



南昌大学实验报告

学生姓名： 杨孟衡 学 号： 8002118240 专业班级： 软工 1809
实验类型： ☒ 验证 ☐ 综合 ☐ 设计 ☐ 创新 实验日期： 5.14 实验成绩： _____

一、 实验项目名称

实验二 Web 应用登录功能

二、 实验要求

有登录页面，登录页面中用户名和密码有非空校验，后台连接数据库，登录成功与否则有信息提示。

实验环境： windows10 操作系统， mysql-8.0.16， Navicat-12.0， IDEA2019-3

三、 主要仪器设备及耗材

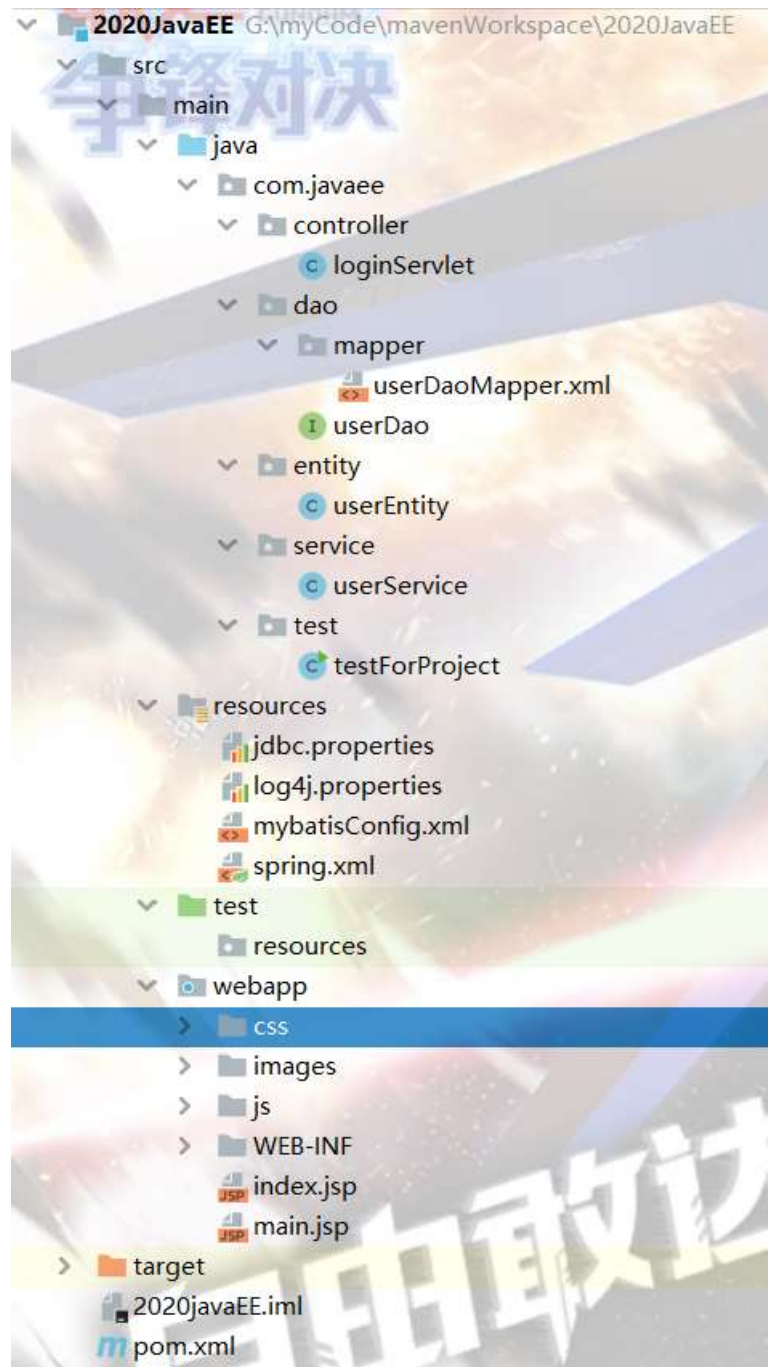
1. 截图主要代码及操作结果图，截图清晰。
2. 学习通平台提交实验报告。
3. 文件名： 班级+姓名+学号
4. 本实验含有工程源代码附件，实验报告仅保留关键代码，详情请查看源代码附件。
5. 本实验含有工程代码效果演示视频，视频文件在同文件夹下，以下是超链接：

