组长:魏志航

组员:兰天阳，彭尹垚，张世纪，叶翔昊，谢泽川

**魏志航**

1.学习git基本命令，并在gitee创建项目库，并帮助组员完成配置、项目下载和项目上传。



2.学习HTML、CSS、Vue、Bootstrap、Layui等前端知识和框架

3.运用前端知识，初步完成活动界面的一些组件



**HTML：**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>活动 | 校园活动平台</title>

<link rel="icon" type="image/png" href="../image/logo.png" sizes="32x32">

<link href="//unpkg.com/layui@2.9.16/dist/css/layui.css" rel="stylesheet">

<link rel="stylesheet" href="activity.css">

</head>

<body>

<div>

<nav>

<div class="logo">

<img src="../image/logo.png" alt="Logo" /> <!-- 替换为您的 logo 图片路径 -->

</div>

<ul>

<li><a href="#">首页</a></li>

<li><a href="#"class="active">活动</a></li>

<li><a href="#">后台管理</a></li>

<li><a href="#">消息</a></li>

<li><a href="#">个人中心</a></li>

</ul>

</nav>

</div>

<div class="mainbody-area">

<div class="mainbody">

<div><h3>默认面包屑</h3>

<span class="layui-breadcrumb">

<a href="">首页</a>

<a href="">演示</a>

<a><cite>导航元素</cite></a>

</span>

<h3>自定义分隔符</h3>

<span class="layui-breadcrumb" lay-separator=">">

<a href="">首页</a>

<a href="">活动</a>

<a href="">分类</a>

<a><cite>详情</cite></a>

</span>

<h3>分类</h3>

<span class="layui-breadcrumb" lay-separator="|">

<a href="">体育</a>

<a href="">学术</a>

<a href="">桌游</a>

<a href="">出游</a>

<a href="">艺术</a>

<a href="">手游</a>

<a href="">端游</a>

</span>

<script src="//unpkg.com/layui@2.9.16/dist/layui.js"></script>

</div>

</div>

</div>

</body>

</html>

**CSS：**

/\* 基本重置 \*/

body {

margin: 0;

