



院 系 数据科学与计算机学院

学 号 15352013 姓 名 蔡泽杰

学 号 15352005 姓 名 蔡景韬

学 号 15352020 姓 名 曾华浪

学 号 15352023 姓 名 曾宇航

移动 Web 应用设计

大作业实验报告

【实验目的】

通过建立一个网站来综合应用本课程学习的知识。

【大作业分工】

组员分工可以按照模块（JSP 程序）分工，也可以按照软件开发过程分工，例如：系统设计、前台(html+CSS)、后台(JSP)、数据库管理、测试、安装、写文档。

【大作业评分标准】

- (1) 有无数据库增改删操作
- (2) 有无文件上传功能
- (3) 核心功能是否实现以及功能是否完整性
- (4) 项目的整体难度
- (5) 画面设计的美观
- (6) 实验报告编写
- (7) 提交时间

同一个组所有成员统一打分。所有 2~4 人的组，打分时不看人数，同等对待。对于单人 1 组的要有充足的理由（在实验体会中进行说明），否则要扣分。4 人以上一组的要求更多功能。

【实验环境】

tomcat, JDK 和 mysql。注意要允许外部访问应用程序需要[设置 Windows 防火墙](#)。

【实验要求】

- (1) 最好有数据库增改删操作，上传文件操作。例如：博客上传照片，增加修改删除博客。
- (2) 最好有管理员后台的操作功能，例如，浏览所有用户的博客内容，删除用户非法博客，某用户禁言。
- (3) 前后台都直接采用底层编程，**不要使用任何框架**。
- (4) 确定需求后，画出用例图（顶层）、类图、E-R 图，然后再编码。采用快速迭代的方法，先做一个小型完整的系统，有时间再不断增加功能或改进设计。
- (5) 整个项目完成后要在教学服务器上安装，安装说明由另一个文件给出。



[下面各项为要填写的内容]

【小组号】

15352005

【项目名称】

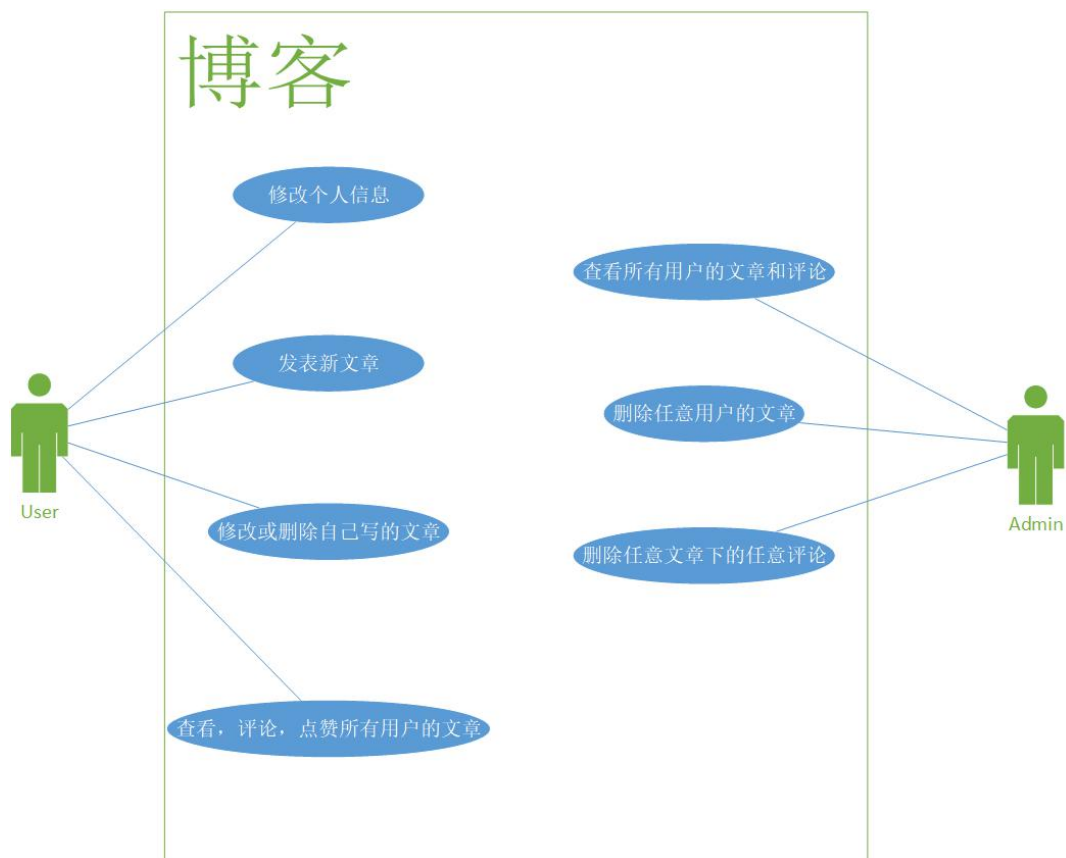
学习汇-博客

【项目描述】

建立一个功能较为齐全的博客网站，所有注册使用的用户都能够分享其自己的文章，也可查看他人分享的文章，不仅如此，用户还能够对任意文章发表评论，同时还能够为自己喜爱的文章进行点赞。而用户所发表的文章能够记录访客数量，增强用户体验。

