8319205241442265013257

本 科 毕 业 设 计 (论 文)

**题目：响应式互联网金融借贷平台设计**

|  |  |
| --- | --- |
| **学生姓名：** | **王 艳** |
| **学 号：** | **1501511717** |
| **班 级：** | **15网络工程2班** |
| **专 业：** | **网络工程** |
| **院 （系） ：** | **网络与通信学院** |
| **指导教师：** | **张 敏** |

年 月 日

毕业设计(论文)原创性声明

本人郑重声明：所呈交毕业论文，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

毕业设计(论文)作者签名： 导师签名：

日期： 年 月 日 日期： 年 月 日

毕业设计(论文)使用授权声明

本人完全了解成都工业学院有关保留、使用论文的规定，即：学校有权保留论文并向国家主管部门或其指定机构送交论文的电子版和纸质版，有权将论文用于非赢利目的的少量复制并允许论文进入学校图书馆、院（系）资料室被查阅，有权将论文的内容编入有关数据库进行检索，可以采用复印、缩印或其他方法保存论文。

论文作者签名： 导师签名：

日期： 年 月 日 日期： 年 月 日

**摘 要**

由于互联网技术发展迅速，以及移动端各种技术的发展和移动端设备快速的普及，网页设计师们就必须得让传统的PC端和各种各样的移动终端设备都可以很完美的呈现出用户体验一致的网页。针对这个需求，本文就基于响应式的Web设计技术的思想，以Bootstrap框架为开发工具，设计并实现了一个适应性很好的网站平台，而这个网站可以自动适应浏览的设备，并且是独立于设备。响应式设计思想搭配Bootstrap响应式的框架为网站开发提供了一种非常高效的方法。

关键词：响应式；P2P；用户体验；Bootstrap框架；媒体查询

Abstract

Due to the rapid development of Internet technology, the development of various technologies on the mobile side, and the rapid spread of mobile devices, web designers must have traditional PCs and a variety of mobile devices that can be perfectly presented. A web page with consistent user experience. In response to this demand, this paper based on the idea of responsive web design technology, using Bootstrap framework as a development tool, designed and implemented a very adaptable website platform, and this website can automatically adapt to the browsing device and is independent of device. Responsive design ideas with Bootstrap's responsive framework provide a very efficient method for website development.

Keywords: responsive;P2P; user experience; Bootstrap framework; media query

目 录

[引 言](#_Toc6036_WPSOffice_Level1) [1](#_Toc6036_WPSOffice_Level1)

[第1章 绪论](#_Toc31526_WPSOffice_Level1) [2](#_Toc31526_WPSOffice_Level1)

[1.1 研究背景](#_Toc31526_WPSOffice_Level2) [2](#_Toc31526_WPSOffice_Level2)

[1.1.1 研究现状](#_Toc31526_WPSOffice_Level3) [2](#_Toc31526_WPSOffice_Level3)

[1.2 研究目的和意义](#_Toc17131_WPSOffice_Level2) [3](#_Toc17131_WPSOffice_Level2)

[1.2.1 智能化更强](#_Toc17131_WPSOffice_Level3) [3](#_Toc17131_WPSOffice_Level3)

[1.2.2 更利于SEO优化](#_Toc19165_WPSOffice_Level3) [3](#_Toc19165_WPSOffice_Level3)

[1.2.3 更容易博得用户的认可](#_Toc30897_WPSOffice_Level3) [4](#_Toc30897_WPSOffice_Level3)

[1.2.4 对用户友好](#_Toc13290_WPSOffice_Level3) [4](#_Toc13290_WPSOffice_Level3)

[1.2.5 响应式网页设计的发展](#_Toc13077_WPSOffice_Level3) [4](#_Toc13077_WPSOffice_Level3)

[第2章 相关开发技术和手段介绍](#_Toc17131_WPSOffice_Level1) [6](#_Toc17131_WPSOffice_Level1)

[2.1 Bootstrap框架技术](#_Toc19165_WPSOffice_Level2) [6](#_Toc19165_WPSOffice_Level2)

[2.2 JavaScript技术](#_Toc30897_WPSOffice_Level2) [6](#_Toc30897_WPSOffice_Level2)

[2.3 JQuery框架技术](#_Toc13290_WPSOffice_Level2) [7](#_Toc13290_WPSOffice_Level2)

[2.4 HTML5和CSS3技术](#_Toc13077_WPSOffice_Level2) [7](#_Toc13077_WPSOffice_Level2)

[2.5 响应式的设计模式](#_Toc26175_WPSOffice_Level2) [7](#_Toc26175_WPSOffice_Level2)

[2.6 响应式布局工具](#_Toc12593_WPSOffice_Level2) [8](#_Toc12593_WPSOffice_Level2)

[第3章 网站分析](#_Toc19165_WPSOffice_Level1) [10](#_Toc19165_WPSOffice_Level1)

[3.1 可行性分析](#_Toc32699_WPSOffice_Level2) [10](#_Toc32699_WPSOffice_Level2)

[3.1.1 对搜索引擎友好，有利于SEO](#_Toc32699_WPSOffice_Level3) [10](#_Toc32699_WPSOffice_Level3)

[3.1.2 访问速度快](#_Toc12264_WPSOffice_Level3) [10](#_Toc12264_WPSOffice_Level3)

[3.1.3 跨平台运行](#_Toc3633_WPSOffice_Level3) [11](#_Toc3633_WPSOffice_Level3)

[3.1.4 统一的后台](#_Toc8279_WPSOffice_Level3) [11](#_Toc8279_WPSOffice_Level3)

[3.2 需求分析](#_Toc12264_WPSOffice_Level2) [11](#_Toc12264_WPSOffice_Level2)

[3.2.1 首页模块](#_Toc13970_WPSOffice_Level3) [12](#_Toc13970_WPSOffice_Level3)

[3.2.2 个人中心模块](#_Toc21196_WPSOffice_Level3) [12](#_Toc21196_WPSOffice_Level3)

[3.2.3 登录注册模块](#_Toc21134_WPSOffice_Level3) [12](#_Toc21134_WPSOffice_Level3)

[3.2.4 投资模块](#_Toc17000_WPSOffice_Level3) [13](#_Toc17000_WPSOffice_Level3)

[3.2.5 借款模块](#_Toc8236_WPSOffice_Level3) [13](#_Toc8236_WPSOffice_Level3)

[3.3 网站业务流程图](#_Toc3633_WPSOffice_Level2) [13](#_Toc3633_WPSOffice_Level2)

[第4章 网站设计与实现](#_Toc30897_WPSOffice_Level1) [16](#_Toc30897_WPSOffice_Level1)

[4.1 接口数据模拟与Ajax请求](#_Toc8279_WPSOffice_Level2) [16](#_Toc8279_WPSOffice_Level2)

[4.2 浏览器兼容](#_Toc13970_WPSOffice_Level2) [16](#_Toc13970_WPSOffice_Level2)

[4.2.1 使用浏览器私有前缀](#_Toc15141_WPSOffice_Level3) [16](#_Toc15141_WPSOffice_Level3)

[4.2.2 跨浏览器的默认样式](#_Toc23745_WPSOffice_Level3) [16](#_Toc23745_WPSOffice_Level3)

[4.2.3 使用我们比较熟悉也比较强大的CSS的hack技术](#_Toc5635_WPSOffice_Level3) [17](#_Toc5635_WPSOffice_Level3)

[4.3 功能模块](#_Toc21196_WPSOffice_Level2) [17](#_Toc21196_WPSOffice_Level2)

[4.3.1 首页模块](#_Toc23277_WPSOffice_Level3) [17](#_Toc23277_WPSOffice_Level3)

[4.3.2 个人中心模块](#_Toc1835_WPSOffice_Level3) [18](#_Toc1835_WPSOffice_Level3)

[4.3.3 登录注册模块](#_Toc2658_WPSOffice_Level3) [19](#_Toc2658_WPSOffice_Level3)

[4.3.4 投资模块](#_Toc3503_WPSOffice_Level3) [22](#_Toc3503_WPSOffice_Level3)

[4.3.5 借款模块](#_Toc5113_WPSOffice_Level3) [23](#_Toc5113_WPSOffice_Level3)

[4.4 响应式布局](#_Toc21134_WPSOffice_Level2) [24](#_Toc21134_WPSOffice_Level2)

[4.4.1 媒体查询方法：](#_Toc22858_WPSOffice_Level3) [24](#_Toc22858_WPSOffice_Level3)

[4.4.2 在文档中里面发Head标签中是需要引入一下代码的：](#_Toc12732_WPSOffice_Level3) [25](#_Toc12732_WPSOffice_Level3)

[4.4.3 字体自适应：](#_Toc27904_WPSOffice_Level3) [25](#_Toc27904_WPSOffice_Level3)

[4.5 开发中的问题及解决方案](#_Toc17000_WPSOffice_Level2) [26](#_Toc17000_WPSOffice_Level2)

[4.5.1 问题1：](#_Toc26523_WPSOffice_Level3) [26](#_Toc26523_WPSOffice_Level3)

[4.5.2 问题2：](#_Toc17415_WPSOffice_Level3) [27](#_Toc17415_WPSOffice_Level3)

[4.5.3 问题3：](#_Toc26920_WPSOffice_Level3) [28](#_Toc26920_WPSOffice_Level3)

[第5章 网站测试](#_Toc13290_WPSOffice_Level1) [29](#_Toc13290_WPSOffice_Level1)

[5.1 网站测试](#_Toc8236_WPSOffice_Level2) [29](#_Toc8236_WPSOffice_Level2)

[5.2 测试目标](#_Toc20182_WPSOffice_Level2) [29](#_Toc20182_WPSOffice_Level2)

[5.3 测试过程](#_Toc15141_WPSOffice_Level2) [29](#_Toc15141_WPSOffice_Level2)

[5.3.1 第一步](#_Toc24078_WPSOffice_Level3) [29](#_Toc24078_WPSOffice_Level3)

[5.3.2 第二步](#_Toc17077_WPSOffice_Level3) [30](#_Toc17077_WPSOffice_Level3)

[5.3.3 第三步](#_Toc32144_WPSOffice_Level3) [31](#_Toc32144_WPSOffice_Level3)

