

编号: _____

桂林电子科技大学信息科技学院

毕业设计(论文)说明书

题 目: 网站后端用户信息管理系统

系 别: 信息工程系

专 业: 软件工程

学生姓名: 余俊晖

学 号: 1652200204

指导教师: 沈岚岚

职 称: _____

题目类型: ☐理论研究 ☐实验研究 ☐工程设计 ☐工程技术研究 ☒软件开发 ☐应用研究

2018 年 12 月 5 日

独 创 性 声 明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期：2018 年 12 月 13 日

关于学位论文版权使用授权的说明

本人完全了解桂林电子科技大学信息科技学院关于收集、保存、使用学位论文的以下规定：学院有权采用影印、缩印、扫描、数字化或其它手段保存论文；学院有权提供本学位论文全文或者部分内容的阅览服务；学院有权将学位论文的全部或部分内容编入有关数据库进行检索、交流；学院有权向国家有关部门或者机构送交论文的复印件和电子版。

学位论文作者签名：

日期：2018 年 12 月 13 日

导 师 签 名：

日期：2018 年 12 月 13 日

目录

摘 要	5
目的背景与意义.....	6
一、需求分析.....	6
1. 用户端信息管理.....	6
2. 管理员管理.....	6
二、系统设计.....	7
1. 结构设计.....	7
2. 功能结构图.....	7
三、数据库的设计.....	8
1. 数据库设计.....	8
2. 数据库详细.....	9
2.1 数据库表结构.....	9
四、详细设计.....	10
1. 系统模块总览.....	10
2. 模块描述细化.....	10
2.1 管理员控制模块如图所示:	10
2.2 用户控制模块如图所示:	11
2.3 Dao 设计模式模块如图所示:	11
2.4 过滤器模块.....	11
2.5 监听器模块.....	11
2.6 管理员类和用户类模块.....	12
2.7 服务端模块.....	12
2.8 servlet 模块.....	12
2.9 工具类.....	12
2.10 重置注册界面的所有用户输入额信息模块	17
五、实现后功能展示.....	18
1. 登陆界面:	18
2. 注册界面:	19
3. 注册成功:	20
4. 数据库存取信息:	20
5. 登录:	20
6. 修改信息:	21
7. 管理员登录:	21
8. 管理员界面:	22
9. 显示所有用户功能:	22
10. 用户被冻结:	22
11. 显示冻结:	23
六、软件测试.....	23
1、登陆界面测试用例.....	23
2、压力测试.....	25
七、课程设计心得.....	30
八、参考文献.....	30

附录	30
----------	----

摘 要

该项目是本人一个网站后端分离出来的用户信息管理系统。使用 java 语言和 mysql 数据库完成，开发工具包含 java、mysql、javaweb、HTML、JavaScript 和 tomcat 等。项目功能齐全且易于操作，界面简洁，清楚的满足了本次实训所要求的内容。

本文介绍了网站后端用户信息管理系统的开发的整个过程，阐述了系统分析、数据库设计和系统实现的全过程。在开发方法上本系统利用了软件工程化的思想和方法，总体上采用结构化生命周期的开发方法。本系统据主要分为用户信息管理和管理员管理两大板块。两个板块之间相互联系，但又互不干扰，这符合了软件模块的开发思想。

本系统完成了用户系统注册，用户信息修改，用户信息重置，冻结用户账号，删除用户，用户查询，用户模糊搜索等功能。小而精。

关键词：开发工具；后端信息管理系统；软件工程。

正文

目的背景与意义

网站后端用户信息管理系统是网站不可缺少的部分。一个功能齐全、简单易用的信息系统不但能有效的减轻网站后端维护人员的工作负担，还对网站决策者和管理者至关重要。所以网站后端用户信息管理系统能很好的满足管理者对用户信息进行管理，实现智能化管理，不用管理员时刻去审核注册用户的相关信息，智能的实现用户的增删改查。

为了更好的保护用户信息，本系统还使用了对用户密码的加密方式，利用 salt 和 hash 算法对用户密码进行了加密处理，保障了用户信息的安全性。使用 mysql 数据库对海量用户信息实现存储。并且成本也低，这些优点可以很好的为网站管理者提供一个科学化、正规化管理、智能化管理的良好条件。