【系统设计】

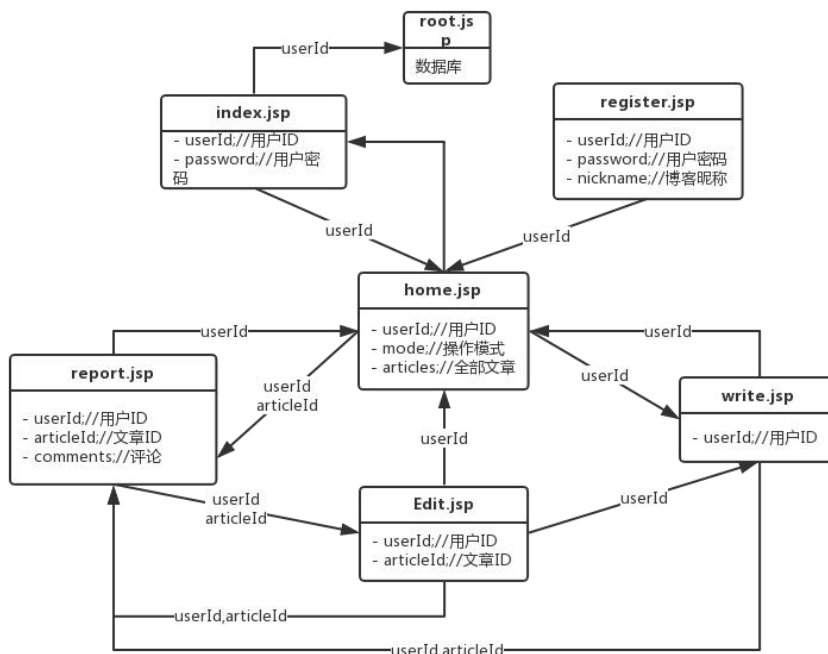
1、顶层用例图





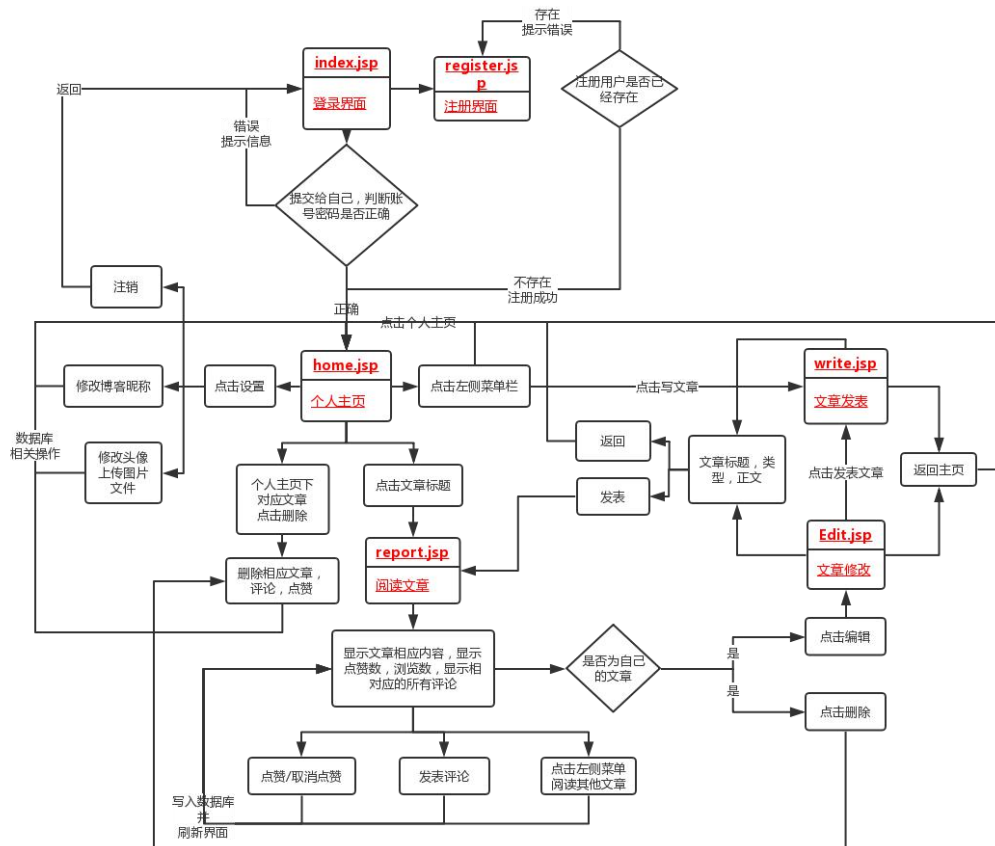
2、类图

网站整体结构：



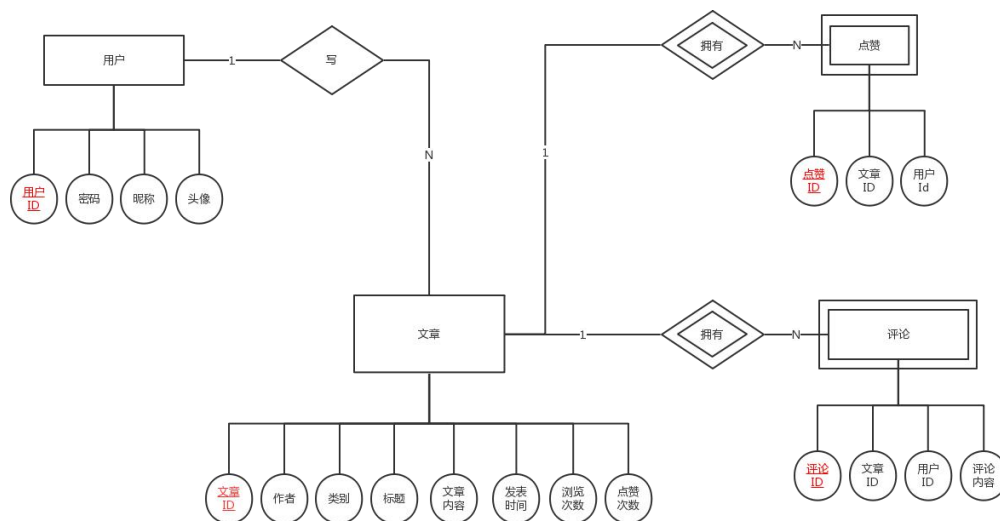
线上的值为传递所需要的参数。在对应各自的 jsp 中，关于数据库的操作处理，建立一个额外没有 html 视图的 jsp 文件进行处理，处理完成后执行强制跳转，跳转到另外界面后回到原有的界面，完成数据库操作以及刷新界面。

详细结构图解：





3、E-R 图



4、数据库设计（表格字段都为 UTF-8 编码）

表名：user			
字段名	中文名	类型	说明
UserId	用户名	Varchar	userId 是表格的主键；用户登录所用
Password	密码	Varchar	用户登录密码
Nickname	博客名	Varchar	用户的博客名称
Picture	头像状态	Varchar	存在为 1，否则为空

表名：article			
字段名	中文名	类型	说明
ArticleId	文章 ID	Int	1、 表格主键； 2、 自动增长
UserId	用户名	Varchar	1、 外键约束, 参照 user 表格的 UserId 2、 删除用户的时候级联删除 3、 文章是谁写的
Belong	文章类型	Varchar	文章属于哪种类型
Title	文章标题	Text	文章的文章标题
Content	文章内容	Text	文章的内容
Time	修改时间	Varchar	文章的最后编辑时间
Count	访客数	Int	文章的访客人数
Like_Count	点赞数	Int	文章的点赞人数

表名：comment			
字段名	中文名	类型	说明
CommentId	评论 ID	Int	1、 CommentId 是表格的主键； 2、 自动增长
ArticleId	文章 ID	Int	1、 外键约束， 参照 article 表格的 articleId 2、 删除文章时级联删除 3、 哪篇文章的评论



UserId	用户名	Varchar	1、 外键约束，参照 user 表格的 UserId 2、 删除用户的时候级联删除 3、 哪个用户名写的评论
Content	评论内容	Text	评论内容