[5.3.4 第四步](#_Toc29644_WPSOffice_Level3) [33](#_Toc29644_WPSOffice_Level3)

[5.3.5 第五步](#_Toc2444_WPSOffice_Level3) [34](#_Toc2444_WPSOffice_Level3)

[5.3.6 第六步](#_Toc32341_WPSOffice_Level3) [34](#_Toc32341_WPSOffice_Level3)

[5.4 本章小结](#_Toc23745_WPSOffice_Level2) [35](#_Toc23745_WPSOffice_Level2)

[第6章 总结与展望](#_Toc13077_WPSOffice_Level1) [36](#_Toc13077_WPSOffice_Level1)

[6.1 总结](#_Toc5635_WPSOffice_Level2) [36](#_Toc5635_WPSOffice_Level2)

[6.2 展望](#_Toc23277_WPSOffice_Level2) [36](#_Toc23277_WPSOffice_Level2)

[参考文献](#_Toc26175_WPSOffice_Level1) [37](#_Toc26175_WPSOffice_Level1)

[致 谢](#_Toc12593_WPSOffice_Level1) [38](#_Toc12593_WPSOffice_Level1)

# **引 言**

随着网络的发展和普及，以及各类建站技术的更新与运用，现在我们要去搭建一个网站的话，需要的时间以及成本已经越来越低了，这就让企业的官方网站得到了非常大的发展。网页内容从以前的一些简单的网页展示，到现在最新的、最主流的网站类型—响应式网站的内容展示，更新迭代的速度非常之快。而基于HTML5技术开发的响应式网站能够自动适应各种分辨率的屏幕大小，实现多中盾覆盖，设计高端，丰富的特效让页面展示更加的精美。这已经是大势所趋，响应式的网站会吸引更多的用户访问，提升用户体验。

1. 绪论
   1. 研究背景

今天，以智能手机为代表的移动智能设备在很多方面已经取得了巨大的进步，硬件性能的突飞猛进和无线网络的日新月异，使得移动智能设备通过手指简单的触碰就可以完成一些以前无法想象的任务，移动浏览器能够呈现与桌面浏览器一样的效果。移动设备的这些进步使得响应式技术不断发展起来。

 而现在传统的Web应用技术也具有了比较完善的功能，和良好的用户体验。在PC端的话，用户使用最多的肯定就是浏览器了，通过访问不同网站，就可以满足用户的购物、社交、娱乐等等的一些需求。但是近几年呢，移动端的技术也发展的很快，各类移动端设备的出现，让移动端的设备用户拥有数很快就超过了PC端的。而移动端的设备分辨率，以及响应网络的速度都和PC端的不同，并且用户登录使用习惯也是大相径庭，那就使得传统的Web技术思维以及设计模式不再适用于移动端。如果是单独的一套Web界面那就无法满足用户的期望，更无法为用户带来良好的体验。所以，Web界面的设计不能再以浏览器主，而应该是以＂用户为中主＂以及＂移动优先＂的理念来进行设计。

* + 1. 研究现状

从2012年开始到去年为止，各个技术社区以及一些技术专家对[Web设计](http://www.codeceo.com/article/9-gif-web-design-history.html" \o "Web设计" \t "http://www.codeceo.com/article/_blank)的趋势预测，基本都提到响应式设计；到现在，很多企业网站依然还是采用的响应式设计理念。就目前情况来说，响应式网页技术的设计模式主要有以下两种： 第一个就是基于设备：他主要是通过一些主流设备的类型、及尺寸来确定布局断点，然后再设计多套样式，最后分别显示到相对应的设备。它的优点就是断点可以相对固定，缺点在于对一些新设备需要重新考虑设计方案。第二种是根据网页自身的内容进行布局的划分。优点是设备的兼容性更好，缺点是对网页样式不好把握。但总体来说，响应式的网站在现在以及未来几年之内还会继续保持，而当响应式设计思想提出到现在，它的概念在不断的扩展，越来越多的人都认识并且开始使用，很多人认可响应式设计的主要原因有：

（1）良好的用户体验。响应式往网站可以适应所有智能设备，并在移动端根据用户习惯做了优化（比如汉堡按钮替代桌面端的网页导航样式）。

（2）易于维护。一个网站即可解决兼容桌面端、移动端等多种设备，能有效降低维护成本。相对于建设独立的移动网站或手机应用程序 ，更易开发和维护。

（3）响应式网站根据屏幕大小获得最适合的图片，可以是不同尺寸或不同清晰度的图片，满足各种屏幕尺寸设备的需求。

（4）桌面端和移动端共用同一个网址，网站管理者无需申请额外的域名，用户也只需记住一个网站入口。

* 1. 研究目的和意义

如今，越来越多的企业在进行建站时，都选择采用H5响应式建站。响应式网站到底有什么魅力所吸引他们呢？很多人刚接触响应式网站，可能是因为它能够适应不同屏幕设备的这一特性。响应式设计建站技术成为了目前主流的建站技术，不但企业喜欢建设响应式网站，连个人建站都开始选择响应式网站。响应式网站的特立独行，能够让你的网站更容易的被别人知道，让人喜欢，所以响应式的网站是具有非常强大的优势所在的，如下几点：

* + 1. 智能化更强

企业想要构建响应式网站，它最大的优点就是具有智能、自动化的响应设计。构建这样的网站是可以自动识别到用户使用的终端设备以及分辨率的大小，然后再进行相应的显示器调整出它认为最合适的内容格式输出给用户，让用户可以很好的浏览到网页的内容。并且这种网站会自动适应设备，可以让内容在任何设备上都不会出现滚动条，用户只用放心的浏览网页，不用去在意网页会出现一些页面布局错乱，或者信息出现有误来影响自己正常浏览网页内容，这会大大提升用户体验效果的。

* + 1. 更利于SEO优化

建站项目开始之后，一般项目经理在建设响应式网站过程中会在内容和代码编辑上都会按照SEO原理进行规划，响应式的网站会提升网站的SEO，因为搜索引擎对于响应式网站是非常友好的，搜索引擎的爬虫蜘蛛机制会抓取到响应式网站的任何内容，这将是直接提高该网站的用户访问率以及点击量的快速有效方式，从而网站的排名以及各类种种都会让更多的用户看到并使用，SEO优化是网站运营重要的一点，但是响应式网站已经可以帮我们很好的做到。

* + 1. 更容易博得用户的认可

响应式网站是最近这两年最新，最流行的技术，也是未来企业网站发展的趋势，目前来说，这项技术并不完善，正处于成长的阶段，因此，当你的网站采用响应式技术制作时，你就已经是可以打败很多竞争对手了，因为你比他们早一步占领了市场，并且在你的网站中还会使用许多新的设计以及一些开发技术，而这些方式都给你的用户带来很好的第一印象。

* + 1. 对用户友好

众所周知，响应式网站设计能够给用户提供非常友好的Web界面，是提升用户体验最有力的方式[1]，毕竟它可以适应几乎所有终端设备的分辨率大小。而且到了现在，计算机技术发展迅速，几乎就是每天都会有新款智能手机发布出来，如果一直都保持有响应式设计的思想，那就是代表你可以让你的网站一直与用户有了互动，不用去因为更换设备而放弃对你的网站的一个访问，而这些都是响应式网站设计理念的一个初衷。

* + 1. 响应式网页设计的发展

    W3C 组织在 CSS3 的规范中，提出了媒体查询的概念，无论用户使用何种设备，都可以获得令人兴奋的体验。响应式网页设计不仅仅是一种跨设备的解决方案，而且是一种“移动优先”的全新设计模式。流动布局：流动布局将不再使用绝对单位，转而采用百分比等相对单位，使网页的布局可以根据视口的不同而发生改变。 媒介查询：它是 CSS3 新增加的一种属性，通常和最大宽度或最小宽度一起使用。这些临界点被称为断点[3]。网页被加载时，让设备浏览器的视口，在与之相符的断点范围内呈现对应的网页样式。从而实现网页在不同设备的不同参数下（分辨率、物理屏幕尺寸等）显示不一样的布局。 弹性图片：图片可以根据视口的变化而发生改变。而响应式图片已经不仅仅是宽度的自适应。还可以通过不同设备显示不同质量的图片，从而提高网站性能。 就目前情况来说，响应式网页技术的设计模式主要有以下两种： 第一种根据设备不同，在 CSS 中分断点把需要呈现不同样式的部分用媒体查询分开。优点在于断点可以相对固定，缺点在于对一些新设备需要重新考虑设计方案。第二种是根据网页自身的内容进行布局的划分。优点是设备的兼容性更好，缺点是对网页样式不好把握。当然，国内外对响应式网页技术的争论一直没有停止，其主要原因是由于响应式网页设计自身的优缺点决定的。

1. 相关开发技术和手段介绍
   1. Bootstrap框架技术

现在一般提到响应式网站构建，就会响应的提到Bootstrap框架技术，基本上面他们两个已经是搭配上了一样。Bootstrap框架是由国外的一个大型社交网站Twitter开源的，并且现在已经是全世界最流行的一个前端组件库了，它是用来开发响应式的布局网站，以及理念是提倡移动优先的原则Web项目[4]。而且Bootstrap技术呢，让我们在开发项目的时候，可以充分利用框架提供的 SASS 变量和大量 mixin、响应式栅格系统、可扩展的预制组件、基于 JQuery 的强大的插件系统，可以快速的上手开发一个完整的项目。框架集成了CSS预编译语言，但是也是可以让开发者自己设计自己加入数熟悉的CSS预编译语言的，开发者可以自己使用LESS、SASS预编译语言进行灵活、快速开发。开发者可以在Bootstrap的帮助下，写一套代码但是却可以在不同的终端设备上面跑，让开发者的后期维护工作变得简单[11]。

