**摘 要**

开发本系统的主要工作目的是将最新的超文本标记语言HTML5、CSS3，以及jQuery、Swiper框架应用到实际项目中，借助这些最新的网站设计技术实现页面布局、色彩显示和控件显示。同时能够很好的解决不同设备之间的兼容性问题，降低网站的开发成本，从而，推进这些互联网新技术的应用。

在整个项目的开发中使用多浏览器兼容的javascript库jQuery处理HTML事件、实现动画效果，以及为网站提供AJAX交互。同时，使用Node.js搭建服务器，通过mongodb创建数据库，express框架里的一般处理程序处理传入到服务器的HTTP请求，以返回我们需要的字符串或者对数据表中的内容进行修改。

最终实现了基于HTML5的校园帮App客户端。该客户端实现了“学习”、“娱乐”、“生活”、“个人”等页面。使同学们不必再在网站上下载手机App客户端文件，就可以直接访问校园帮App，降低了访问成本，节约了手机内存。

实践证明，虽然目前的框架还存在一些需要完善的地方，且目前人们对基于HTML5的移动Web开发的认识还不够深刻，但是借助于这些新的技术确实可以有效的提高项目开发的效率。

**关键词**：综合服务系统；javascript；jQuery；node.js；mongodb；express；HTML5

# 引言

1.1应用背景

最近几年发展最快的产业莫过于移动互联网，移动互联网的迅速发展带来了许多商机。从而，吸引了很多企业和个人的加入，而这恰恰又推动了移动互联网相关的新技术的推广和更新。移动互联网的发展离不开智能手持设备的迅速推广，而智能手持设备则因其相对廉价的硬件成本和使用的便捷性而得到广泛的应用。根据2014年智能手机的出货量数据统计，使用Android和IOS操作系统的智能手机占全球智能手机出货量的96.3%。因此，Google公司的Android和苹果公司的IOS占据了主要的市场份额。那么，我们在开发移动应用的时候至少要考虑这两种情况，而这势必增加企业或个人的开发成本。而基于浏览器的Web App的出现无疑会大大降低企业的开发成本，同样的，对于用户也不必再下载不同版本的手机应用程序，只要打开对应的网址即可。作为社会成员的一部分，我校的学生也有同样的需求。因此，非常有必要开发一款基于手机Web浏览器的校园帮APP，供同学们使用。

Web App，即网页应用． 它是一种依赖于 Web 浏览器，通过网络进行访问的应用程序． HTML5 的强势发展、智能手机的迅速普及，以及优质的用户体验都极大地促进了 Web App的发展．与原有的 Native App 相比，Web App 具有明显的优势，比如:可以一次开发多平台使用，应用开发成本较低;在支持HTML5 的浏览器上运行，直接适配多种移动终端;方便服务提供商随时发布更新;无需下载安装，打开浏览器就能使用．但也存在一些不足，比如:短时间内，用户体验不能与 NativeApp 相媲美;不能充分发挥移动设备的硬件功能;不能在离线状态下工作．为了证明用 Web App 取代 Native App 的可行性，微软与ZeptoLab 合作推出了基于 Internet Explorer 9 和 HTML5 框架的《Cut the rope》(即“割绳子游戏”，其 iOS 版本在 AppleStore 的下载量超过 5000 万)． 在国内，手机端新浪微游戏平台登场之初主打的也是 HTML5 游戏，中国移动游戏基地也表示将把 HTML5 作为其未来游戏平台发展的重要方向 ［1］ ．通过游戏这种形象直观的展示来证明 Web App 同样可以有良好的用户体验，这对于浏览器厂商来说是一件非常具有说服力的事情［2］ ．功能日益完善的浏览器在未来不再只是一个浏览、获取信息的入口，它还将拥有 PC 所有的功能，甚至成为一个新的

应用入口，取代操作系统．HTML5 的迅速发展推动了 WebApp 时代的到．

# 2 可行性分析

## 2.1技术可行性

自万维网联盟在2012年宣布HTML5规范正式定稿以来，HTML5在网站开发方面的应用铺天盖地席卷而来。与此同时，支持HTML5的浏览器也越来越多，从国外的Chrome、IE9、Safari、Opera到国内的傲游浏览器、搜狗浏览器、QQ浏览器等都开始具备支持和兼容HTML5的能力。由此可见，开发基于HTML5的校园帮App可以在较广泛的浏览器中运行。

## 2.2经济可行性

在经济方面，借助于现有的笔记本电脑、图书馆里的资料以及互联网上的资料完全可以在有限的时间内开发出满足同学们需求的webapp。不需要额外购买更多的硬件设备或软件。具备经济可行性。