font-family: Arial, sans-serif; /\* 字体设置 \*/

}

/\* 导航栏样式 \*/

nav {

background-color: #f9e71e; /\* 浅黄色背景 \*/

padding: 20px 20px; /\* 内边距 \*/

display: flex; /\* 使用 Flexbox 以便于对齐 logo 和列表 \*/

align-items: center; /\* 垂直居中对齐内容 \*/

justify-content: center; /\* 水平居中对齐所有内容 \*/

height: 50px; /\* 高度 \*/

}

.mainbody-area

{

background-color: hsl(60, 65%, 91%); /\* 浅色背景 \*/

padding: 20px 0; /\* 顶部和底部的内边距 \*/

}

.mainbody

{

background-color: #fff; /\* 白色背景 \*/

margin: 0 auto; /\* 居中对齐 \*/

padding: 20px; /\* 内容内边距 \*/

border-radius: 10px; /\* 圆角边框 \*/

max-width: 1000px; /\* 最大宽度, 根据需要调整大小 \*/

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); /\* 阴影效果 \*/

}

/\* Logo 样式 \*/

.logo img {

width: 50px; /\* 设置 logo 图片的宽度，您可以根据需要进行调整 \*/

height: auto; /\* 保持比例 \*/

margin-right: 10px; /\* logo 与导航项之间的间距 \*/

}

/\* 列表样式 \*/

nav ul {

list-style-type: none; /\* 去掉默认列表样式 \*/

padding: 0; /\* 去掉列表内边距 \*/

margin: 0; /\* 去掉列表外边距 \*/

display: flex; /\* 使用 Flexbox 进行布局 \*/

justify-content: center; /\* 水平居中 \*/

}

/\* 列表项样式 \*/

nav li {

margin: 0 15px; /\* 列表项之间的横向间距 \*/

}

/\* 链接样式 \*/

nav a {

text-decoration: none; /\* 去掉下划线 \*/

color: #333; /\* 文字颜色 \*/

padding: 10px 15px; /\* 内边距，使链接更大 \*/

border-radius: 5px; /\* 圆角效果 \*/

font-size: 22px; /\* 设置更大的字体大小，这里提高到原来的两倍 \*/

