

# 软件工程课程设计



课 题：\_\_\_\_\_ 个人博客系统 \_\_\_\_\_

班 级：\_\_\_\_\_ 科技二班 \_\_\_\_\_

组 号：\_\_\_\_\_ 第三组 \_\_\_\_\_

姓 名：\_\_\_\_\_ 于义好 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614092 \_\_\_\_\_

姓 名：\_\_\_\_\_ 贾鹏成 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614034 \_\_\_\_\_

姓 名：\_\_\_\_\_ 张光正 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614060 \_\_\_\_\_

姓 名：\_\_\_\_\_ 杨金字 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614048 \_\_\_\_\_

姓 名：\_\_\_\_\_ 陈菲儿 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614107 \_\_\_\_\_

姓 名：\_\_\_\_\_ 葛 声 \_\_\_\_\_ 学 号：\_\_\_\_\_ E11614094 \_\_\_\_\_

指导教师：\_\_\_\_\_ 罗 耀 \_\_\_\_\_

完成日期：\_\_\_\_\_ 2019 年 12 月 \_\_\_\_\_

**摘 要：**随着计算机及网络技术的飞速发展，人们对 Internet 应用的需求也越来越多，其中网上交友或利用网络和朋友家人联系，在线管理自己的相片等已经成为广大网民的共同需求，在这个背景下，我们的博客系统应需而出。

一般一个博客就是一个页面，它通常由简短而且经常更新的帖子构成，所有的文章都是按照年份和日期排列，有些类似斑竹的日记。看上去平淡，但它可以让更多的人零成本、零维护地创建自己的网络媒体，每个人都可以把自己的思想和灵感更新到博客站点上。博客的人性化和平民视角使得它提供的消息更贴近人们的生活，极大的丰富了人们业余生活，同时也给人与人之间的沟通架起一座桥梁。

主要开发环境是 mysql, phpstorm, wampserver, 代码使用语言 php, html。

**关键词：**个人博客、web

# 目 录

一、 可行性分析.....	1
1.1 经济可行性.....	1
1.2 技术可行性.....	1
1.3 操作可行性.....	1
二、 需求分析.....	2
2.1 功能需求.....	2
2.2 性能需求.....	3
2.3 数据库设计.....	3
三、 概要设计.....	4
3.1 设计目的.....	4
3.2 模块划分.....	4
3.2.1 用户注册与登录模块.....	4
3.2.2 用户信息设定模块.....	4
3.2.3 发表文章模块.....	4
3.2.4 发表评论和点赞模块.....	4
3.2.5 用户关注模块.....	4
3.3 数据库表结构设计.....	5
3.4 任务分配.....	6
四、 详细设计.....	7
4.1 用户注册与登录模块.....	7
4.2 用户信息设定模块.....	8
4.3 发表文章模块.....	10
4.4 发表评论模块.....	11
4.5 点赞模块.....	12
4.6 用户关注模块.....	12
五、 测试.....	15
5.1 用户注册测试.....	15
5.2 用户登录测试.....	15
5.3 用户信息设定测试.....	16
5.4 发表文章功能测试 .....	17
5.5 发表评论测试.....	18
5.6 用户关注功能测试 .....	18
5.7 屏蔽字功能测试.....	19
5.7.1 测试内容.....	19
附录.....	20

# 一、 可行性分析

## 1.1 经济可行性

经济可行性主要是对项目的经济效益进行评价。本系统对系统开发者来说并不需要太高的成本支出，而且开发周期不需要太长，节省了人力、物力、财力资源，所以本系统在经济上是可行的。

## 1.2 技术可行性

技术上的可行性分析主要是分析技术条件能否顺利完成开发工作，硬、软件能否满足开发者的需求等。在软件方面，本系统是基于 html 与 PHP 的个人博客系，数据库采用了 MYSQL;在硬件方面，本系统对机器本身没有太高要求，一般个人电脑完全满足。通过分析，具有技术上的可行性。

## 1.3 操作可行性

界面设计时充分考虑用户的习惯，使得操作简单。

## 二、需求分析

### 2.1 功能需求

该系统为个人博客，具有一般博客系统的功能。游客只能浏览前台页面，注册称为用户后，可以进入个人中心发表或者删除博文，对个人资料进行管理，浏览、评论、点赞、收藏博文，以及基于文章名或用户名的搜索。根据以上分析，该系统要实现的功能为：

#### 2.1.1 前台显示文章部分

游客注册以及注册用户的登录；博文发布；查看关注的人；个人信息修改；搜索博文和用户；评论；点赞；收藏。

#### 2.1.2 后台管理部分

登录验证；添加、编辑、删除文章功能；评论、点赞、收藏管理功能；审核功能。

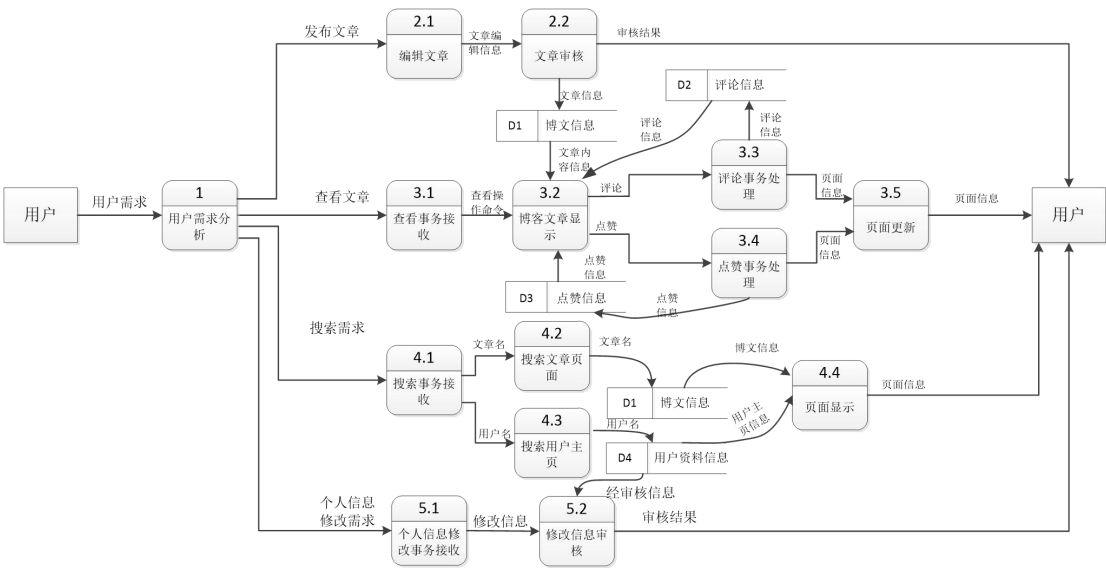


图 2.1 博客系统数据流图

## 2.2 性能需求

- (1) 操作简单、界面友好。完全控件式的页面布局，使得文章的录入工作更简便，许多选项只需要点击鼠标就可以完成；另外，跟踪出现的提示信息也让用户随时清楚自己的操作情况。对常见网站的博客管理的各个方面：上传博客、浏览、删除、修改等方面都大体实现，实现了网站对即时文章的管理要求；
- (2) 即时可见：对文章的处理（包括添加、编辑、删除）将立即在主页的对应栏目显示出来，达到“即时发布、即时见效”的功能；
- (3) 系统运行应该快速、稳定、高效和可靠；
- (4) 在结构上应具有很好的可扩展性，便于将来的功能扩展和维护。