表名: love			
字段名	中文名	类型	说明
Likeld	点赞 ID	Int	1、 Likeld 是表格的主键; 2、 自动增长
ArticleId	文章 ID	Int	1、 外键约束，参照 article 表格的 articleId 2、 删除文章时级联删除 3、 哪篇文章的点赞
UserId	用户名	Varchar	1、 外键约束，参照 user 表格的 UserId 2、 删除用户的时候级联删除 3、 哪个用户名点的赞

【运行截图】

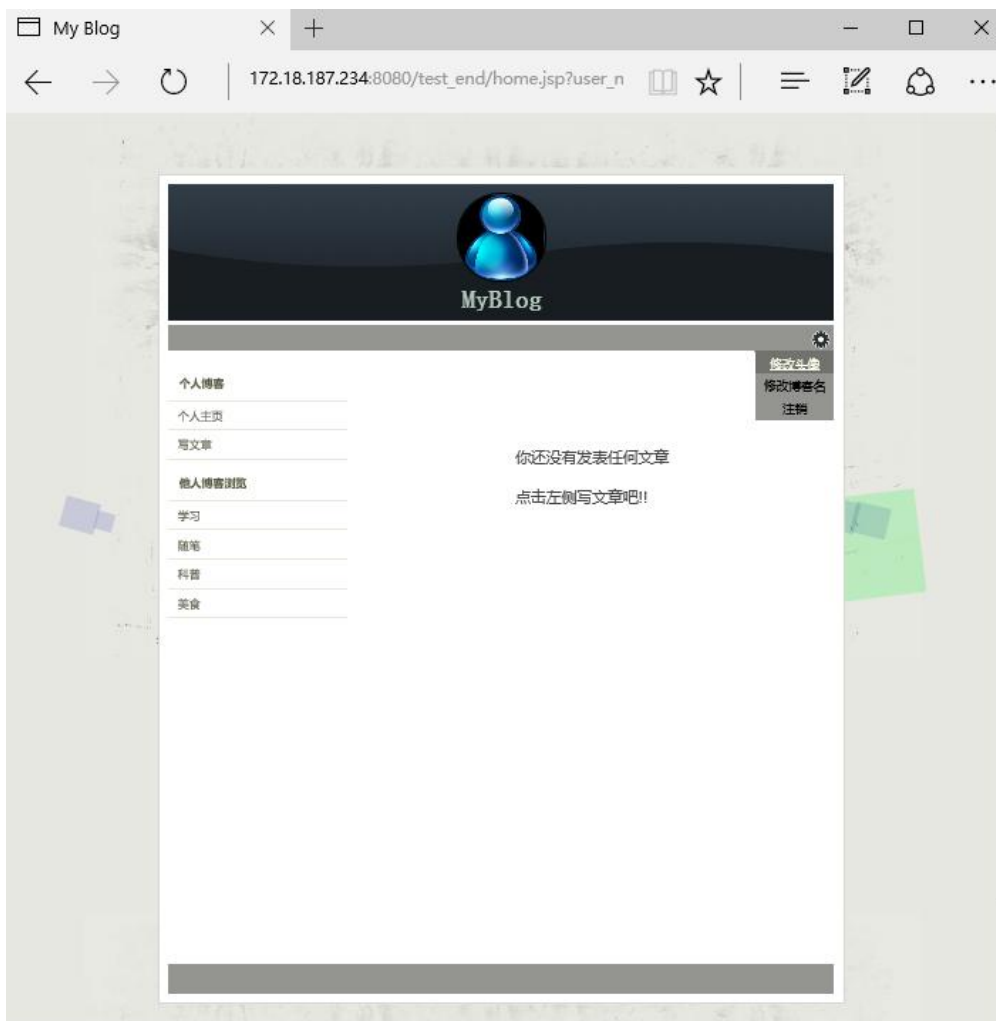
1.登录:



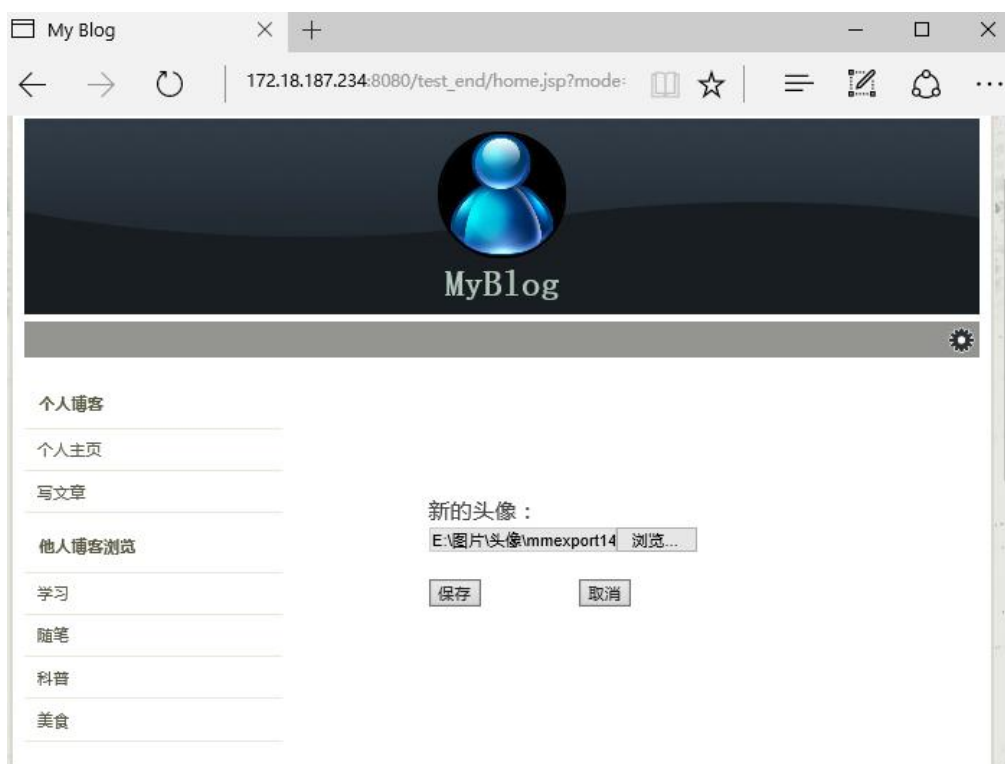
输入已经注册过的用户名以及密码即可登录自己相应的博客。如果没有注册过，可以点击注册按钮跳转到注册页面，按照要求注册完成后会和登录一样跳转到个人主页：

