

成绩:

江西科技师范大学

毕业论文（设计）

题目（中文）：基于 Web 客户端技术的个性化 UI 设计和实现

（外文）：Web client based customized UI design and Programming

院（系）：元宇宙产业学院

专 业：计算机科学与技术

学生姓名：李国庆

学 号：20213575

指导教师：李建宏

2024 年 6 月 19 日

目录

1. 前言	1
1.1 学习计划	1
1.2 项目的技术路线	1
1.3 研究方法	2
1.3.1 文献法	2
1.3.2 模型研究法	3
2. 技术总结和文献综述	4
2.1 Web 平台和客户端技术概述	4
2.1.1 历史	4
2.1.2 一个关于万维网的联盟	4
2.1.3 Web 平台和 Web 编程	5
2.2 项目的增量式迭代开发模式	5
3. 内容设计概要	7
3.1 分析和设计	7
3.2 项目的实现和编程	7
3.3 项目的运行和测试	9
3.4 项目的代码提交和版本管理	10
4. 移动互联时代的 UI 开发初步--窄屏终端响应式设计	10
4.1 分析和设计	10
4.2 项目的实现和编程	11
4.3 项目的运行和测试	12
5. 应用响应式设计技术开发可适配窄屏和宽屏的 UI	13
5.1 分析和设计	13
5.2 项目的实现和编程	13
5.3 项目的运行和测试	15
6. 个性化 UI 设计中对鼠标交互的设计开发	16
6.1 项目的实现和编程	16
6.2 项目的运行和测试	18
7. 对触屏和鼠标的通用交互操作的设计开发	19
7.1 项目的实现和编程	19
7.2 项目的运行和测试	21
8. UI 的个性化键盘交互控制的设计开发	22
8.1 分析和设计	22
8.2 项目的实现和编程	22
8.3 项目的运行和测试	23
9. 谈谈本项目中的高质量代码	24
10. 用 gitBash 工具管理项目的代码仓库和 http 服务器	25
10.1 经典 Bash 工具介绍	25
10.2 通过 gitHub 平台实现本项目的全球域名	25
10.3 创建一个空的远程代码仓库	25
10.4 设置本地仓库和远程代码仓库的链接	26
参考文献	28

摘要：Web 技术以其跨操作系统平台的优势成为了广泛流行的软件开发手段，为了适应移动互联网时代软件项目的前端需求，本项目以 Web 客户端技术为研究学习内容，广泛查阅了技术资料与相关文献，尤其是 mozilla 组织的 MDN 社区的技术实践文章，探索了 HTML 内容建模、CSS 样式设计和 JavaScript 功能编程的基本技术和技巧。通过集成上述技术，再应用本科的相关课程的知识，实现了一个个性化的用户界面（UI: uer interface）的项目，该用户界面以响应式技术为支撑做到了最佳适配用户屏幕，程序可以动态适用于当前 PC 端和移动设备；在功能上以 DOM 技术和事件驱动模式的程序为支撑实现了对鼠标、触屏、键盘的底层事件响应和流畅支持，为鼠标和触屏设计了一个对象模型，用代码实现了对这类指向性设备的模拟（这是本项目模型研究法的一次创新实践，也是本项目的亮点。）。为了处理好设计和开发的关系，项目用了工程思想管理，使用了软件工程的增量式开发模式，共做了 6 次项目迭代开发，每次迭代都经历了开发 4 个经典开发阶段（A:Analysis, D:Design, I: Implementation, T:Testing），以逐步求精的方式编写了本 UI 的应用程序。为了分享和共享本代码，与网上的开发者共同合作，本项目还使用了 git 工具进行代码和开发过程日志记录，一共做了 12 次提交代码的操作，详细记录和展现了开发思路和代码优化的过程，最后通过 gitbash 把项目上传到 github 上，建立了自己的代码仓库，并将该代码仓库设置成为了 http 服务器，实现了本 UI 应用的全球便捷访问。

1. 前言

1.1 学习计划

运用本科阶段学习的计算机科学技术知识，尤其是程序设计和软件工程领域学习的方法、训练的代码能力，构架感兴趣的技术路线，结合探求的问题形成软件需求，有条理第系统落实分析问题、建立模型、软件设计、系统实施、测试调试等传统软件工程的全部流程。

毕业论文（设计）从字面上来看是以毕业论文为主，毕业设计作品没有做硬性要求。这对于以软件开发为方向的毕设模式描述不太准确，因为一般是先要做出毕设软件才可以总结写出毕业论文。另外，在计算机专业软件开发的毕业论文中，插入截图配合文本来演示软件的效果证明真实性，还将在论文中提供网址让读者可以访问和使用毕设作品。用这种方式也可以证明论文的可信度，这也将是该专业论文的特色所在。

1.2 项目的技术路线

所谓技术路线特指从事理工科的人们为实现复杂的工程而制定的计划或规划，技术二字强调的是专业性，一般用于本专业或行业的属于表达，而路线二字

则强调逻辑性，一般表达了事物的因果关系或层次关系，我们常用技术路线来证明工程项目或者研究的可行性。

毕设主要分为两个阶段完成，首先选择一条自己感兴趣的技术实践路线，把核心的技术加以整合学习，以导师的案例项目为参考，主要理解好各个技术之间的关系，在项目中的作用和分工，更重要的是在项目实施中提升自己的写高质量的代码的能力。当仿造导师的案例的技术基本实现后，则可以视为实践和理论基本打通，此时就可进入第二个阶段，开始真正的做毕设软件。