[工程代码效果演示.wmv](#)

四、 实验内容和步骤

(1) 搭建实验环境，打开项目空间 ideaWorkSpace，新建一个 module 名为 2020JavaEE 的 maven 项目：

➤ 创建项目结构如下图所示：



➤ 其中，java 文件夹是用来编写后端服务器的，包括 controller（服务器端），dao（javabean 代理包），dao.mapper（方法映射文件，用于 mybatis 实现 jdbc 操作），entity（javabean 实体包），service（javabean 方法包），test（测试包），resources（源文件包，包含各种配置文件以及 xml 文件），test（其他测试包），webapp（包含 css 样式文件，js 文件，图片资源文件等），jsp 文件放在 main 文件夹下，方便我们访问；

（2）本实验采用 sspring-mybatis 框架的方式实现 jdbc 操作，首先引入项目依赖：

➤ pom.xml 文件代码请参看工程源代码，这里不再展示，主要就是 spring 与 mybatis 的集成环境；

（3）编写登录页面 index.jsp：

➤ 这里我是用 HBuilder X5 编写的 jsp 文件，是之前上课时编写的改了一些地方，然后添加到项目中，以下是源代码：

```

<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<head>
    <meta charset="utf-8" />
    <title>敢达 Club 登录</title>
    <meta name="keywords" content="敢达,高达,机甲,二次元,动漫,自由高达,奥布, 海盗 X1FC 高达, OOR 高达, 账号登录">
    <meta name="description" content="敢达粉丝俱乐部网">
    <meta name="author" content="旗木白哉(Gerald)">
    <link rel="stylesheet" href= "./css/login.css">
    <link rel="icon" href= "./images/图
层%200.ico" type="image/x-icon">
    <link rel="shortcut icon" href= "./images/图
层%200.ico" type="image/x-icon">
    <script type="text/javascript" src= "./js/login.js" charset="U
TF-8"></script>
    <script type="text/javascript">
        function checkLogin(){
            var obj1 = document.getElementById("form_username_in"
);
            var obj2 = document.getElementById("form_password_in"
);

            var flag1 = checkUserName(obj1);
            var flag2 = checkPass(obj2);
            var flag3 = document.getElementById("form_footer_in")
.checked;

            if(!flag1 || !flag2) {
                alert("Error:The userName or Password of input yo
u entered is wrong!Pls check and input again.");
                obj1.value = "";
                obj2.value = "";
                document.getElementById("form_footer_in").checked
= false;

                return false;
            }
            if(!flag3){
                alert("Error:The agreement needed to be chosen!")
;

                return false;
            }
            return true;
        }
    </script>

```

```

        <base href="login.html" target="_parent">
</head>
<body>
<div id="wrapper" class="">
    <div id="head"></div>
    <div id="navigation"></div>
    <div id="content">
        <div id="backgroundForIndex" class="login_background">
            
        </div>
    </div>
    <div id="body_footer" class="login_footer">
        <span class="footer_help">
            <a id="helpAnchor" class="footer_ache
">帮助中心</a>
            <span class="footer_help_line"></span>
        >
            <span class="footer_help_copyright">2
020 ©GundamClub</span>
        </span>
    </div>
    <div id="windowForRegister" class="login_window set_pos">
        <div class="login_title">
            <h3>登录</h3>
        </div>
        <div id="log_form" class="login_form">
            <form autocomplete="off" id="log_formbody" class="log
_01" action="http://localhost:8080/javaee/ee"
            target="_self" onsubmit="return checkLogin()" m
ethod="get">
                <p id="form_username" class="log_form_input set_p
os">
                    <label for="form_username_in" id="form_userna
me_label" class="form_username_in_label form_label_square">用户名
</label>
                    <input id="form_username_in" autocomplete="of
f" type="text" name="userAccount" class="form_input_index form_in
_square"
                        placeholder="请输入用户名
(6<=, <=12)" required="required" value=""/>
                </p>
                <p id="form_password" class="log_form_input set_p
os">