一、需求分析

网站后端管理系统的功能总结起来，共需要一下几个方面：

1. 用户端信息管理

用户通过输入网址进入登录注册界面，登录界面包括填写用户账号、密码和记住密码信息，注册界面包括用户自行按要求填写符合注册要求的各方面信息等。如下图展示：

2. 管理员管理

网站为管理员设置了管理员操作的一系列权限，管理员的功能权限包括：一

键显示所有用户、按用户名、邮箱和昵称搜索用户、模糊搜索用户等。管理员还可以一键重置用户密码、冻结/恢复用户登录和删除用户。

二、系统设计

1. 结构设计

根据对系统的需求分析，本系统主要分为两大板块，外加一前端板块：

1.1 用户端信息管理

用户通过输入网址进入登录注册界面，登录界面包括填写用户账号、密码和记住密码信息，注册界面包括用户自行按要求填写符合注册要求的各方面信息等。如下图展示：

1.2 管理员管理

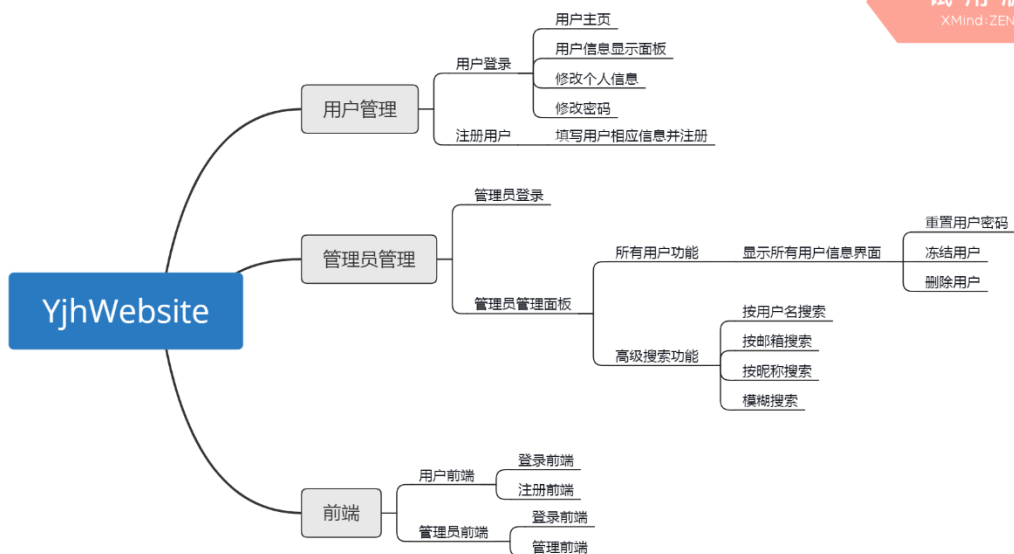
网站为管理员设置了管理员操作的一系列权限，管理员的功能权限包括：一键显示所有用户、按用户名、邮箱和昵称搜索用户、模糊搜索用户等。管理员还可以一键重置用户密码、冻结/恢复用户登录和删除用户。

1.3 前端

展示给用户并提示操作的一系列界面。

2. 功能结构图

系统的功能思维导图如图所示：



三、数据库的设计

1. 数据库设计

此系统使用的是 mysql 数据库，此系统的数据库名称是 websitedatabase。创建语句代码如下：

```
CREATE DATABASE WebsiteDatabase;
```

```
USE WebsiteDatabase;
```

```
CREATE TABLE user(
    id CHAR(10) PRIMARY KEY,
    salt CHAR(16) NOT NULL,
    pass CHAR(64) NOT NULL,
    freeze BIT NOT NULL,
    name CHAR(10) NOT NULL,
    email VARCHAR(60) UNIQUE NOT NULL,
    sex CHAR(2) NOT NULL,
    intro VARCHAR(256) NOT NULL
);
```

```
CREATE TABLE admin(
    id CHAR(10) PRIMARY KEY,
    pass CHAR(16) NOT NULL
);
```



```
/* 管理员 */
INSERT INTO admin VALUE('admin', 'admin888');
```

2. 数据库详细

2.1 数据库表结构

本次创建的数据库中包含两张表，分别为 admin 表和 user 表，如图所示：

```
mysql> show tables;
+-----+
| Tables_in_websitedatabase |
+-----+
| admin                      |
| user                      |
+-----+
2 rows in set (0.00 sec)
```

表结构展示如图所示：

```
mysql> desc admin;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | char(10)  | NO   | PRI | NULL    |       |
| pass  | char(16)  | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Admin 表

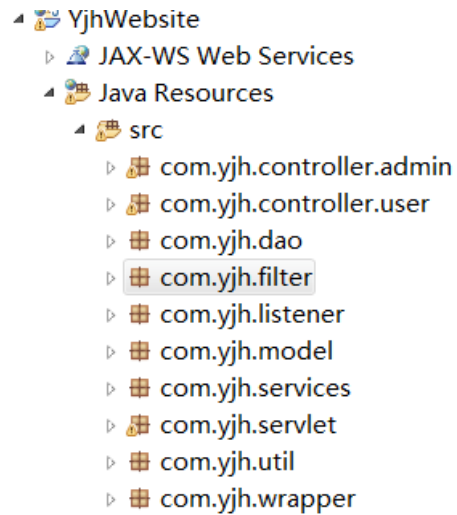
```
mysql> desc user;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | char(10)  | NO   | PRI | NULL    |       |
| salt  | char(16)  | NO   |     | NULL    |       |
| pass  | char(64)  | NO   |     | NULL    |       |
| freeze | bit(1)    | NO   |     | NULL    |       |
| name  | char(10)  | NO   |     | NULL    |       |
| email | varchar(60) | NO   | UNI | NULL    |       |
| sex   | char(2)   | NO   |     | NULL    |       |
| intro | varchar(256) | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)
```

User 表

四、详细设计

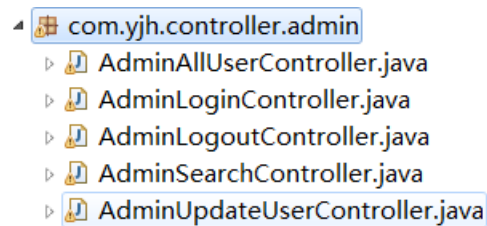
1. 系统模块总览

本系统包含 10 个包，详细如图所示：



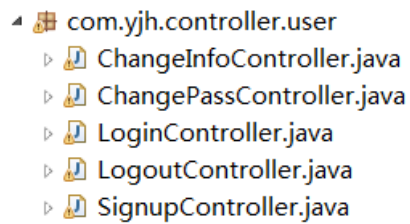
2. 模块描述细化

2.1 管理员控制模块如图所示：



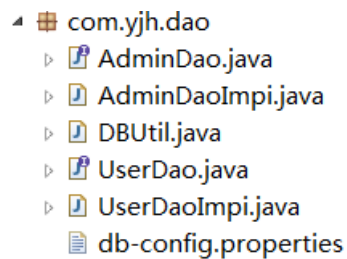
各个模块功能为：管理员一键显示所有用户、管理员登录、管理员注销、管理员所有用户和管理员修改用户信息模块。

2.2 用户控制模块如图所示：



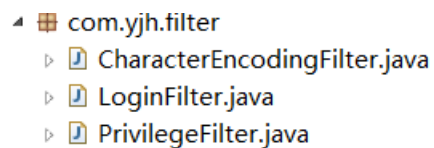
各个模块功能为：修改信息、修改密码、登录、注销和注册。

2.3 Dao 设计模式模块如图所示：



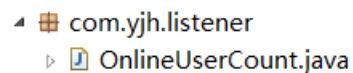
主要是数据库方面配置信息及操作的板块。

2.4 过滤器模块



主要是设计注册登录等字符相关的过滤操作。

2.5 监听器模块



主要是对前端按钮的一系列监听操作。

2.6 管理员类和用户类模块

- └─ com.yjh.model
 - └─ Admin.java
 - └─ User.java

2.7 服务端模块

- └─ com.yjh.services
 - └─ AdminServices.java
 - └─ AdminServicesImpi.java
 - └─ UserServices.java
 - └─ UserServicesImpi.java

主要是管理员和用户类服务的类操作

2.8 servlet 模块

- └─ com.yjh.servlet
 - └─ ErrorHandler.java
 - └─ InitServlet.java

初始化和错误消息操作。

2.9 工具类

- └─ com.yjh.util
 - └─ CookieSessionParam.java
 - └─ DigestUtils.java
 - └─ FormParam.java
 - └─ Privilege.java
 - └─ UserPaging.java
 - └─ ValidateUserInfo.java

各种与网络链接有关的类。

◎重点的类的实现功能

○MD5 or SHA 加密工具类

其中 DigestUtils.java 实现了用户注册时密码的加密。具体代码如下：