## 2.3 数据库设计

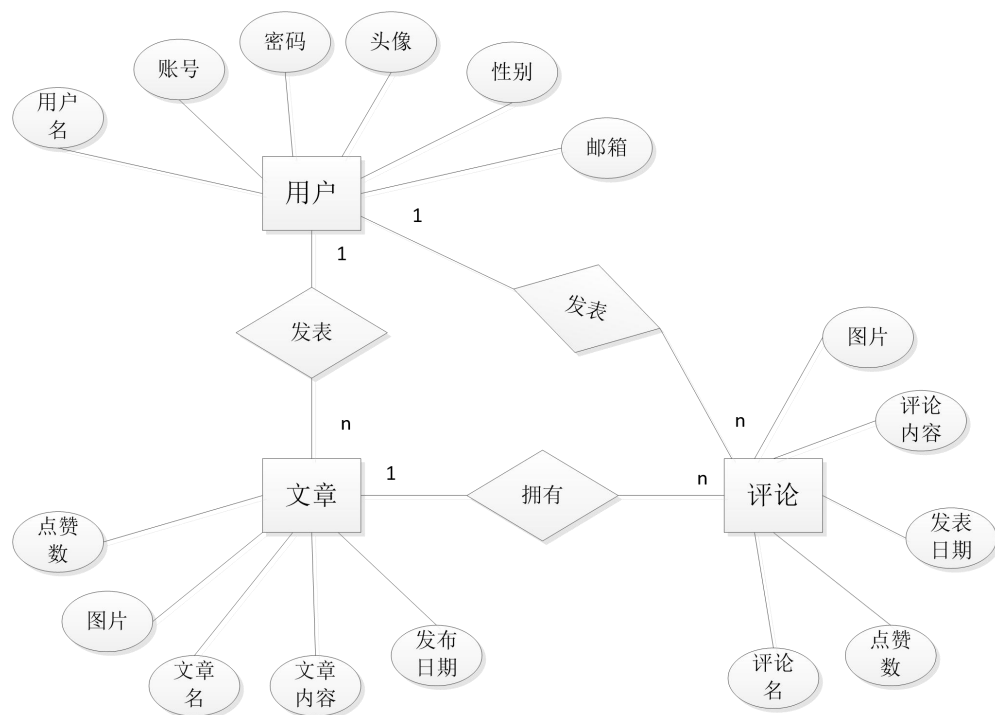


图 2.2 博客系统 E-R 图

## 三、 概要设计

### 3.1 设计目的

设计出一个界面友好的个人博客系统，具有一般博客系统的功能。游客只能浏览前台页面，注册称为用户后，可以进入个人中心发表或者删除博文，对个人资料进行管理，浏览、评论、点赞、收藏博文，以及基于文章名或用户名的搜索。

### 3.2 模块划分

本次程序设计中，总体设置 5 个子模块：用户注册与登录模块、用户信息设定模块、发表文章模块、发表评论和点赞模块、用户关注模块。

#### 3.2.1 用户注册与登录模块

实现用户的注册与登入功能，实现判断并存储用户的注册信息，判断用户的登入信息是否匹配成功，导向用户的个人博客主页。

#### 3.2.2 用户信息设定模块

实现用户对信息的设定，包括头像、昵称、城市、性别、个性签名等。

#### 3.2.3 发表文章模块

实现用户发布新文章的功能，使用时在博客主页引入文章的编辑栏实现文章设计，具体功能包含对文章最大字数的限制，文章在数据库中的存储，文章引用图片的上传与存储，在博客主页的显示方式。

#### 3.2.4 发表评论和点赞模块

实现对任意文章进行符合格式的评论以及点赞的功能。

#### 3.2.5 用户关注模块

操作用户可以实现对未关注用户的关注以及对已经关注的用户的取消关注。

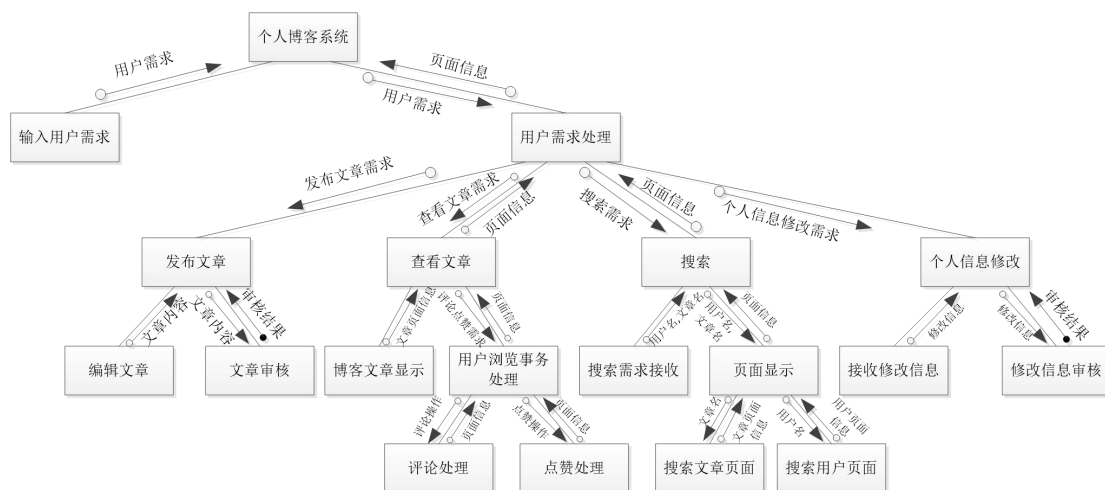


图 3.1 博客系统软件结构图

## 3.3 数据库表结构设计

### 3.3.1 登入信息表

序号	字段	说明	数据类型	允许为空	主键	备注
1	user_name	用户名	varchar[20]	Not null	P	
2	account_number	账号	varchar[20]	Not null	P	
3	password	密码	varchar[20]	Not null	x	