* 1. JavaScript技术

JavaScript ( JS )标准是ECMAScript ，从互联网技术疯狂发展的2012年开始，几乎所有的浏览器都已经开始支持JavaScript5.1了，旧的浏览器版本也至少支持ECMAscript3.0了。而在15年的6月，ECMA组织正式发布了ECMAScrip的第六个版本，所以这个版本的名字叫做ES6或者说是ECMAScrip2015。JavaScript是一种高级的，解释型语言的编程语言。JavaScript是一门基于原型、函数先行的语言，是一门多范式的语言，它支持面向对象编程，命令式编程，以及函数式编程[5]。它提供语法来操控文本、数组、日期以及正则表达式等，不支持I/O，比如网络、存储和图形等，但这些都可以由它的宿主环境提供支持。它已经由ECMA（欧洲计算机制造商协会）通过ECMAScript实现语言的标准化。它被世界上的绝大多数网站所使用，也被世界主流浏览器（Chrome、IE、Firefox、Safari、Opera）支持[12]。

* 1. JQuery框架技术

JQuery是一个快速、简洁的JavaScript框架，是继Prototype之后又一个优秀的JavaScript代码库[6]。很多人说也可以叫做框架，其实我觉得只能叫做库。库和框架的区别是在于，库是给你实现一套方法，让你去调用，你的知道怎么用，但是框架就是给你封装好方法之后，还给你指定用户，你只管按它的方式去使用，别管它怎么实现的。大家也都知道，JQuery设计的宗旨是“write Less，Do More”，（意思就是写少的代码，但是却做更多的事嘛）其实就是说它使用原生js帮我们封装了一大批我们经常可能用到的方法，然后让我们调用。它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互。

* 1. HTML5和CSS3技术

以现在的技术为标准，HTML5添加了许多新的语法特征，其中就是包括<video>、<audio>和<canvas>等元素等等内容这些元素是为了让现在的网页中的内容根据有可读性，并且内容更加的能被搜索引擎识别并抓取的。其它新的元素如<section>、<article>、<header>和<nav>则是为了丰富文档的数据内容[7]。当然，虽然增加的新的属性，但是也是为了相同的目的。与此同时，还把一些比较不规范的标签元素已经废除，将HTML更加标准合理化。然后就是还有一些元素，比如<a>、<cite>和<menu>被修改，重新定义或者说是标准化了。同时和DOM已经成为HTML5中的基础部分了。

而现在非常流行甚至已经大批量使用的CSS3， 是层叠样式表语言的最新版本，主要目的是扩展CSS2.1的一些功能。它带来了许多期待已久的新特性，例如圆角、阴影、gradients(渐变) 、transitions(过渡) 与 animations(动画) 。以及新的布局方式，如 multi-columns 、 flexible box 与 grid layouts[8]。CSS3中的新特性让Web网页更加具有活力，并且以往很多需要图片的地方，现在CSS3技术都可以实现了，网页中的图片加载一旦少了，整个网站的速度就提起来了。

* 1. 响应式的设计模式

现在，很多网站都会去选择响应式的一个设计模式，那响应式网站的设计模式到底有哪些呢？到现在为止，基本上就是下面的两种情况：

第一个就是基于设备：他主要是通过一些主流设备的类型、及尺寸来确定布局断点，然后再设计多套样式，最后分别显示到相对应的设备。就像图2-1基于设备示：那这种模式的一个优点就在于，断点可以相对的固定起来，对于提炼设计模式来说也是非常的方便；但是他的缺点就在于，由于设备的更新是非常快的，所以总有么有办法覆盖到的设备。

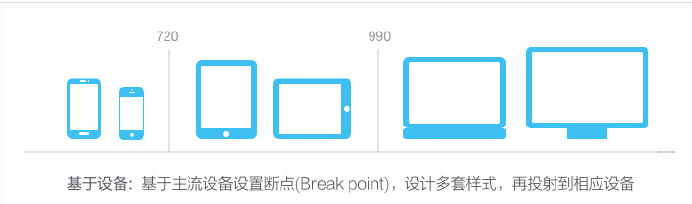


图2-1 基于设备

第二个呢就是基于内容：它主要是根据内容的可读性、易读性作为确定断点的标准，也就是说对内容进行布局的考虑的时候，先可以不考虑设备，然后有内容再去决定什么时候需要采用不同的展现形式。如图2-2示基于内容：而这种设计模式的优点就是在于它的适应性就会更加的强，几乎是可以弥补基于设备的缺陷，可以覆盖很多设备。但是也是有缺点的，就是很难提炼标准的设计模式，断点可能会根据内容不同而有所不同，需要考虑设备的物理尺寸。

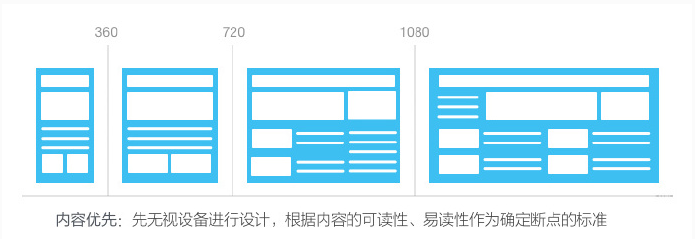


图2-2 基于内容

* 1. 响应式布局工具

我们构建响应式网站，最重要的就是考虑怎么去实现它的一个布局，以下几个工具是我们在开发中最经常用到的开发布局工具：

1. 第一个就是我们常见的流动布局：之前是专门指以百分比为度量单位的布局技术。我就此统为一个比较大的概念：在响应式设计的布局设计当中呢中，就不再像之前一样，专门以像素作为唯一单位，而是采用百分比或者混合百分比为单位，目的是设计出更加灵活的布局方式。
2. 第二个就是CSS中提到比较多的媒体查询[9]了：它可以让你根据在专门的环境下呢、查询到的各种各样的属性值。就比如之前经常提到的什么设备类型啊、分辨率啊、屏幕物理尺寸及色彩等等等一些因素来决定你的应用应该是什么样子的。通过使用媒介查询，基本上就可以获取到设备、设备的特性，并且可以给出一些求同存异的方法设计，这样子就可以解决传统Web技术解决不了的问题啦。
3. 接下来的就是弹性图片了：因为页面布局的弹化处理，图片是作为网站最能直接展示信息以及给用户带来最直观的的感受的元素，当然必须是更加的弹性了。所以我个人认为弹性图片是Ethan Marcotte提出设计产品时提出的概念，我们在后续的开发中会以图片为主，扩大研究范围：除了图片，还应该包括图标、图表、视频等信息内容的响应方式研究。
4. 网站分析
   1. 可行性分析
      1. 对搜索引擎友好，有利于SEO

在网上搜集到的观点：基于Flash的网站在搜索引擎上基本上是搜不到的，但是搜索引擎的爬虫蜘蛛就能抓取你的HTML5站点和获取你的网站内容。所有你写进网站的内容都会被访问到，这将会给我们的网站带来巨大并且不可预估的点击量，提升网站访问量。那就让我们跟着产品经理的角度理解，搜索引擎之所以被大家广泛使用，是因为在上面能找到我们想要的结果，而这个结果是通过网页展现出来的。所以为什么搜索引擎要对H5响应式网站更友好呢？互联网发展的时代，搜索引擎也是必须要做客户体验的：现在大家使用搜索引擎的智能设备多样。假如出来的网站是响应式的，搜索用户更容易得到自己满意的结果，这就是响应式网站最大的特点之一，能根据屏幕分辨率调整布局，以达到最佳的浏览效果。这是响应式最大的可行性之一，网站必须是面向用户开发的，永远以提升用户体验为最终目的[1]。

* + 1. 访问速度快

“天下武功，唯快不破”，网站的访问速度是毋庸置疑的重要。特别是在这个信息爆炸的时代，没有人愿意为你的网站等待10s，研究表明如果3秒之内网站无法加载出来，57%的网络用户会选择直接离开。HTML5+CSS3实现的响应式网站，加载速度非常之快。从我们阅站无数的经验中。我们会发现太多的老网站为了有设计感，或者布局，都牺牲了网站的质量——直接就是设计图稿切出来的图片贴在网站上。这种做法不仅网站画质感官不好，最重要的是保证网站足够的清晰图，图片大小一定不会太小，导致网站页面数据加载过多，打开速度自然慢了许多。而响应式网站都是静态页面，数据除了必要的产品图，BANNER图，都是文字的数据大小，几乎都是秒开[10]！

* + 1. 跨平台运行

跨平台运行这个优势跟上面说到的第一点里面的观点是一致的，正因为响应式网站有跨平台运行的优势，搜索引擎对响应式网站才更加友好。

因为响应式网站能适应不同类型的终端设备，开发者已经写好了代码，会让整个网站根据设备类型以及分辨率的大小自动调整网站的布局，来展示最合适的视觉内容，很好的解决了各种手机终端对传统的PC网站不友好的问题。本身企业性的网站就是为了一个官方形象和企业宣传，拥有响应式网站就丰富了更多的场景，还能应用于微信平台。可以直接绑定在微信公众号的自定义菜单[11]。

* + 1. 统一的后台

最后则是在后期维护上，网站做好了只是一个开始，真正好的网站是需要时常打理和维护的。传统网站因为PC、手机是独立后台管理，因此更新内容时需要分开进行，可以说是维护起来也特别困难，开发者的工作量也是变多了，但是HTML5的响应式网站就不会再这样子操作了。这对于整个开发流程来说节约了不少时间以及人力成本。

根据以上几个以用户角度、开发者角度、产品角度、后期维护角度，尤其是提升用户体验角度分析可见，响应式网站是时代趋势。不仅用户体验很好，搜索引擎也十分“偏爱”，这正是做网站的最终目标。所以，响应式网站的可行性是非常之大的[8]。

* 1. 需求分析

整个项目的需求分析是必须要做的，不然设计师和开发工程师没有一个标准，项目的需求分析也是为了让项目的开发的过程变得更加便利，一个优秀的前端工程师必须要有一个项目需求分析的能力。本文中这个项目就是主要以金融借贷平台作为一个列子，主要还是研究网站的响应式功能，项目主要以Bootstrap框架为主要技术选型搭建一个适应多分辨率的响应式平台。网站希望是可以实现在不同的终端设备，比如PC、手机、平板这些设备上，都可以把一些网站自动稍有区别但是主体内容不变的展示出来，让用户察觉不到。