```
package com.yjh.util;

import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

/**
 * MD5 or SHA 加密工具类
 *
 * @author yjh
 */

public class DigestUtils {

    public static final String MD5 = "MD5", SHA1 = "SHA-1", SHA256 =
"SHA-256";

    /**
     * 使用 MD5 或 SHA 加密<br/>
     *
     * @return 返回加密后的 16 进制表示字符串，出错返回 null
     */
    public static String digest(String mess, String method) {
        MessageDigest md;
        try {
            md = MessageDigest.getInstance(method);
            md.update(mess.getBytes());
            byte[] digest = md.digest();

            // 将 digest 转换为十六进制字符串形式
            char[] hexDigits = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
                'A', 'B', 'C', 'D', 'E', 'F' };
            char[] result = new char[digest.length * 2];
            int k = 0;
            for (int i = 0; i < digest.length; i++) {
                byte byte0 = digest[i];
                result[k++] = hexDigits[byte0 >>> 4 & 0xf];
                result[k++] = hexDigits[byte0 & 0xf];
            }
        } catch (Exception e) {
            return null;
        }
    }
}
```

```

    }

    return new String(result);

} catch (NoSuchAlgorithmException e) {
    e.printStackTrace();
}

return null;
}

/**
 * 使 SHA-256 加密<br/>
 *
 * @return 返回加密后的 16 进制表示字符串，出错返回 null
 */
public static String sha256Digest(String mess) {
    return digest(mess, SHA256);
}
}

```

○正则表达式加密

ValidateUserInfo.java 使用正则表达式对用户名、邮箱和密码进行了格式的匹配，具体代码如下：

○验证邮箱的合法性

```

public class ValidateUserInfo {
    /**
     * 验证邮箱正则表达式
     */
    private static final String REGEX_EMAIL = "^[a-z0-9A-Z]+[-|\\.|?)+[a-z0-9A-Z]@([a-z0-9A-Z]+(-[a-z0-9A-Z]+)?\\.)+[a-zA-Z]{2,}$";

    /**
     * 对所有信息进行验证
     *
     * @return 合法返回 null，否则返回错误信息
     */
    public static String validate(String id, String pass, String name, String email, String sex, String intro) {

```

```

StringBuilder sb = new StringBuilder();
String temp;
if((temp = validateID(id)) != null) {
    sb.append(temp + "; ");
}
if((temp = validatePassword(pass)) != null) {
    sb.append(temp + "; ");
}
if((temp = validateName(name)) != null) {
    sb.append(temp + "; ");
}
if((temp = validateEmail(email)) != null) {
    sb.append(temp + "; ");
}
if((temp = validateSex(sex)) != null) {
    sb.append(temp + "; ");
}
if((temp = validateIntro(intro)) != null) {
    sb.append(temp + "; ");
}

String result = sb.toString();
if("").equals(result)) {
    return null;
} else {
    return result;
}
}

```

○验证用户名的合法性

```

/**
 * 检验用户名是否合法<br/>
 * 用户名：1 到 10（含）个字符。由数字、英文字母、连字符 构成
 *
 * @return 合法返回 null。当 id 为 null 或不符合要求时返回错误信
息
 */
public static String validateID(String id) {
    if(id == null || "".equals(id)) {
        return "用户名为空";
    }
}

```

```

    }

    if(id.length() > 10) {
        return "用户名为 1 到 10 个字符";
    }

    for(int i = 0; i < id.length(); i++) {
        char c = id.charAt(i);
        if(!(c == '-' || Character.isLetter(c) || Character.isDigit(c))) {
            return "用户名包含非法字符，应由数字、英文字母、连
字符 构成";
        }
    }

    return null;
}

```

○验证昵称的合法性

```

/**
 * 判断昵称是否合法<br/>
 * 昵称：1 到 10（含）个字符。
 *
 * @return 合法返回 null。当 name 为 null 或不符合要求时返回错
误信息
 */
public static String validateName(String name) {
    if(name == null || "".equals(name) || name.length() > 10) {
        return "昵称为 1 到 10 个字符";
    }
    return null;
}

```

○验证密码的合法性