font-weight: bold; /\* 加粗字体 \*/

transition: background-color 0.3s; /\* 背景颜色过渡效果 \*/

}

/\* 选中的链接样式 \*/

nav a.active {

background-color: #f5f3ef; /\* 选中状态的背景颜色 \*/

color: #000; /\* 选中的文字颜色 \*/

}

/\* 鼠标悬停效果 \*/

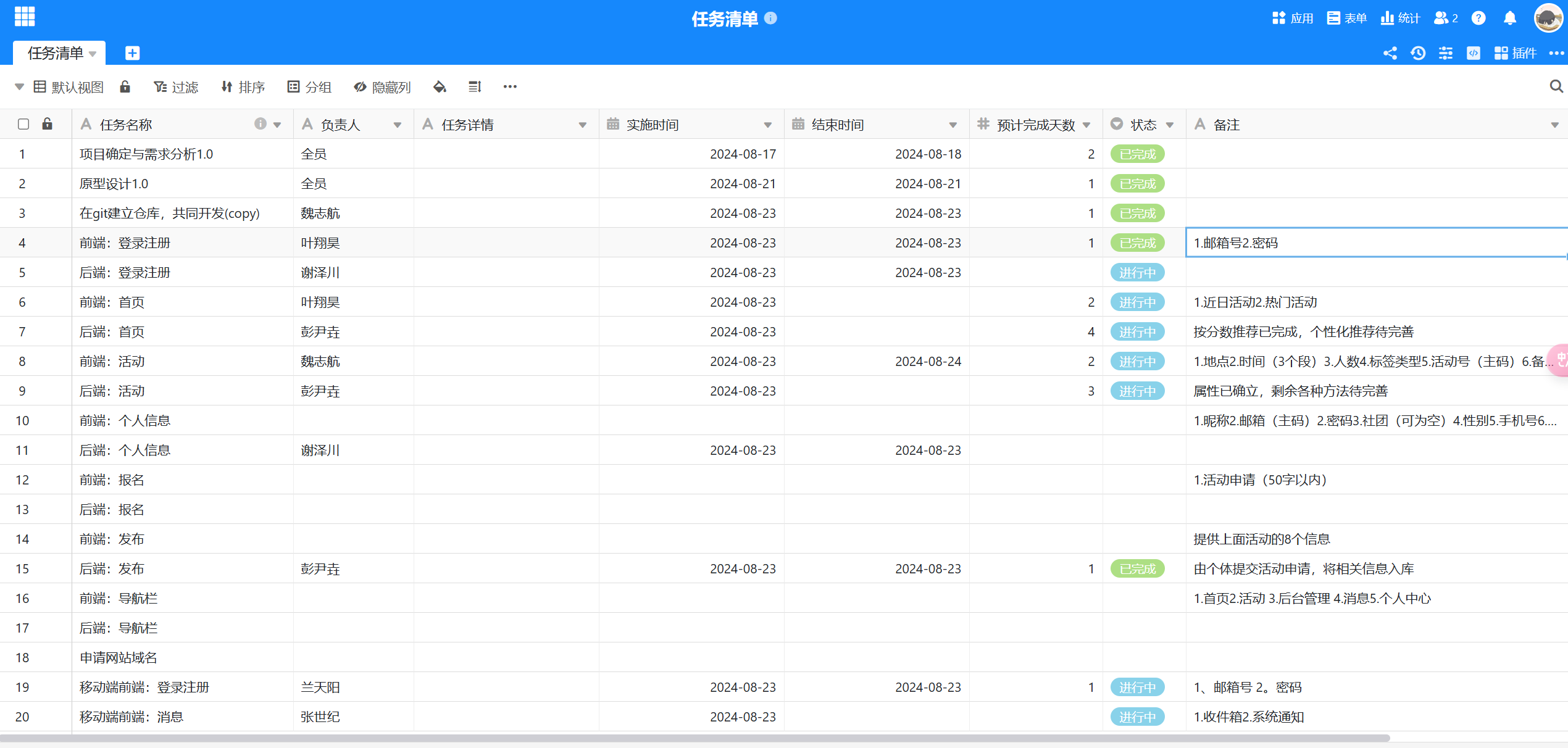
nav a:hover {

background-color: #e8f5e9; /\* 鼠标悬停时改变背景颜色 \*/

color: #000; /\* 鼠标悬停时改变文字颜色 \*/

}

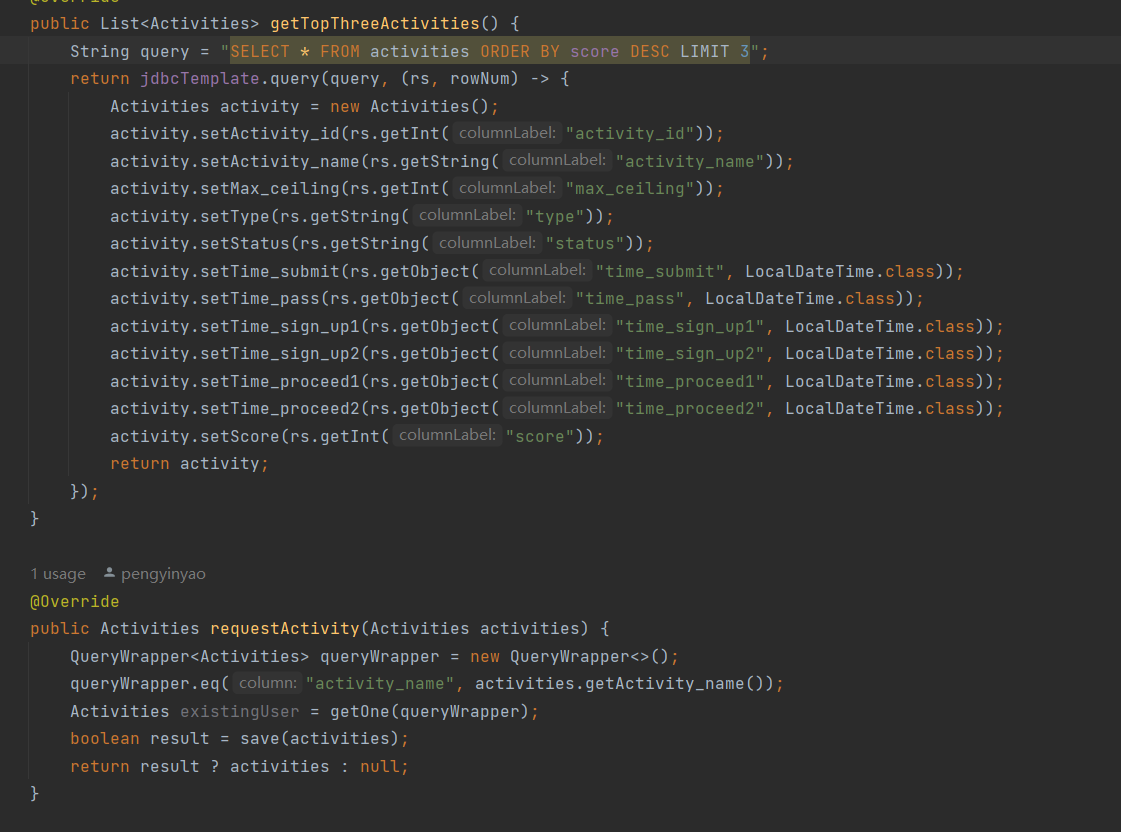
4.创建初步的任务清单

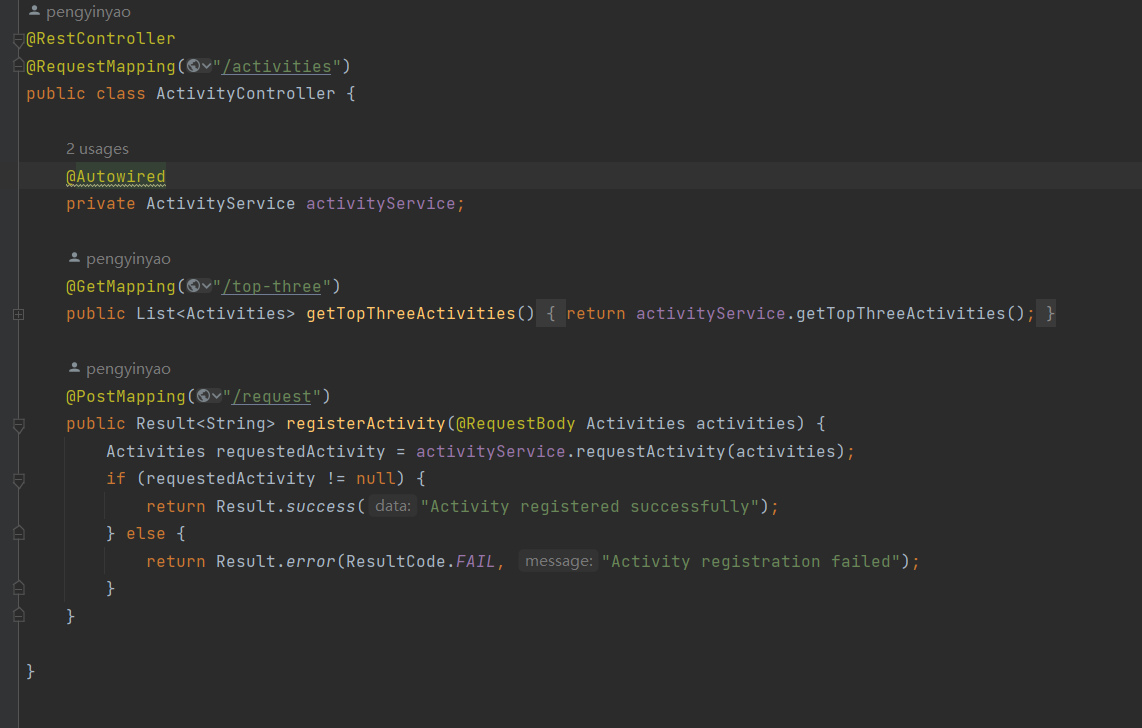


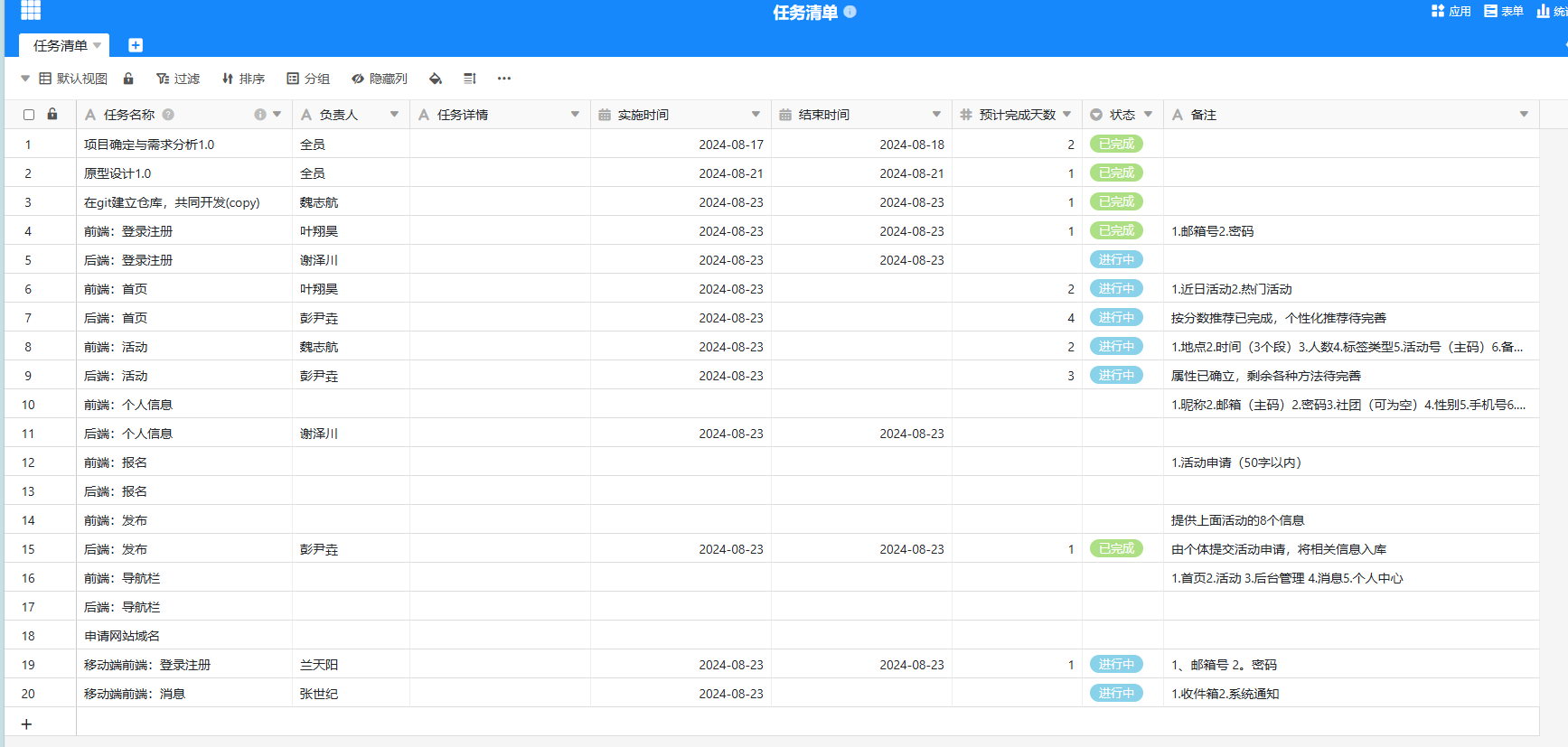
**彭尹垚**

Java日志

今天针对activity对象进行了建库，对活动上报的方法分activity接口，activityimpl实现，activitycontroller与网页对接，实现了活动申请和按评分输出前三的活动的功能。