```

```

        <label for="form_password_in" id="form_passwo
rd_label" class="form_pass_in_label form_label_square">密码
</label>

        <input id="form_password_in" autocomplete="of
f" type="password" name="userPwd" class="form_input_index form_in
_square"

                placeholder="请输入登录密码
(6<=)" required="required" value="" />
        </p>
        <p id="form_submit" class="log_form_sub set_pos">
                <input id="form_submit_b" type="submit" name
="login" value="登录" class="form_submit_s" />
        </p>
        <p id="form_footer" class="form_ft set_pos">
                <input id="form_footer_in" class="form_ft_in"
type="checkbox" autocomplete="off" />
                <label for="form_footer_in">阅读并接受
</label>

        <a target="_blank" href="G:/GUNDAMHTML5/index
/index.html">《高达俱乐部用户协议》</a>
        <a target="_blank" href="G:/GUNDAMHTML5/index
/index.html">《高达俱乐部隐私权保护声明》</a>
        </p>
    </form>
</div>
</div>
</div>
<div id="footer"></div>
</div>
<script type="text/javascript">
    var helpBtn = document.getElementById("helpAnchor");
    helpBtn.onmouseover = function(){
        helpMOver(this);
    }
    helpBtn.onmouseout = function(){
        helpMOut(this);
    }
    helpBtn.onclick = function(){
        urlOpen("G:/GUNDAMHTML5/index/index.html");
    }
</script>
</body>
</html>

```

➤ 验证 jsp 页面编写无误，打开页面可以观察到如下图所示的页面：



(4) 接下来编写简易个人主页 main.jsp:

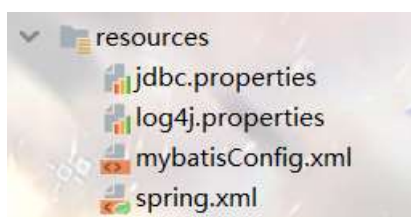
➤ 因为是一次简单的实验，在这里就不好好写了，写一个简易版本的 main.jsp，源代码如下：

```
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<body>
<h1>This is the main page!</h1>
<br>
<h2>User Account: ${user.userEntityAccount}</h2>
<h2>User Name: ${user.userEntityName}</h2>
<h2>User Password: ${user.userEntityPwd}</h2>
</body>
</html>
```

(5) 前端部分的最后准备工作，就是编写登录页面的渲染，包括 login.js, login.css，以及图片资源文件也不要忘了哦，在这里就不贴源代码了，详情查看附件：

(6) 现在来准备后端工作，JDBC 的连接，javaBean 的映射操作，数据库数据的映射操作等等：

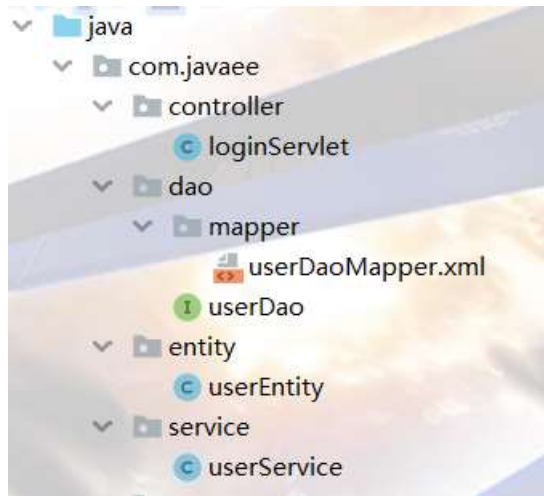
➤ 在 resources 文件夹下，新建四个配置文件，数据库的配置文件 jdbc，日志的配置文件，mybatis 的配置文件，spring 的配置文件：



➤ 具体编写参看附件源代码：

(7) 编写 java 代码部分：

➤ 编写 javabean 实体类 userEntity，javabean 代理类 userDao，javabean 服务类 userService，代理类的映射文件 userDaoMapper，后端服务器类 loginServlet：



➤ 在这里仅展示部分关键代码，比如代理类的映射文件，代理类 userDao，服务类 userService：

➤ 代理类的映射文件：

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
    PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
    "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.javaee.dao.userDao">
    <select id="queryUserByAccount" parameterType="String" resultType="userEntity">
        SELECT username 'userEntityName',userpwd 'userEntityPwd',useraccount 'userEntityAccount' FROM test.usertest WHERE useraccount=#{userAccount};
    </select>
</mapper>
```

➤ 代理类：

```
package com.javaee.dao;

import com.javaee.entity.userEntity;
import org.apache.ibatis.annotations.Param;

public interface userDao {
    public userEntity queryUserByAccount(@Param("userAccount") String userAccount);
}
```

➤ 服务类：

