自动显示在文章下方

#### 4.2.3 输入举例

登录

序号	输入信息
----	------

1	空白
2	合理长度，正确用户名
3	合理长度，错误用户名
4	超长长度
5	已登录
6	未登陆

发表文章

1	空白标题+合法标签及内容，发表文章
2	标题“软件架构”+标签“软件架构”+内容“abcd”发表文章
3	标题“abcd”+超长标签+内容“abcd”发表文章
4	超长标题+标签“1234”+内容“你好”发表文章
5	标题“abcd”+标签“abcd”+超长内容，发表文章
6	未登陆
7	标题“abcd”+标签“abcd”+内容“abcd”，发表文章
8	标题“abcd”+标签“abcd”+内容粗体，发表文章
9	标题“abcd”+标签“abcd”+内容下划线，发表文章
10	复制图片到内容，发表文章
11	空白标题+合法标签及内容，存为草稿
12	合法标题+标签“软件架构”+内容“abcd”存为草稿
13	标题“abcd”+超长标签+内容“abcd”存为草稿
14	超长标题+标签“1234”+内容“你好”存为草稿
15	标题“abcd”+标签“abcd”+超长内容，存为草稿
16	未登陆
17	标题“abcd”+标签“abcd”+内容“abcd”，存为草稿
18	标题“abcd”+标签“abcd”+内容粗体，存为草稿
19	标题“abcd”+标签“abcd”+内容下划

	线，存为草稿
20	复制图片到内容，存为草稿

修改文章

1	空白标题+合法标签及内容，发表文章
2	标题“软件架构”+标签“软件架构”+内容“abcd”发表文章
3	标题“abcd”+超长标签+内容“abcd”发表文章
4	超长标题+标签“1234”+内容“你好”发表文章
5	标题“abcd”+标签“abcd”+超长内容，发表文章
6	未登陆
7	标题“abcd”+标签“abcd”+内容“abcd”，发表文章
8	标题“abcd”+标签“abcd”+内容粗体，发表文章
9	标题“abcd”+标签“abcd”+内容下划线，发表文章
10	复制图片到内容，发表文章
11	空白标题+合法标签及内容，存为草稿
12	合法标题+标签“软件架构”+内容“abcd”存为草稿
13	标题“abcd”+超长标签+内容“abcd”存为草稿
14	超长标题+标签“1234”+内容“你好”存为草稿
15	标题“abcd”+标签“abcd”+超长内容，存为草稿
16	未登陆
17	标题“abcd”+标签“abcd”+内容“abcd”，存为草稿
18	标题“abcd”+标签“abcd”+内容粗体，存为草稿
19	标题“abcd”+标签“abcd”+内容下划线，存为草稿
20	复制图片到内容，存为草稿

管理友情链接

1	Name 为空
---	---------



2	Name= 'github', URL 为空
3	Name= 'github', URL= 'github.com', 描述为空
4	都为空
5	Name 超长
6	Name= 'github', URL 超长,描述为空
7	Name= 'github', URL= 'github.com', 描述超长

## 4.3 输出对每项输出作出说明

### 4.3.1 输出数据的现实背景

说明输出数据的现实背景，主要是：

- 使用——数据提供给游客以及博客拥有者，方便管理员进行博客管理和阅读；
- 使用频度——不定期使用；
- 媒体——显示屏，浏览器，
- 支配——发表文章时，根据用户选择文章状态的管理。

### 4.3.2 输出格式

1	输入用户账号	符合格式则不显示输出，不符合格式则确认时报错,显示相应错误
2	输入用户密码	符合格式则不显示输出，不符合格式则确认时报错,显示相应错误
3	显示文章总数量	发表了 X 篇文章
4	显示当前日期时间	日期为 XXXX/X/X
5	显示友情链接数量	友链了 X 个网站
6	显示最新文章	最新文章:各文章,每文章一行
7	输入文章标题	符合格式则不显示输出，不符合格式则确认时报错,显示相应错误

8	输入文章标签	符合格式则不显示输出,不符合格式则确认时报错,显示相应错误
9	文章编辑	符合要求后上传至数据库, 页面提示保存成功, 否则提示相应错误
10	存为草稿	将编辑框内内容上传至服务器, 上传成功则提示“已保存”
11	显示文章信息列表	显示表格, 列信息为: 文章标题-ID-发布时间-标签-状态-操作
12	修改文章	将服务器上的文章信息显示在编辑框中, 等待编辑, 完成后检查长度等格式, 符合要求后更新到服务器上
13	删除文章	若确认则显示“删除成功”, 同时删除服务器端文章内容。取消则无显示
14	文章预览	显示文章内容
15	添加新连接	若接收到成功信息, 页面显示“添加成功”, 否则显示“添加失败, 请刷新后重试”
16	链接编辑	若接收到成功信息, 页面显示“编辑成功”, 否则超时时显示“编辑失败, 请刷新后重试”
17	链接删除	若接收到成功信息, 页面显示“删除成功”, 否则超时时显示“删除失败, 请刷新后重试”
18	显示文章列表	在页面上显示各文章的标题, 发布时间, 标签信息
19	显示文章摘要	显示指定文章的前 20 字。
20	显示文章标签	显示文章的标签信息
21	查看友情链接	显示友情链接信息, 包括名称和链接, 若无友情链