第二阶段一般按软件工程的标准来规范开发：结合自己的问题做出定义和分析；设计一套合适的技术解决方案；按解决方案设计流程和编写相关代码，实现技术部署；调试代码、测试代码、性能调优。其中按解决方案设计流程和编写相关代码，实现技术部署；调试代码、测试代码、性能调优可以发现前面步骤的问题，因此可能会在设计一套合适的技术解决方案；按解决方案设计流程和编写相关代码，实现技术部署；调试代码、测试代码、性能调优这三步多次循环，发现和解决设计一套合适的技术解决方案的设计失误或者按解决方案设计流程和编写相关代码，实现技术部署的代码错误。当然大部分工作是用在按解决方案设计流程和编写相关代码，实现技术部署的构建代码体系和落实软件架构的具体实施和细节。

本科毕设与个人开发者类似，项目的设计和具体实现都没有经验丰富的团队，很多时候为了提高效率，方案设计的细化优化和写代码具体部署两个步骤其实是交替进行的。前者是工程师落实微观和细节层面，而后者则是设计师的工作，确保宏观层面的设计不偏离需求。

1.3 研究方法

不管在哪个领域学习研究，其实学习方法很多都是相通的，方法可以互相借鉴的。

1.3.1 文献法

对于 Web 应用方向而言，在代码能力成长的任何阶段我们都无法离开文献法，然而承载知识的媒体形式非常丰富，包括：书籍、在线文档、社区论坛、期刊、

会议报告等等，Web 应用技术方向的资料来源于以下几类：

一是被中国 IT 邻域诙谐的取名为“衣冠禽兽”：“O Reilly Media”、“Manning Publications”，这两大著名的出版集团出版了大量的经典计算机科学技术书籍，全球开发者也把有大量自己私藏的这类书籍以电子文件的形式，广泛地发布在互联网的各个开发者社区。

二是在在线文档和社区论坛的文章，主要是 W3C 国际组织的 Web 标准和 Mozilla 基金会的 MDN 文档网站，二者分别站在科学家和工程师的两个层次，都有二十多年的积累和知识沉淀，他们在线上发表了大量公开文档，描绘了 Web 的过去、现在、将来。W3C 主要以 Web 国际标准和最佳实践为主，而 MDN 则偏向 Web 的最佳实践和各从层面应用的教学和讨论。

三则是用的最多的知网平台，在这里主要可以学习写论文的规范、了解我选择的毕设选题他人在理论上的研究和探索，对本科生而言可提供理论的指导和展示学术的规范，对于梦像成为开发者的本科生而言，这些文章对于我们毕设的软件开发能力的意义不大，但对于毕业论文还是很有参考价值的。当然，文字材料还有其他的来源，在线信息如此丰富，也是因为我们的 Web 技术广泛而深刻的应用在 Internet 上，让知识可以跨越国界、跨越种族、跨越阶层。

1.3.2 模型研究法

模型研究方法非常具体，比如承载我们 Web 应用的台式机、笔记本、手机、平板，传递在线信息要用到的互联网、服务器，沟通硬件和我们的代码之间的操作系统、浏览器、代码编译器、编译器，这些软硬件对象，对我们而言，都值得从写代码的角度去研究，我们笼统的称它们为对象，这些对象最终会在我们大脑中被理解为抽象的模型，我们再通过分析把这些模型程序化、数据化，最后写出来，这种行为本质上就是先在思维上“建模”，再用 OOP 语言表述出来。

在 OOP 分析和开发过程中，在毕设中试图解决的问题也被定义成为了各级各种模型。模型研究只是更为抽象，与具体的计算机语言无关，在毕设中也尝试使用国际标准 UML 语言来建立抽象模型。采用 UML 模型研究法和面向对象的程序设计方法的目标是一致的，只是在不同层面分析表述问题而已。因此采用模型研究法，毕设在用例设计采用了类似的 UML 对问题建模，景观 UML 比较抽

象，设计准确有一定难度，而使用 OOP 程序对画好的模型开展程序设计则更为具体直观，通过熟悉的 OOP 语言和代码运行环境和调试模型，我们甚至可以推导出模型设计的问题和缺陷。对于我们本科生而言，可能直接写代码建立模型研究模型，代码跑通后，再利用 UML 语言绘制模型，作为代码的文档资料则更合理。

2. 技术总结和文献综述

2.1 Web 平台和客户端技术概述

Web 之父 Tim Berners-Lee 在发明 Web 的基本技术架构以后，就成立了 W3C 组织，该组织在 2010 年后推出的 HTML5 国际标准，结合欧洲 ECMA 组织维护的 ECMAScript 国际标准，几乎完美缔造了全球开发者实现开发平台统一的理想，直到今天，科学家与 Web 行业也还一直在致力于完善这个伟大而光荣的理想[1]。学习 Web 标准和 Web 技术，学习编写 Web 程序和应用有关工具，最终架构一套高质量代码的跨平台运行的应用，是我的毕设项目应用的技术路线。

2.1.1 历史

1989 年，蒂姆·伯纳斯-李爵士发明了万维网（见最初的提案）。他在 1990 年 10 月创造了“万维网”一词，并写下了第一个万维网服务器“httpd”和第一个客户端程序（一个浏览器和编辑器）“世界万维网”。他编写了“超文本标记语言”（HTML）的第一个版本，这是一种具有超文本链接功能的文档格式化语言，后来成为了 Web 的主要发布格式。他对 uri、HTTP 和 HTML 的最初规范随着网络技术的传播，在更大的圈子中得到了改进和讨论。