```
package com.javaee.service;

import com.javaee.dao.userDao;
import com.javaee.entity.userEntity;
import org.springframework.beans.factory.annotation.Autowired;
```

```

import org.springframework.stereotype.Service;

@Service
public class userService {
    @Autowired
    private userDao userDao;

    public boolean userLogin(String userAccount, String userPwd)
    {
        userEntity user = userDao.queryUserByAccount(userAccount
    );
        System.out.println(user);
        if(user != null && user.getUserEntityPwd().equals(userPw
d)) return true;
        return false;
    }

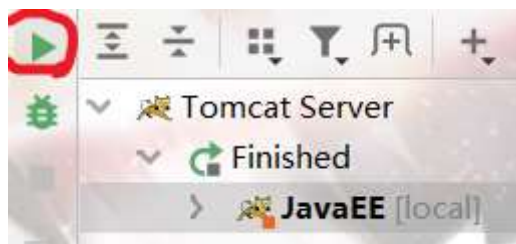
    public com.javaee.dao.userDao getUserDao() {
        return userDao;
    }
}

```

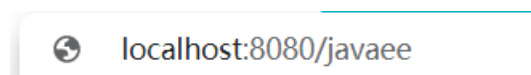
➤ 至此，项目前后端开发基本完毕，下一步就是配置 Tomcat 服务器即可，详细配置过程参看实验一报告：

五、实验数据及处理结果

➤ 实践项目开发，首先第一步运行服务器：

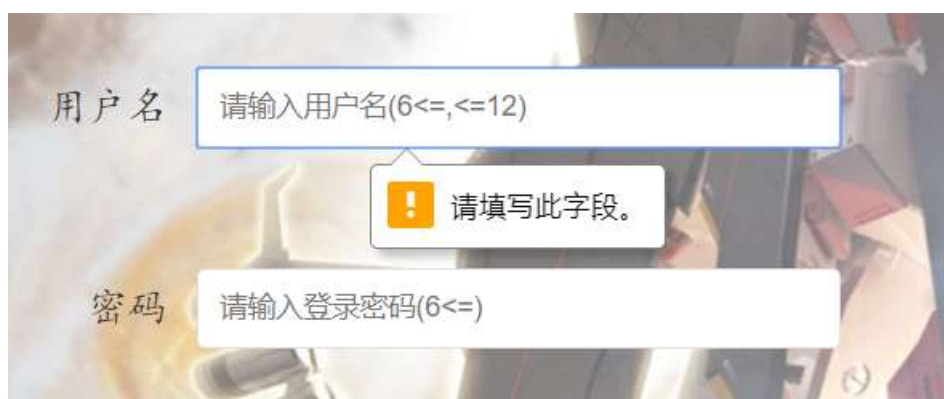


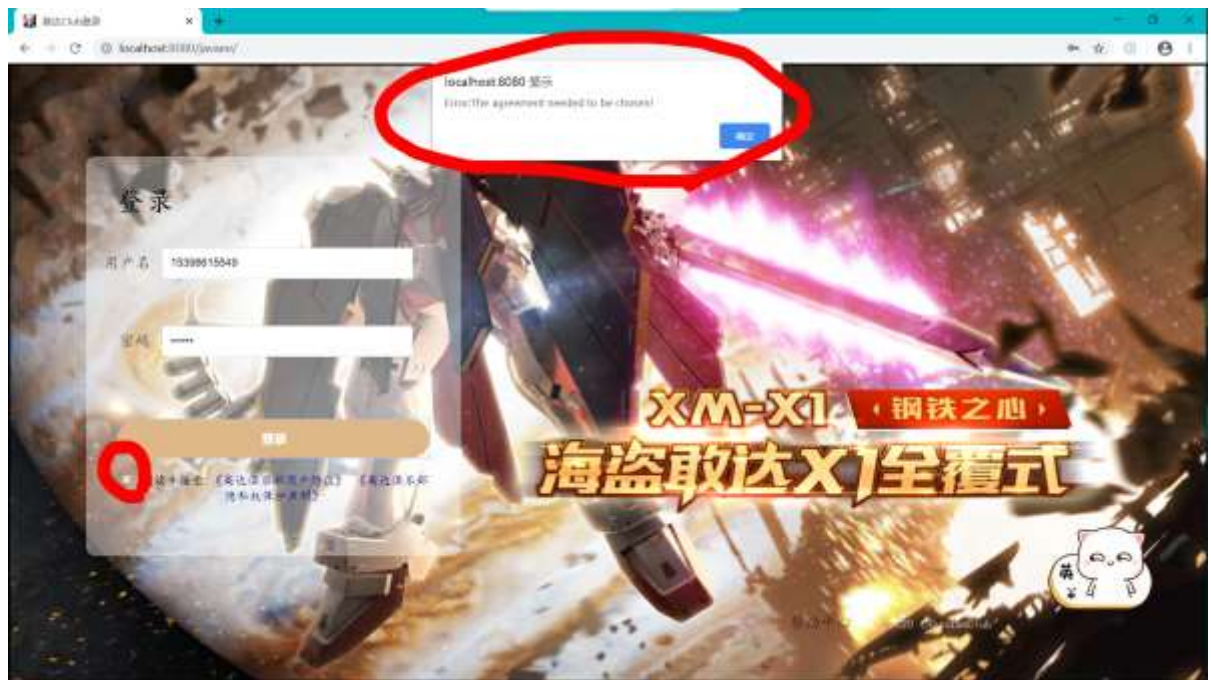
➤ 打开 Chrome，输入如图所示的网址，访问开发项目：



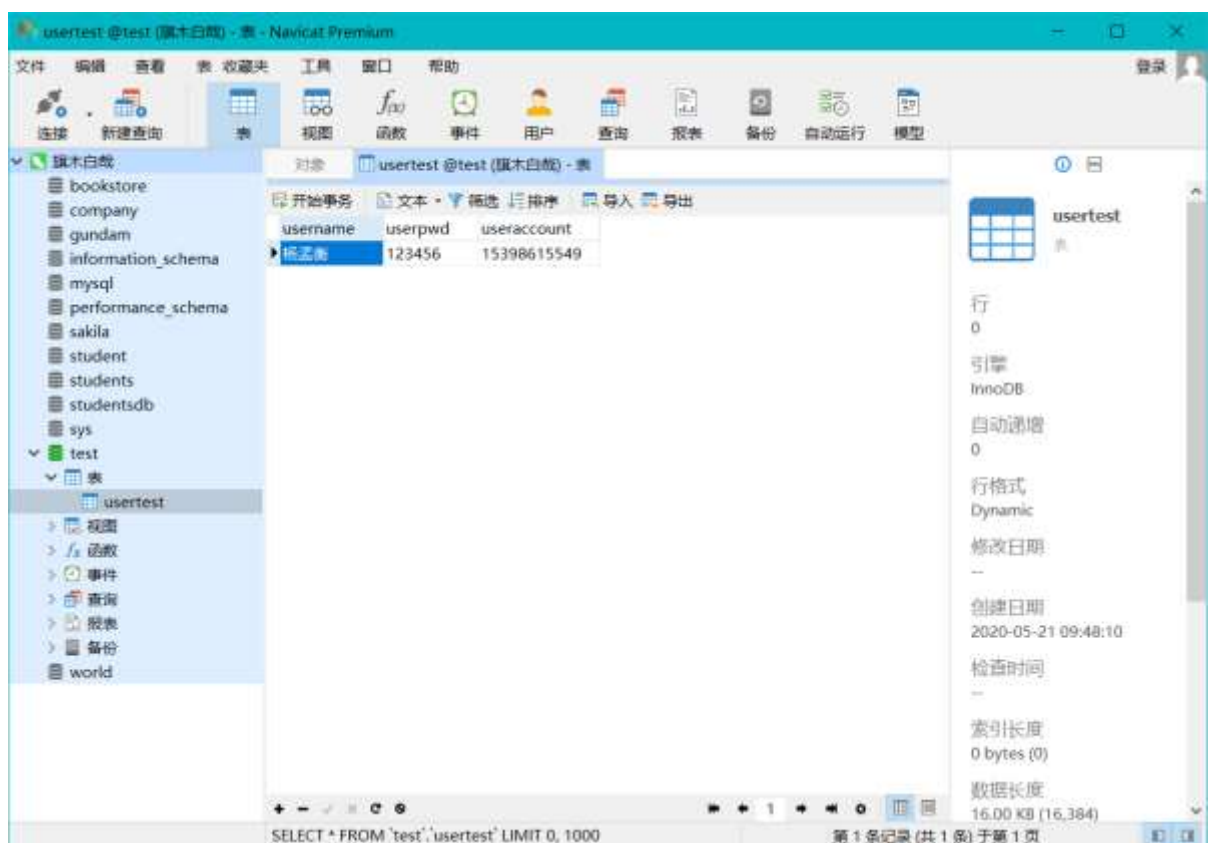


➤ 测试是否能正确如我们编写的那样，对用户的用户名，密码，协议等等有登录检查，并且有提示：