```

/**
 * 判断密码是否合法<br/>
 *
 * 1. 由英文字母、数字 0-9 、特殊符号! @ # $ % ^ & * ( ) [ ] { } - _
= + 构成<br/>
 * 2. 8 到 16 位<br/>

```



```

* 3. 至少包含一个字母、一个数字、一个特殊符号<br/>
*
* @return 合法返回 null。当 password 为 null 或不符合要求时返回
错误信息
*/
public static String validatePassword(String password) {
    if(password == null || "".equals(password)) {
        return "密码为空";
    }
    if(password.length() < 8 || password.length() > 16) {
        return "密码为 8 到 16 位";
    }

    String symbol = "!@#$%^&*()[]{}-_=+";
    boolean haveLetter = false, haveDigit = false, haveSymbol = false;
    for(int i = 0; i < password.length(); i++) {
        char c = password.charAt(i);
        if(Character.isLetter(c)) {
            haveLetter = true;
        } else if(Character.isDigit(c)) {
            haveDigit = true;
        } else if(symbol.indexOf(c) != -1) {
            haveSymbol = true;
        } else {
            return "密码包含非法字符" + c;
        }
    }

    if(haveLetter && haveDigit && haveSymbol) {
        return null;
    } else {
        return "密码至少包含一个字母、一个数字、一个特殊符号";
    }
}

```

2.10 重置注册界面的所有用户输入额信息模块