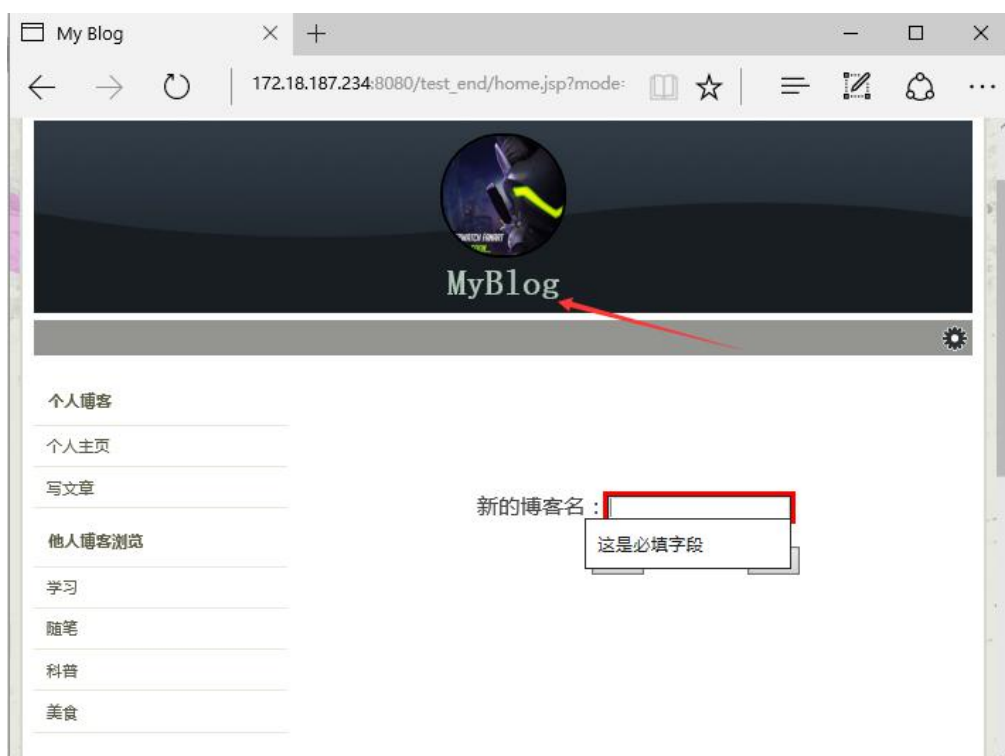


2.个人主页:

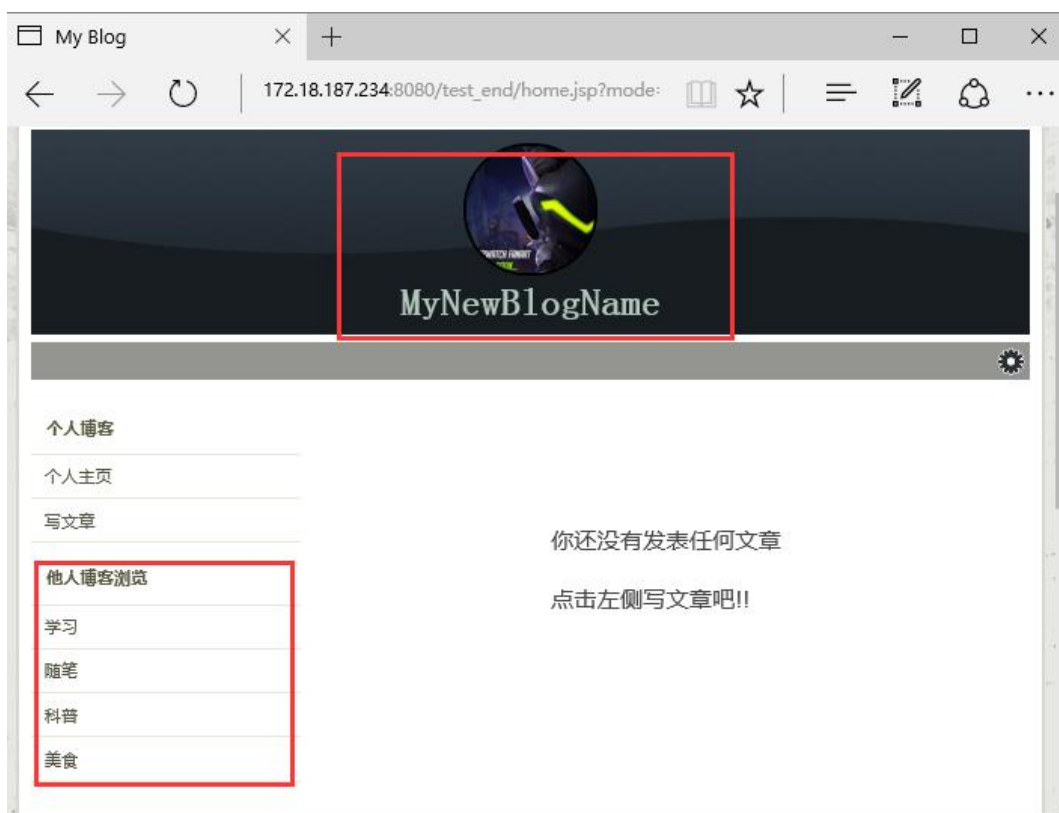


光标悬浮在设置图标上显示出菜单，可以进行个人资料的修改：

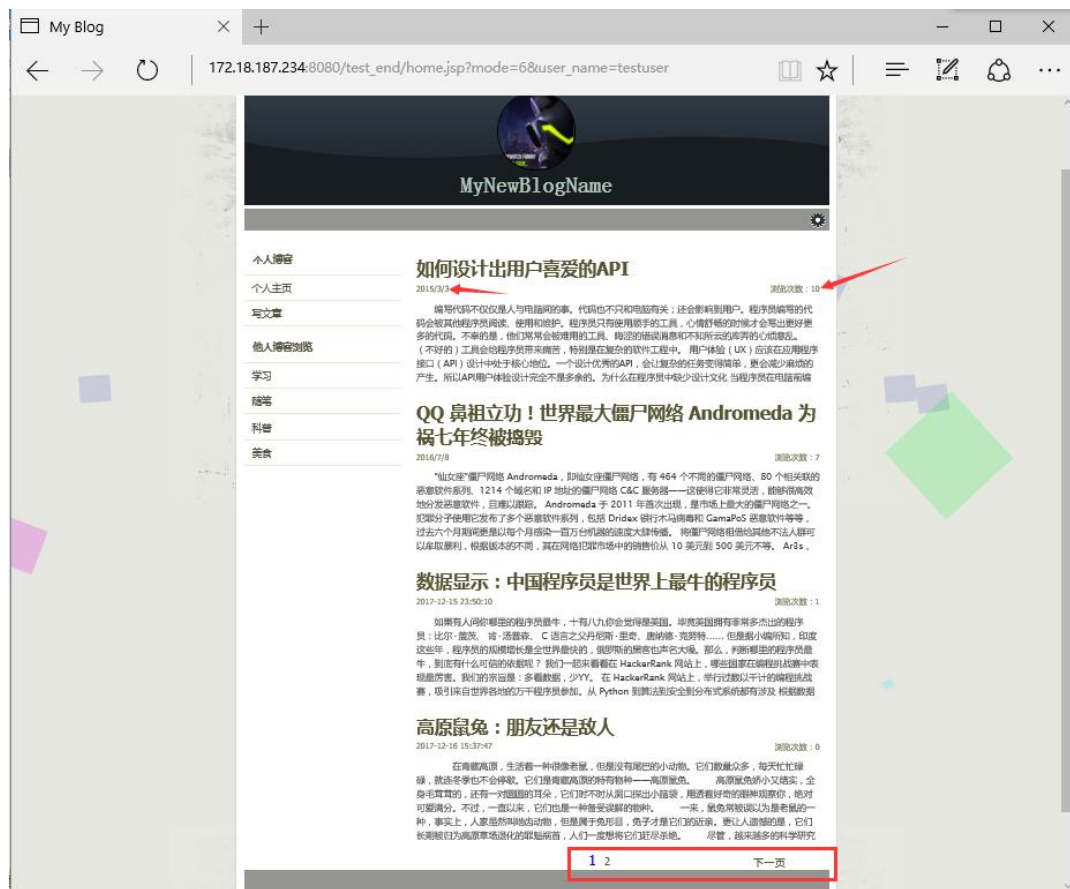




若点击保存会要求必须输入相应字段，点击取消则返回原有的个人主页：

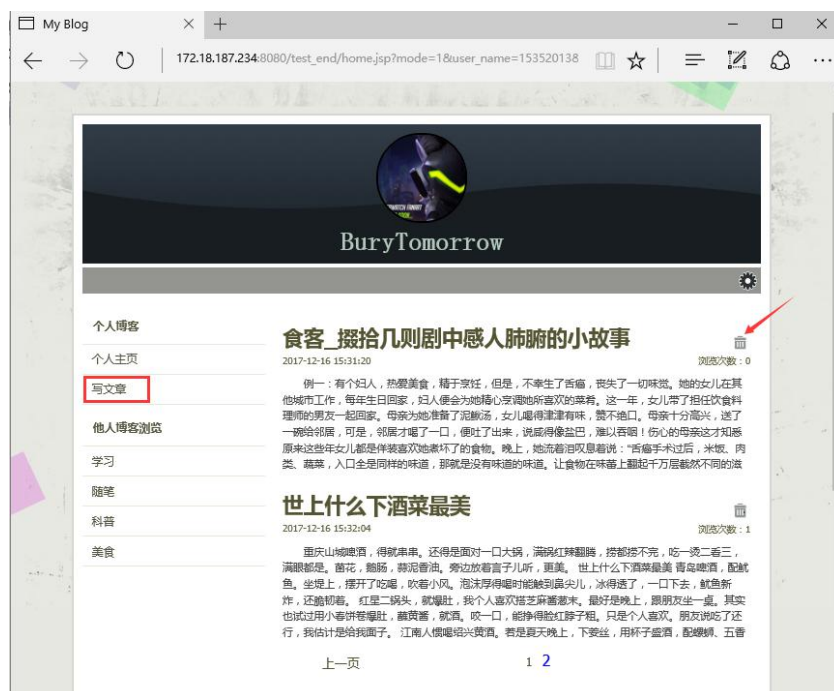


(个人头像及博客昵称修改成功)左侧菜单栏中的他人博客浏览，能够浏览其他人的博客，并按照所选的类型进行分类，如科普类：



进行分页管理,每篇文章均显示其最后修改时间以及浏览次数(如果登录用户是原作者则显示删除按钮,提供删除该文章的操作),分页操作可以点击上一页或下一页进行跳转,也可以点击数字直接跳转到某一页的内容,如果没有上一页则不显示'上一页',同样如果没有下一页则不显示'下一页'。

为了方便操作,接下来切换用户进行展示:

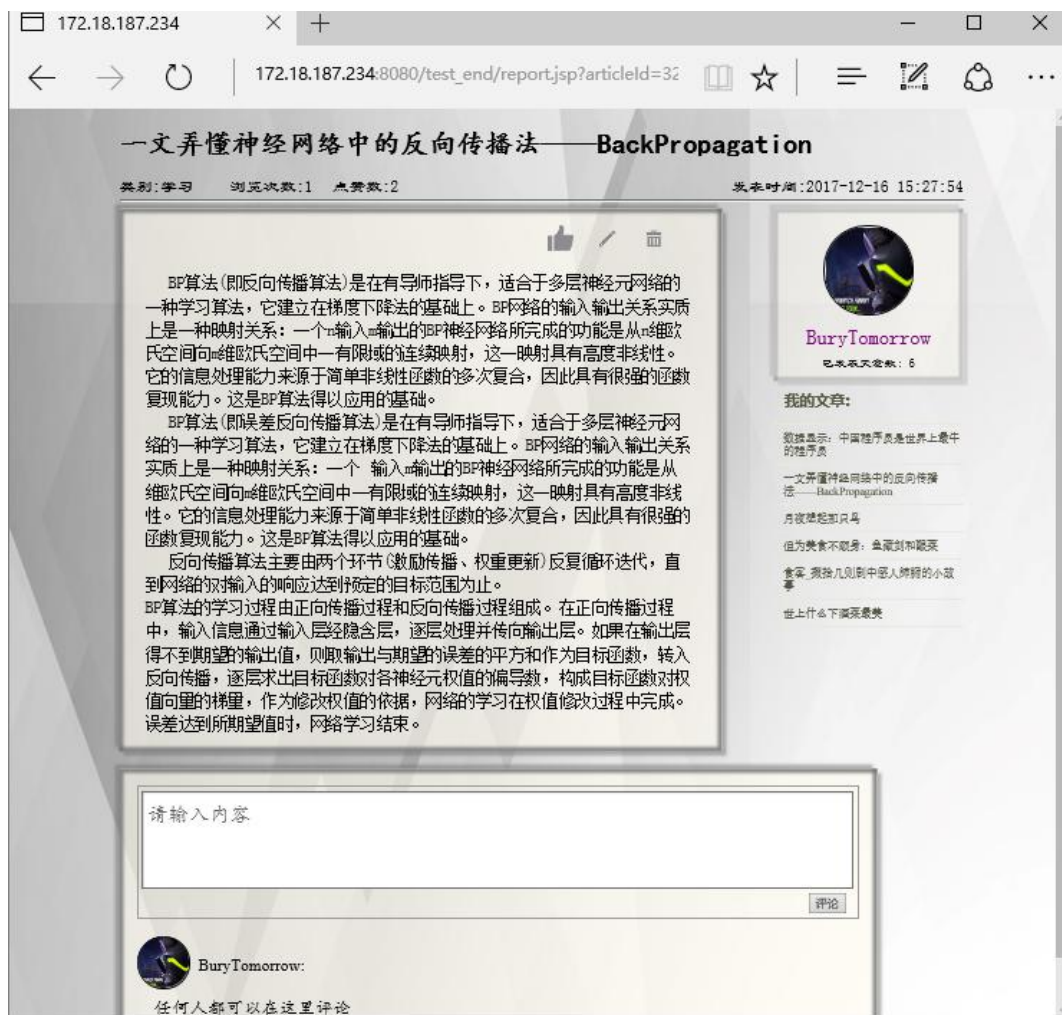


如图中所示如果是登录用户写的文章则可以删除操作。

点击任意文章的标题,阅读文章:



3. 阅读文章界面：



在上方：





点击点赞图片能够相应的改变图标(已赞以及未赞,可以与上两图对比),相对应的,文章上方的点赞数会做相应的更改,上方的浏览次数只会统计非作者浏览的次数。右方显示现登录的用户,以及登录用户的博客昵称,点击博客昵称可以返回到个人主页。在登录用户信息的下方显示自己写的文章,点击可以直接进入自己的相对应的文章。

点赞按钮旁边的两个编辑和删除图标:



这两个图标只有登录用户是作者的时候才会显示,比如下方不是作者的时候显示为:



可以注意到自己编写的文章中并没有这篇文章,所以不能够进行修改或删除操作。同样点击删除按钮会删除文章并且返回到个人主页。

在显示文章界面的下方为评论区:显示该文章下各个评论用户的头像、博客昵称以及评论内容:





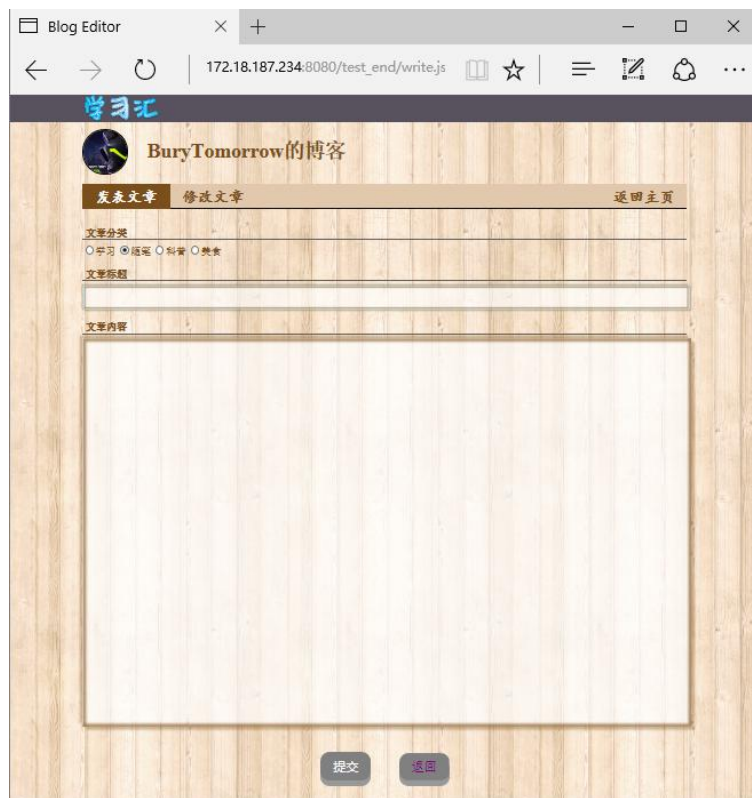
在文章显示页面中点击编辑的图标可以进入编辑该文章的界面：

4.文件编辑界面：



点击上方的返回主页则返回到个人主页，点击下方的返回则返回到上一层的文章显示界面，点击提交保存自己修改的内容，并回到阅读文章界面显示相应的文章。

点击上方的'发表文章'或者在个人主页中点击'写文章'按钮，可以进入发表新文章的界面：





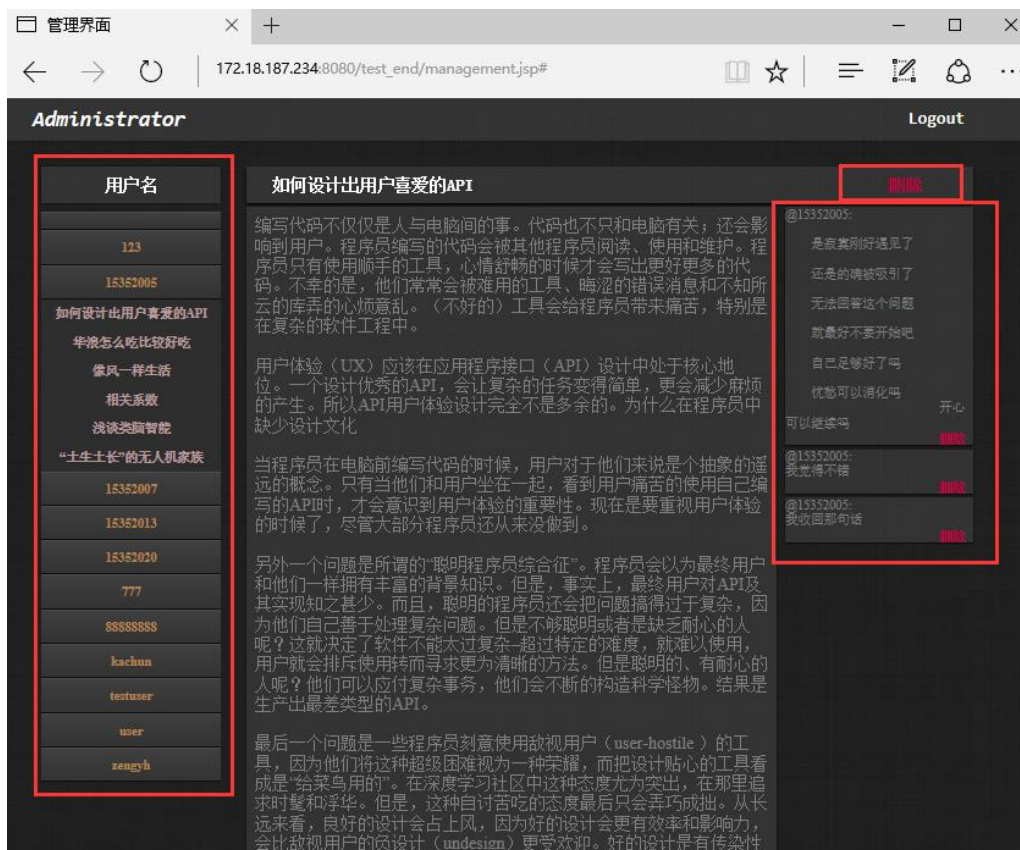
界面内容与编辑文章界面相同，但操作不同，这里不能点击修改文章，点击返回主页或者点击下方的返回都是回到个人主页，点击提交则保存新的文章并跳转到文章显示界面，显示刚编写的新文章。