➤ 下一步检查后台能否正确连接数据库并且判断用户名与密码对应关系，如下图所示是测试数据库的数据表：



➤ 在用户名输入错误的用户名，密码输入错误的密码，若数据库不存在该用户，应刷新当前页面：

登录

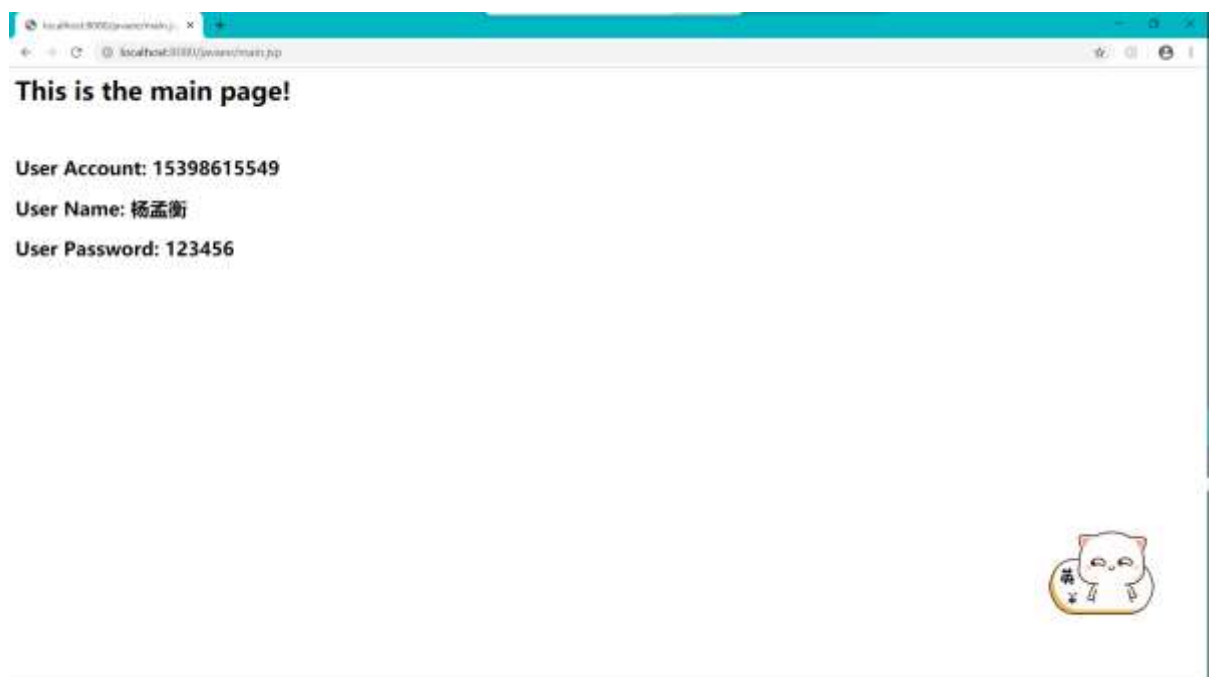
用户名 123456

密码

登录

☒ 阅读并接受 《高达俱乐部用户协议》 《高达俱乐部隐私权保护声明》

- 最后一步，正确登录用户，输入正确的用户名以及密码观察能否跳转到我们的简易主页：



➤ 至此，实验二测试完毕，正确实现了实验要求！

六、 思考、讨论或体会对改进实验的建议

本次实验是本学期 javaee 学习的第二次试验，难度适中，个人感受良好，从这次实验里复习了 IDEA 的 spring-mybatis 环境搭建基本操作，以及 servlet 服务器使用以及概念等等知识点，加深了我们对服务器后端的使用经验，进一步增强了对于 spring-mybatis 框架的应用能力。本次实验也没

有花太多的时间，主要还是一个复习巩固以及实操 `servlet`+框架的作用。反正作为一个程序猿，就是要抓细节！

七、 参考资料

[1]李建民 朱俊炎. 《JAVAE 工程实训教程》(学院自编), 中国商业出版社, 2011