```

com.yjh.wrapper
  MyCharacterEncodingRequest.java

```

五、实现后功能展示

1. 登陆界面：

YjhWebsite
欢迎你：游客

用户登录
登录用户名：

密码：

☐ 下次自动登录

还没有帐号？[注册一个吧](#)
管理员登录？[Click here!](#)

访问[我的github主页](#)获取项目源代码和教程
Copyright © 2018-12-5

2. 注册界面：

YjhWebsite

欢迎你：游客

用户注册

用户名：

1到10个字符，由数字、英文字母、连字符构成

密码：

确认密码：

8到16位密码；至少包含一个字母、一个数字、一个特殊符号

昵称：

1到10个字符，由汉字、数字、英文字母、连字符构成

邮箱：

性别：☐ 男 ☐ 女 ☒ 保密

简介：

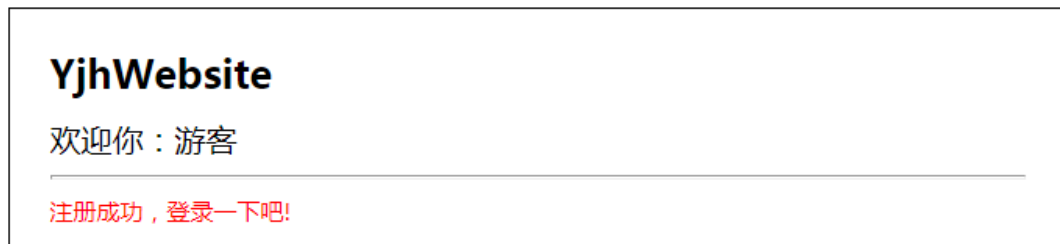
介绍下自己吧!!! (256个字符之内)

已有帐号？[点击登录](#)

访问[我的github主页](#)获取项目源代码和教程

Copyright © 2018-12-5

3. 注册成功:



4. 数据库存取信息:

The screenshot shows a database table with the following data:

id	salt	pass	freeze	name	email	sex	intro
111	\)5\$ej2f]_5^BC96	39CB07362E8DAF8DB1420F00936A886649B2E0F3967AA235A3B1F398453E0667	b'0'	111	1254094880@qq.com	女	测试用户，测试
222	/kTw)=w4k*8I<O9s	2922F205A399DF9F72147F76432B14B60C586755B26F83B881FCF10DFBAE3682	b'0'	222	222@qq.com	男	注册