* + 1. 首页模块

首页模块是整个网站的核心模块，所有二级页面的入口都会集中在首页模块，除此之外，首页是一个网站最先让用户看到并且感知到的，是用户最大的感官触发页面。所以企业的相关亮点信息要以界面美观简洁为标准展现给用户浏览。首页模块主要实现导航，轮播，借款列表，新闻展示，页脚展示企业或者公司的联系方式以及地址信息等。整个页面效果需要根据屏幕分辨率的不同，展示相应分辨率下最合适的展示效果。头部导航之内的控件，需实现在屏幕尺寸大小和手机分辨率大小大致相持的时候隐藏，并收起为一个菜单按钮，点击导列表出现。除此之外，实现部分鼠标滑入滑出的动态效果，提升用户体验。

* + 1. 个人中心模块

个人中心模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。内容区分在中间部分。采用左右结构，左边为个人中心模块下的子级页面菜单选项，分大类和小类，采用树形结构。动画效果是可以点击某一级页面时候，该菜单对应打开，其余关闭。动态关闭打开，左边的内容部分跟随菜单选择动态切换。最终还是适应多终端设备。

* + 1. 登录注册模块

登录注册模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。登录注册模块结构相似，都是在创建一个表单，我们的用户是需要通过它最初设置的用户名和她自己设置的密码进行登录，而注册是需要用户新添加用户名和密码，并且密码必须提供两次，为了确保用户的密码是他记住的并且是正确的，登录注册页面之间随时进行流畅切换。当然了，我们的登录注册用到的表单，用户输入的用户名以及它的密码之后是需要，在前端就进行验证的，验证格式是否是我们本网站要求的格式，如果都验证成功之后呢，才可以向后台发起请求进行数据库的用户比对，比对成功之后就登录或者注册成功才可以在本网站进行其他的操作，这样的验证，会大大减少用户对服务器发起的请求，减轻服务器压力。校验方式我们是自己编写一些正则规则，对于用户输入符合格式的用户名和密码，我们就首先进行一个错误的提示，让用户进行重新输入。当然，用户在用户名和密码都正确之后，是需要同意网站协议书，该协议通过蒙层方式弹出，用户体验较好。

* + 1. 投资模块

投资模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。投资内容显示一个投资列表，投资列表进行分状态查看，分别是全部、招标中、还款中三个状态进行分类显示。此处的数据为自己模拟的三个列表，通过前端模板框架将数据显示在页面中，三个状态切换以动画交互的样式提示用户已经切换或直接可以看到目前处于哪个状态下。最终整体页面效果同样实现多种终端浏览差异不大，用户体验良好。

* + 1. 借款模块

投资模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。借款模块主要向用户展示平台可执行的借贷方式，以列表的方式进行展示。并展示每个借贷产品需要提供的借贷资料。整体效果实现响应式的效果，适应多终端。

* 1. 网站业务流程图

不仅是网站，任何一个开发项目的流程图都是可以最快速度让客户、开发者、产品经理理解该项目的一个工作安排的方式。此项目主要针对响应式研究方向，仅仅实现前端页面之间的交互逻辑以及网站在不同的设备下的视觉效果。后台业务没有涉及。整体的一个前端页面之间交互逻辑流程如图3-1所示：

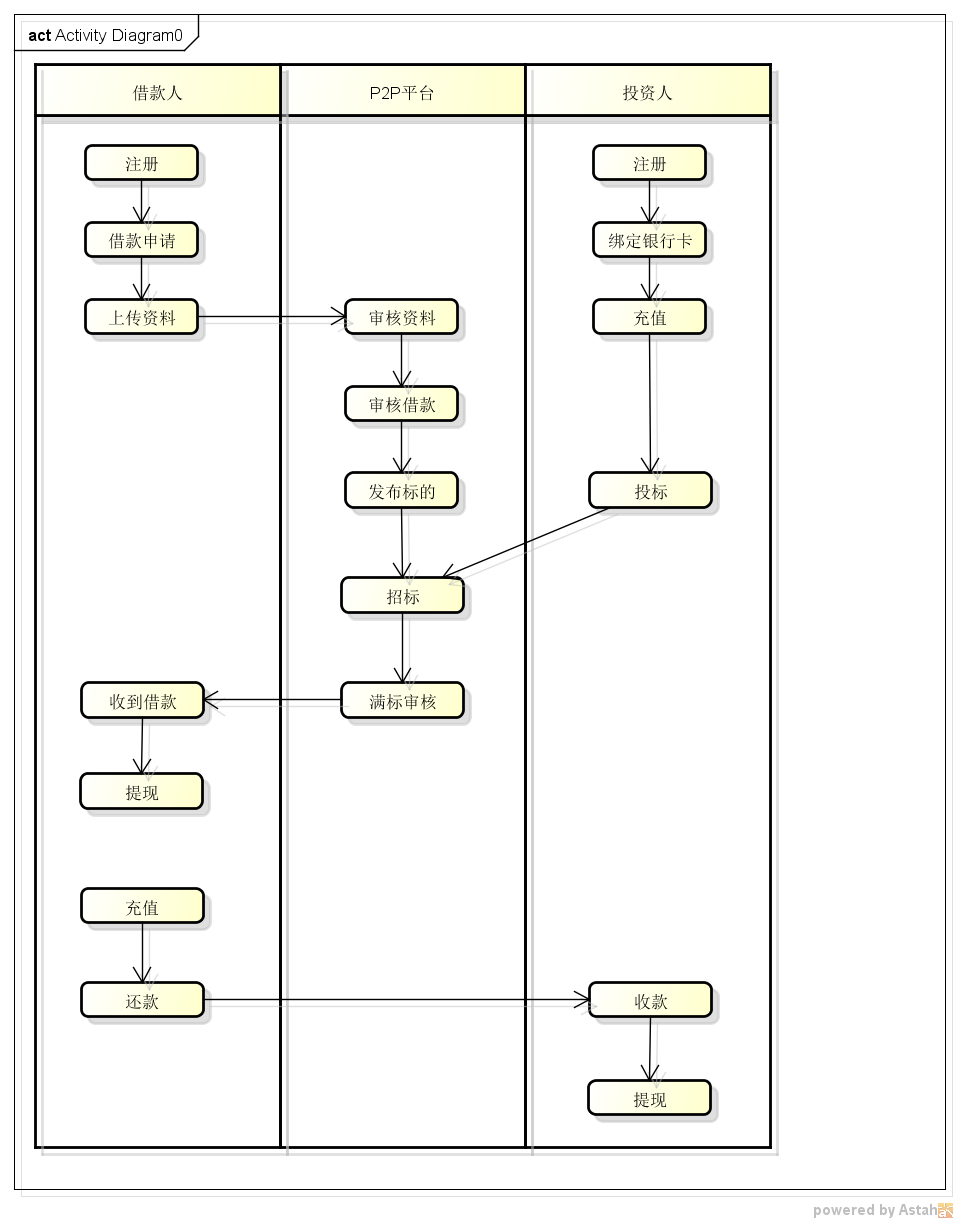


图3-1 金融借贷平台项目整体流程

主要分为几个流程：

1、借款流程：需要借款的人首先进行账号注册，同时需要提交借款资料（比如借款人的一些借款信誉、到底能不能还清债务的能力啊这些等，主要是为了管控风险度），之后平台风控会进行相应的审核、最终审核通过，发布借款；

2、投资流程：投资人（有钱人）还是首先进行账号注册，然后进行实名认证，之后通过就可以相应进行银行卡的绑定，接着对账户本网站账户进行充值，充值完成之后投资人就可以进行随意的投标了，

3、放款流程：平台满标审核，借款人收款，借款人提现；

4、还款流程：借款人会账户进行充值，借款人可以进行还款，投资人对应的进行收款，并且投资人继续投资、或者进行体现也可以

1. 网站设计与实现
   1. 接口数据模拟与Ajax请求

本网站采用的是前后端分离的开发模式，重点在于研究网站的响应式，即在不同的终端设备下的表现形式是否会适应，来达到最终的一个较好的视觉效果。所以没有搭建一个真正的后台与前端进行数据交互，该网站的所有的数据都是模仿数据接口自行模拟了一系列数据，数据都是存于本地，通过Ajax异步请求本地JSON文件，返回数据，请求路径是文件的相对路径，利用JavaScript拼接技术、JQuery的API有html()、text()将数据拼接到页面中进行渲染[6]。

* 1. 浏览器兼容

响应式网站主要的技术采用的HTML5和CSS3技术，技术更新迭代比较容易也就比较快也比较碎片，但是浏览器的更新速度不可能像更新技术一样迅速，它是一个慢慢迭代的过程。而HTML和CSS以及JavaScript的宿主环境都是浏览器，所以新技术面临的兼容性问题是开发者不可避免的一大问题。针对浏览器兼容，我这边采用了以下几个解决方案：

* + 1. 使用浏览器私有前缀

-moz内核- 火狐浏览器的主要内核

-Webkit内核- 是360、苹果、猎豹、搜狗、QQ、 Chrome等浏览器的主要内核

-o- 是Opera/欧朋浏览器的主要内核

-ms- IE、百度等浏览器

* + 1. 跨浏览器的默认样式

1、本网站是采用了默认样式库Normalize.css去代替 Reset.css，它的优势在于，会去保护有用的浏览器默认的一些友好的样式，并且主动修改浏览器自身带来的一些BUG，甚至还优化CSS的可用性，Normalize.cs它整体是模块化的，拥有详细的文档。兼容到浏览器IE8以上了。

2、Reset.css只会去移除掉浏览器上面的默认样式，然后再根据需要把样式再加回来最为补充[9]。

* + 1. 使用我们比较熟悉也比较强大的CSS的hack技术

常用CSS内 hack 浏览器兼容写法，这次浏览器的兼容我使用的是选择器的前缀法，条件注释IE10不支持了且条件注释里的代码是不仅仅对CSS有效，其它兼容问题也可以使用条件注释方法，浏览器的兼容只有CSS的样式有问题：