2.1.2 一个关于万维网的联盟

1994 年，在许多公司的敦促下，决定成立万维网联盟。蒂姆·伯纳斯-李爵士开始领导网络联盟团队的基本工作，以培养一个一致的架构，以适应网络标准的

快速发展，以构建网站、浏览器和设备，以体验网络所提供的一切。在创立万维网联盟的过程中，蒂姆·伯纳斯-李爵士创建了一个同行社区。Web 技术已经如此之快，以至于组装一个组织来协调 Web 标准至关重要。蒂姆接受了麻省理工学院举办 W3C 课程的邀请。他从一开始就要求 W3C 拥有全球的足迹。

2.1.3 Web 平台和 Web 编程

让我们从对网络的简要描述开始，它是万维网的缩写。大多数人说“是网络”而不是“万维网”，我们会遵循这个惯例。Web 是一个文档的集合，被称为网页，它们由世界各地的计算机用户（在大部分时间内）共享。不同类型的网页可以做不同的事情，但至少，它们都能在电脑屏幕上显示内容。我们所说的“内容”是指文本、图片和用户输入机制，如文本框和按钮^[2]。

Web 编程是一个很大的领域，通过不同的工具实现不同类型的 Web 编程。所有的工具都与核心语言 HTML 一起工作，所以几乎所有的 web 编程书籍都在某种程度上描述了 HTML。这本教科书涵盖了 HTML5，CSS，和 JavaScript，所有的深入。这三种技术被认为是客户端网络编程的支柱。使用客户端 web 编程，所有网页计算都在终端用户的计算机（客户端计算机）上执行^[3]。

Web 应用的程序设计体系由三大语言有机组成：HTML, CSS, JavaScript。这三大语言的组合也体现了人类社会化大生产分工的智慧，可以看作用三套相对独立体系实现了对一个信息系统的描述和控制，可以总结为：HTML 用来描述结构（Structure）、CSS 用来描述外表（presentation）、Javascript 用来描述行为（Behavior）^[3]；这也可以用经典的 MVC 设计模式来理解 Web 平台架构的三大基石，Model 可以理解为 HTML 标记语言建模，View 可以理解为用 CSS 语言来实现外观，Controller 则可理解为用 JavaScript 结合前面二个层次，实现了在微观和功能层面的代码控制。

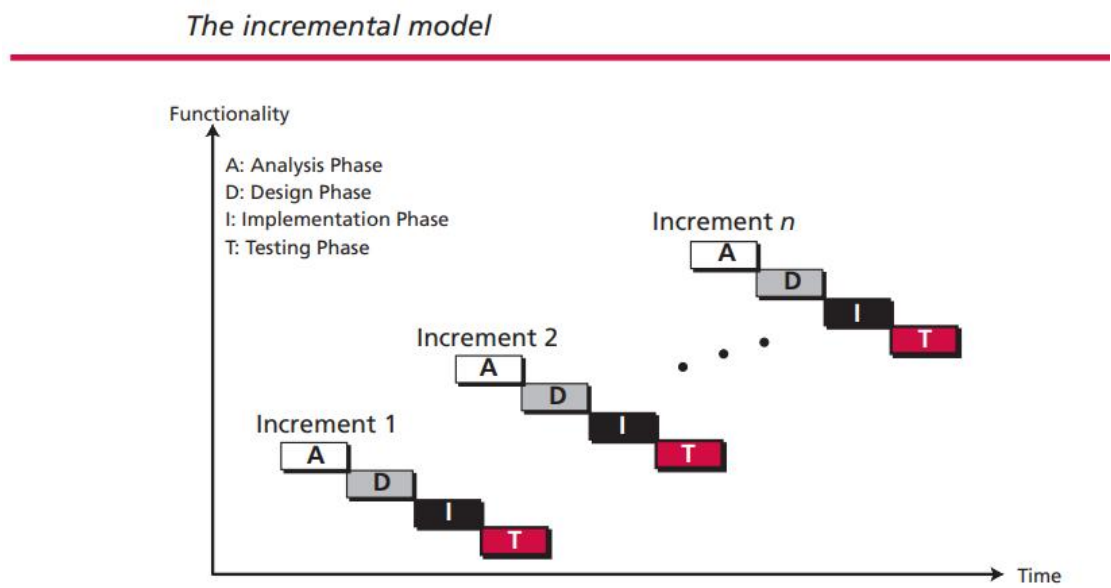
2.2 项目的增量式迭代开发模式

本项目作为一个本科专业学生毕业设计的软件作品，与单一用途的程序相比较为复杂，本项目所涉及的手写代码量远超过简单一二个数量级以上，从分析问

题的到初步尝试写代码也不是能在几天内能落实的,可以说本项目是一个系统工程,因此需要从软件工程的管理视角来看待和规范项目的编写过程。

而本项目考虑选择的软件工程开发过程管理模式有两种经典模型:瀑布模型(The waterfall model)和增量式迭代模型(The incremental model)。而任何开发模式则都必须同样经历四个阶段:分析(Analysis)、设计(Design)、实施(Implementation)、测试(test)。

瀑布模型需要专业团队完美的配合,从分析、设计到实施,最后到测试,任何阶段的开始必须基于上一阶段的完美结束。而这对于我们大多数普通开发者是不太现实的,作为小微开发者由于身兼数职,其实无法1次就能完美完成任何阶段的工作,比如在实施过程中,开发者会发现前面的设计存在问题,则必须在下一次迭代项目时改良设计。在当今开源的软件开发环境中,开发者在软件的开发中总是在不断地优化设计、重构代码,持续改进程序的功能和代码质量。因此在本项目的开发中,也采用了增量模型的开发模式^[5]。本项目中我一共做了六次项目的开发迭代,如下图所示:



增量模型

在增量模型中,软件分一系列步骤进行开发。开发人员首先完成了整个系统的一个简化版本。这个版本表示整个系统,但不包括详细信息。图中显示了增量模型的概念。

在第二个版本中,添加了更多的细节,而一些没有完成,系统再次测试。如

果有问题，开发人员就知道问题在于新功能。在现有的系统正常工作之前，它们不会添加更多的功能。这个过程继续进行，直到需要的所有功能都添加到^[5]。

3. 内容设计概要

3.1 分析和设计

这一步是项目的初次开发，本项目最初使用人们习惯的“三段论”式简洁方式开展内容设计，首先用一个标题性信息展示 logo 或文字标题，吸引用户的注意力，迅速表达主题；然后展现主要区域，也就是内容区，“内容为王”是项目必须坚守的理念，也是整个 UI 应用的重点；最后则是足部的附加信息，用来显示一些用户可能关心的细节变化。如图 3-1 用例图所示：

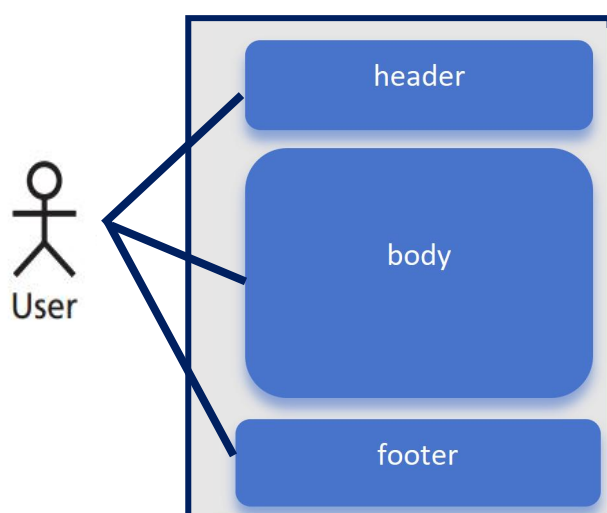


图 3-1 用例图

3.2 项目的实现和编程

一、HTML 代码编写如下：

```

<title>"读名著、练代码、爱学习"</title>
</head>
<body>
  <header>
    <p id="book">
      现代电影赏析
    </p>
  </header>
  <nav>
    <button>导航一</button>
    <button>导航二</button>
    <button>导航三</button>
  </nav>
  <main id='main'>
    软件内容区域
  </main>
  <footer>
    <p id="statusInfo">
      Copyright from 李国庆 江西科技师范大学 2022--2025
    </p>
  </footer>

```

二、CSS 代码编写如下:

```

<style>
*{
margin:10px;
text-align:center;
}
header{
border: 2px solid blue;
height: 80px;
}
main{
border: 2px solid blue;
height: 300px;
}
nav{
border: 2px solid blue;
height: 50px;
}
footer{
border: 2px solid blue;
height: 50px;
}

```

3.3 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 3-2 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 3-3 的二维码，运行测试本项目的第一次开发的阶段性效果。



图 3-2 PC 端运行效果图



图 3-3 移动端二维码

3.4 项目的代码提交和版本管理

本项目的文件通过 gitBash 工具管理，作为项目的第一次迭代，在代码提交和版本管理环节，我们的目标是建立项目的基本文件结构，还有设置好代码仓库的基本信息：如开发者的名字和电子邮件。

进入 gitBash 命令行后，按次序输入以下命令：

```
$ cd /  
$ mkdir webUI  
$ cd webUI  
$ git init  
$ git config user.name 江科师大李国庆  
$ git config user.email 2538587008@qq.com  
$ touch index.html myCss.css
```

编写好 index.html 和 myCss.css 的代码，测试运行成功后，执行下面命令提交代码：

```
$ git add index.html myCss.css  
$ git commit -m 项目第一版：“三段论”式的内容设计概要开发
```

成功提交代码后，gitbash 的反馈如下所示：

```
$ git commit -m 项目第一版：“三段论”式的内容设计概要开发  
[master (root-commit) 32de024] 项目第一版：“三段论”式的内容设计概要开发  
2 files changed, 46 insertions(+)  
create mode 100644 index.html  
create mode 100644 myCss.css
```

项目代码仓库自此也开启了严肃的历史记录，我们可以输入日志命令查看，

```
$ git log
```

4. 移动互联时代的 UI 开发初步--窄屏终端响应式设计

4.1 分析和设计

响应性设计—适应显示硬件

在计算机上使用的显示器硬件差别很大，显示器的大小和分辨率取决于成本。设计师并没有选择每个网页的版本，而是选择让网页给出总体布局指南，并允许浏览器选择如何在给定的计算机上显示页面。因此，一个网页并没有提供很多细

节。例如，一个网页的作者可以指定一组句子组成一个段落，但作者不能指定细节，如一行的确切长度或是否缩进段落的开头^[1]。

允许一个浏览器选择显示细节有一个有趣的结果：当通过两个浏览器或在两个硬件不同的计算机上查看时，一个网页可能会出现不同的外观。如果一个屏幕比另一个屏幕宽，那么一行文本的长度或可以显示的图像的大小就会有所不同。重点是：网页给出了关于所需演示文稿的一般指南；浏览器在显示页面时选择详细信息。因此，当同一网页在两台不同的计算机或不同的浏览器上显示时，可能会出现略有不同^[1]。

用 JavaScript 开动态读取显示设备的信息，然后按设计，使用 js+css 来部署适配当前设备的显示的代码。

4.2 项目的实现和编程

一、HTML 代码编写如下：

```
<title> "读名著、练代码、爱学习" </title>
</head>
<body>
  <header>
    <p id="book">
      现代电影赏析
    </p>
  </header>
  <nav>
    <button>导航一</button>
    <button>导航二</button>
    <button>导航三</button>
  </nav>
  <main id = 'main'>
    软件内容区域
  </main>

  <footer>
    <p id="statusInfo">
      李国庆 江西科技师范大学 2025
    </p>
  </footer>
```