yjh	mOg\y3to\O\K9t#*	B665011689EB1BD51D494D00AD0FF04AD22C1730CB49A35DC06965F72125DA75	b'0'	yjh110	123@qq.com	男	java+mysql
*	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

5. 登录:



6. 修改信息：

YjhWebsite
欢迎你：222 在线人数：2 [我的主页](#) [退出](#)

修改个人信息
用户名：
昵称：
邮箱：
简介：

注册

* 用户名：1到10个字符，由汉字、数字、英文字母、连字符构成

访问[我的github主页](#)获取项目源代码和教程

Copyright © 2018-12-5

7. 管理员登录：

管理员登录
登录用户名

密码

用户登录？[Click here!](#)

用户名：admin 密码：admin888
可开启禁用删除用户功能
重置密码：123456

8. 管理员界面：

YjhWebsite管理系统
欢迎你：管理员 [我的主页](#) [退出](#)

[所有用户](#) | [高级搜索](#) | [可以添加功能](#)

访问[我的github主页](#)获取项目源代码和教程
Copyright © 2018-12-5

9. 显示所有用户功能：

YjhWebsite管理系统
欢迎你：管理员 [我的主页](#) [退出](#)

用户名	昵称	邮箱	性别	重置密码	冻结/恢复	删除
111	111	1254094880@qq.com	女	重置密码	冻结	删除用户
222	222	222@qq.com	男	重置密码	冻结	删除用户
yjh	yjh110	123@qq.com	男	重置密码	冻结	删除用户

共找到3条记录

访问[我的github主页](#)获取项目源代码和教程
Copyright © 2018-12-5

10. 用户被冻结：

YjhWebsite
欢迎你：游客

出错了！！
该账户已被冻结，请联系管理员

用户登录

11. 显示冻结：

YjhWebsite管理系统