* 1. 功能模块
     1. 首页模块

首页模块是整个网站的核心模块，所有二级页面的入口都会集中在首页模块，除此之外，首页是一个网站最先让用户看到并且感知到的，是用户最大的感官触发页面所以得企业的相关亮点信息要以界面美观简洁为标准展现给用户浏览。首页模块主要实现导航，轮播，借款列表，新闻展示，页脚展示企业或者公司的联系方式以及地址信息等。整个页面效果需要根据屏幕分辨率的不同，展示相应分辨率下最合适的展示效果。头部导航之内的控件，需实现在屏幕尺寸大小和手机分辨率大小大致相持的时候隐藏，并收起为一个菜单按钮，点击导列表出现。除此之外，实现部分鼠标滑入滑出的动态效果，提升用户体验。原型图设计如图4-1所示以及后续效果图实现如图4-2和图4-3所示：

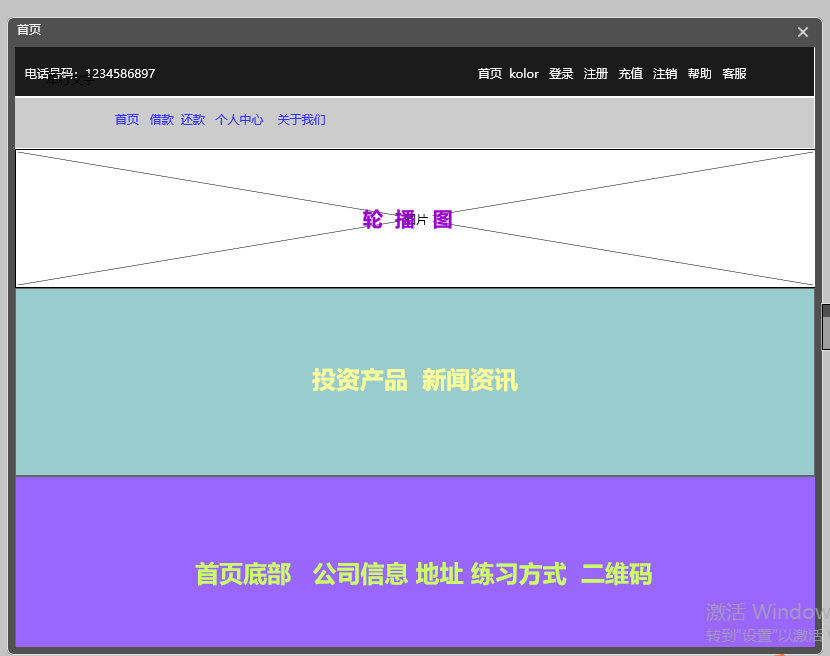


图4-1 首页原型设计图



图4-2首页效果实现图顶部



图4-3 首页效果实现图底部

* + 1. 个人中心模块

个人中心模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。内容区分在中间部分。采用左右结构，左边为个人中心模块下的子级页面菜单选项，分大类和小类，采用树形结构。动画效果是可以点击某一级页面时候，该菜单对应打开，其余关闭。动态关闭打开，左边的内容部分跟随菜单选择动态切换。最终还是适应多终端设备。原型图设计如图4-4以及后续效果图实现如图4-5的图片所示：



图4-4 个人中心原型图设计

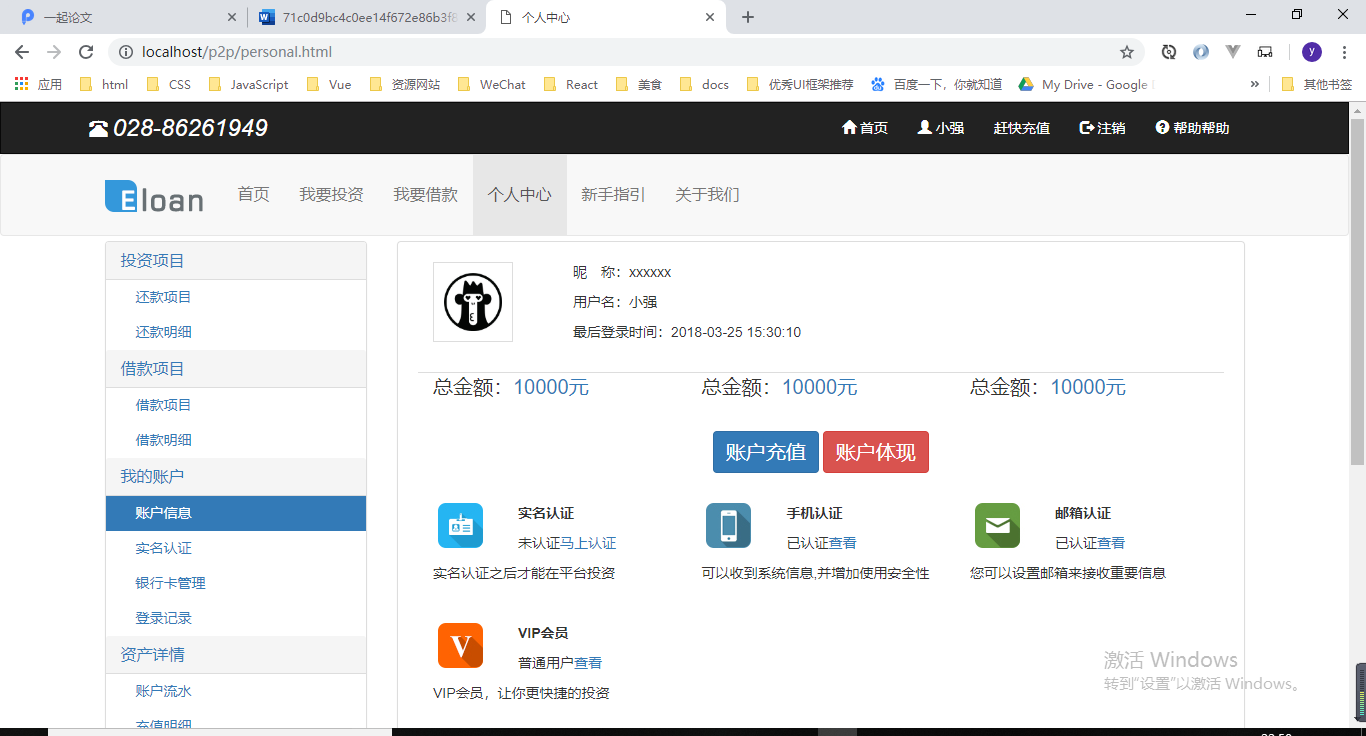


图4-5 个人中心效果图实现

* + 1. 登录注册模块

登录注册模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。登录注册模块结构相似，都是在创建一个表单，我们的用户是需要通过它最初设置的用户名和她自己设置的密码进行登录，而注册是需要用户新添加用户名和密码，并且密码必须提供两次，为了确保用户的密码是他记住的并且是正确的，登录注册页面之间随时进行流畅切换。当然了，我们的登录注册用到的表单，用户输入的用户名以及它的密码之后是需要，在前端就进行验证的，验证格式是否是我们本网站要求的格式，如果都验证成功之后呢，才可以向后台发起请求进行数据库的用户比对，比对成功之后就登录或者注册成功才可以在本网站进行其他的操作，这样的验证，会大大减少用户对服务器发起的请求，减轻服务器压力。校验方式我们是自己编写一些正则规则，对于用户输入符合格式的用户名和密码呢，我们就睡首先进行一个错误的提示，让用户进行重新输入。当然，用户在用户名和密码都正确之后呢，是需要同意网站协议书，该书通过蒙层方式弹出，用户体验较好。注册原型图设计如图4-6、登录原型设计图如图4-7以及后续效果图实现如图4-8和图4-9所示：



