

信息工程学院

程序设计类课程设计报告

**设计名称： 前端开发课程设计**

**设计题目： 青少年体态健康中心**

**学生学号： 2507230105**

**专业班级： 软件工程2301班**

**学生姓名： 严泽昊**

**学生成绩：**

**指导教师： 任强**

**工作时间： 2025-05-06 至 2025-05-20**

**说明**

1．报告中的任务书由指导教师在课程设计开始前填写并发给每个学生。

2．学生成绩由指导教师根据学生的设计情况给出各项分值及总评成绩。

3．所有学生必须参加课程设计的答辩环节，凡不参加答辩者，其成绩一律按不及格处理。答辩的时间、地点、流程由指导教师实施。

4．报告正文字数一般应不少于3000字，也可由指导教师根据本门综合设计的情况另行规定。

**信息工程学院 程序设计类课程设计任务书**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **学生姓名** | 严泽昊 | **学号** | 2507230105 | **专业班级** | 软件工程2301 |
| **主要任务及设计要求：**  1.主题。制作一个健康、积极、实用的自定主题的网站。内容较详实并围绕主题。  2.界面。网站所有页面风格统一、布局合理、简洁美观，每个页面都应由页头(网站名、导航栏)、内容区、侧边栏(可选项)、页尾等部分组成。  3.结构。每人网页文件(.html)至少5个，按照功能或文件类别建立子文件夹。CSS和JavaScript必须是外部文件形式引入HTML。CSS和JavaScript外部文件数自定但需要规划清晰。  4. HTML要求。元素结构合理，标签按语义使用恰当且多样，属性运用得当多样。  5. CSS要求。选择器和属性运用合理无冗余，使用的知识点多，代码可维护性高。流、Flex、定位均至少使用一次。动画、透明度、阴影、圆角边框等应有体现。  6. JavaScript要求。知识点运用的越多越好，每人实现5个功能或效果：  (1) 使用DOM和事件操作元素、属性和内容，改变外观、提高用户体验等共1次。例如增删留言、Tab选项卡等。  (2) 使用表单和控件的常用DOM属性、方法、事件，表单前端验证至少1次。  (3) 使用BOM调用浏览器功能至少1次，例如页面滚动、秒杀倒计时、打印、通知、本地存储等。  (4) 使用Ajax或fetch获取第三方网站数据并更新本网页内容至少1次。  (5) 使用模块脚本文件实现上述效果至少1次。  7.使用第三方库和框架的要求。  (1) 使用ECharts库(https://echarts.apache.org)实现数据可视化1次。  (2) 使用Swiper框架(https://www.swiper.com.cn)实现轮播图1次。  (3) 使用Vue框架的v-if、v-for、v-on、v-model之一实现1个简单效果。  8. 代码的30%可参考其它网站，70%应由自己编写。每人JS代码至少300行。  9. 报告字数不少于3000字，结构合理，文字表达清晰，格式规范。 | | | | | |
| **指导教师：**   **日 期：**2025年04月30日 | | | | | |

**成绩评定表**

学生姓名： 严泽昊 学号： 2507230105 专业/班级： 软件工程2301

|  |  |  |  |
| --- | --- | --- | --- |
| 类别 | 合计  分值 | 评分标准 | 实际得分 |
| 代码清单 | 20 | 完成了任务书的全部要求。  代码结构清晰，涉及的知识面广，具有一定的可维护性和扩展性。  算法合理且经过适当的优化。 |  |
| 报告质量 | 40 | 报告文字通顺，内容翔实，论述充分、完整，立论正确，结构严谨合理；报告字数符合相关要求，工整规范，整齐划一。  课题背景介绍清楚，综述分析充分。  设计方案合理、可行，论证严谨，逻辑性强，具有说服力。  符号统一；图表完备、符合规范要求。  能对整个设计过程进行全面的总结，得出有价值的结论或结果。 |  |
| 答辩情况 | 40 | 在规定时间内能就所设计的内容进行阐述，言简意明，重点突出，论点正确，条理清晰。  在规定时间内能准确、完整、流利地回答教师所提出的问题。 |  |
| 指导教师：  （签字） 总评成绩： 分  日 期： 2025 年 05 月 22 日 | | | | |

