

实验 15：Web 前端测试

一、实验目的

- 1、掌握 Web 前端界面测试的基本方法
- 2、了解浏览器兼容性测试流程
- 3、学习使用 Lighthouse 进行性能测试与分析
- 4、能够根据测试结果进行代码修订和优化

二、实验学时

2 学时

三、实验类型

综合性

四、实验需求

1、硬件

每人配备计算机 1 台，建议优先使用个人计算机开展实验。

2、软件

安装 Visual Studio Code，以及 Edge 浏览器。

3、网络

本地主机能够访问互联网和实验中心网络。

4、工具

无。

五、实验任务

- 1、完成界面测试
- 2、完成兼容性测试
- 3、完成性能测试
- 4、完成测试修订

六、实验内容及步骤

1、界面测试

步骤 1：界面元素验证

打开实验 14 发布的管理系统，检查页面所有按钮、链接、表单是否正常显示，验证导航菜单功能是否正常，检查数据表格、图表等组件显示是否完整。

步骤 2：响应式测试

打开浏览器开发者工具（F12），切换到设备模拟模式，分别测试手机（375px）、平板（768px）、桌面（1200px）三种尺寸下的显示效果，记录显示异常的界面元素。



图1 开发者工具的设备模拟模式

步骤 3：交互测试

测试所有表单的提交功能，验证模态框、下拉菜单等交互组件，检查页面跳转和路由切换。

2、兼容性测试

步骤 1：浏览器测试

(1) 在 Edge 浏览器中运行管理系统，记录运行状态，测试功能

(2) 在 360 浏览器中运行管理系统，记录运行状态，测试功能

步骤 2：控制台检查

查看浏览器控制台是否有错误或警告。

3、性能测试

在 Edge 浏览器中打开要测试的管理系统页面，按 F12 打开开发者工具，切换到"Lighthouse"标签页。

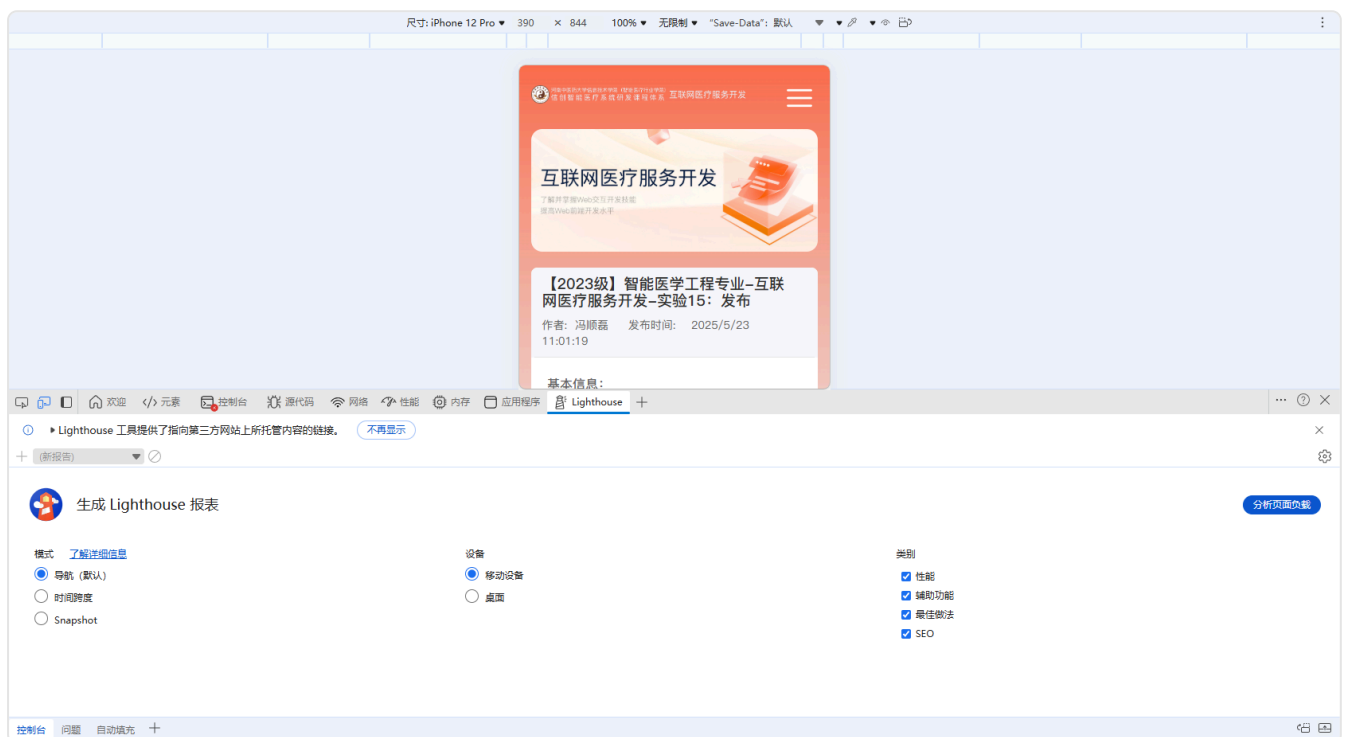


图2 Lighthouse标签页

步骤 2：生成性能报告

(1) 配置测试选项：

- 设备：Desktop（桌面端）
- 类别：勾选 Performance、Accessibility、Best Practices、SEO