### 3.3.2 用户信息表

序号	字段	说明	数据类型	允许为空	主键	备注
1	user_name	用户名	varchar[20]	Not null	P	
2	sex	性别	varchar[10]	Not null	X	
3	mail	邮箱	varchar[20]	Not null	X	
4	icon	头像	varchar[20]	Not null	X	存储图片名
5	article_num	文章数量	varchar[10]	Not null	X	
6	comment_num	评论数量	varchar[10]	Not null	X	
7	commented_count	被评论数量	int	Not null	X	用作回复提醒
8	user_age	用户使用时间	data	Not null	X	可以没有

### 3.3.3 文章信息表



序号	字段	说明	数据类型	允许为空	主键	备注	
1	user_name	用户名	varchar[20]	Not null	P		
2	icon	头像	varchar[20]	Not null	X	存储图片名	
3	article_name	文章名	varchar[20]	Not null	X		
4	picture	图片	varchar[200]	Null	X	存储多个图片路径	
5	article	文章内容	varchar[1000]	Not null	X	能否存储还不确定	
6	usefull_num	被点赞数量	int	Not null	X		
7	published_date	发布日期	date	Not null	X		

### 3.3.4 评论信息表

序号	字段	说明	数据类型	允许为空	主键	备注	
1	user_name	用户名	varchar[20]	Not null	P		
2	icon	头像	varchar[20]	Not null	X	存储图片名	
3	comment_name	评论名	varchar[20]	Not null	X		
4	picture	图片	varchar[200]	Null	X	存储多个图片路径	
5	comment	评论内容	varchar[500]	Not null	X	能否存储还不确定	
6	usefull_num	被点赞数量	varchar[10]	Not null	X		
7	published_date	发布日期	date	Not null	X		

### 3.3.5 管理数据表

序号	字段	说明	数据类型	允许为空	主键	备注
1	user_num	用户数	varchar[20]	Not null	X	
2	icon_data	头像数据	varchar[20]	Not null	X	
3	comment_num	评论数	varchar[20]	Not null	X	
4	article_num	文章数	varchar[20]	Not null	X	
6	run_date	运行日期	date	Not null	X	

## 3.4 任务分配

角色	职责	负责人
项目经理	负责分配资源，确定优先级，协调开发人员直接以及与用户的交往	于义好
前端设计	负责网站界面的设计，与数据库的连接	于义好、贾鹏成
后端设计	负责编写相应的 url，编写数据库，对用户、网页数据进行交互与存储以及主要逻辑模块设计	张光正、杨金宇
文档撰写	进行需求分析以及开发过程中相应文档的撰写	陈菲儿、葛声

## 四、详细设计

### 4.1 用户注册与登录模块

#### 4.1.1 功能

- (1) 对所有未持有账号的用户提供注册功能；
- (2) 保证注册信息合法；
- (3) 对所有持有账号的用户提供登录页面；
- (4) 检查账号密码的一致性，做出相关反馈

#### 4.1.2 内部算法流程

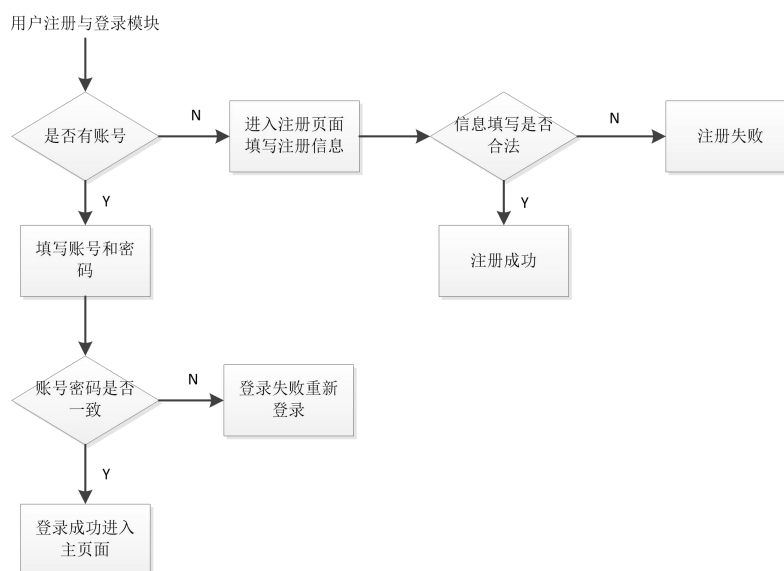
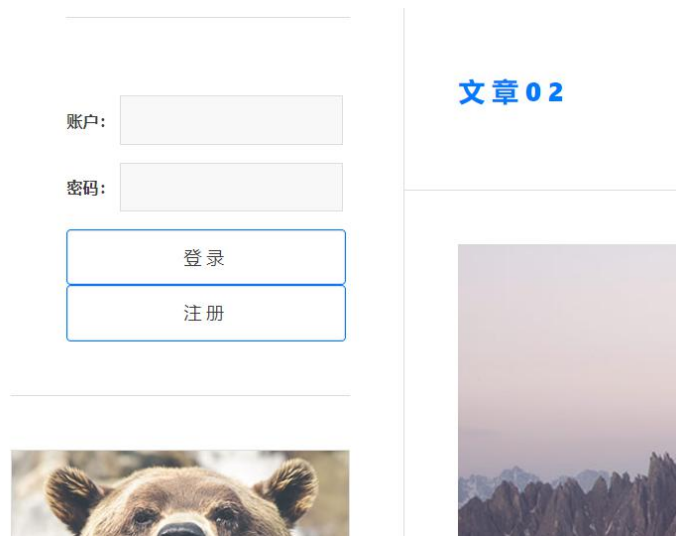


图 4.1 用户注册与登录模块流程图

### 4.1.3 界面截图



The screenshot shows a user login interface. On the left, there are two input fields labeled '账户:' (Account) and '密码:' (Password). Below these are two buttons: '登录' (Login) and '注册' (Register). To the right of the input fields, there is a blue text label '文章 02' (Article 02). Below the input fields, there is a small image of a brown bear's head. To the right of the bear image, there is a larger image of a mountain range at sunset or sunrise.

图 4.2 用户登录界面



The screenshot shows a user registration interface. It includes input fields for '用户名:' (Username), '密码:' (Password), and '个性签名:' (Personal Signature). Below the password field, there is a gender selection section with radio buttons for '保密' (Privacy), '男' (Male), and '女' (Female). At the bottom, there is a blue button labeled '确认注册' (Confirm Registration).