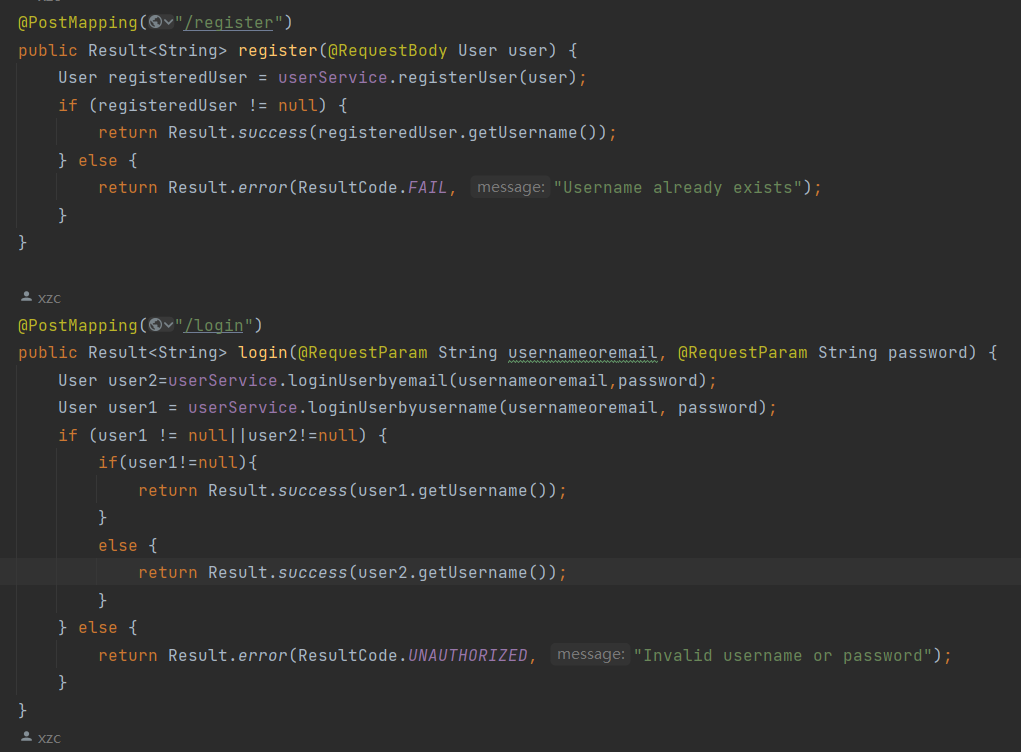
**谢泽川**

##### Java学习日志 - JAVASpringboot对校园活动的编写实现

##### 日期：2024年8月23日

##### 实现内容：

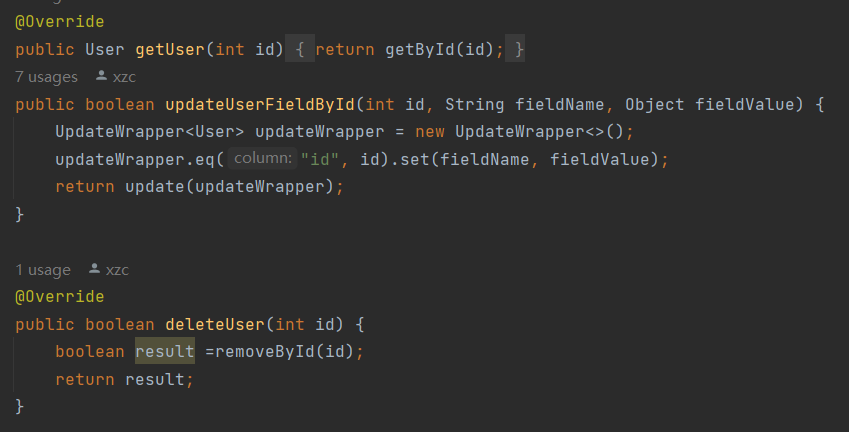
用户登录注册内容的实现：





个人中心界面的实现：

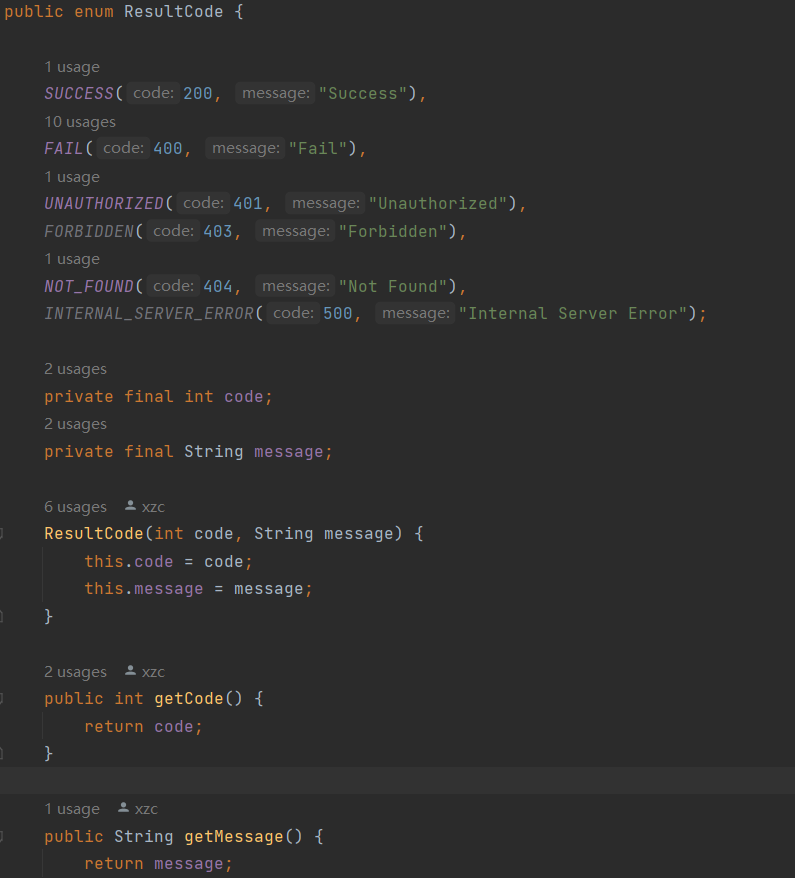
账户信息显示和账户的注销：



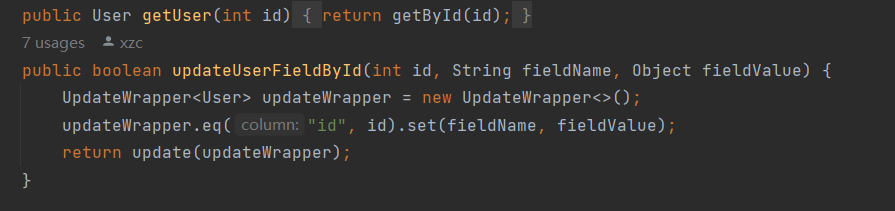


输出码的编写和统一：