(2) 点击"Generate report"开始分析

(3) 等待测试完成（约 30-60 秒）

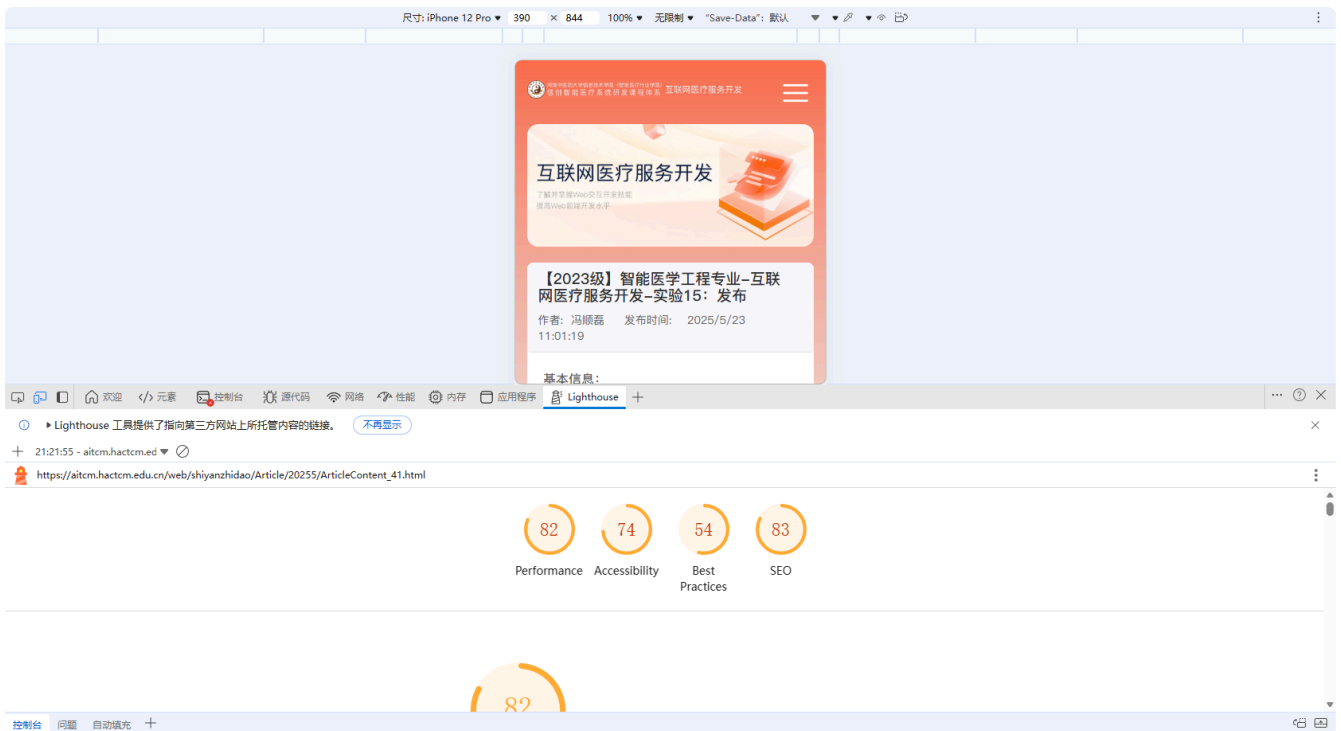


图3 Lighthouse测试结果

步骤 3: 分析测试结果

(1) 查看性能评分和关键指标:

- First Contentful Paint (首次内容绘制)
- Largest Contentful Paint (最大内容绘制)
- Cumulative Layout Shift (累积布局偏移)

(2) 记录"Opportunities"中的优化建议

(3) 查看"Best Practices"和"Accessibility"建议

4、测试修订

步骤 1: 问题分析

整理前三个任务中发现的问题,按优先级排序:功能问题 > 兼容性问题 > 性能问题,为每个问题制定解决方案。

步骤 2: 代码修订

在VS Code 中打开项目,修复界面显示问题:

- 调整 CSS 样式
- 修复布局错位

解决兼容性问题:

- 添加浏览器前缀
- 使用 polyfill (如有需要)

性能优化：

- 压缩图片资源
- 优化 JavaScript 代码
- 减少 HTTP 请求

步骤 3：验证修订结果

重新运行界面测试，确认问题已修复，再次进行兼容性测试，使用 Lighthouse 重新测试性能，对比优化前后的评分。

步骤 4：测试报告整理

记录测试过程中的问题，记录解决方案和优化措施，对比优化前后的性能数据，总结测试经验。

七、实验考核

本实验考核采用【实验随堂查】方式开展。

每个实验完成后，在实验课上通过现场演示的方式向实验指导教师进行汇报，并完成现场问答交流。

每个实验考核满分 100 分，其中实验成果汇报 60 分，现场提问交流 40 分。

实验考核流程：

- (1) 学生演示汇报实验内容的完成情况，实验指导老师现场打分。
- (2) 指导老师结合实验内容进行提问，每位学生提问 2-3 个问题，根据回答的情况现场打分。
- (3) 实验考核结束后，进行公布成绩。