**目 录**

[1 设计简介 1](#_Toc198876456)

[1.1 背景和意义 1](#_Toc198876457)

[1.2 主要技术 1](#_Toc198876458)

[1.3 安装和使用 2](#_Toc198876459)

[1.4 页面效果功能图 2](#_Toc198876460)

[2 详细设计 4](#_Toc198876461)

[2.1 首页 4](#_Toc198876462)

[2.2 生活习惯页面 6](#_Toc198876463)

[2.3饮食与体态界面 7](#_Toc198876464)

[2.4 趣味知识页面 9](#_Toc198876465)

[2.5体态问题咨询页面 10](#_Toc198876466)

[3 项目总结 12](#_Toc198876467)

# 1 设计简介

1.1 背景和意义

青少年群体的体态健康问题已成为当前社会关注的重要公共卫生议题。随着电子设备的普及和学习压力的增大，我国青少年体态异常的检出率已超过30%，其中脊柱侧弯、驼背、骨盆前倾等问题尤为突出。这些体态问题不仅影响青少年的外在形象，更可能导致慢性疼痛、呼吸功能障碍等严重后果。

通过这个平台，我们希望能够提升全社会对青少年体态健康的重视程度，让更多青少年养成良好的体态习惯，以更健康、更自信的姿态迎接未来挑战。

1.2 主要技术

主要使用了Html,CSS,JavaScript三种语言，此外，还引入Vue.js框架开发动态组件，使用Echarts库实现数据可视化展示，并集成Swiper库打造流畅的轮播效果。这些技术的有机结合，既保证了功能完整性，又提升了用户体验。

1.2.1 Vue.js框架

Vue.js 是一个轻量级的前端 JavaScript 框架，用于构建用户界面和单页应用（SPA）。它采用​​响应式数据绑定​​和​​组件化开发​​，使开发者能够高效地创建可复用的 UI 组件。Vue 的核心优势在于其**​​**渐进式架构**​**​，可以灵活地从小型项目扩展到大型企业级应用，同时提供​**​**虚拟 DOM**​**​ 优化渲染性能。Vue 生态丰富，支持 Vuex（状态管理）、Vue Router（路由）等工具，适合开发 Web 应用、管理后台和动态网站.

1.2.2 ECharts​库

ECharts 是由百度开源的一款强大的​**​**数据可视化库**​**​，用于创建交互式图表和大屏数据展示。它支持多种图表类型，如折线图、柱状图、饼图、地图等，并提供丰富的配置选项，包括动画、主题、数据缩放等。ECharts 特别适合处理​**​**大规模数据**​**​，并能在 PC 和移动端流畅运行，广泛应用于数据分析、实时监控和商业报表系统。

1.2.3 Swiper​库

Swiper 是一个专注于​**​**滑动交互​​的 JavaScript 库，常用于实现轮播图、图片画廊、幻灯片等效果。它支持​**​**触摸滑动、手势操作、自动播放**​**​，并内置分页器、懒加载、3D 过渡动画等功能。Swiper 兼容多种前端框架（如 Vue、React），同时也能在原生 JavaScript 项目中使用，是开发响应式滑动组件的首选工具。

1.2.4 HTML

HTML是用于构建网页内容结构的基础语言，它通过一系列的标签来定义网页中的各种元素，例如标题、段落、链接、图片等。这些标签就像是网页的骨架，将文本、图像和其他媒体内容组织起来，为用户呈现出清晰的页面结构。HTML的核心作用是确保网页内容的语义化表达，让浏览器能够正确理解并展示网页的基本信息。

1.2.5 CSS

CSS是专门用于控制网页视觉效果和样式的语言。它通过定义一系列的样式规则，来设置网页中元素的外观，比如字体大小、颜色、背景、布局等。CSS能够让网页从简单的文本和图像组合，变成具有丰富视觉效果的页面。CSS的层叠特性使得样式规则可以相互继承和覆盖，从而实现高效且灵活的样式管理，让网页在不同设备和屏幕尺寸上都能保持美观和一致的视觉效果。