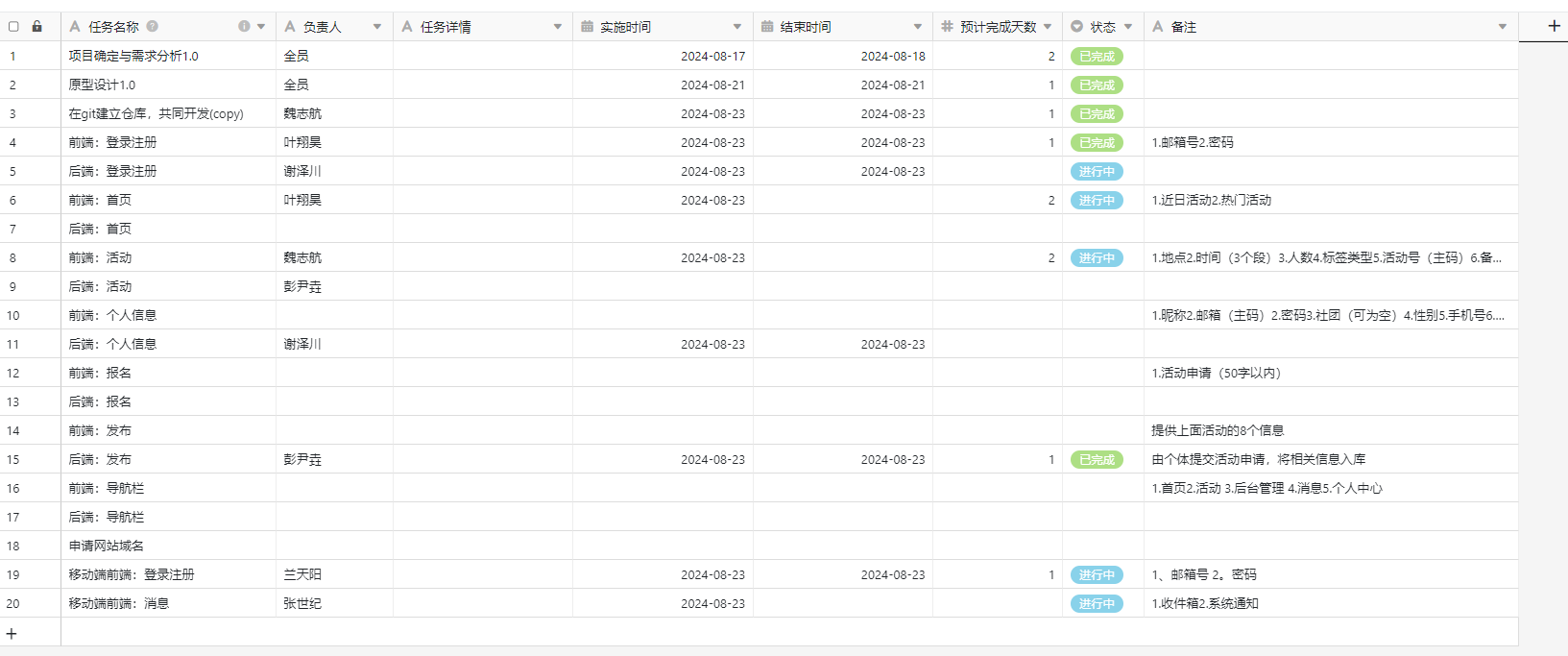
个人信息的修改：





**张世纪**

初步拟定分工，进度寥寥



**叶翔昊**

8·21实训日志

总体情况：作为前端开发完成了登录注册界面，并成功与后端完成对接。

代码：

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>用户登录</title>

<style>

html, body {

height: 100%;

margin: 0;

padding: 0;

font-family: Arial, sans-serif;

overflow: hidden;