## 2.3操作可行性

为了提高用户体验，并且在视觉效果上能够进一步的满足用户视觉体验，界面将设计非常符合当代大学生的审美风格。并且同时，校园帮App的操作使用也符合大学生对于一贯App的操作习惯。所以，对用户来说，只要会使用手机浏览器上网，就有足够的能力轻松地使用这款校园帮App。

# 3.需求分析

## 3.1 需求分析

目前，虽然存在有各类校园APP，但是在校大学生的很多需求依然没有比较合适的实现方法。对于大学生来说，无论是在生活上，还是在学习上，都会遇到各种各样的问题，可能在大部分的时间里，都很难快速得寻找到解决问题的相对来说比较合理的途径。尤其是在大学生刚步入大学的时候，处于一个完全陌生的环境中，对一些没有接触过的事情感到力不从心，这时候，如果有一个平台，在你需要帮助的时候，可以将自己的问题发布到一个其他人都能看到的某个地方，这样，别人就能看到你的问题，就会提供解决方法给你。 这样就能借助他人的能力来完成自己力所不能及的事情。这款“校园帮“APP就是立足于校园内部互助的需求从而为广大学生提供了这么一个高效互助的平台，这个平台供生活在同一个生活圈的大学生在上面发布需求、共享需求、接受需求，快速地满足需求。这款APP的功能模块一共有三个模块，分别是学习互助模块，校园生活互助模块，娱乐共享模块。

在学习互助模块中，有四个区域，1）期末考试区，2）四六级英语区，3）考研区，4）日常问答区。

在生活互助模块中 ，有四个区域，1）跳搔市场区，2）兼职消息发布区，3）日常问答区,4)图书阅读借还区。

娱乐共享模块中有2个区域，1）随聊区，2）文艺爱好区。

最后就是一个个人信息的页面。

这些功能能够给大学生的学习，生活和娱乐带来相对比较全面地帮助，在一定程度上方便了大学生日常生活及其各方面的需求。

## 3.2 界面的设计与规划

1、界面的设计

在界面设计阶段，为了提高用户体验，能够有一个合理的布局和符合审美规则的色彩搭配，设计人员全面地了解和调查了其他校园App的设计风格和样式布局，尤其是在色彩方面，做了比较多的对比和研究，然后再结合当代大学生对色彩的敏感和偏好，以及设计规则，和布局规范，最后设计出了整体的页面布局。整个布局格调相对来说比较活泼，功能设计也比较符合大学生的使用习惯。

2、 对页面的样式文件和脚本文件统一部署

因为用了express框架，这个框架有一个public的文件夹，里面一般用来放置js文件，css文件，img图片和其他模块或者框架文件。因为这个项目比较大，所以对于这些文件的合理管理就比较重要。我将所有的js文件放在了public文件下面的javascript文件夹里，所有的图片按照所属页面的不同放在了不同的文件中，然后所有的图片文件夹放在了public文件夹下面的img文件夹里，所有的css样式文件放在了public文件夹下面的css文件夹里。因为有的页面样式比较相似，就会用到同一个css的文件，为了在页面样式调试的时候能够快速的找到相关的样式文件，我专门用一个文档专门记录了每个不同的页面所使用的 css样式文件的名称。这有助于网站的后期维护，同时有助于统一网站的编码风格，同时也可以减少重复CSS代码的编写。

## 3.3 系统结构图

## 3.4软件要求

**3.4.1 HTML5**

HTML5是用于取代1999年所制定的 HTML 4.01 和 XHTML 1.0 标准的 HTML 标准版本，现在仍处于发展阶段，但大部分浏览器已经支持某些 HTML5 技术。HTML 5有两大特点：首先，强化了 Web 网页的表现性能。其次，追加了本地数据库等 Web 应用的功能。广义论及HTML5时，实际指的是包括HTML、CSS和JavaScript在内的一套技术组合。它希望能够减少浏览器对于需要插件的丰富性网络应用服务，如Adobe Flash、Microsoft Silverlight，与Oracle JavaFX的需求，并且提供更多能有效增强网络应用的标准集。 HTML5的一些特性： 1.表单验证轻松搞定。以前，制作一个表单提交页面，需要各种验证与判断，现在则只需要在标签内使用自带的属性和正则表达式，就能解决问题。 2.数据类型扩大一倍。大量新增的input属性带来的更加友好的用户界面，是以前需要花费巨大的精力与成本去实现的。可以毫不夸张地说，从一个小小的input标签属性的变化就可以预见HTML5应用的美好前景。 3.动画简单易懂。以前，制作一个简单的动画效果必须要依赖JavaScript和Flash，编写大量的脚本，或者制作动画帧。而现在，依赖强大的CSS3动画属性将快速高效地开发出各种动画效果，甚至3D效果的制作都将是一件轻而易举的事情。 4.原生支持无需插件。以前网页中对音频和视频的处理都会依赖各种插件来完成，这一直是使用HTML 4标准进行网页设计时的一个遗憾。但是，现在HTML5对音频和视频的处理有了强大的支持，audio和video标签就是专门为此而诞生的。各种可控的属性、预载、插页功能都令人激动不已，因为我们知道原生的意义重大。[3]