图 4.3 用户注册界面

## 4.2 用户信息设定模块

### 4.2.1 功能

- (1) 登录后的用户可以对个人信息进行设定，包括头像、昵称、城市、性别、个性签名；
- (2) 点击保存后根据信息填写是否合法做出相关反馈。

### 4.2.2 内部算法流程

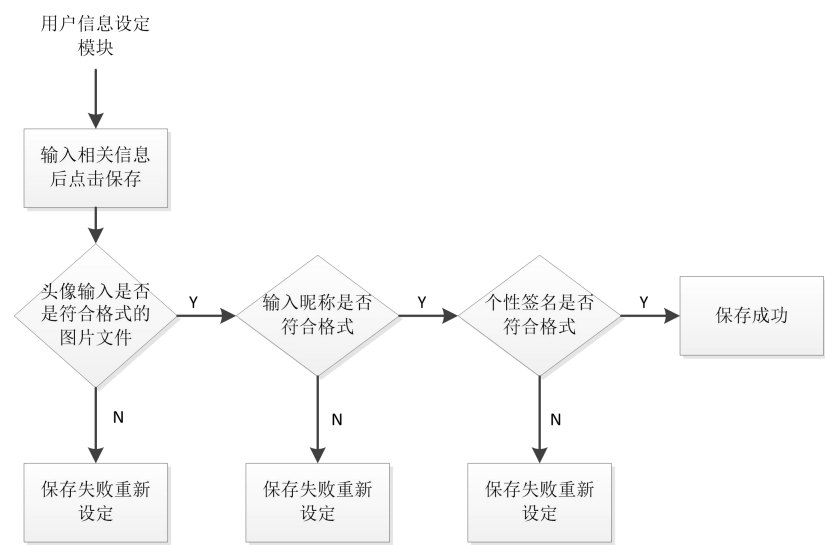


图 4.4 用户信息设定模块流程图

### 4.2.3 界面截图

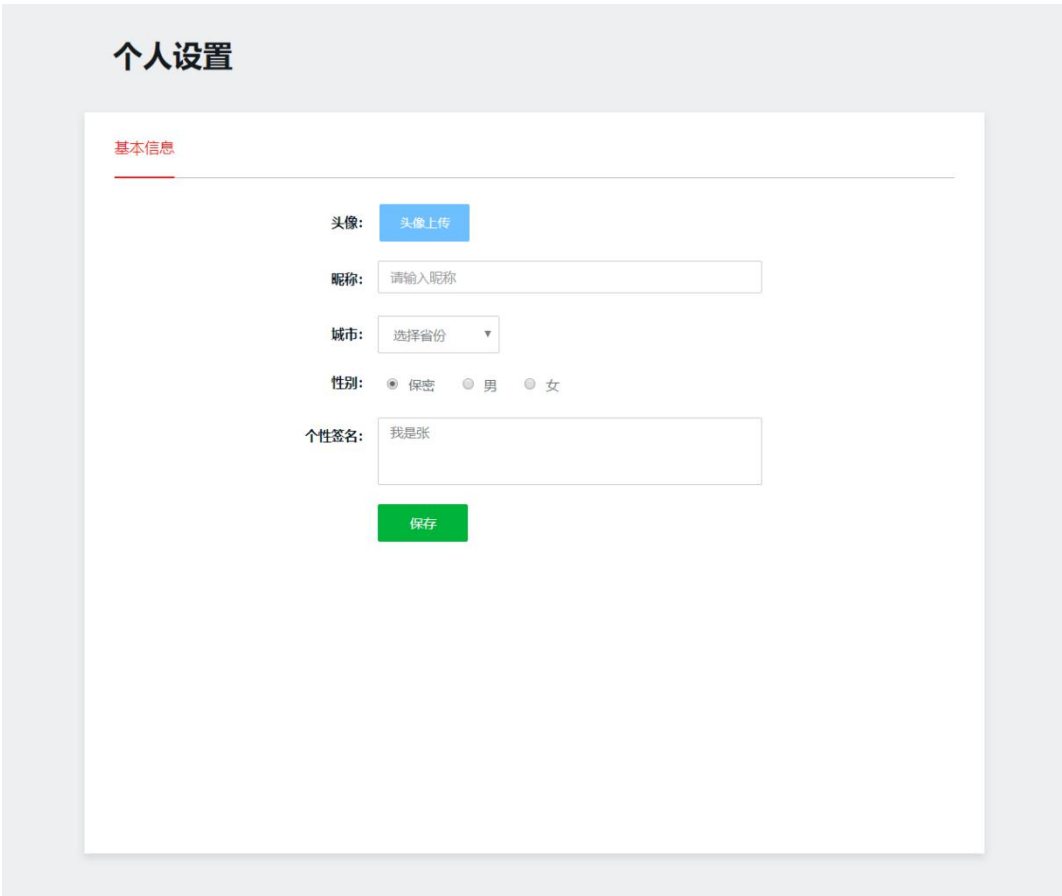


图 4.5 用户信息设定界面

## 4.3 发表文章模块

### 4.3.1 功能

- (1) 保证发表的文章标题、封面、内容符合格式；
- (2) 确保文章标题的唯一性；
- (3) 若文章标题和内容中存在预先设定的屏蔽字会自动进行替换
- (4) 发表的文章出现在博客个人界面的文章展示区。

### 4.3.2 内部算法流程

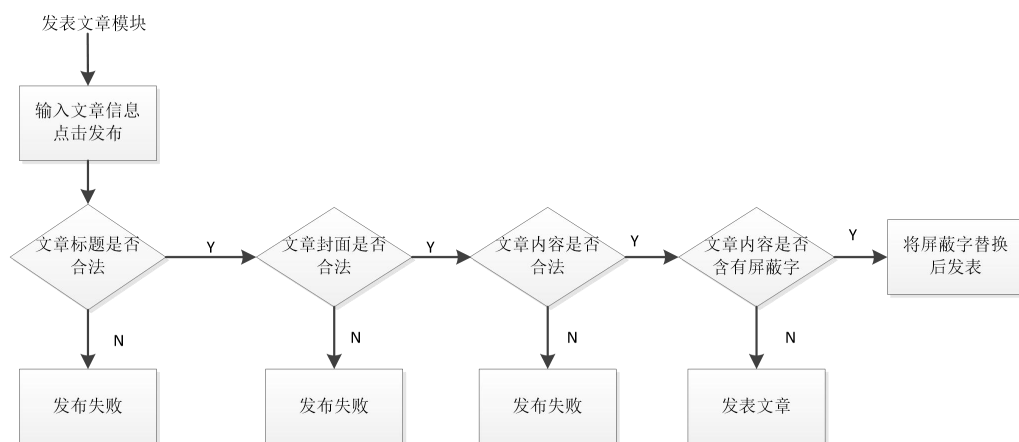


图 4.6 发表文章模块流程图

### 4.3.3 界面截图

标题:

封面:

字体

字数统计

图 4.7 发表文章界面

## 4.4 发表评论模块

### 4.4.1 功能

- (1) 在任意文章页面输入评论内容点击发布按钮若评论内容符合格式则在文章页面的评论区展示；
- (2) 若评论内容中有预先设定的屏蔽字会自动进行替换。

### 4.4.2 内部算法流程

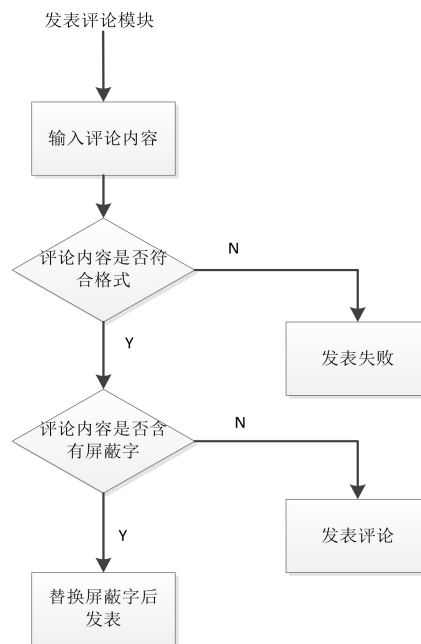


图 4.8 发表评论模块流程图

### 4.4.3 界面截图

The screenshot shows the user interface for posting a comment. At the top, there is a section labeled '评论区' (Comment Area). Below this, there is a text input area with the label '发布评论:' (Post comment:). At the bottom, there is a '发布' (Post) button.

图 4.9 发表评论界面

## 4.5 点赞模块

### 4.5.1 功能

- (1) 在任意文章页面，点击文章内容下方的“爱心”按钮，文章当前的点赞数加 1。

### 4.5.2 内部算法流程



图 4.10 点赞模块流程图

### 4.5.3 界面截图

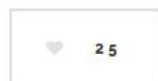


图 4.11 点赞界面

## 4.6 用户关注模块

### 4.6.1 功能

- (1) 关注任意未关注的用户，该用户出现在操作用户的好友列表中；
- (2) 取消关注已经关注过的用户。

## 4.6.2 内部算法流程

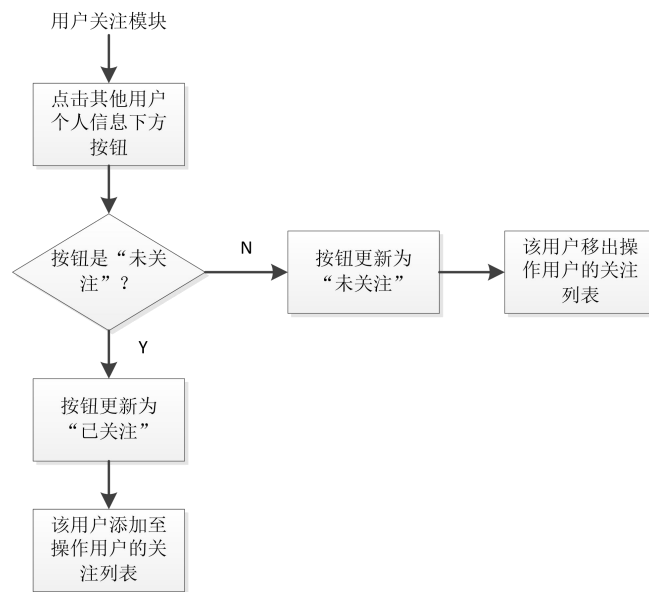


图 4.12 用户关注模块流程图



### 4.6.3 界面截图

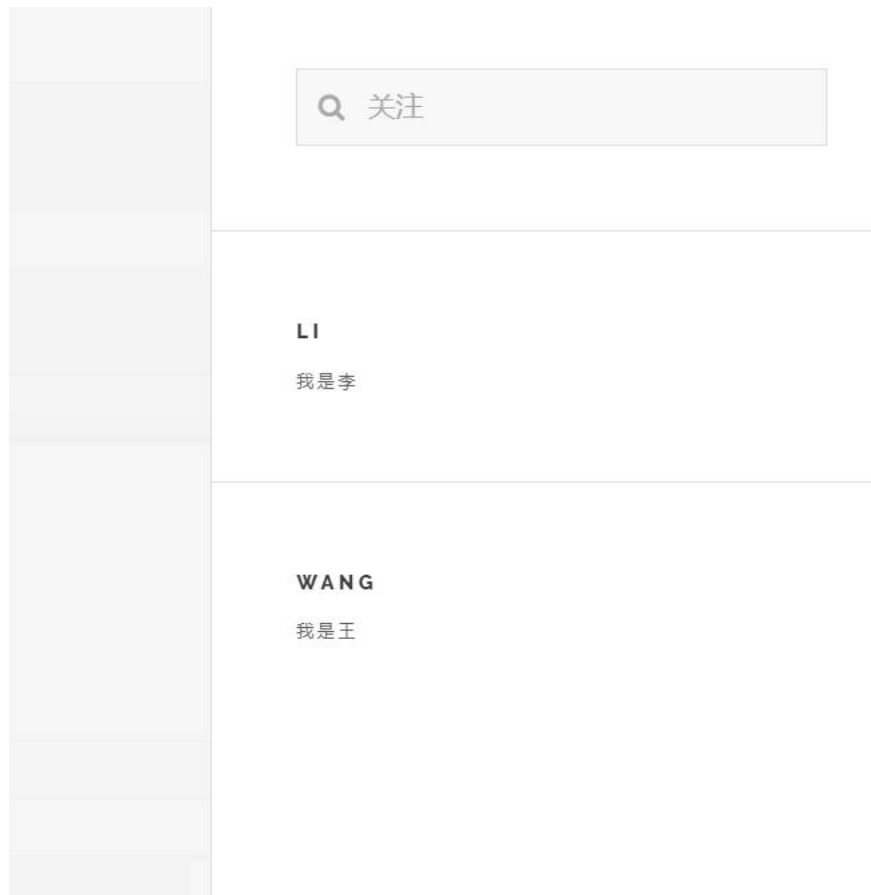


图 4.13 用户关注列表



图 4.14 用户取消关注界面

## 五、测试

### 5.1 用户注册测试

#### 5.1.1 测试内容

输入		结果
账号	密码	
“aaaa”（不存在账号）	“123456”	注册成功，账号与密码存储到数据库，转到个人博客主页界面
“aaaa”	“”	注册失败，给与反馈信息
“”	“123456”	注册失败，给与反馈信息
“aaaa”（已存在账号）	“123456”	注册失败，给与反馈信息