}

body {

background: url('your-image-url.jpg') no-repeat center center fixed;

background-size: cover;

position: relative;

min-height: 100vh;

}

body::before {

content: "";

position: absolute;

top: 0;

left: 0;

width: 100%;

height: 100%;

background-color: rgba(0, 0, 0, 0.5);

z-index: -1;

}

.header {

text-align: center;

color: white;

margin-top: 50px;

font-size: 2em;

text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);

}

.container {

max-width: 400px;

margin: 100px auto;

padding: 20px;

background-color: rgba(255, 255, 255, 0.9);

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

position: relative;

width: 100%;

height: auto;

}

h2 {

text-align: center;

}

label {

display: block;

margin-top: 10px;

}

input {

width: 100%;

padding: 8px;

margin-top: 5px;

box-sizing: border-box;

}

button {

width: 100%;

padding: 10px;

background-color: #28a745;

color: white;

border: none;

margin-top: 20px;

cursor: pointer;

}

button:hover {

background-color: #218838;

}

.error {

color: red;

margin-top: 10px;

text-align: center;

}

.switch-page {

text-align: center;

margin-top: 20px;

}

.switch-page a {

color: #007bff;

text-decoration: none;

}

.switch-page a:hover {

text-decoration: underline;

}

</style>

</head>

<body>

<div class="header">学生活动发布系统</div>

<div class="container">

<h2>用户登录</h2>

<form id="loginForm">

<label for="loginUsername">用户名：</label>

<input type="text" id="loginUsername" placeholder="请输入用户名" required>

<label for="loginPassword">密码：</label>

<input type="password" id="loginPassword" placeholder="请输入密码" required>

<button type="submit">登录</button>

<p class="error" id="loginError"></p>

</form>

<div class="switch-page">

<p>没有账户？<a href="register.html">点此注册</a></p>

</div>

</div>

<script>

document.getElementById('loginForm').addEventListener('submit', async function (event) {

event.preventDefault();

const username = document.getElementById('loginUsername').value;

const password = document.getElementById('loginPassword').value;

try {

const response = await fetch('/auth/login', {

method: 'POST',

headers: {

'Content-Type': 'application/json'

},

body: JSON.stringify({ username, password })

});

const result = await response.json();

if (response.ok) {

alert('登录成功，欢迎：' + result.data);

} else {

document.getElementById('loginError').textContent = result.message || '登录失败';

}

} catch (error) {

document.getElementById('loginError').textContent = '网络错误，请稍后重试';

}

});

</script>

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>用户注册</title>

<style>

html, body {

height: 100%;

margin: 0;

padding: 0;

font-family: Arial, sans-serif;

overflow: hidden;

}

body {

background: url('your-image-url.jpg') no-repeat center center fixed;

background-size: cover;

position: relative;

min-height: 100vh;

}

body::before {

content: "";

position: absolute;

top: 0;

left: 0;

width: 100%;

height: 100%;

background-color: rgba(0, 0, 0, 0.5);

z-index: -1;

}

.header {

text-align: center;

color: white;

margin-top: 50px;

font-size: 2em;

text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);

}

.container {

max-width: 400px;

margin: 100px auto;

padding: 20px;

background-color: rgba(255, 255, 255, 0.9);

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

position: relative;

width: 100%;

height: auto;

}

h2 {

text-align: center;

}

label {

display: block;

margin-top: 10px;

}

input {

width: 100%;

padding: 8px;

margin-top: 5px;

box-sizing: border-box;

}

button {

width: 100%;

padding: 10px;

background-color: #28a745;

color: white;

border: none;

margin-top: 20px;

cursor: pointer;

}

button:hover {

background-color: #218838;

}

.error {

color: red;

margin-top: 10px;

text-align: center;

}

.switch-page {

text-align: center;

margin-top: 20px;

}

.switch-page a {

color: #007bff;

text-decoration: none;

}

.switch-page a:hover {

text-decoration: underline;

}

</style>

</head>

<body>

<div class="header">学生活动发布系统</div>

<div class="container">

<h2>用户注册</h2>

<form id="registerForm">

<label for="registerUsername">用户名：</label>

<input type="text" id="registerUsername" placeholder="请输入用户名" required>

<label for="registerEmail">邮箱：</label>

<input type="email" id="registerEmail" placeholder="请输入邮箱" required>

<label for="registerPassword">密码：</label>

<input type="password" id="registerPassword" placeholder="请输入密码" required>

<label for="confirmPassword">确认密码：</label>

<input type="password" id="confirmPassword" placeholder="请再次输入密码" required>

<button type="submit">注册</button>

<p class="error" id="registerError"></p>

</form>

<div class="switch-page">

<p>已有账户？<a href="login.html">点此登录</a></p>

</div>

</div>

<script>

document.getElementById('registerForm').addEventListener('submit', async function (event) {

event.preventDefault();

const username = document.getElementById('registerUsername').value;

const email = document.getElementById('registerEmail').value;

const password = document.getElementById('registerPassword').value;

const confirmPassword = document.getElementById('confirmPassword').value;

if (password !== confirmPassword) {

document.getElementById('registerError').textContent = '两次输入的密码不一致';

return;