**3.4.2 CSS3**

CSS即层叠样式表（Cascading StyleSheet）。在网页制作时采用层叠样式表技术，可以有效地对页面的布局、字体、颜色、背景和其它效果实现更加精确的控制。 只要对相应的代码做一些简单的修改，就可以改变同一页面的不同部分，或者页数不同的网页的外观和格式。CSS3是CSS技术的升级版本，CSS3语言开发是朝着模块化发展的。以前的规范作为一个模块实在是太庞大而且比较复杂，所以，把它分解为一些小的模块，更多新的模块也被加入进来。这些模块包括： 盒子模型、列表模块、超链接方式 、语言模块 、背景和边框 、文字特效、多栏布局等。 CSS3将完全向后兼容，所以没有必要修改的设计来让它们继续运作。网络浏览器也还将继续支持CSS2。CSS3主要的影响是将可以使用新的可用的选择器和属性，这些会允许实现新的设计效果（譬如动态和渐变），而且可以很简单的设计出现在的设计效果（比如说使用分栏）。

HTML5的优点： 1）提高可用性和改进用户的友好体验； 2）有几个新的标签，这将有助于开发人员定义重要的内容； 3）可以给站点带来更多的多媒体元素(视频和音频)； 4）可以很好的替代FLASH和Silverlight； 5）当涉及到网站的抓取和索引的时候，对于SEO很友好； 6）将被大量应用于移动应用程序和游戏。

**3.4.3 javascript**

Javascript是一种由Natscape的LiveScript发展而来的原型化继承的面向对象的动态类型的区分大小写的客户端脚本语言，主要目的是为了解决服务器端语言，比如Perl,遗留的速度问题，为客户提供更流畅的浏览效果。当时服务端需要对数据进行验证，由于网络速度相当缓慢，只有28，8kbps,验证步骤浪费的时间太多。于是Netscape的浏览器Navigator加入了JavaScript,提供了数据验证的基本功能。由于Ajaxde chuxian ,JavaScript的功能被更新的越来越丰富，能做的事业越来越多，是Web2.0时代的象征。

一个完整的JavaScript实现是由以下3个不同部分组成的：核心（ECMAScript）、文档对象模型（Document Object Model，简称DOM），浏览器对象模型（Browser Object Model,简称BOM）。

JavaScript的一个重要功能就是面向对象的功能，通过基于对象的程序设计，可以用更直观、模块化和可重复使用的方式进行程序开发。

**3.4.4 jquery**

JQuery是继prototype之后又一个优秀的Javascrip库。它是轻量级的js库 ，它兼容CSS3，还兼容各种浏览器（IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+），jQuery2.0及后续版本将不再支持IE6/7/8浏览器。jQuery使用户能更方便地处理HTML（标准通用标记语言下的一个应用）、events、实现动画效果，并且方便地为网站提供AJAX交互。jQuery还有一个比较大的优势是，它的文档说明很全， 而且各种应用也说得很详细，同时还有许多成熟的插件可供选择[2]。jQuery能够使用户的html页面保持代码和html内容分离，也就是说，不用再在html里面插入一堆js来调用命令了，只需要定义id即可。JQuery是一个兼容多浏览器的javascript库，核心理念是write less,do more(写得更少,做得更多)。jQuery在2006年1月由美国人John Resig在纽约的barcamp发布，吸引了来自世界各地的众多JavaScript高手加入，由Dave Methvin率领团队进行开发。如今，JQuery已经成为最流行的javascript库，在世界前10000个访问最多的网站中，有超过55%在使用JQuery。JQuery 是免费、开源的，使用MIT许可协议。JQuery的语法设计可以使开发更加便捷，例如操作文档对象、选择DOM元素、制作动画效果、事件处理、使用Ajax以及其他功能。除此以外，jQuery提供API让开发者编写插件。其模块化的使用方式使开发者可以很轻松的开发出功能强大的静态或动态网页。JQuery，顾名思义，也就是JavaScript和查询（Query），即是辅助JavaScript开发的库。