表 5.1 用户注册测试

#### 5.1.2 步骤及操作

进入博客主界面，在左上角输入框中输入账号与密码，并点击“注册”按钮，将信息存入数据库表中。

### 5.2 用户登录测试

#### 5.2.1 测试内容

输入		结果
账号	密码	
“aaaa”（正确账号）	“123456”	登陆成功，转到个人博客主页界面
“aaaa”	“”	登陆失败，给与反馈信息
“”	“123456”	登陆失败，给与反馈信息
“aaaav”（不存在账号）	“123456”	登陆失败，给与反馈信息

表 5.2 用户登录测试

#### 5.2.2 步骤及操作

进入博客主界面，在输入框中输入账号与密码，并点击“登陆”按钮。

## 5.3 用户信息设定测试

### 5.3.1 测试内容

#### (1) 头像设定测试

输入	结果
符合格式的图片文件	点击“保存”按钮后，头像信息上传并设置成功
非图片文件	点击“保存”按钮后，头像信息设置失败，给与反馈信息

表 5.3 头像设定测试

#### (2) 昵称设定测试

输入	结果
符合格式的昵称（字符串）	点击“保存”按钮后，昵称信息存储并设置成功
不符合格式的昵称	点击“保存”按钮后，昵称信息设置失败，给与反馈信息

表 5.4 昵称设定测试

#### (3) 城市设定测试

输入	结果
在下拉框中选择城市名	点击“保存”按钮后，城市信息存储并设置成功

表 5.5 城市设定测试

#### (4) 性别设定测试

输入	结果
在选择框中选择性别	点击“保存”按钮后，性别信息存储并设置成功

表 5.6 性别设定测试

#### (5) 个性签名设定测试

输入	结果
符合格式的个性签名（字符串）	点击“保存”按钮后，个性签名信息存储并设置成功
不符合格式的个性签名	点击“保存”按钮后，个性签名信息设置失败，给与反馈信息

表 5.7 个性签名设定测试

### 5.3.2 步骤及操作

登录后，进入博客个人界面，点击在右上角的设置菜单，通过键盘输入和鼠标点选进行用户信息设定，在所有个人信息设定完成后，点击“保存”按钮，上传个人信息。

## 5.4 发表文章功能测试

### 5.4.1 测试内容

#### (1) 文章标题编写测试

输入	结果
符合格式的文章标题（字符串）	点击“发布”按钮后，文章标题信息存储并设置成功
不符合格式的文章标题	点击“发布”按钮后，文章标题信息设置失败，给与反馈信息
已存在的文章标题	点击“发布”按钮后，文章标题信息设置失败，给与反馈信息

表 5.8 文章标题编写测试

#### (2) 文章封面上传测试

输入	结果
符合格式的图片文件	点击“发布”按钮后，文章封面信息上传存储并设置成功
非图片文件	点击“发布”按钮后，文章封面信息设置失败，给与反馈信息

表 5.9 文章封面上传测试

#### (3) 文章内容编写测试

输入	结果
符合格式的文章内容（字符串）	点击“发布”按钮后，文章内容信息存储并设置成功
不符合格式的文章内容	点击“发布”按钮后，文章内容信息设置失败，给与反馈信息

表 5.10 文章内容编写测试

### 5.4.2 步骤及操作

登录后，进入博客个人界面，在右上部的文本编辑器部分，通过键盘输入和鼠标点选进行具体文章编写，所有文章编写完成后，点击“发布”按钮将所有文章信息以及发布时间，发布用户的信息存入数据库并发表文章。

## 5.5 发表评论测试

### 5.5.1 测试内容

#### (1) 评论内容编写测试

输入	结果
符合格式的评论内容（字符串）	点击“发布”按钮后，评论内容信息存储并发布
不符合格式的评论内容	点击“发布”按钮后，评论内容信息存储失败，给与反馈信息

表 5.11 评论内容编写测试

#### (2) 点赞功能测试

输入	结果
点击一下文章页面，文章内容下方的“爱心”按钮	文章的当前点赞数加 1，并存入数据库中

表 5.12 点赞功能测试

### 5.5.2 步骤及操作

在任意文章页面，在文章内容下方的文本编辑器输入评论内容后，点击“发布”按钮将所有评论内容信息以及发布时间，发布用户的信息存入数据库并发表评论。在任意文章页面，点击一下在文章内容下方的“爱心”按钮。

## 5.6 用户关注功能测试

### 5.6.1 测试内容

输入	结果
点击“未关注”	关注成功，该用户出现在关注列表中
点击“已关注”	取消关注，该用户从关注列表中移出

表 5.13 关注功能测试

5.6.2 步骤及操作

在任意用户进入任意其他用户的文章页面后，点击一下在其他用户个人信息内容下方的“未关注”按钮，成功后将关注信息存入数据库，该用户将出现在操作用户的好友列表中，“未关注”按钮将更新为“已关注按钮”；

点击一下在其他已关注后的用户的个人信息内容下方的“已关注”按钮，成功后将更新数据库中的关注信息，该用户将不在出现在操作用户的好友列表中，“已关注”按钮将重置为“未关注按钮”。

5.7 屏蔽字功能测试

5.7.1 测试内容

输入	结果
“有一个“屏蔽字符串” ”	“有一个*”
“ “屏蔽字符串” ”	“*”
“有一个苹果”	“有一个苹果”