1.2.6 JavaScript

JavaScript是一种强大的脚本语言，它为网页带来了动态交互功能。通过JavaScript，开发者可以在用户与网页交互时实时响应，例如点击按钮、输入文本、滚动页面等。它可以操作HTML元素，动态地修改页面内容和样式，甚至可以与服务器进行交互，实现无需刷新页面的数据更新。例如，一个简单的表单验证功能可以通过JavaScript来实现，当用户输入不符合要求的内容时，立即给予提示。此外，JavaScript还能够创建复杂的动画效果、实现拖放功能、开发单页面应用等，极大地丰富了网页的功能性和用户体验。

1.3 安装和使用

该网站程序采用纯前端技术实现，以网页形式呈现，无需安装任何额外软件即可使用。由于项目是为了满足考试需求，并未部署到云服务器或进行端口映射配置，因此暂时无法通过互联网直接访问。

如果要使用该网站，需要通过下载源码并在相应的IDE上进行编译运行。

1.4 页面效果功能图

青少年体态健康中心的网页构成与主要功能由功能结果图总结，另：所有网页也都使用了common.css和common.js，不在图中列出。

功能结果图如图1.1所示。

图1.1 功能结构图

# 2 详细设计

2.1 首页

2.1.1 轮播图功能

主要展示体态正确姿势内容，包括正确坐姿，正确玩手机姿势，正确背书包姿势，该功能可以通过单击轮播图箭头或左右滑动图片，可以实现轮播图的切换，界面如图2.1所示。

图2.1 体态科普界面

轮播图效果通过引入Swiper库实现，核心代码如下：

 const **initSwiper** = () => {

        const swiper = new Swiper('.health-swiper', {

            loop: true,

            autoplay: {

                delay: 5000,

                disableOnInteraction: false,

            },

            pagination: {

                el: '.swiper-pagination',

                clickable: true,

            },

            navigation: {

                nextEl: '.swiper-button-next',

                prevEl: '.swiper-button-prev',

            },

            effect: 'slide',

            speed: 800,

            slidesPerView: 1,

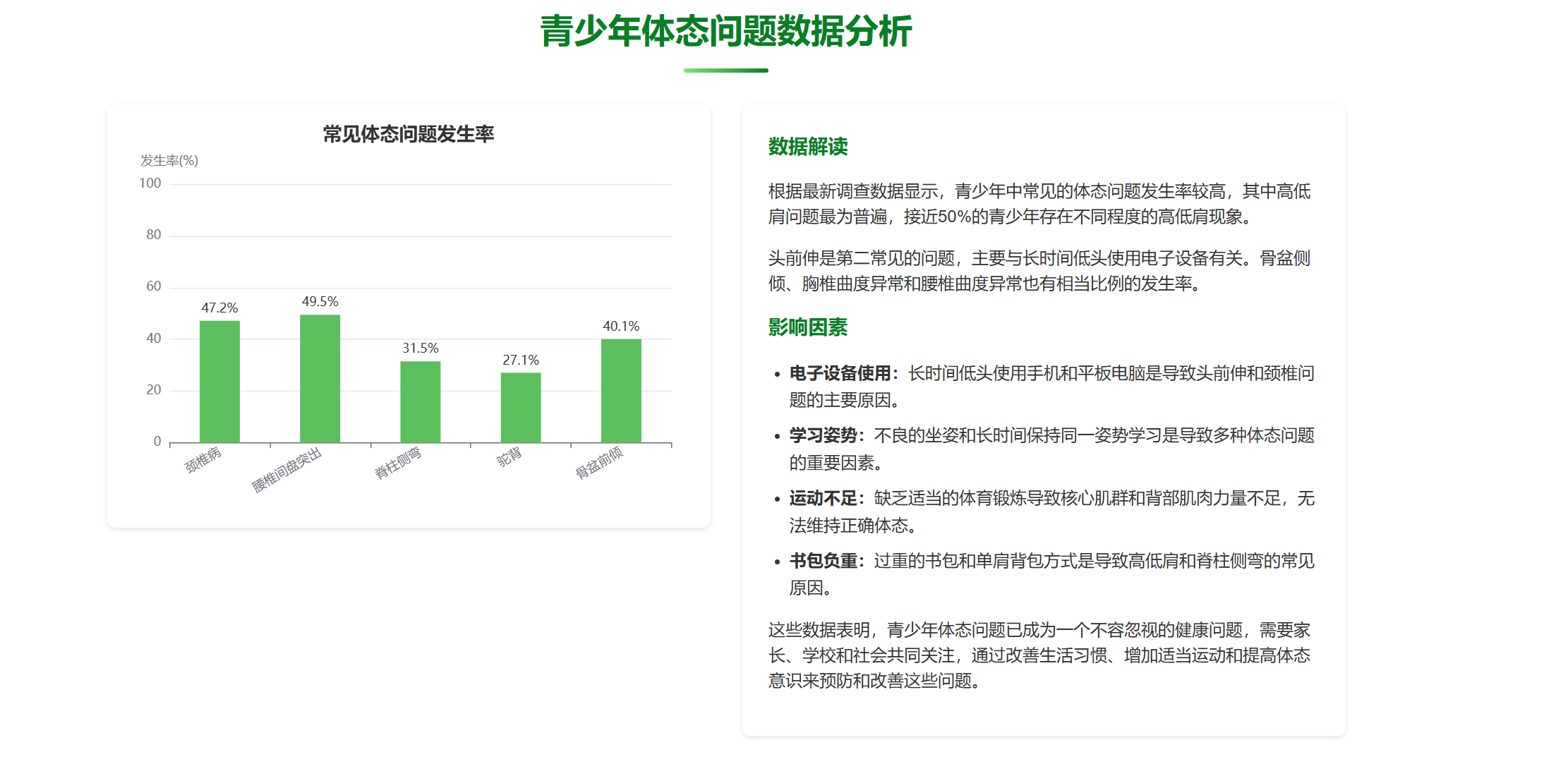
            spaceBetween: 30,

        });

    };

2.1.2 数据可视化功能

主要展示体态问题发生率内容，使用了Echarts库进行该图表的展示。鼠标悬停至图表上时数据会另外显示，界面如图2.2所示。

图2.2数据可视化界面

数据可视化功能通过ECharts库实现，算法是使用ECharts库的柱状图渲染算法，核心代码如下：

const initChart = () => {