欢迎你：管理员 [我的主页](#) [退出](#)

用户名	昵称	邮箱	性别	重置密码	冻结/恢复	删除
111	111	1254094880@qq.com	女	重置密码	<button>恢复</button>	删除用户
222	222	222@qq.com	男	重置密码	<button>冻结</button>	删除用户
yjh	yjh110	123@qq.com	男	重置密码	<button>冻结</button>	删除用户

共找到3条记录

访问[我的github主页](#)获取项目源代码和教程

Copyright © 2018-12-5

六、软件测试

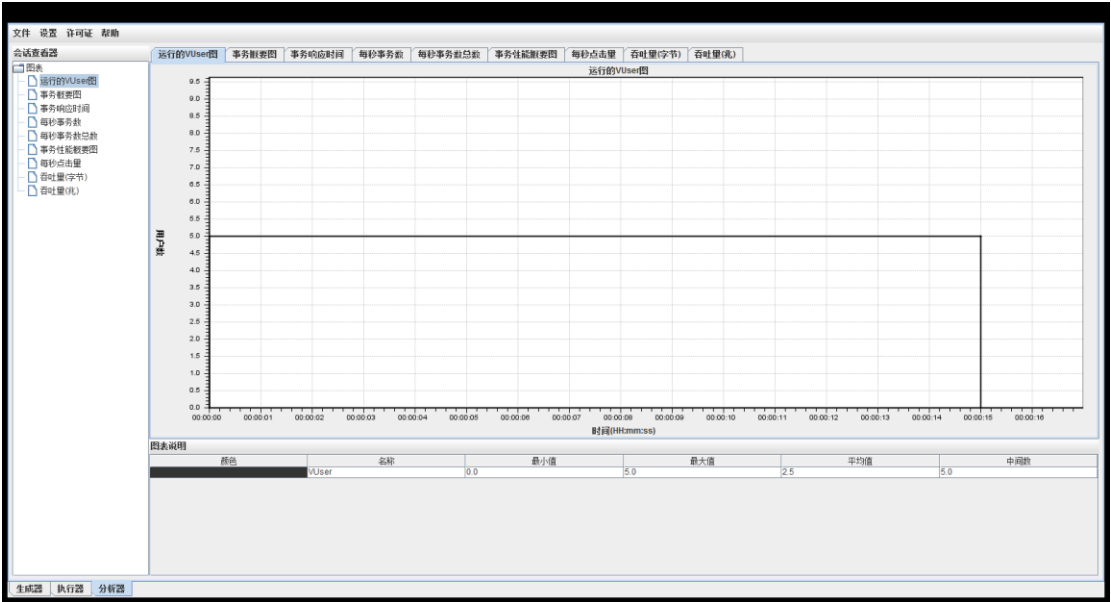
1、登陆界面测试用例

用例 ID	XXXX-XX-XX	用例名称	系统登录	
用 例 描 述	系统登录 用户名存在、密码正确的情况下，进入系统 页面信息包含：页面背景显示 用户名和密码录入接口，输入数据后的登入系统接口			
用 例 入 口	打开 tomcat 服务器 Chrome，在地址栏输入相应的地址 进入该系统的登录界面			
测试用例 ID	场景	测试步骤	预期结果	备注
TC1	初始界面显示	从用例入口进入	页面元素完整，显示与详细设计一致	
TC2	用户名录入-验证	输入已经存在的用户:yjh	输入成功	

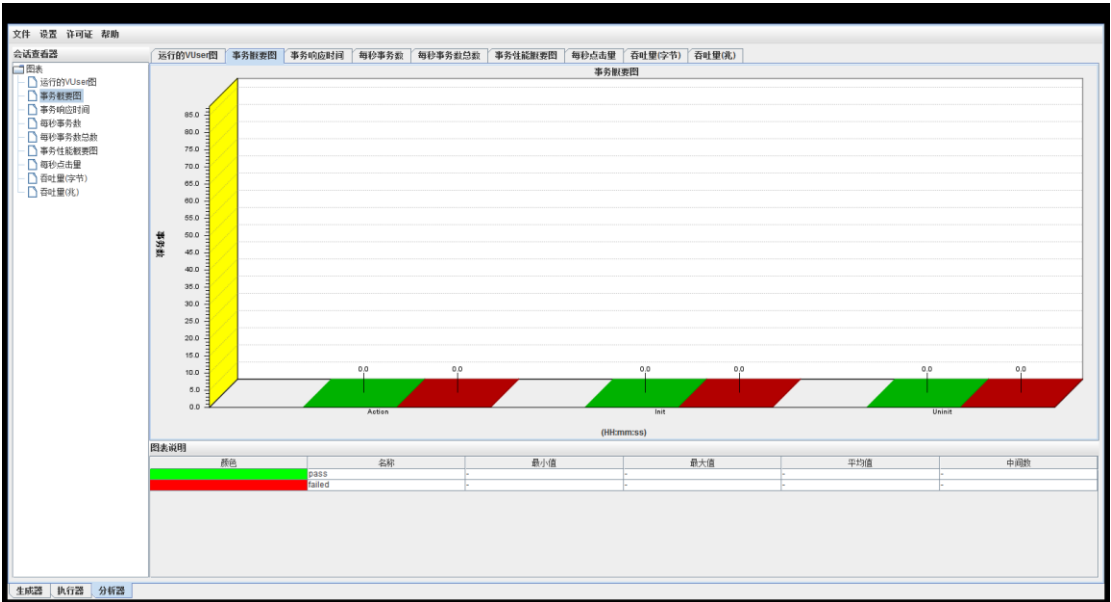
TC3	用户名-容错性验证	输入：aaabbbccc	输入到超出最大输入长度时，拒绝输入	输入超过规定的长度
TC4	密码-密码输入	输入与用户名相关联的数据： qqqq1111#	输入成功	
TC5	系统登录	TC2, TC4, 单击登录按钮	登录系统成功	
TC6	系统登录-用户名、密码校验	没有输入用户名、密码，单击登录按钮	系统登录失败并提示：用户名、密码不能为空	
TC7	系统登录-密码校验	输入用户名，没有输入密码，单击登录按钮	系统登录失败并提示：用户名、密码不能为空	