表 5.14 屏蔽字功能测试

5.7.2 步骤及操作

对所有存储在数据库中的文章标题，文字内容，评论与预先设定好的屏蔽字符串相同部分自动进行替换。

## 附录

### 注册与登录模块

```
<!-- 登陆功能 -->
<?php
session_start();
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);
if ($conn->connect_error)
{
    die("连接失败: " . $conn->connect_error);
}
if($conn)
{
    $name = $_POST['username'];
    $pwd = $_POST['password'];

    $check_query = "SELECT * FROM 用户信息表 WHERE 用户名='$name'
AND 密码='$pwd' ";
    $result = mysqli_query($conn,$check_query);
    if ($row = mysqli_fetch_array($result))
    {//登录成功

        $_SESSION['username']=$_POST['username'];
        header('Location:../WebBlog/index.php');//跳转页面，注意路径
        exit;
    }
    else {
        exit('登录失败! 点击此处 <a href="javascript:history.back(-1);">返回</a>
重试');
    }
}
?>
```

```

<!-- 登陆功能 -->
<?php
session_start();
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);
if ($conn->connect_error)
{
    die("连接失败: " . $conn->connect_error);
}
if($conn)
{
    $name = $_POST['username'];
    $pwd = $_POST['password'];
    $sex = $_POST['sex'];
    $sign = $_POST['sign'];
    $avatar="/WebBlog/images/avatar.jpg";
    $check_query = "INSERT INTO `用户信息表` (`用户名`,`密码`,`性别`,`头像`,`个性签名`) VALUES ('$name', '$pwd', '$sex', '$avatar', '$sign'); ";
    $result = mysqli_query($conn,$check_query);
    if ($result)
    {//登录成功
        header('Location:../WebBlog/shouye.php');//跳转页面，注意路径
        exit;
    }
    else {
        exit('注册失败! 点击此处 <a href="javascript:history.back(-1);">返回</a>重试');
    }
}
?>

```



## 用户信息设定模块

```
<!-- 设置头像功能-->
<?php
date_default_timezone_set('prc');
session_start();
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);
$data=$_POST['aboutme'];
$name = $_SESSION['username'];
$sex=$_POST['sex'];
$image=$_POST['image'];
$pattern='/src="(\\.*?(jpg|jpeg|gif|png))"/';
preg_match_all($pattern,$image,$image_url);
$url=$image_url[1][0];
echo $data;
echo PHP_EOL;
echo (string)$image_url[0][0];
echo PHP_EOL;
echo (string)$image_url[1][0];
$date=date("Y-m-d H",time());

if (!$conn)
{
    die("连接失败: ");
}
if($conn)
{
    $check_query = "UPDATE 文章信息表 SET 头像='$url' WHERE 用户名='$name'";
    $check_query1 = "UPDATE 用户信息表 SET 头像='$url' WHERE 用户名='$name'";
    $check_query2 = "Update 用户信息表 SET 个性签名='$data',性别='$sex' where 用户名='$name'";
    $result = mysqli_query($conn,$check_query);
    $result1 = mysqli_query($conn,$check_query1);
    $result2 = mysqli_query($conn,$check_query2);
}

header('Location:../WebBlog/index.php');
?>
```

```

<!-- 设置签名功能-->
<?php
session_start();
$data=$_POST['aboutme'];
$name=$_SESSION['username'];
$sex=$_POST['sex'];
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);

if (!$conn)
{
    die("连接失败: ");
}
if($conn)
{
    $check_query2 = "Update 用户信息表 SET 个性签名='$data',性别='$sex'
where 用户名='$name'";
    $result2 = mysqli_query($conn,$check_query2);
    header('Location:../WebBlog/Setting.php');
}
?>

```

## 文章模块

```
<!-- 文章界面 -->
<!DOCTYPE HTML>
<html>
<head>
    <title>文章</title>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1"/>
    <script src="assets/js/ie/html5shiv.js"></script>
    <link rel="stylesheet" href="assets/css/main.css"/>
    <link rel="stylesheet" href="assets/css/ie9.css"/>
    <link rel="stylesheet" href="assets/css/ie8.css"/>
    <!-- 新 Bootstrap4 核心 CSS 文件 -->
    <link rel="stylesheet"
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
    <!-- jQuery 文件。务必在 bootstrap.min.js 之前引入 -->
    <script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js"></script>
    <!-- popper.min.js 用于弹窗、提示、下拉菜单 -->
    <script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js"></script>
    <!-- 最新的 Bootstrap4 核心 JavaScript 文件 -->
    <script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js"></script>
    <!--图标库-->
    <link rel="stylesheet" href="assets/css/font-awesome.min.css">
    <!--Ueditor 编辑器-->
    <script src="assets/ueditor/ueditor.config.js">/*引入配置文件*/</script>
    <script src="assets/ueditor/ueditor.all.js">/*引入源码文件*/</script>
</head>
<body>

<!-- Wrapper -->
<div id="wrapper">
    <!-- Main -->
    <!-- Post -->
    <div id="main">
        <?php
            session_start();
            $servername = "localhost";
            $username = "root";
            $password = "123456";
            $dbname = "boke";
            $conn = mysqli_connect($servername, $username, $password, $dbname);
```

```

$title=$_REQUEST["title"];
$savatar22=$_SESSION['avatar22'];
$sign22=$_SESSION['sign22'];
$ownername = $_SESSION['username'];
$check_query = "SELECT * FROM 文章信息表 WHERE 文章名
=$title";
$sql_arr = mysqli_fetch_assoc(mysqli_query($conn,$check_query));
$name=$sql_arr['用户名'];
$savatar = $sql_arr['头像'];
$cover=$sql_arr['图片'];
$content = $sql_arr['文章内容'];
$like = $sql_arr['点赞数'];
$date = $sql_arr['日期'];
$new_like=$like;
$_SESSION['owner_name']=$name;
$_SESSION['owner_article']=$title;
$check_query1 = "SELECT * FROM 评论信息表 WHERE 文章名
=$title' AND 用户名=$name";
$result1=mysqli_query($conn,$check_query1);
$datarow = mysqli_num_rows(mysqli_query($conn,$check_query1));
echo "
<center><section id=\"sidebar\" >
    <!-- Intro -->
    <section id=\"intro\">
        <a href=\"#\" class=\"logo\"><img src=\"$savatar22\"
alt=\"\"/></a>
        <header>
            <h2>个性签名</h2>
            <p>$sign22</p>
        </header>
    </section>
</center>
<article class=\"post\">
    <header>
        <div class=\"title\">
            <a name=\"\"><h2>$title</h2></a>
        </div>
        <div class=\"meta\">
            <time class=\"published\" datetime=\"\">$date</time>
            <a href=\"#\" class=\"author\"><span
class=\"name\">$name</span><img src=\"$savatar\" alt=\"\"/></a>
        </div>
    </header>
    <a href=\"#\" class=\"image featured\"><img src=\"$cover\"