二、CSS 代码编写如下：

```

<style>
*{
margin: 10px;
text-align: center;
}
header{
border: 2px solid blue;
height: 15%;
font-size: 1.66em;

}
main{
border: 2px solid blue;
height: 70%;
font-size: 1.2em;
background-image: url(../lesson/CS.jpg);
background-size: cover;
}
nav{
border: 2px solid blue;
height: 10%;
font-size: 1.1em;
}
footer{
border: 2px solid blue;
height: 5%;
}

```

4.3 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 4-1 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 4-2 的二维码，运行测试本项目的第一次开发的阶段性效果。

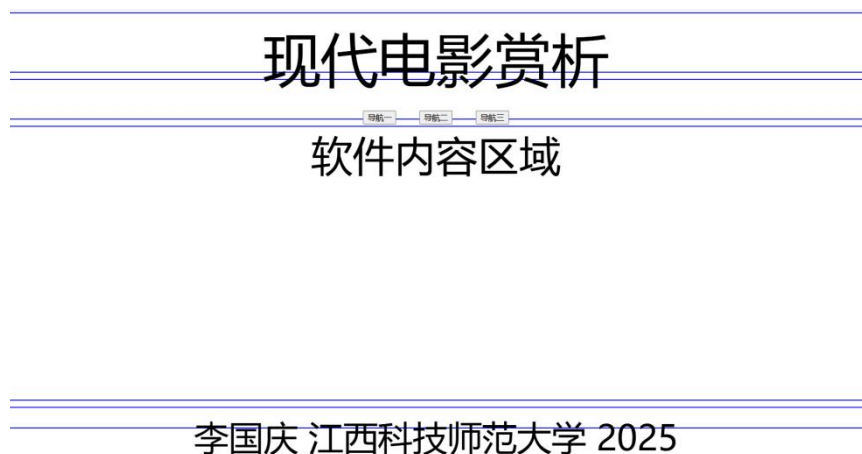


图 4-1 PC 端运行效果图



图 4-2 移动端二维码

5. 应用响应式设计技术开发可适配窄屏和宽屏的 UI

5.1 分析和设计

响应式就是同一个代码可以自动适应不同设备和屏幕尺寸，以提供最佳的用户体验。响应式设计的目标是确保网站在不同设备上呈现内容和布局时能够自动调整，以适应屏幕的大小和方向。这意味着网站可以在台式电脑、平板电脑、手机和其他各种设备上有效显示，而不需要为每种设备创建不同版本的网站。响应式设计使用 HTML、CSS 和 JavaScript 技术来实现，以确保网站内容能够根据访问设备的特征进行动态调整。

5.2 项目的实现和编程

一、HTML 代码编写如下：


```

<header>
  <p id="book">
    《现代电影赏析》
  </p>
</header>
<nav>
  <button>向前</button>
  <button>向后</button>
  <button>暂停</button>
</nav>

<main id="main">
  <div id="bookface">
    书的封面图
  </div>
</main>

<footer>

  Copyright from 李国庆 江西科技师范大学 2022--2025

</footer>
<div id="aid">
  <p>用户键盘响应区</p>

</div>

```

二、CSS 代码编写如下：

```

<style>
  *{
    margin: 10px;
    text-align: center;
  }
  header{
    border: 3px solid green;
    height: 10%;
    font-size: 1em;
  }
  nav{
    border: 3px solid green;
    height: 10%;
  }
  main{
    border: 3px solid green;
    height: 70%;
    font-size: 0.8em;
    position: relative;
  }
  #box{
    position: absolute;
    right: 0;
    width: 100px;
  }

```



```

    footer{
        border: 3px solid green;
        height:10%;
        font-size: 0.7em;
    }
    body{
        position: relative;
    }
    #aid{
        position: absolute;
        border: 3px solid blue;
        top:0px;
        left:600px;
    }
    #bookface{
        width: 80%;
        height: 80%;
        border:1px solid red;
        background-color: blanchedalmond;
        margin:auto;
    }
}
/style>

```

5.3 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 5-1 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 5-2 的二维码，运行测试本项目的第一次开发的阶段性效果。



图 5-1 PC 端运行效果图



图 5-2 移动端二维码

6. 个性化 UI 设计中对鼠标交互的设计开发

6.1 项目的实现和编程

一、HTML 代码编写如下：

```
<body >
  <header>
    <p id="book">
      《现代电影赏析》
    </p>
  </header>
  <nav>
    <button>向前</button>
    <button>向后</button>
    <button>暂停</button>
  </nav>

  <main id="main">
    <div id="bookface">
      书的封面图
    </div>
  </main>

  <footer>

    Copyright from 李国庆 江西科技师范大学 2022--2025

  </footer>
  <div id="aid">
    <p>用户键盘响应区</p>
  </div>
```

二、CSS 代码编写如下：

```

<style>
*{
    margin: 10px;
    text-align: center;
}
header{
    border: 3px solid green;
    height: 10%;
    font-size: 1em;

}
nav{
    border: 3px solid green;
    height: 10%;
}
main{
    border: 3px solid green;
    height: 70%;
    font-size: 0.8em;
    position: relative;
}

#box{
    position: absolute;
    right: 0;
    width: 100px;
}

footer{
    border: 3px solid green;
    height: 10%;
    font-size: 0.7em;
}
body{
    position: relative;
}
#aid{
    position: absolute;
    border: 3px solid blue;
    top: 0px;
    left: 600px;
}
#bookface{
    position: absolute;
    width: 80%;
    height: 80%;
    border: 1px solid red;
    background-color: blanchedalmond;
    left: 7% ;
    top: 7% ;
}