**3.4.5 Node.js+Express**

Node.js是一个Javascript运行环境。实际上它是对Google V8引擎进行了封装。V8引 擎执行Javascript的速度非常快，性能非常好。Node.js对一些特殊用例进行了优化，提供了替代的API，使得V8在非浏览器环境下运行得更好。同时Node.js是一个基于Chrome JavaScript运行时建立的平台， 用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用[事件驱动](http://baike.baidu.com/item/%E4%BA%8B%E4%BB%B6%E9%A9%B1%E5%8A%A8" \t "_blank)， 非阻塞[I/O](http://baike.baidu.com/item/I%2FO/84718" \t "_blank) 模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。所以用node.js来搭建本app的系统是非常高效和方便的。Express 是一个简洁而灵活的 [node.js](http://baike.baidu.com/item/node.js) Web应用框架, 提供一系列强大特性帮助你创建各种Web应用。Express 不对 node.js 已有的特性进行二次抽象，我在它之上扩展了Web应用所需的功能。Express框架中丰富的HTTP工具以及来自Connect框架的中间件随取随用还有其对路由的定义，结合后台的mongodb数据库使用起来非常的灵活。尤其是在页面使用Ajax进行数据验证的时候，结合express的路由实现起来非常地简单和方便。而页面的动作特效使用javascript和其封装的函数库Jquery来实现。 部分动画效果是用jquery来实现的。JQuery的很大的特点就是快速、简单的 Java Scriptlibrary，其中jQuery拥有简洁的语法和很好的兼容性，非常大的简化了程序员的工作量，这也就是更好的映射了jQuery所一直强调的理念：“写得少，做的多。”jQuery是以 JavaScript为核心的编程语言。可以实现的功能有：操作 HTML文档、捕捉事件、对动画效果的修饰、与 Ajax的交换等等功能。页面布局是用html+css来实现的，页面布局的实现比较简单，我先根据设计图将页面分成相应的几部分。然后在具体每部分的内容再具体给相应的样式。

**3.4.4 Mongodb:**

MongoDB是一个基于分布式文件存储的数据库。由[C++](http://baike.baidu.com/item/C%2B%2B" \t "_blank)语言编写。旨在为WEB应用提供可扩展的高性能数据存储解决方案。同时它也是一个介于[关系数据库](http://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E6%95%B0%E6%8D%AE%E5%BA%93)和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。他支持的数据结构非常松散，是类似[json](http://baike.baidu.com/item/json)的[bson](http://baike.baidu.com/item/bson" \t "_blank)格式，因此可以存储比较复杂的数据类型。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立[索引](http://baike.baidu.com/item/%E7%B4%A2%E5%BC%95" \t "_blank)。

## 3.6 运行环境

3.6.1、Chrome浏览器

Chrome浏览器是一款简约，蕴涵了尖端的技术，可以让用户更快速、轻松且安全的使用网络的浏览器，嘎浏览器是基于其他开放原始软件所撰写，包括Webkit和Mozilla，目标是提升稳定性、速度和安全性，并向创造出简单且有效的使用界面。软件的名称是来自乘坐Chrome的网络浏览器图形使用着界面（GUI）。目前市场上的浏览器虽然品牌众多，但是其都是根据几大开源内核来构建的，主要可以分为两大类：IE系列和非IE系列。Chrome浏览器属于非IE系列，与IE和Mozilla等浏览器相比，chrome浏览器在javascript的执行速度上最快，对Css的渲染也是最快的。Chrome浏览器能够很好地支持HTML5，同时又是功能强大的调试工具。所以在开发过程中，重要用Chrome来调试app的功能。

3.6.2、移动设备

所有支持html5的安卓手机和iphone6及iphone6之前的版本的苹果手机，都可以通过在浏览器里面输入相关网址或者域名来登录校园帮app，就可以使用这款app了。

# 4总体设计

## 4.1 app系统设计

参照现有的校园app以及其他app,本系统对一些功能的实现做了一些优化和补充。首先，登录模块是必不可少的，在登录模块中设置了用户名和密码的验证，只有当用户名和密码都验证成功后方可进入系统的引导页，在引导页点击用户体验，就进入到系统的选择模块的页面。其中有学习模块，娱乐模块和生活模块可供你选择。最后还有个人信息的模块。页面下面的导航栏中分别有学习、娱乐、生活和我。点击学习，就会跳转到学习模块的页面；点击娱乐，跳转到娱乐模块的页面；点击生活，跳转到生活模块的页面；点击我，跳转到个人信息的页面。（每个模块都可以画一个逻辑图进行相关的说明）