```

```

alt="/" /></a>
    <p>$content</p>
    <footer>
        <ul class="stats">
            <form action="Set_Like.php?New_Like=$like title = \"
method="post">
                <li><button style="left: 500px" name="like" type =
\"submit" class ="icon fa-heart" id ="olike"
onclick="onlike()">$like</button></li>
            </form>
        </ul>
    </footer>
</article>";
?>
<article class="post">
    <header>
        <div class="title">
            <a href="./Article.html"><h2>评论区</h2></a>
            <table class="table">
                <thead>
                </thead>
                <tbody>
                    <?php
                    for ($i = 0; $i < $datarow; $i++) {

                        $sql_arr1 =

mysqli_fetch_assoc($result1);

                        $article = $sql_arr1['文章名'];
                        $name = $sql_arr1['用户名'];
                        $date = $sql_arr1['发布时间'];
                        $reviewer = $sql_arr1['评论人'];
                        $avatar = $sql_arr1['头像'];
                        $review = $sql_arr1['评论内容'];

                        if($title==$article&&$ownername==$name) {
                            echo "<tr><td></td>
<td>$reviewer</td> <td><img src='$avatar' /></td> <td>$review</td>
<td>$date</td> <td></td> <br>";
                        }
                    }
                </tbody>
            </table>
        </div>

```

```

        </header>
    </article>
    <?php
    echo "
    <form id=\"profile\" action=\"Review.php\" method=\"post\">
        <div class=\"wlf-g-wrap clearfix\">

            <label class=\"label-name\" for=\"aboutme\">发布评论:
        </label>

            <div class=\"rlf-group\">
                <div class=\"pr\">
                    <textarea name=\"review\" id=\"aboutme\"
    rows=\"3\" class=\"noresize js-sign moco-form-control\"
    maxlength=\"50\"></textarea>
                    <p class=\"numCanInput js-numCanInput
    \"></p>
                </div>
            </div>
        </div>
        <div class=\"wlf-g-wrap clearfix\">
            <label class=\"label-name\"></label>
            <div class=\"rlf-group\">
                <input type=\"submit\" name=\"submit\"
    value=\"发布\" class=\"rlf-btn-green btn-block profile-btn\"/>
            </div>
        </div>
    </form>
    ";
    ?>
    </div>
</div>
<!-- Scripts -->
<script src=\"assets/js/jquery.min.js\"></script>
<script src=\"assets/js/skel.min.js\"></script>
<script src=\"assets/js/util.js\"></script>
<!--[if lte IE 8]>
<script src=\"assets/js/ie/respond.min.js\"></script><![endif]-->
<script src=\"assets/js/main.js\"></script>
</body>
</html>

```

## 评论模块

<!-- 评论自己文章功能 -->

<?php

date\_default\_timezone\_set('prc');

session\_start();

\$servername = "localhost";

\$username = "root";

\$password = "123456";

\$dbname = "boke";

\$conn = mysqli\_connect(\$servername, \$username, \$password, \$dbname);

\$data=\$\_POST['review'];

\$reviewer = \$\_SESSION['username'];

\$name=\$\_SESSION['owner\_name'];

\$title=\$\_SESSION['owner\_article'];

\$avatar=\$\_SESSION['avatar22'];

\$date=date("Y-m-d H-i-s",time());

function Get\_Keywords(\$file\_path)

{

if(file\_exists(\$file\_path))

{

\$str = file\_get\_contents(\$file\_path);//将整个文件内容读入到一个字符串

中

\$pattern = "/".str\_replace("\r\n", "|", \$str)."/";//定义正则表达式

\$pattern = iconv('GB2312','UTF-8',\$pattern);//编码转换

return \$pattern;

}

}

function Sensitive(\$pattern, \$string)

{

\$count = 0;//违规词的个数

\$sensitiveWord = "";//违规词

\$stringAfter = \$string; //替换后的内容

if(preg\_match\_all(\$pattern, \$string, \$matches))

{ //匹配到了结果

\$patternList = \$matches[0];//匹配到的数组

\$count = count(\$patternList);

\$sensitiveWord = implode(',', \$patternList); //敏感词数组转字符串

\$replaceArray =

array\_combine(\$patternList,array\_fill(0,count(\$patternList),'\*')); //把匹配到的数组进行合并，替换使用

```

        $stringAfter = strstr($string, $replaceArray); //结果替换
    }
    return $stringAfter;
}
function Filter($str)
{
    $pattern=Get_Keywords("D:\wamp64\www\WebBlog\keywords.txt");
    $result = Sensitive($pattern, $str);
    return $result;
}
$final_review=Filter($data);
if (!$conn)
{
    die("连接失败: ");
}
if($conn)
{
    $check_query = "INSERT INTO `评论信息表` (`文章名`,`用户名`,`发布时间`,`评论人`,`头像`,`评论内容`) VALUES ('$title', '$name', '$date', '$reviewer', '$avatar', '$final_review') ";
    $result = mysqli_query($conn,$check_query);
    $_SESSION['avatar22']=$_SESSION['avatar22'];
    $_SESSION['sign22']=$_SESSION['sign22'];
    $_SESSION['username']=$_SESSION['username'];
    $_SESSION['title']=$_SESSION['owner_article'];
}
header('Location:../WebBlog/Article_1.php');
?>

```



## 点赞模块

```
<!-- 点赞功能-->
<?php
date_default_timezone_set('prc');
session_start();
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);
$_SESSION['avatar22']=$_SESSION['avatar22'];
$_SESSION['sign22']=$_SESSION['sign22'];
$_SESSION['username']=$_SESSION['username'];
$_SESSION['title']=$_SESSION['owner_article'];
if (!$conn)
{
    die("连接失败: ");
}
if($conn)
{
    $New_Like=$_REQUEST["New_Like"];
    $name=$_SESSION['owner_name'];
    $title=$_SESSION['owner_article'];
    $New_Like=$New_Like+1;
    $check_query = "UPDATE 文章信息表 SET 点赞数='$New_Like' WHERE
用户名='$name' AND 文章名='$title'";
    $result = mysqli_query($conn,$check_query);
}
header('Location:../WebBlog/Article_1.php');
?>
```

## 关注模块

```
<!-- 关注功能 -->
<?php
date_default_timezone_set('prc');
session_start();
$servername = "localhost";
$username = "root";
$password = "123456";
$dbname = "boke";
$conn = mysqli_connect($servername, $username, $password, $dbname);
$data=$_POST['if_focus'];
$username=$_SESSION['username'];
$name=$_SESSION['owner_name'];
$date=date("Y-m-d H-i-s",time());

if (!$conn)
{
    die("连接失败: ");
}
if($conn)
{
    if($data==1)
        $check_query = "DELETE FROM `关注列表` WHERE `关注人` =
'$username' AND `被关注人` = '$name'";
    else
        $check_query = "INSERT INTO `关注列表` (`关注人`,`被关注人`)
VALUES ('$username', '$name')";

    $result = mysqli_query($conn,$check_query);
    echo $data;
}
header('Location:../WebBlog/index.php');
?>
```