```

6.2 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 6-1 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 6-2 的二维码，运行测试本项目的第一次开发的阶段性效果。



图 6-1 PC 端运行效果图



图 6-2 移动端二维码

7. 对触屏和鼠标的通用交互操作的设计开发

7.1 项目的实现和编程

一、HTML 代码编写如下：

```
<header>
  <p id="book">
    《我的毕设题目》
  </p>
</header>
<nav>
  <button>向前</button>
  <button>向后</button>
  <button>其他</button>
</nav>

<main id="main">
  <div id="bookface">
    这是书的封面图<br>
    在此对象范围拖动鼠标/滑动触屏<br>
    拖动/滑动超过100像素，视为有效UI互动！
  </div>
</main>

<footer>

  Copyright 李国庆 江西科技师范大学 2024--2025

</footer>
<div id="aid">
  <p>用户键盘响应区</p>
</div>
```

二、CSS 代码编写如下：

```

<style>
*{
    margin: 10px;
    text-align: center;
}
header{
    border: 3px solid green;
    height: 10%;
    font-size: 1em;

}
nav{
    border: 3px solid green;
    height: 10%;
}
main{
    border: 3px solid green;
    height: 70%;
    font-size: 0.8em;
    position: relative;
}

#box{
    position: absolute;
    right: 0;
    width: 100px;
}

footer{
    border: 3px solid green;
    height: 10%;
    font-size: 0.7em;
}
body{
    position: relative;
}
#aid{
    position: absolute;
    border: 3px solid blue;
    top: 0px;
    left: 600px;
}
#bookface{
    position: absolute;
    width: 80%;
    height: 80%;
    border: 1px solid red;
    background-color: blanchedalmond;
    left: 7% ;
    top: 7% ;
}

```

7.2 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 7-1 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 7-2 的二维码，运行测试本项目的第一次开发的阶段性效果。

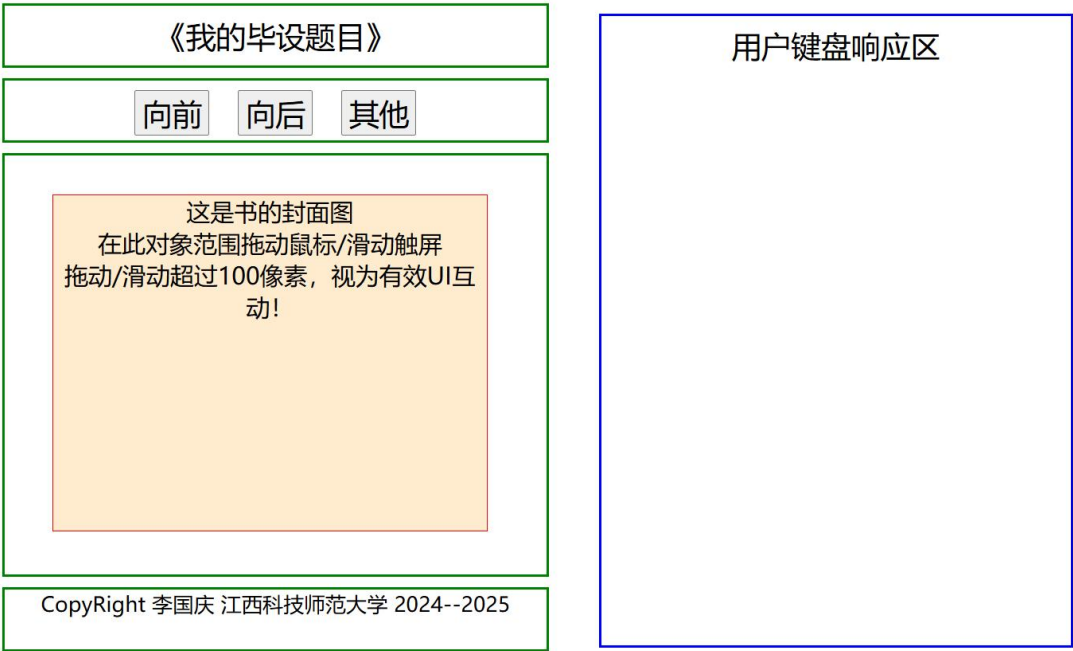


图 7-1 PC 端运行效果图



图 7-2 移动端二维码

8. UI 的个性化键盘交互控制的设计开发

8.1 分析和设计

当用户按下键盘上的任意键时触发。这个事件会一直触发，如果用户长按某个键，就会持续触发 `keydown` 事件。当用户释放键盘上的键时触发 `keyup` 事件。键盘事件是用户与计算机交互的重要方式之一，了解和掌握键盘事件的种类、触发顺序以及它们在 DOM 中的表现，可以帮助我们更好地设计和实现交互体验良好的网页和应用程序。在实际应用中，需要根据具体的需求和场景选择合适的事件处理方式，并注意跨浏览器兼容性和用户体验等问题。