5.管理员登录：



在登录界面中使用管理员账号能够跳转到管理员登录界面：

登录后进入：



点击左侧栏的用户名能够显示该用户编写的所有文章，点击文章可以显示对应的文章内容以及其相关评论，管理员有权限删除任意的文章以及任意的评论。



【实验体会】

写出实验过程中遇到的问题，解决方法和实验体会，每位同学单独写。

[同学 1 姓名 蔡景韬]

这次合作中我主要负责登录注册界面、管理员界面与数据库管理部分，除了基本的界面设计外，我觉得感受最深的便是对 java script 更加理解与能较熟练的使用 js，以及 jsp 和 js 之间的变量传输。

登录注册界面

问题描述	解决方法
设置 Hint	1、 使用 HTML5 的 placeholder 属性; 2、 但是由于 HTML5 给定的 Hint 只有在点击的时候才会消失，于是设置点击后背景色与原 Hint 颜色相同，可以感觉 Hint 已经消失
当用户名不存在或密码不匹配时修改 Hint 内容	1、 在链接了 css 后，还是可以继续使用 style 标签 2、 另外使用 id 选择器，保证重新增加的 style 能够生效（优先级最大） 3、 在 jsp 代码的 if 判断中，直接插入 style 标签即可 <pre>if (rs.getRow() == 0) { userIdHint = true; } <style type="text/css"> #userId::webkit-input-placeholder{color: red;font-weight: 1000;} #userId::moz-placeholder{color: red;font-weight: 1000;} #userId::ms-placeholder{color: red;font-weight: 1000;} #userId::ms-input-placeholder{color: red;font-weight: 1000;} </style> <%</pre>
获得查询结果的行数，以判断用户名是否存在	1、 先使用 ResultSet.last()跳至 ResultSet 的结尾，使用 getRow()即可获得查询结果的行数 2、 如果需要重新使用，只要用 ResultSet.first()即可，值得注意的是，如果想遍历所有的 result，且使用 ResultSet.next()应该使用 do while(ResultSet.next())。

管理界面

问题描述	解决方法
点击用户名，显示其写过的文章； 点击文章名，右侧显示文章内容及其评论	1、 基本思路：通过数据库拿到所有的用户名和写过的文章以及内容，并平铺到界面上，设置其 display 属性为 none。最后通过点击事件设置 display 属性 2、 使用 document.getElementsByTagName("li")可以获得所有的元素 3、 使用 element.nextElementSibling 可以获得 element 的下一个兄弟节点对象 4、 使用 element.onclick = function(){} 可以为 element 设置点击事件
循环绑定点击事件时的作用域问题 <pre>for (var i=0; i<ali.length ; i++) { all[i].onclick = function () { console.log(i) } }</pre> 此时输出的都=ali.length	1、 原因：当点击函数 function 作用域中找不到 i 变量，此时解析引擎会寻找父级作用域，此时父级作用域中有 i，且 for 循环捆绑点击事件结束后 i 已经赋值为 ali.length，所以输出的都是 ali.length 2、 解决：使用包装函数，并传入参数即可解决作用域的问题（经过了漫长的 debug 才发现==） <pre>for (var i=0; i<ali.length ; i++) { var element = ali[i].nextElementSibling ; // console.log(element); (function(x,ul){ ali[x].onclick = function(){ if (ul.style.display=="none") ul.style.display="block"; else ul.style.display="none"; // console.log(ul); } })(i,element); }</pre>
使用 document.getElementBy***()与 document.getElementsByTagName***() 函数相关问题	1、 getElementByClassName 是通过 class 寻找标签对象，返回的是对象数组，函数名为**Elements** 2、 getElementsByTagName 是通过标签名寻找标签对象（如，），返回的是对象数组，函数名为**Elements** 3、 getElementById 是通过 ID 寻找标签对象，返回的是单个对象（如果多个 id 相同，返回的是第一个匹配的对象），函数名为**Element**
使用 Js 跳转页面	window.location.href (url) 将当前的页面跳转到 url 所在的界面
标签对象相关属性的获得	1、 **.style.** 可以获得 style 样式中的属性



	<p>(如, element.style.display 可以获得 element 当前的 display 属性取值)</p> <p>2、 **.* 可以获得其他属性取值, 特别的, 可以自己创造属性 (如, element.id; element.class; element.创造的属性名)</p>
JS 代码运行顺序的问题	<p>1、 JS 是按照代码块来进行编译和执行的, 代码块之间是独立的 (即某个代码块报错不会影响另外代码块的执行), 而代码块间的变量是共享的。</p> <p>2、 而 JS 标签的执行顺序则是根据 HTML 文档流的执行顺序</p> <p>3、 需要在页面元素渲染前执行的 js 代码应该放在<body>前</p> <p>4、 而需要在页面元素加载完后的 js 放在</body>元素后</p> <p>5、 body 标签的 onload 事件是在最后执行的。</p> <p>6、 如果想保证渲染完毕再执行 JS 代码 (即 document.getElementById 时不会取到 null 值), 则可以在代码前加 window.onload = function(){}</p>
JS 和 JSP 变量的传输	<p>1、 JS 代码运行在浏览器端, 而 JSP 代码运行在服务器端, 之间不能直接传输</p> <p>2、 JS 变量传输给 JSP 有两种方法:</p> <ul style="list-style-type: none">a、放入一个隐藏区域, 隐藏提交b、使用 ajax <p>3、 JSP 变量传输给 JS, 与 HTML 使用 JSP 中变量一样使用 <%= %></p> <p>(如, JS 获得 JSP 中的数组变量)</p> <pre>var userList = new Array(); var titleList = new Array(); <% for (int i = 0; i < userList.length; i++) { %> userList[<%=i%>]=<%=userList[i]%>; titleList[<%=i%>]=new Array(); <% for (int j = 0; j < titleList[i].length; j++) { %> titleList[<%=i%>][<%=j%>]=<%=titleList[i][j]%>; <% } %> <% } %></pre>

[同学 2 姓名 蔡泽杰]

负责个人主页模块以及协助他人页面设计:

1.界面显示问题, 因为在同一个界面中有许多不同的操作, 比如说显示不同类别的文章, 修改个人信息, 而如果分成多个页面则过于冗余:

添加一个 mode 参数, 页面的点击都是提交给自己来负责处理, 然后根据这个参数来对 html 正文执行不同的输出, 主要是利用 JSP 的优势: java 语言与 html 共同处理:

```
<div class="main_right">
<% if(mode.equals("2")){ %>
<% }else if(mode.equals("3")){ %>
<% }else if(mode.equals("1")||mode.equals("4")||mode.equals("5")||mode.equals("6")||mode.equals("7"))
{ %>
<%}%>
</div>
```