图4-6 注册模块原型设计图



图4-7 登录原型图设计

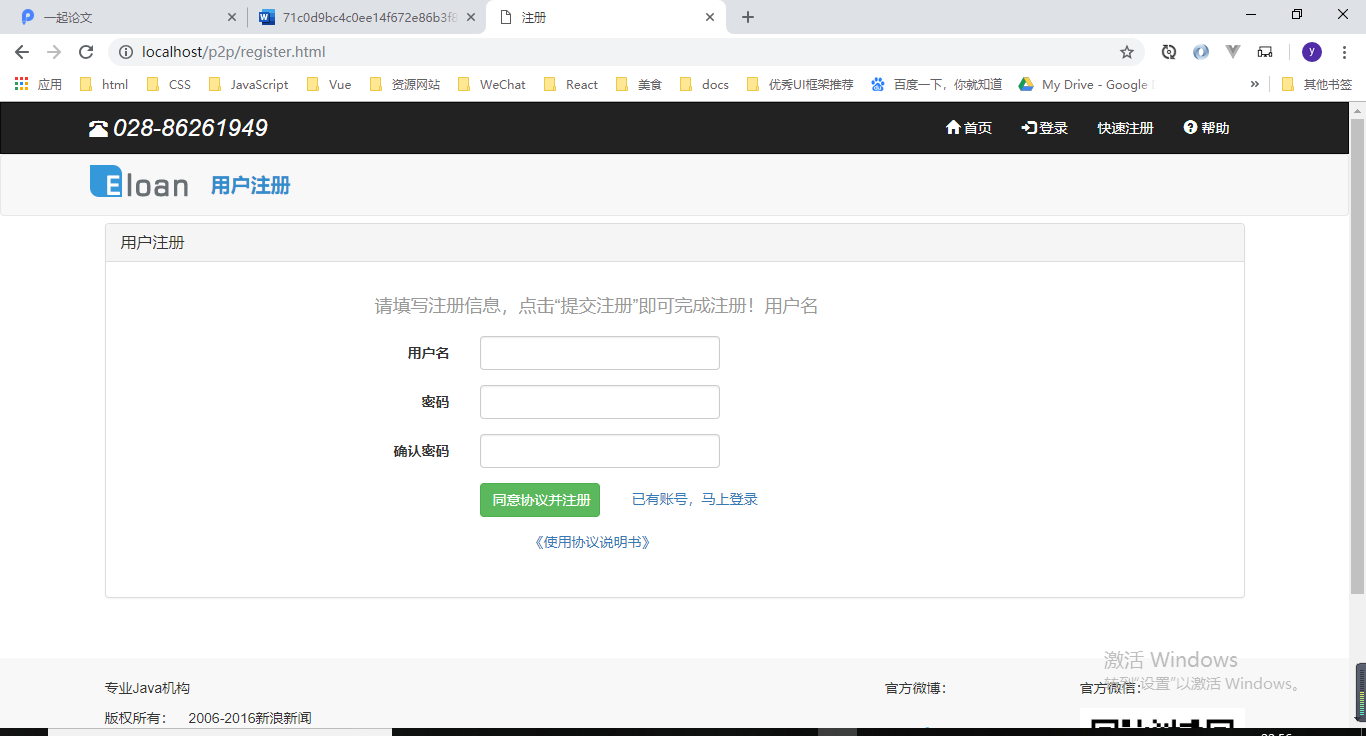


图4-8 注册效果图实现

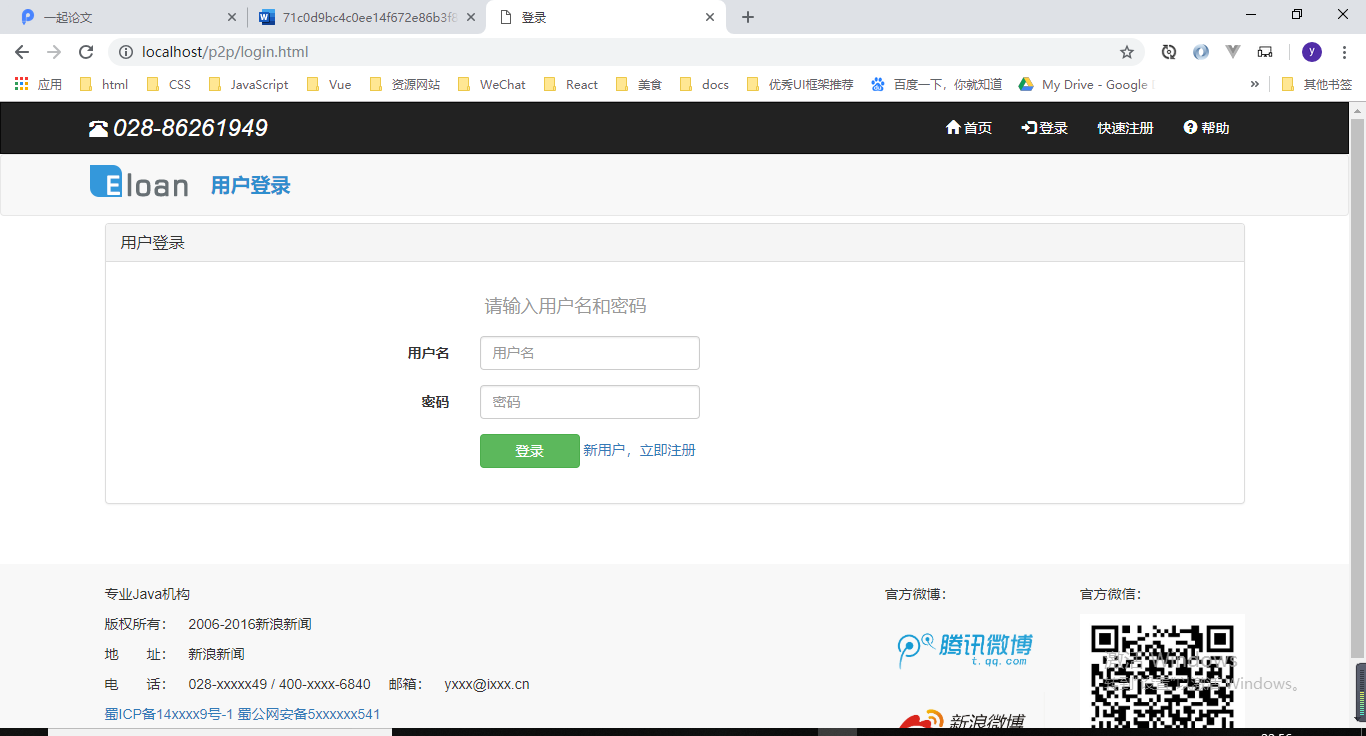


图4-9 登录效果图实现

* + 1. 投资模块

投资模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。投资内容显示一个投资列表，投资列表进行分状态查看，分别是全部、招标中、还款中三个状态进行分类显示。此处的数据为自己模拟的三个列表，通过前端模板框架将数据显示在页面中，三个状态切换以动画交互的样式提示用户已经切换或直接可以看到目前处于哪个状态下。最终整体页面效果同样实现多种终端浏览差异不大，用户体验良好。原型设计图如图4-10所示，效果实现如图4-11所示：



图4-10 投资效果图设计

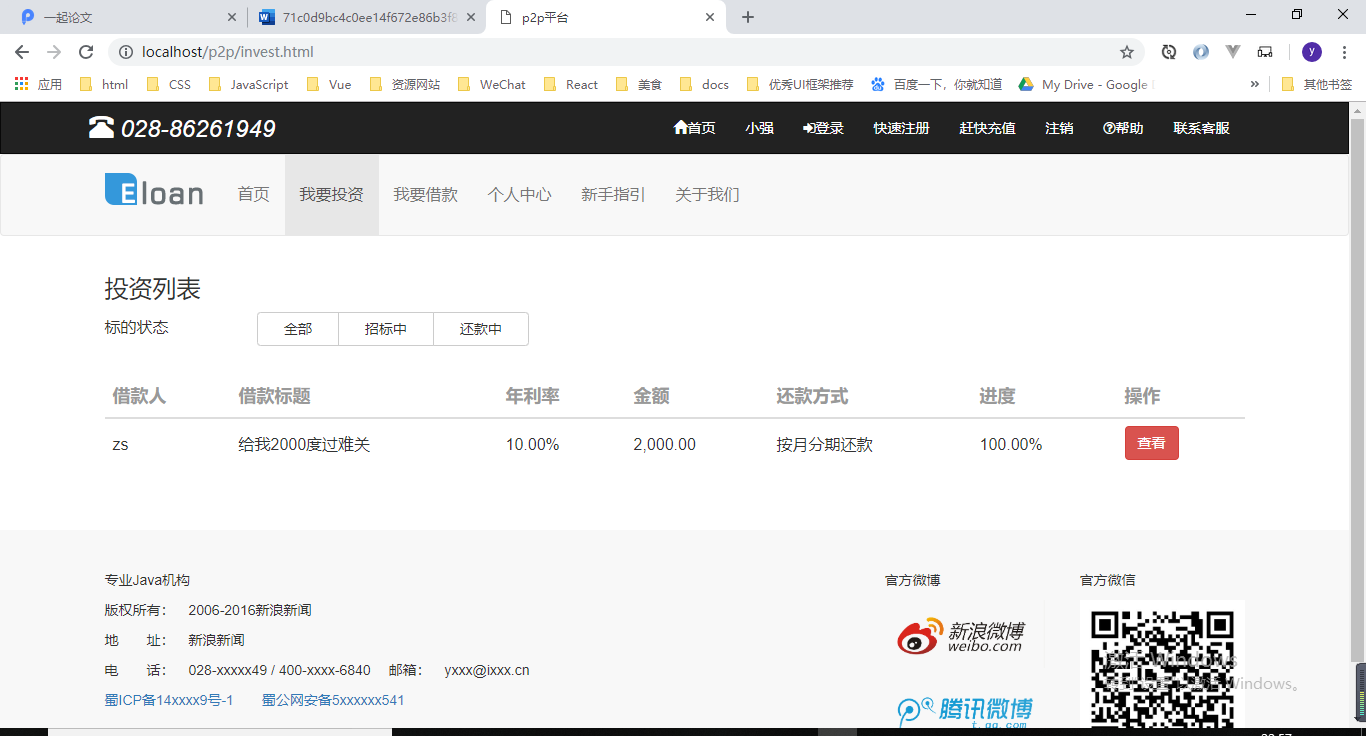


图4-11 投资效果图实现

* + 1. 借款模块

投资模块头部和底部选择和首页大致相同，实现效果以及动画效果也是保持相同。借款模块主要向用户展示平台可执行的借贷方式，以列表的方式进行展示。并展示每个借贷产品需要提供的借贷资料。整体效果实现响应式的效果，适应多终端。原型设计如图4-12和效果实现图4-13所示：