		接则显示"暂无友情链接"
22	显示文章内容	在浏览器中显示文章内容
23	显示最后编辑时间	在浏览器中显示文章最后编辑时间
24	添加留言	接收到服务器成功信息则显示“发表成功”，超时未接收到信息则提示“发表失败，请刷新重试”
25	显示留言	显示留言信息，若无留言则显示空白

### 4.3.3 输出举例

#### 登录

序号	输出结果
1	提示长度错误
2	检测密码正确与否
3	提示登录信息错误
4	提示长度错误

#### 发表文章

编号	输出结果
1	提示标题长度错误
2	提示发表成功
3	提示标签超长
4	提示标题长度错误
5	提示内容超长
6	跳转登录界面
7	提示发表成功
8	提示发表成功
9	提示发表成功
10	提示发表成功
11	提示标题长度错误
12	提示发表成功
13	提示标签超长
14	提示标题长度错误

15	提示内容超长
16	跳转登录界面
17	提示发表成功
18	提示发表成功
19	提示发表成功
20	提示发表成功

修改文章

编号	输出结果
1	提示标题长度错误
2	提示发表成功
3	提示标签超长
4	提示标题长度错误
5	提示内容超长
6	跳转登录界面
7	提示发表成功
8	提示发表成功
9	提示发表成功
10	提示发表成功
11	提示标题长度错误
12	提示发表成功
13	提示标签超长
14	提示标题长度错误
15	提示内容超长
16	跳转登录界面
17	提示发表成功
18	提示发表成功
19	提示发表成功
20	提示发表成功

管理友情链接

1	提示名称长度错误
2	提示 URL 长度错误
3	提交成功
4	报告任意长度错误
5	提示名称超长
6	提示 URL 超长
7	提示描述超长
8	提示登录信息错误
9	提示名称长度错误
10	提示 URL 长度错误

## 4.4 文卷查询

数据库建表

```
CREATE TABLE `articles` (  
  `art_id` int(11) NOT NULL AUTO_INCREMENT,  
  `title` varchar(255) NOT NULL,  
  `created_date` date NOT NULL,  
  `modified_date` date DEFAULT NULL,  
  `content` varchar(10000) DEFAULT NULL,  
  `categories` varchar(50) DEFAULT NULL,  
  `comments_num` int(11) DEFAULT '0',  
  `status` int(11) DEFAULT '0',  
  PRIMARY KEY (`art_id`)  
)  
  
CREATE TABLE `comments` (  
  `cmt_id` int(11) NOT NULL AUTO_INCREMENT,  
  `created_date` date NOT NULL,  
  `art_id` int(11) NOT NULL,  
  `user_id` int(11) NOT NULL,  
  `content` varchar(200) NOT NULL,  
  PRIMARY KEY (`cmt_id`)  
)  
  
CREATE TABLE `links` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `url` varchar(200) DEFAULT NULL,  
  `name` varchar(100) DEFAULT NULL,  
  `description` varchar(300) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
)  
  
CREATE TABLE `users` (  
  `user_id` int(11) NOT NULL,  
  `username` varchar(64) NOT NULL,  
  `password` varchar(64) NOT NULL,  
  `email` varchar(100) DEFAULT NULL,  
  `home_url` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`user_id`),  
  UNIQUE KEY `user_id` (`user_id`),  
  UNIQUE KEY `username` (`username`)
```

)

## 4.5 出错处理和恢复

对于网页端的输入，脚本会提供错误处理机制，检测输入值是否符合规范。对于超出规范范围的输出，网页将会报错；对于异常输出未被检测的情况，输入值传至数据库时数据库会报错或返回异常，该异常将返回给网页端 `fail` 信息，网页仍然会弹错提示。

对于意外情况导致程序崩溃或异常运行，则需要用户进行以下操作：

1. 对于程序崩溃无法打开页面的情况，关闭 `tomcat` 服务器并重新启动。
2. 对于初次运行时程序无法正常运行的情况，请重新配置 `tomcat` 和 `mysql`，确保信息链接正常。
3. 对于数据异常，如文章数量异常或 `id` 异常等情况，请保存文章等信息并清空数据库，利用建表文件重新建立数据库表项，然后再次运行 `tomcat`。