TC8	系统登录-密码有效性检验	输入用户名，输入密码与用户名不一致，单击登录按钮	系统登录失败并提示：出错了!! 用户名或密码错误	
TC9	系统登录-输入有效性检验	输入不存在的用户名、密码，单击登录按钮	系统登录失败并提示：出错了!! 用户不存在	

2、压力测试

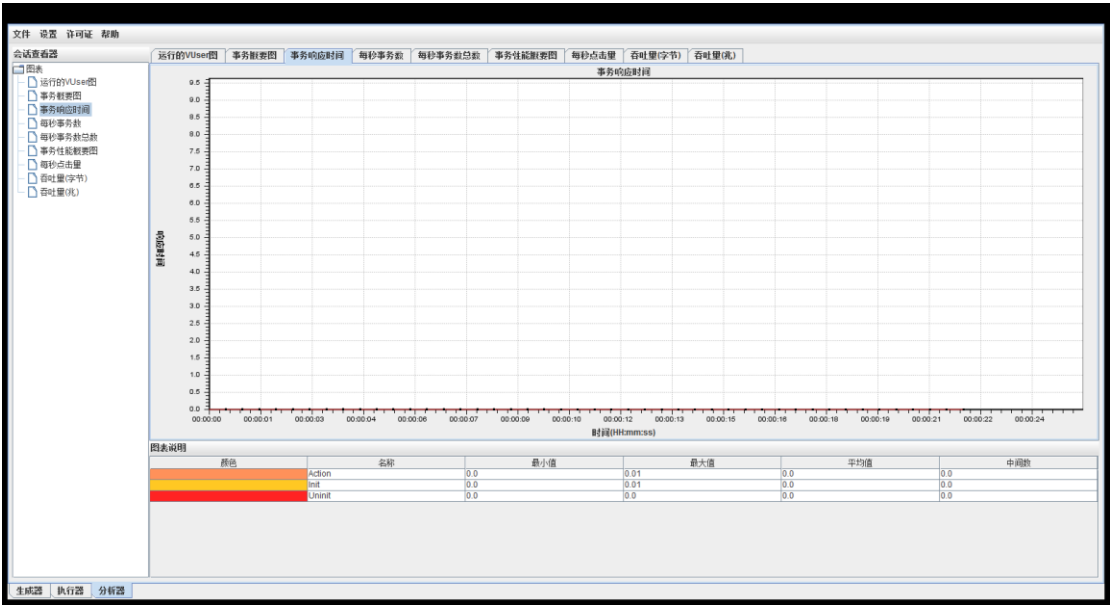
2.1、运行的 Vuser 图



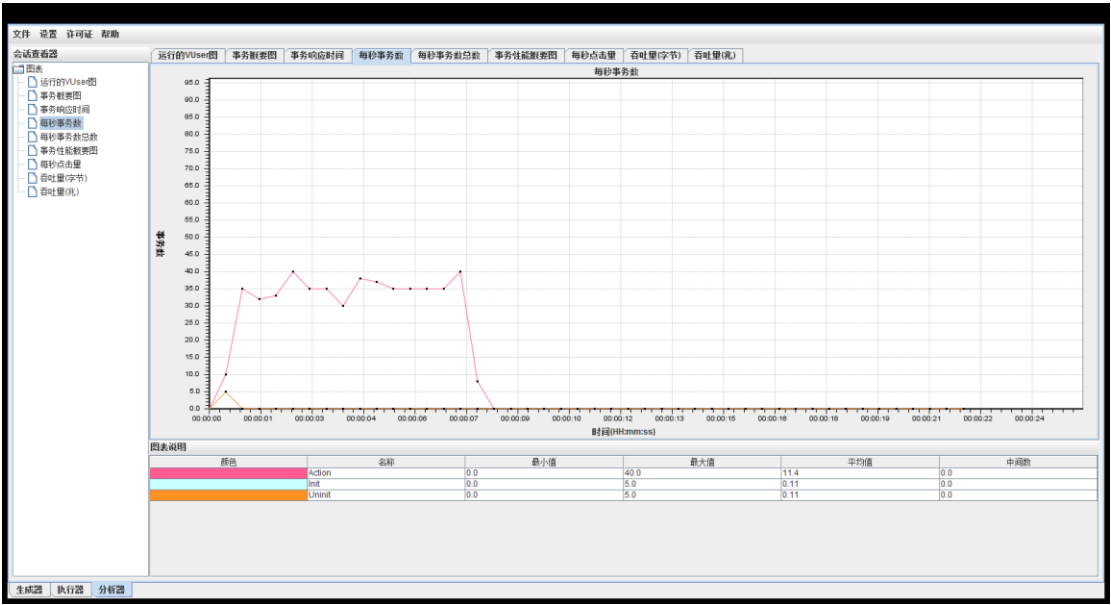
2.2、事物概要图



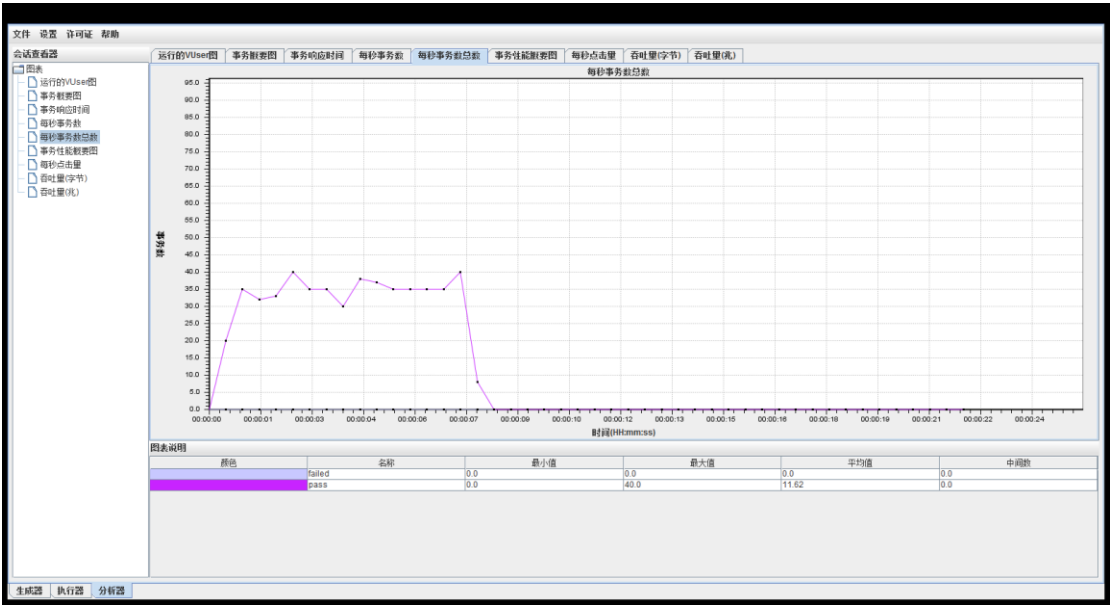
2.3、事物响应时间



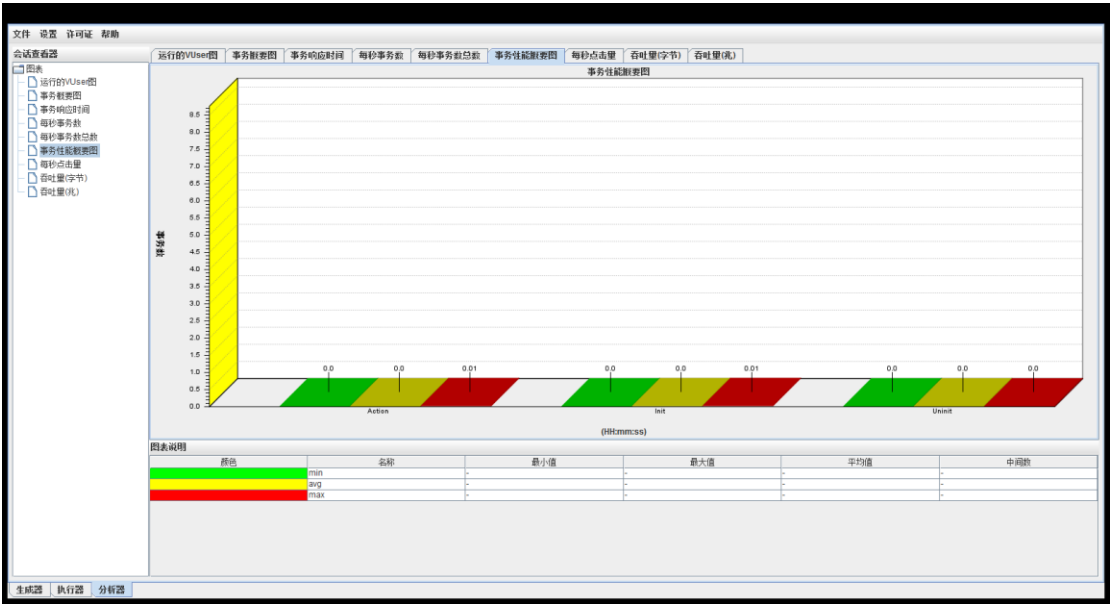
2.4、每秒事务数



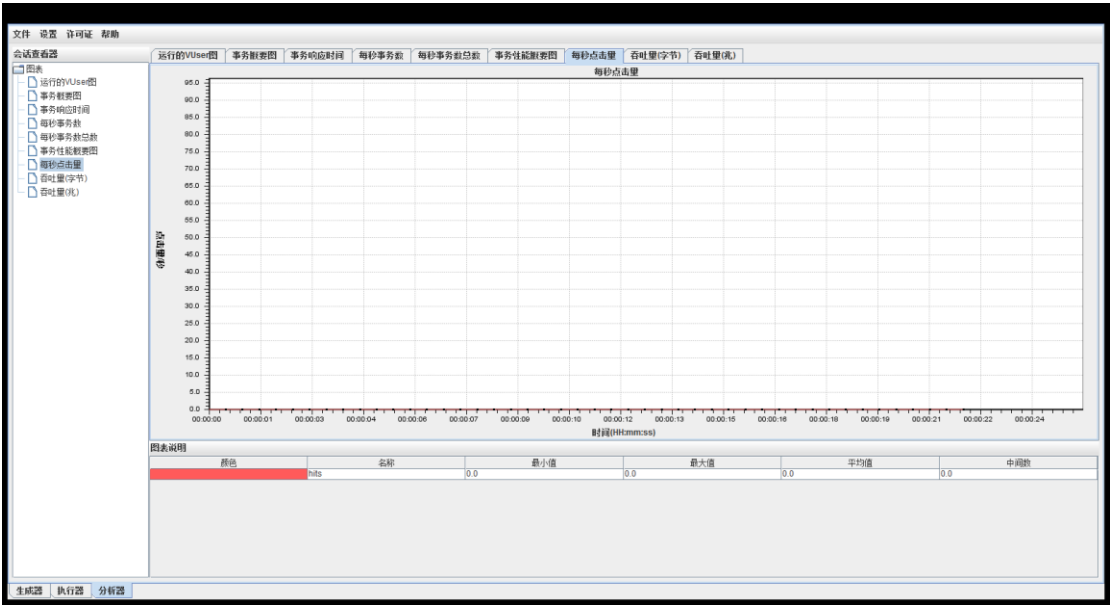
2.5、每秒事务总数



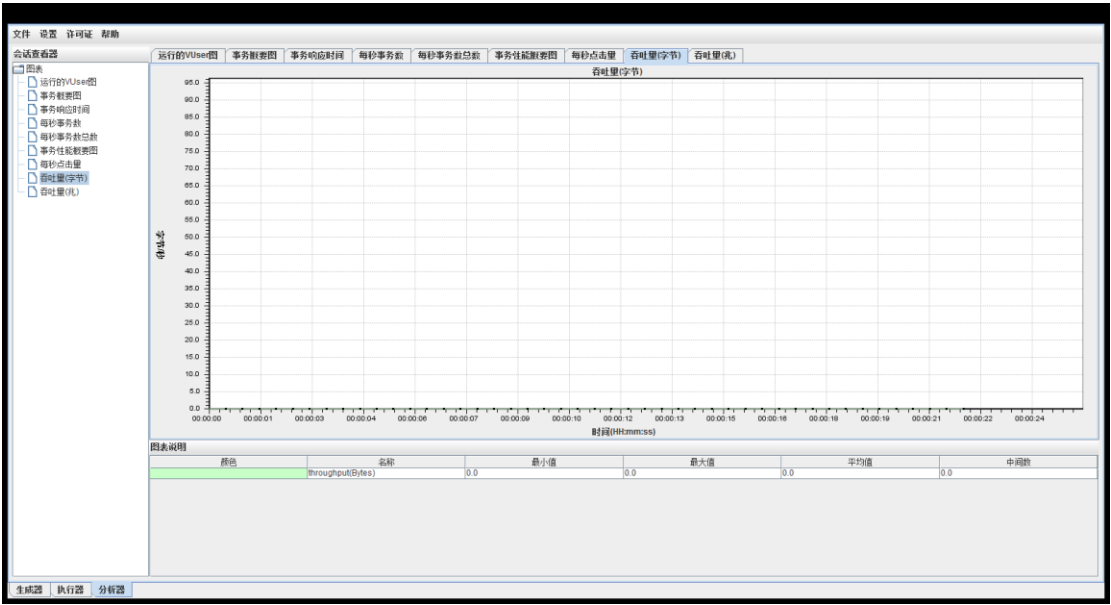
2.6、事务性能概要图



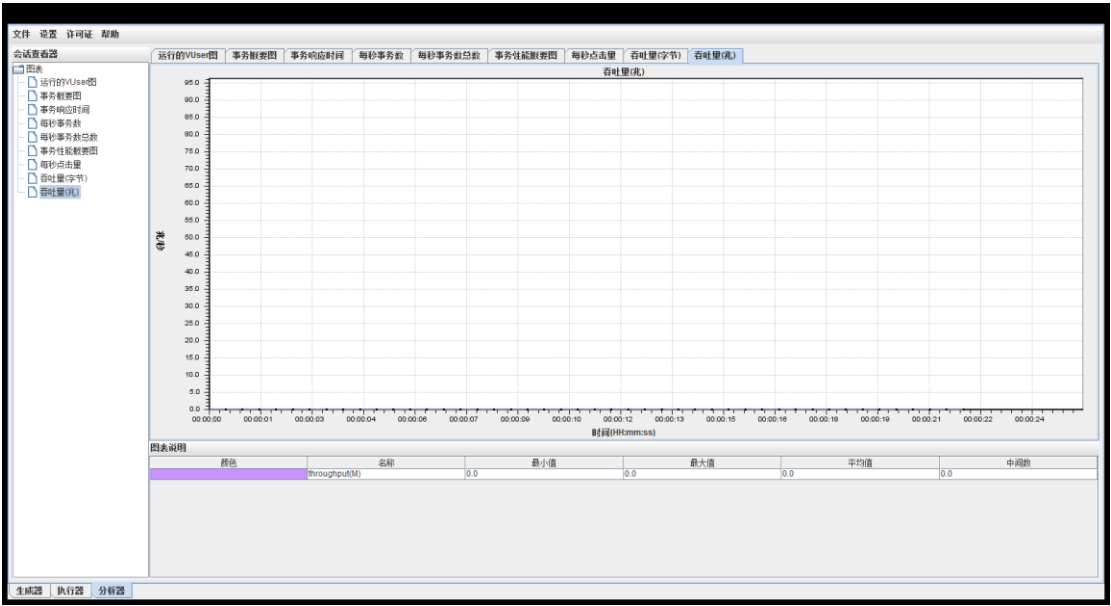
2.7、每秒点击量



2.8、吞吐量（字节）



2.9、吞吐量（兆）



七、课程设计心得

本次系统开发过程中，本项目完全由本人独立完成，在实现功能的时候，难免也会碰到很多问题，通过查阅相关资料得到顺利解决。通过本次实训，深刻了解了软件开发过程中软件设计和需求分析分中重要性，掌握了软件工程的设计思维方式。

尽管本系统很简洁，但是“麻雀虽小，五脏俱全”。用到的知识是自己通过日常的 java 及 Javaweb 学习所表现的。

从开始入手到写完实训报告，我收获很多，希望自己在以后的学习生活中，继续保持对中国软件事业保持高度的激情，为自己将来发展继续前行助力。

八、参考文献

[1] Robert Sedgewick.算法（谢路云） [M].北京：人民邮电出版社，2018

附录

本项目本人已上传至 github，欢迎 star。

地址：<https://github.com/yjh1227/websiteInfoManager>