const chartContainer = document.getElementById('posture-chart');

if (!chartContainer) return;

const myChart = echarts.init(chartContainer);

const option = {

//内容略

}

]

};

myChart.setOption(option);

window.addEventListener('resize', () => {

myChart.resize();

});

2.2 生活习惯页面

2.2.1计时器提醒功能

生活习惯页面主要展示姿势体态计时器和睡眠与体态内容，通过点击相应的分钟按钮或者点击自定义来设置多久后进行提醒，界面如图2.3所示。

图2.3生活习惯界面图

计时器效果通过DOM和BOM等技术结合实现，核心代码如下：

const playAlertSound = () => {

try {

const audioContext = new (window.AudioContext || window.webkitAudioContext)();

const oscillator = audioContext.createOscillator();

const gainNode = audioContext.createGain();

oscillator.type = 'sine';

oscillator.frequency.value = 880;

gainNode.gain.value = 0.3;

oscillator.connect(gainNode);

gainNode.connect(audioContext.destination);

oscillator.start();

setTimeout(() => {

oscillator.stop();

const oscillator2 = audioContext.createOscillator();

oscillator2.type = 'sine';

oscillator2.frequency.value = 1046.5;

oscillator2.connect(gainNode);

oscillator2.start();

setTimeout(() => oscillator2.stop(), 500);

}, 500);

} catch (error) {

console.log('无法播放提示音:', error);

alert('姿势提醒：是时候检查你的姿势了！');

}

};

2.3饮食与体态界面

2.3.1BMI计算器功能

饮食与体态页面主要展示BMI计算器，体态健康关键营养素和饮食误区，通过输入身高体重即可计算自己的BMI，并弹出BMI的参考范围，界面如图2.4所示。

图2.4 饮食与体态界面

BMI计算功能通过​​表单验证+数学计算**​**​技术实现，核心代码如下：

const calculateBMI = (height, weight) => {

return weight / Math.pow(height / 100, 2);

};

const getBMIStatus = (bmi) => {

return bmi < 18.5 ? '体重过轻' :

bmi < 24 ? '正常范围' :

bmi < 28 ? '超重' : '肥胖';