图4-12 借款原型图设计

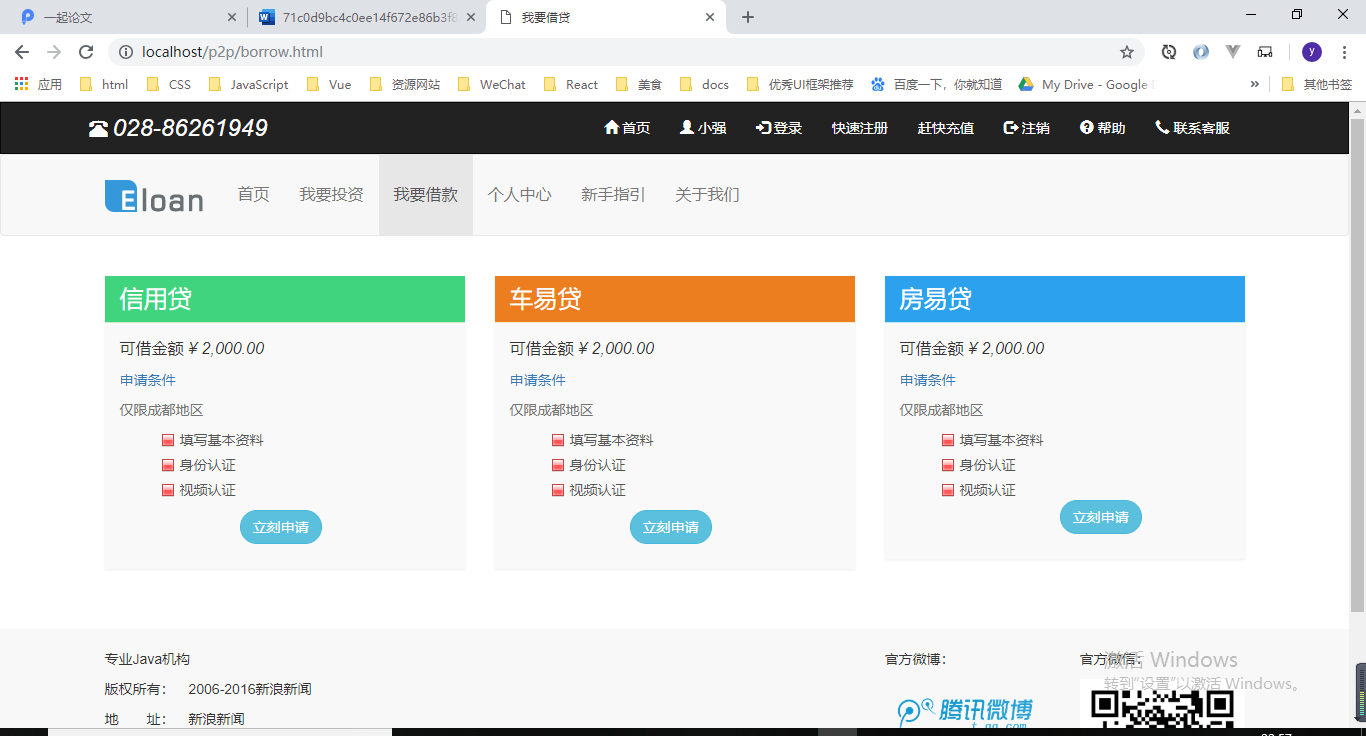


图4-13借款效果图实现

* 1. 响应式布局

想要做成响应式布局的方法是不止一种的，这里有很多种解决方案：比如百分比布局，bootstrap栅格系统，Flex伸缩盒布局，媒体查询，最优的选择就是流式布局和弹性布局及配合媒体查询。

* + 1. 媒体查询方法：

1、终端设备发屏幕需要使用screen;设备需要使用print;手机设备是需要使用device-width的

2、显示设备的信息是包含在了Window.screen 命令里面的

3、用来了解设备的分辨率的是Screen.height / screen.width 两个命令，可以成功的了解到设备的分辨率，

* + 1. 在文档中里面发Head标签中是需要引入一下代码的：

<metaname=’viewport’content=width=device-width,initial-scale=1,maximum-scale=1,user-scalable=no” />具体含义如下：

Width: 控制viewport的大小。如device-width为设备的宽度。

Height: 和width相对应，指定高度。

initial-scale: 最开始的缩放比例，（也就是说页面第一次加载时的缩放比例）

maximum-scale：终端设备允许用户缩放到的最大比例，（这个属性的范围从0到10.0）

minimum-scale: 是指终端设备允许用户缩放到的最小比例，（而这个属性的范围从0到10.0）

user-scalable： 是指的是用来判断用户是否可以手动缩放，属性值可以是: yes,ture 代表允许用户缩放页面；no,false 这个则就相反，代表用户不可以缩放页面[10]；

* + 1. 字体自适应：

字体自适应通常的话指的是给文档的根元素html,body设置基本的一个字体样式，单位vw:html { font-size: calc(112.5% + 4 \* (100vw - 600px) / 400); }或者也可以用JS来根据屏幕宽度去进行字体大小的一个计算：

(function (doc, win) {

vardocEl=doc.documentElement,

resizeEvt='orientationchange'inwindow?'orientationchange':'resize',

recalc=function(){

varclientWidth=docEl.clientWidth;

if(!clientWidth)return;

if(!doc.addEventListener)return;

win.addEventListener(resizeEvt,recalc,false);

在以前，如果我们想同样的一个网站去实现针对手机端啊、PC端啊、平板电脑这些等等不同的一些终端设备的显示支持，同城就是只有针对不同的终端设备设计出来多套不同的网页出来，然后再在页面被用户访问的时候增加一层判断，判断用户的设备属于哪一种类型，然后再相应的去调对应设备的页面展示给用户，来达到用户用户不同设备访问网页的一致性。但是，这样子的做法是弊端是非常明显的，不管对于设计师们还是开发者来说，都是非常繁琐的，因为额外的实现，后续设计师可开发者之间针对对应网站发维护和开发代价是非常高的。这个时候，记得考虑我们有没有一种方法，只需要实现一份设计和代码就可以针对不同终端设备的分辨率自动的做出展示上面的调整呢。Media Query为这种思路的实现提供了支持。

@media screen and (min-width: 961px) and (max-width: 1200px) {p {color: pink;}}我们上面的这段代码的意思就是，当设备的展现页面的宽度大于960px而且小于1200px时，就把文章中所有的p标签的字体颜色设置成为粉色。对于Media Query针对各大浏览器的兼容性，除了IE8及其以下的浏览器不支持，其他的主流浏览器基本上都支持。

* 1. 开发中的问题及解决方案
     1. 问题1：

bootstrap的模态框产生的问题，在登录页面的用户协议以及注销登录的地方项目有用到模态框展示内容，但是在最开始用的时候，因为不是很熟悉文档，所以老是出现问题，就经常会出现弹出模态框的时候它的遮挡层会把整个模态框都给遮住这样子。

解决方案：我不停的在各大网站上面搜索了很多的问题解决方案以及资料，但是还是没有解决，因为上面描述问题和解答都是一支半截，并不是完整的我遇到的情况。所以，最终没有办法还是去看官网文档，仔细看了文档很久，结果才发现问题出在模态框的HML代码的放置位置，人家官网文档明确说明了一定要将模态框的代码放在整个文档的最高的级别当中之中，这样才能保证模态框功能的完整性。并且，如果我们由于某些什么不确定的原因而没有把模态框放在body的直接子元素中去的话，那也可以通过js或者JQuery代码把模态框直接移动到body之中。通过这个遇到的问题就很明显的提醒了我，认真仔细查看文档真的很重要，因为你去搜索别人的解决方案，那很多都是别人的一知半解，你不熟悉文档，就算用起来也是一团懵。

* + 1. 问题2：

在优化首页的导航栏的时候遇到了一个比较简单的问题，但是还是由于不熟悉文档而蹲坑蹲了很久。希望实现的效果是导航在浏览器正常或者说设备宽度足够的情况下是这样子的效果，如图4-14所示：

图41

图4-14首页导航

浏览器和设备分辨率缩小了之后变成这样子，如图4-15所示：

图42

图4-15首页导航

点击右侧下拉之后可以如图4-16所示：

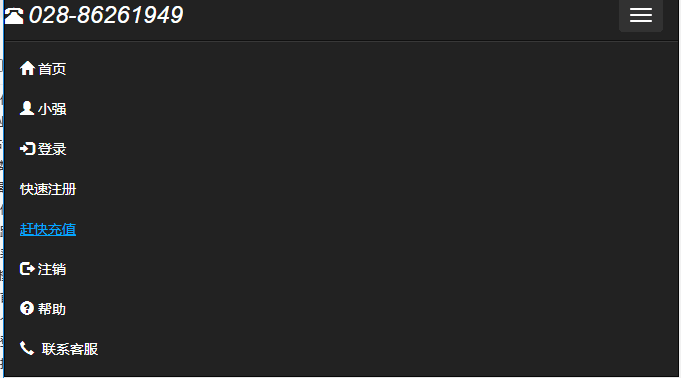


图4-16首页导航

但是结果却变成了如图4-17所示：

图44

图4-17首页导航

解决方案：找了很久，一直以为是bootstrap-resposive.css这个文件没有引进去，最后才发现这个文件已经引入，而且下面的内容都显示正常，主要是bootstrap-resposive.css文件里控制这个下拉按钮的代码出了问题，他原本的代码是：.navbar.btn-navbar { display: block; }，而我的这个button外面并没有一个.navbar的标签，因此我加上了.navbar.btn-navbar{ display: block; } .nav>.navbar-inner>.container>.btn{ display: block;margin-top: 7px; }这样就好了。

* + 1. 问题3：

我本来一开始是想要建一个collapse组件，然后再折叠内容展开之后呢，页面可以根据内容的多少去自动决定决定该容器的大小，但是遇到个文图就是，在加入collapse组件之后，折叠内容展开却出现了预料之外的滚动条，我就通过去设置盒子的overflow-Y属性但是还是不能去掉滚动条，最终思考了很久，那就只能通过设置外层容器的高度来把盒子height:auto，才最终解决了问题。

1. 网站测试
   1. 网站测试

本网站开发出来不仅需要顾及不同的浏览器，还要同时满足在电脑，平板电脑及行动装置各种尺寸荧幕都能正常成显示，但是真正测试的时候不可能都手持一款这些设备进行测试，那是不可取不切实际并且会大大增加开发时间的措施。通过在网上查阅资料以及结合自身实际情况，我最终选择了一款很实用的开发工具-Responsive Web Design Tester进行测试，它可以让我们在浏览器轻松模拟、切换不同发装置类型，协助测试自适应网页设计是否能正确显示内容。Responsive Web Design Tester是一个浏览器扩充功能，支援Google Chrome、Firefox及Opera ，使用非常简单，安装后就能快速切换要模拟的行动装置类型，内建包括Android手机、Nexus平板电脑、BlackBerry、Kindle Fire 、iPad和iPhone，也能依照开发者需求来自订加入要测试的装置解析度。因为我们不可能找来这么多的装置实际测试网页显示情形，就由 Responsive Web Design Tester 便能将浏览器模拟成各种行动装置解析度，实际测试网页显示效果。以往我都是使用User-Agent Switcher功能来自订浏览器的User Agent，不过这方法其实没那么好用，又要经常切来切去实在很不方便，使用Responsive Web Design Tester彻底解决开发上的问题，让开发者在修改网页时能更加得心应手。

* 1. 测试目标

在目前市场流行的几大浏览器上面进行测试，以及各个厂商的移动端设备上面进行测试，最终目标达到所展示的内容根据设备自动适应，以最优的方式把内容进行显示或者隐藏。

* 1. 测试过程
     1. 第一步

安装Responsive Web Design Tester，因为Responsive Web Design Tester是浏览器的扩展功能，以Chrome为例，我是通过在Chrome的应用商店里面搜索这个插件功能，和Chrome其他产插件的方法一样，把这个插件也添加到Chrome浏览器中去，看到浏览器导航栏出现如图5-1所示的按钮表示成功安装Responsive Web Design Tester。