8.2 项目的实现和编程

因为系统中只有一个键盘，所以我们在部署代码时，把键盘事件的监听设置在 DOM 文档最大的可视对象——`body` 上，通过测试，不宜把键盘事件注册在 `body` 内部的子对象中。代码如下所示：

```
$("#body").addEventListener("keydown",function(ev){
    ev.preventDefault() ; //增加“阻止事件对象的默认事件后”，不仅 keypress 事件将不再响应，而且系统的热键，如“F5 刷新页面/Ctrl+R”、“F12 打开开发者面板”等也不再被响应
    let k = ev.key;
    let c = ev.keyCode;
    $("#keyStatus").textContent = "按下键 : " + k + " , "+ "编码 : " + c;
});
```

```
$("#body").addEventListener("keyup",function(ev){
    ev.preventDefault() ;
    let key = ev.key;
    $("#keyStatus").textContent = key + " 键已弹起" ;
    if (printLetter(key)){
        $("#typeText").textContent += key ;
    }
}
```

```
function printLetter(k){
```



```

    if (k.length > 1){ //学生须研究这个逻辑的作用
        return false ;
    }
    let puncs =
['~','`','!','@','#','$','%','^','&','*','(',')','-','_','+','=',' ','.',
',',';','<','>','?','/',' ','\','\"'];
    if ( (k >= 'a' && k <= 'z') || (k >= 'A' && k <= 'Z')
|| (k >= '0' && k <= '9')) {
        console.log("letters") ;
        return true ;
    }
    for (let p of puncs ){
        if (p === k) {
            console.log("puncs") ;
            return true ;
        }
    }
    return false ;
    //提出更高阶的问题，如何处理连续空格和制表键 tab?
} //function printLetter(k)
});

```

8.3 项目的运行和测试

项目的运行和测试至少要通过二类终端，本文此处仅给出 PC 端用 Chrome 浏览器打开项目的结果，如下图 8-1 所示。由于本项目的阶段性文件已经上传 github 网站，移动端用户可以通过扫描图 8-2 的二维码，运行测试本项目的第一次开发的阶段性效果。



图 8-1 PC 端运行效果图



图 8-2 移动端二维码

9. 谈谈本项目中的高质量代码

这是一本关于指导计算机的书。如今，电脑和螺丝刀一样常见，但它们相当复杂，让它们做你想让它们做的事情并不总是那么容易。

如果你的计算机任务是一个常见的，很容易理解的任务，比如显示你的电子邮件或像一个计算器，你可以打开适当的应用程序，并开始工作。但是对于唯一

的或开放式的任务，可能没有应用程序。

这就是编程可能会出现的地方。编程是构建一个程序的行为——一组精确的指令，告诉计算机要做什么。因为计算机是愚蠢的，迂腐的野兽，编程从根本上来说是乏味和令人沮丧的。幸运的是，如果你能克服这个事实，甚至可以享受到愚蠢的机器能够处理的严格思考，那么编程可能是值得的。它可以让你在几秒钟内完成一些永远需要手工完成的事情。这是一种让你的电脑工具做一些它以前不能做的事情的方法。它提供了一个很好的抽象思维的练习^[6]。

10. 用 gitBash 工具管理项目的代码仓库和 http 服务器

10.1 经典 Bash 工具介绍

当我们谈到命令行时，我们实际上指的是 shell。shell 是一个程序，并将它们传递给操作系统执行。几乎所有的 Linux 发行版都提供了一个来自 GNU 项目的 shell 程序。这个名字是伯恩-再次外壳的首字母缩写，指的是 bash 是 sh 的增强替代品，最初的 Unix 外壳程序由史蒂夫·伯恩写。

和 Windows 一样，类似 inix 的 Linux 操作系统以分层目录结构组织文件。这意味着它们被组织成一个树状的目录模式（在其他系统中有时称为文件夹），其中可能包含文件和其他目录。文件系统中的第一个目录称为根目录。根目录包含文件和子目录，其中包含更多的文件和子目录等等^[7]。

10.2 通过 gitHub 平台实现本项目的全球域名

10.3 创建一个空的远程代码仓库

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * Repository name *

 masterLijh /

✓ userName.github.io is available.

Great repository names are short and memorable. Need inspiration? How about [expert-rotary-phone](#) ?

Create repository

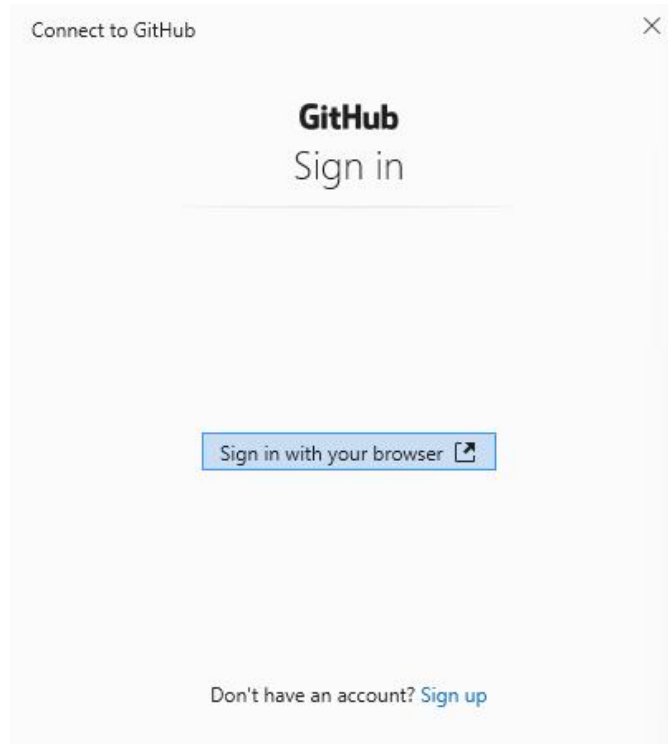
点击窗口右下角的绿色“Create repository”，则可创建一个空的远程代码仓库。