};

bmiForm.addEventListener('submit', (e) => {

e.preventDefault();

if (!validateForm()) return;

const height = parseInt(document.getElementById('height').value);

const weight = parseInt(document.getElementById('weight').value);

const bmi = calculateBMI(height, weight);

document.getElementById('bmi-result').textContent = bmi.toFixed(1);

document.getElementById('bmi-status').textContent = getBMIStatus(bmi);

resultsSection.classList.add('show');

});

2.4 趣味知识页面

2.4.1体态问题展示

趣味知识页面主要展示体态健康小贴士和常见体态问题内容，通过点击体态问题相应的区域可以查看该体态问题的详细内容，并且可以通过选择框选择相应的内容，界面如图2.5所示。

图2.5 趣味知识页面

体态问题展示功能通过​​Vue.js响应式数据绑定+计算属性过滤​​技术实现，核心代码如下：

new Vue({

el: '#vue-questions-app',

data: {

selectedCategory: 'all',

categories: ['脊柱问题', '骨盆问题', '背部问题'],

problems: [

{

}

]

},

computed: {

filteredProblems() {

return this.selectedCategory === 'all'

? this.problems

: this.problems.filter(p => p.category === this.selectedCategory);

}

},

methods: {

toggleDetails(index) {

this.$set(this.problems[index], 'isOpen', !this.problems[index].isOpen);

}

}

});

2.5体态问题咨询页面

2.5.1 AI体态顾问功能

体态问题咨询页面主要展示AI体态顾问内容，通过在对话框中输入问题，发送消息后会使用fetch API将内容传递给DeepSeek接口，经过处理后，DeepSeek会将解答传回，并在页面上进行显示，界面如图2.6所示。

图2.6体态问题咨询页面

AI体态顾问功能通过fetch API接入​​DeepSeek实现，核心算法如下：

async function sendMessage() {

const message = userInput.value.trim();

if (!message) return;

userInput.disabled = true;

sendButton.disabled = true;

conversationHistory.push({ role: "user", content: message });

addMessage('user', message);

userInput.value = '';

try {

const response = await fetch(API\_URL, {

method: 'POST',

headers: {

'Content-Type': 'application/json',

'Authorization': `Bearer ${API\_KEY}`

},

body: JSON.stringify({

model: MODEL\_NAME,

messages: conversationHistory,

temperature: 0.7,

max\_tokens: 1000

})

});

if (!response.ok) throw new Error(`HTTP错误: ${response.status}`);

const data = await response.json();

const aiResponse = data.choices[0].message.content;

conversationHistory.push({ role: "assistant", content: aiResponse });

addMessage('ai', aiResponse);

} catch (error) {

addMessage('ai', `抱歉出错了: ${error.message}`);

} finally {

userInput.disabled = false;

sendButton.disabled = false;

userInput.focus();

}

# 3 项目总结

在本次青少年体态健康中心网站的开发过程中，我深刻体会到前端开发的系统性和复杂性。通过实践，我不仅掌握了HTML、CSS和JavaScript的基础知识，还学习了Vue.js框架、ECharts和Swiper等第三方库的应用。项目开发初期，面对多页面统一风格、响应式布局等需求时，我通过建立公共CSS文件解决了样式复用问题。在实现轮播图和数据可视化功能时，Swiper和ECharts库的文档学习让我认识到第三方工具的高效性，同时也意识到参数调优需要反复测试。

在技术实现过程中，我特别注意到用户体验优化的重要性。例如在BMI计算器模块，不仅实现了基础的计算功能，还添加了表单验证、实时结果反馈和健康建议提示，使简单的计算工具变得更加友好实用。这种细节优化虽然增加了开发工作量，但显著提升了用户的使用体验。

其中最令我兴奋的是我使用了fetch API接入了DeepSeek，在以后的相关项目开发中，也可以援用该项目中的方式（但不要再把API\_KEY放到前端了）。

未来可从以下方面进行优化升级：

1. 引入后端技术，实现用户数据存储和个性化健康建议功能
2. 使用Vuex进行状态管理，提升复杂组件的开发效率
3. 增加体态自测功能，通过上传照片进行初步体态评估
4. 优化移动端交互体验，添加更多手势操作支持

通过这次项目实践，我认识到前端开发不仅是实现功能，更需要从用户角度思考，在细节处下功夫。这种工程思维将指导我未来的学习和开发工作。