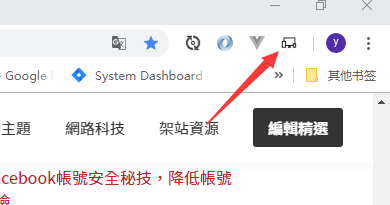


图5-1 浏览器插件测试工具

* + 1. 第二步

在浏览器安装Responsive Web Design Tester后，点选该按钮会显示工具的主功能，我们从「Select Device」选择要测试模拟的装置名称类型，如图5-2所示：

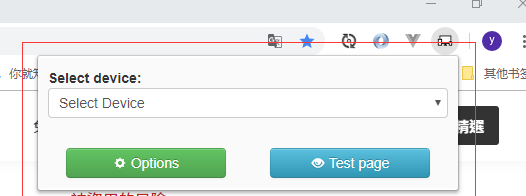


图5-2 插件按钮

可以看到 Responsive Web Design Tester 预设已提供约24 种的装置类型，包括大家熟悉的HTC One、Nexus 7、BlackBerry、Kindle Fire、iPad、iPad Pro、iPad Mini 及iPhone 。每一种装置都会有两种选项，分别为横幅式（Landscape）和直立式（Portrait）。这也会模拟手机或平板电脑在直立或横放时呈现出的不一样的解析度。如图5-3所示：

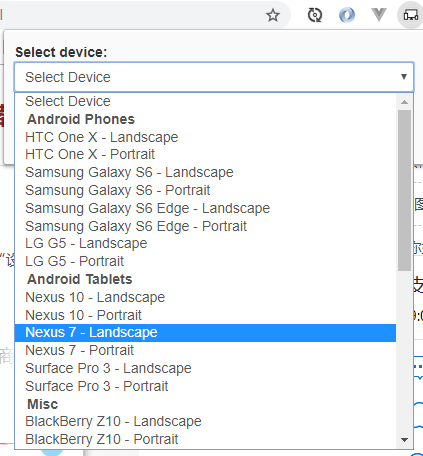


图5-3模拟设备选项

* + 1. 第三步

试着选择直立的「iPhone 7」来模拟浏览本网站首页的样式，Responsive Web Design Tester 会在浏览器内再开启一个适当大小的视窗。比较特别的是在测试中我发现网页被正确载入为行动版页面，这也表示Responsive Web Design Tester并不是只有调整视窗解析度，而是连User Agent也一并修改了！展示效果如图5-4和图5-5所示：

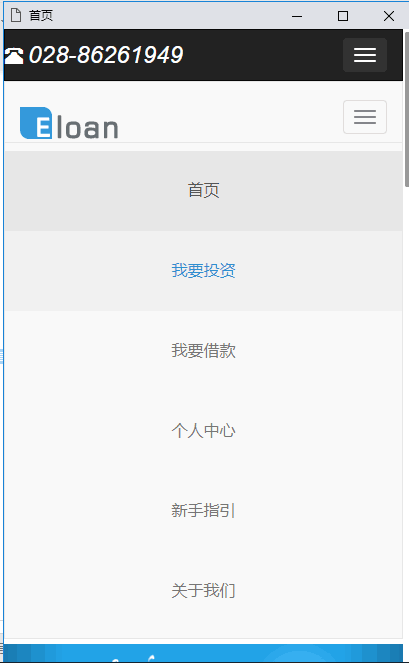


图5-4iPhone7顶部



图5-5iPhone底部

接下来我再使用「iPad」模拟浏览网页的效果，这次使用横幅式的装置，可以看到网页依旧能在该解析度范围内正确显示，这也就是前面提到的使用「响应式网页设计优势。展示效果如图5-6所示：



图5-6Ipad设备横屏

* + 1. 第四步

当然，测试环境可能会依照每个人的情形而有所不同，内建的行动装置类型并不一定符合每一位开发者。还好 Responsive Web Design Tester 内建选项设定，能让我们自由调整要模拟测试的装置类型和解析度，同时也能编辑群组来管理更多的装置，非常强大！如图5-7所示：

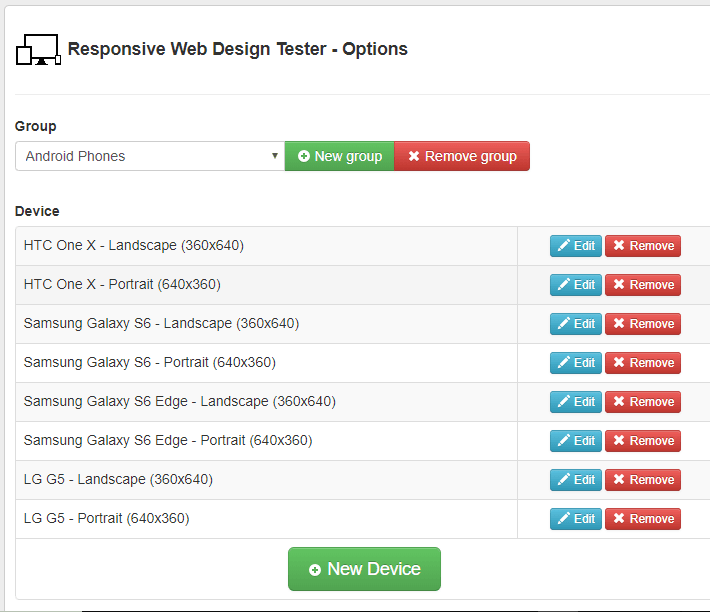


图5-7自定义设备

* + 1. 第五步

自定义PC端浏览器横向宽度像素为1200px的时候，测试效果如图5-8所示：



图5-8 PC端

* + 1. 第六步

在又在iPhone Plus的竖屏状态下进行测试，显示内容如图5-9所示：内容都是正常显示：



图5-9 iPhone Plus竖屏

* 1. 本章小结

整个测试过程还是耗费了大概一天的时间，从测试的效果来看，网站整体都是符合响应式网站的理念的，在不同的设备下显示对应的内容来保证用户体验。大方向上面都是没有问题的，只是有一些小地方的样式错误，专门针对的稍微调整一下就没有问题了。但是从整个测试以及调优的过程中来看，整个开发模式还稍有些欠缺，最优的是采取边开发边自主简单进行测试，这样的话才好针对问题马上进行调试。集中测试的话，定位问题会花掉比较多的时间。本身网站涉及的页面较多，所以开发者遗忘性比较大，这样会拖慢整个开发进度。

1. 总结与展望
   1. 总结

整个项目开发下来，学习到了很多技术，之前看到过的技术以及知识点基本上都在这次项目中用到了。前端技术知识点很多，靠背是不可能记住的，只有不断的练习，才会慢慢积累很多经验。而且从这次项目开发中深深的体会到JavaScript是一门弱类型的语言，很多地方都不会报错提示开发者，或者提示的不明确，这种问题只有经常练习才能快速定位到问题所在进行快速调试。还有就是遇到解决不了的问题，第一时间应该是看官方文档，在文档中详细查找问题解决方案，而不是在百度、Google中一通无方向的寻找，这样只会让自己更加无力。

* 1. 展望

整个项目围绕着响应式的方向实现一个比较完整的网站开发，响应式的效果也实现的比较好，用户体验得到充分发挥。当然，整个项目还有很多需要完善的地方，由于个人能力以及时间的限制，这个网站还应该配合真实的后台，才算是一个完整的网站。只是因为主要研究方向为网站的响应式设计，所以不用强制依赖后台也可以完成研究。但是后期应该去学习后台知识，把整个项目弄完整之后可以提交到GitHub仓库上面进行保存。还有就是需要多去深入了解偏网络层方面的知识，这些都是计算机基础原理和一些语言的底层原理支持，掌握之后，才会更加清楚出现的问题所在，更好的去解决问题。

**参考文献**

[1] Smashing Magazine.众妙之门──用户体验设计的秘密[M].邱胤焱，林本杰.北京：人民邮电出版社，2014.17-18.

1. Thoriq Firdaus.Responsive Web Design by Example [M].2nd Edition. PacktPublishing.2014.52-54.
2. Tim Kadlec.响应式 web 设计实践[M].侯鸿儒.人民邮电出版社 2013.2-4.
3. Tom Barker. High Performance Responsive Design Building Faster Sites AcrossDevices[M].O'Reilly.2014.1-2.
4. 孙萍.基于 Boot Strap 的响应式设计在 WEB 图书馆中的应用.内蒙古科技与经济[J] .2017 年20 期.60-61.
5. 吕媛媛.基于响应式体验的网络性能测量系统移动客户端的设计与实现[D]北京邮电大学.2016.1-3.
6. Benjamin La Grone.响应式 Web 设计 HTML 和 CSS3 实践指南[M]黄博文，饶勋荣.机械工业出版.2014.2-3.
7. 陈曦.面向多设备响应式设计的机制研究[D]北京邮电大学.2015.3-4.
8. 赵璐.基于响应式布局的移动端学生后勤系统的设计[D]东华大学.2016.12-14.
9. 王愉，潘明明.响应式网页设计初探.北京印刷学院学报[J].2014 年 03 期.14-16.
10. 张聪聪.基于响应式 Web 设计中用户界面的分析与探究[D].中南民族大学.2015.2-4.

[12] 雒仪. JavaScript语言在农田灌溉用水量计算中的应用[D]. 甘肃省平凉水文水资源勘测局, 2018.

**致 谢**

在成都工业学院的学习生活乃至我整个的学校生活即将结束了，在做毕业设计的这段时间里面，我的老师、同学给予了我很多细心的帮助，在此，对他们表示深深的感谢。首先，非常感谢我的指导老师张敏老师，对于整个毕业设计中不懂的问题张老师会给我们讲解清楚，并且一直督促着每个同学按时间安排完成相关的工作，尤其是论文的周记，我有两次忘记提交论文，张老师也会一直耐心的提醒按时完成，并且在百忙中对我们的任务进行审查，批改。其次，还要特别感谢王同学，给我介绍几个学习网站以及书籍资源，以及给我讲解业务上的逻辑，让我对我的整个项目有了更深刻的认识，开发加快。