10.4 设置本地仓库和远程代码仓库的链接

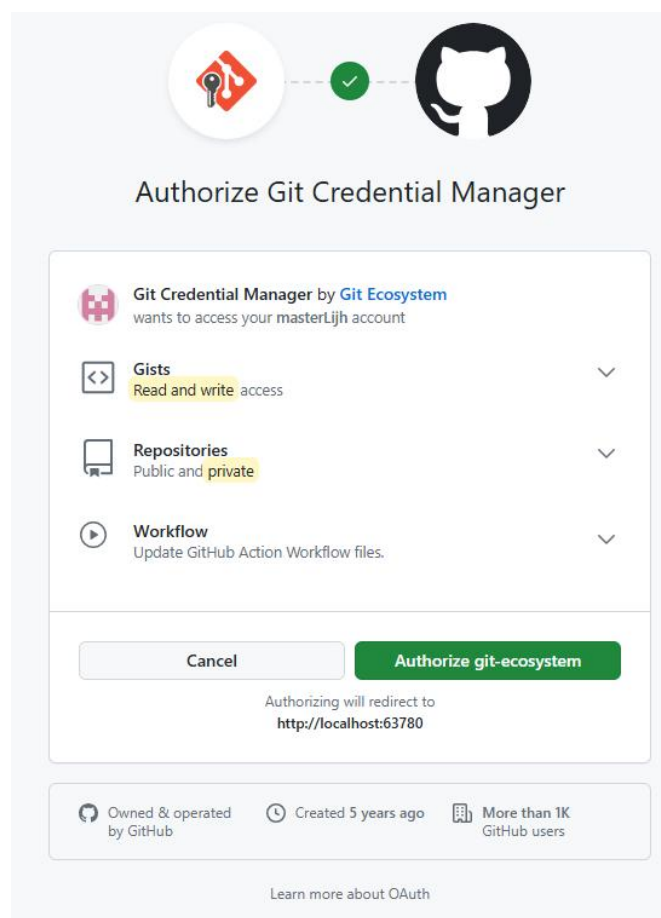
进入本地 webUI 项目的文件夹后，通过下面的命令把本地代码仓库与远程建立密钥链接

```
$ echo "WebUI 应用的远程 http 服务器设置" >> README.md
$ git init
$ git add README.md
$ git commit -m "这是我第一次把代码仓库上传至 gitHub 平台"
$ git branch -M main
$ git remote add origin
    https://github.com/applelucky/applelucky.github.io.git
$ git push -u origin main
```

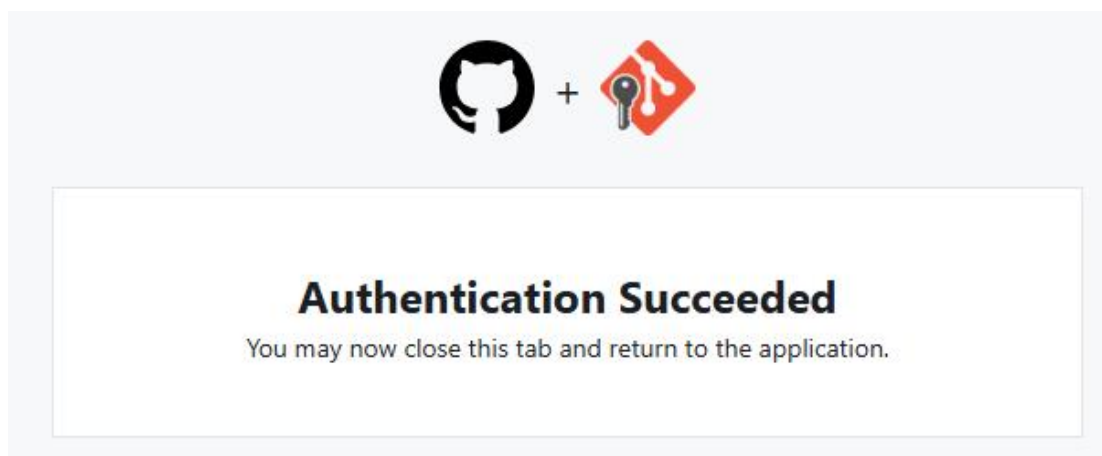
本项目使用 window 平台，gitbash 通过默认浏览器实现密钥生成和记录，第一次链接会要求开发者授权，如下图所示：



再次确认授权 gitBash 拥有访问改动远程代码的权限，如下图所示：



最后，GitHub 平台反馈：gitBash 和 gitHub 平台成功实现远程链接。



从此，我们无论在本地做了任何多次代码修改，也无论提交了多少次，上传远程时都会把这些代码和修改的历史记录全部上传 github 平台，而远程上传命令则可简化为一条：`git push`，极大地方便了本 Web 应用的互联网发布。

远程代码上传后，项目可以说免费便捷地实现了在互联网的部署，用户可以通过域名或二维码打开，本次使用 PC 的微软 Edge 浏览器打开。

参考文献

- [1]W3C. W3C's history. W3C Community. [EB/OL]. <https://www.w3.org/about/.https://www.w3.org/about/history/>. 2023.12.20
- [2]Douglas E. Comer. The Internet Book [M] (Fifth Edition). CRC Press Taylor & Francis Group, 2019: 217-218
- [3]John Dean,PhD. Web programming with HTML5,CSS,and JavaScript[M]. Jones & Bartlett Learning,LLC. 2019: 2
- [4]John Dean,PhD. Web programming with HTML5,CSS,and JavaScript[M]. Jones & Bartlett Learning,LLC. 2019: xi
- [5]Behrouz Forouzan. Foundations of Computer Science[M](4th Edition). Cengage Learning EMEA,2018: 274--275
- [6]Marijn Haverbeke. Eloquent JavaScript 3rd edition. No Starch Press,Inc, 2019.
- [7]William Shotts. The Linux Command Line, 2nd Edition [M]. No Starch Press, Inc, 245 8th Street, San Francisco, CA 94103, 2019: 3-7