}

// 简单的邮箱格式验证

const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

if (!emailRegex.test(email)) {

document.getElementById('registerError').textContent = '邮箱格式不正确';

return;

}

try {

const response = await fetch('/auth/register', {

method: 'POST',

headers: {

'Content-Type': 'application/json'

},

body: JSON.stringify({ username, email, password })

});

const result = await response.json();

if (response.ok) {

alert('注册成功，用户名：' + result.data);

} else {

document.getElementById('registerError').textContent = result.message || '注册失败';

}

} catch (error) {

document.getElementById('registerError').textContent = '网络错误，请稍后重试';

}

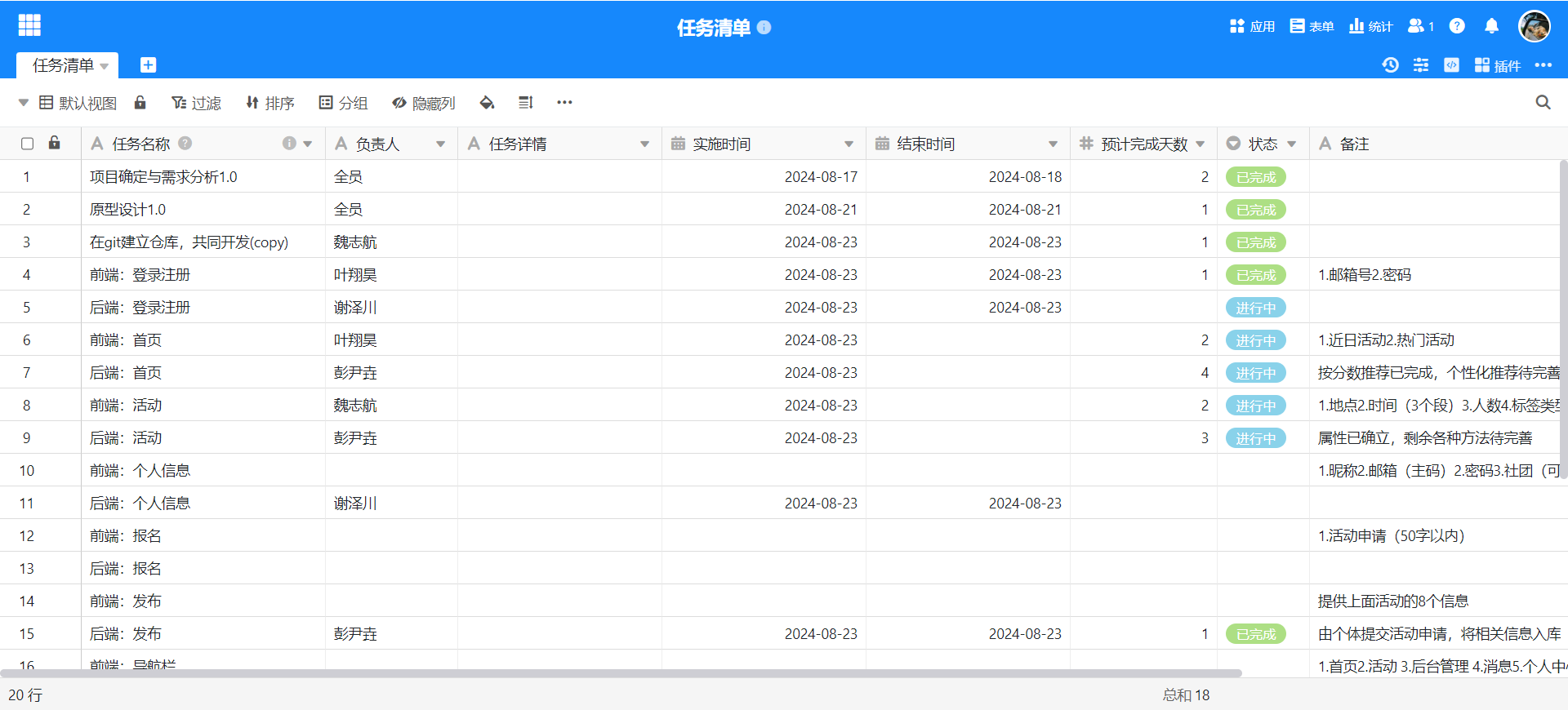
});

</script>

</body>

</html>

任务清单完成情况：



**兰天阳**

2024.8.23

编写前端登录与注册页代码，实现界面跳转