在不同模式下输出不同的 html 即可达到相应的页面模式转换效果

2.显示文章列表时, 怎么做到如果是自己的文章则显示相应的删除按钮, 允许执行删除操作

在对输出语句进行操作的时候, 额外多加一个判断, 判断文章的作者是否为登录用户, 如果是则加入相应的图片语句:

```
if(rs.getString("userId").equals(user_name)){
articles.append("<a href='handler.jsp?mode=-2&articleId="
+rs.getString("articleId")+&user_name="+user_name
+"'><img src='images/delete.png'/></a>");
}
```

3.如何实现分页

从数据库筛选出相应的全部文章后, 每次只显示一定量的文章, 超过的文章则不进行输出处理

```
while(rs.next()){
if(cnt>pgno*pgcnt&&cnt<(pgno+1)*pgcnt){
articles.append("<h1><a href='report.jsp?userId="+user_name+"&articleId="+rs.
getString("articleId")+"'>"+rs.getString("title")+ "</a>");
if(rs.getString("userId").equals(user_name)){
articles.append("</h1><div class='date'>"+rs.getString("Time")
}
cnt++;
}
```

关键是显示文章的下方如何处理上一页下一页以及对应的页码, 这个其实比较简单, 同样也是在 html 区域穿插进 java 语言进行判断输出:



```
<div id="page">
  <%if(pgno!=0){%>
  <a href="home.jsp?mode=<%=mode%>&user_name=<%=user_name%>&pgno=<%=pgno-1%>">上一页</a>
  <%}%><div><%
    for(int i=1;i<=num;i++){
      if(i==pgno+1){
        out.print("<span>"+i+"</span>");
      }else{
        out.print("<a href='home.jsp?mode="+mode+"&user_name="+user_name+"&pgno="+
        (i-1)+"'>"+i+"</a>");
      }
    }%>
  </div>
  <%if(pgno!=num-1&&num!=0){%>
  <a href="home.jsp?mode=<%=mode%>&user_name=<%=user_name%>&pgno=<%=pgno+1%>">下一页</a>
  <%}%>
</div>
```

判断是否还有上一页或者下一页再输出相应的元素，然后对于中间的页码，其中 num 代表总文章数/每页显示文章数 的上取整整数，如果是当前页码，则只是输出一个 span 元素，如果不是，则输出一个 a 元素让用户对应跳转到相应的页码显示。

4.个人头像如何给予刚注册用户一个默认头像：

可以在用户表当中再增加一列属性，用于判断该用户是否修改过头像，如果没有修改头像，则输出的头像 url 则使用一个默认的头像：

```
if (picture == null) {
    picture = "images/默认.jpg";
} else {
    picture = "files/" + user_name + ".jpg";
}
```

如果设置过头像，那么该列属性就不会为 NULL，那么就设置为自己的头像，文件上传管理当中，对于头像的管理则将所有用户上传的头像命名均改为其用户 ID.jpg，因为用户的 ID 是唯一的，所以如果用户多次更改头像，也不会保留上一次的头像文件在服务器中，避免的内存的占用，同时也方便管理。

5.体会：

这次的大作业中，在 JSP 操作中花了相当长的时间，因为需要实现较为复杂的逻辑操作，通过一次次的 debug 更加地了解了 JSP 语言的好处，也更能理解到两种语言同时使用的好处。在实验过程中，代码编写其实更重要的是 CSS 样式，一个页面的给人的感受好坏很大程度上取决于其页面的美观程度，所以在 CSS 上同样花了很多时间，去使用动画，使用这学期学过的所有内容，全都运用到本次的大作业中，同时也提高了自己的审美，能够独立设计出一个较好的页面。在编写好相应的页面之后，只需要慢慢添加 JSP 功能即可完成一个完整的页面。

[同学 3 姓名 曾华浪]

遇到的问题：

一开始发现在执行一个 sql 指令之中执行另一个 sql 语句，会导致运行错误。经过请教同学之后了解到每一次对数据库的操作必须要完成了才能完成下一个操作，将其分成两块就没有问题了。

ResultSet rs = stat.executeQuery(sql); 之后，并不代表着 stat 调用数据库已经完成，在接下来的语句中，如果还想要查询数据库，则必须要新设一个 Statement stat2 对象才行，否则会使原来的语句停止

一开始没有强制换行，在测试的时候发现对于连续的有英文字母，会出现文本溢出的情况，加上了 word-wrap 属性之后就解决了。

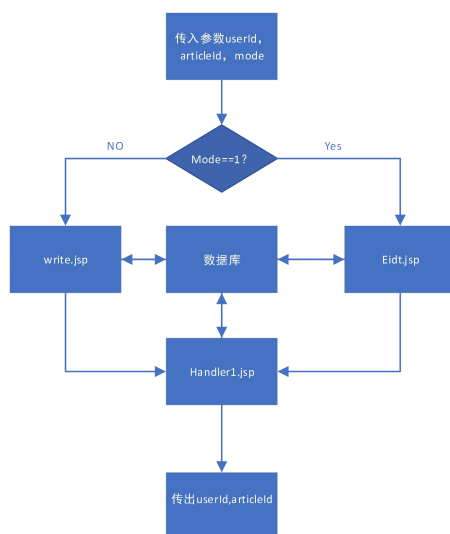
在设定浏览次数的时候，一开始只是简单的设计成访问了就+1，因为点赞和评论都是通过 jsp 来实现的，所以如果点赞或者评论的话都会导致浏览次数上升，这是不合理的。后来加上了一个参数来判断是否是点赞和评论之后，就解决了这个问题。

[同学 4 姓名：曾宇航]



本次项目我负责的是写文章和修改文章这一部分，可以相当于一个简单的封闭系统。

基本框图如下：



因为 `userId` 和 `articleId` 都是公开的，所以网页之间的参数传递可以直接通过 URL 来传递。

与数据库的交互过程总的来说类似于重新做了一遍 JSP1 那次实验，在功能的实现方面没有遇到很大的问题。

修改和新增文章这两个 jsp 可以只用一个实现的，debug 的时候分开了，后面也就没合起来了。

Debug 的时候根据 PPT 上写一个简单的 jsp 获取传出的参数并且输出。

在 CSS 设计方面自己做的不是很到位，总的来说自己做出来的自己都觉得难看，后面队友找了个网站，模仿者设计出 CSS 才感觉好看点，最后队友也帮忙优化了边框。

本来还想做一些其他功能的，比如草稿箱，回收站等等的，但是因为时间问题所以放弃了（建立新的数据库，设计新的网页样式，增加与其他队员之间的接口等等方面都很花时间）