## **学习模块，此模块中有4个区域，分别是期末考试区，四六级英语区，日常问答区和考研区。**

期末考试区，点进去之后，在搜索框中输入你的专业名称，点击搜索按钮，页面中会出现你的专业所对应的相关课程。点击你所需要复习的课程，就会进入课程复习的页面。在这个课程复习的页面中，分别有继续上次，平常练习，同学交流和复习资料等四个部分。点击平常练习，进入选择这门课的章节，选择章节，进入所选章节对应的练习的题目；点击复习资料，同理，进入你所选择的这门课的章节选择页，然后点击你所选择的章节，进入所选章节对应的复习题目的页面；点击继续上次，会进入到一个记录你上次复习或者练习的题目位置；点击同学交流部分，会进入到同学相互交流的页面中。在这个界面中，你可以提出任何上学上的问题，如果看到你问题的同学能够解答你，就会在你的问题下方给出相应的答复。同时，这个页面下方还提供了一些学习方法和答题技巧。这些内容会定期的更新。

四六级英语区，点进去之后，是一个选择四级或者六级的页面。点击英语四级，就进入英语四级的学习模块；点击英语六级，就进入英语六级的学习模块。两个模块中的内容模块分区是相同的。但是，每个模块的分区的里面的具体内容是不一样的。四级英语里面是四级英语的相关学习资料和题目，六级英语里面是六级英语的相关学习资料和题目。只是两个模块的框架和结构是一样的。就以四级英语为例来介绍其模块功能。

英语四级中有四个模块分区，分别是答题技巧，整卷练习，继续上次以及课程推送。点击答题技巧，进入到答题技巧的页面，这个页面里面是基于四级英语中不同的题目类型，分别给出的一些学习方法和建议，还有针对每类题型给出的一些答题技巧等；点击整卷练习，会进入到一个题目类型选择的页面。分别是听力，阅读，写作和翻译。选择你需要练习题目类型，进入到这类题目的章节选择页面，点击你需要的章节，然后进入这个章节的题目的页面。就可以开始做题练习了；点击继续上次，会进入到一个记录你上次练习的题目的页面；点击课程推送，跳转到课程推送的页面。在这个页面中有一些相关的视频教程可供你来学习和提升自己。这个页面下方也有相关的学习方法和技巧的推送。

日常问答区，在这个页面里，你可提出自己的问题。点击我要提问，会跳转到一个可以写出你的问题的页面，写完你的问题之后，点击发送按钮，这时候会回到日常问答区的页面，刷新页面，点击最新提问，你的问题就会出现在这个页面里。在提问题的页面中，万一你的问题你写完之后并不想提交，点击取消就可以了。这个页面中的问题是任何一个登录者都可以看到的。每个登录者都可以回答这个页面中的问题。回答问题的方法是，点击这个问题，会进入回答问题的页面，你在页面下方空白框中，输入你的答案，点击提交按钮，这个时候，你的回答就会出现在这个页面中，点击这个页面的返回箭头，从而返回到日常问答区的页面，刷新页面，点击最新回答，被回答的问题下面就会出现其所对应的答案。这样，你的问题就可以得到相关的解答。

考研区，这个页面里有六个部分，他们分别是校园资讯，政策新闻，头条新闻，调剂专题，时时热点以及常见问题。点击校园资讯，进入校园资讯页面，这个页面里是一些院校关于录取研究生的政策的一些改变，以及教育部所发布的一些有关硕士学位授予点的增加或者减少的消息；点击政策新闻，进入政策新闻页面，这个页面中详细的介绍了关于考研的相关政策和近期的一些有关考研的新闻报道。点击头条新闻：进入到头条新闻页面，此页面中罗列了近期三天内重要的考研新闻消息，有助于同学们能够及时，快速的了解到一些相关政策和消息；点击调剂专题，进入到调剂专题的页面，此页面中有一些学校相关考研调剂消息的发布；点击实时热点，进入实时热点页面，这个页面中是一些实时热点消息的发布；点击常见问题，进入到常见问题页面，此页面中是对于一些考研政策和专业成绩以及其他方面的问题的一些详细的解答。例如：可以参加调剂的考生所需要求，全国初试成绩基本要求分区的划定以及调剂的主意事项等。

**娱乐模块，此模块中有2个区域，分别是随聊区和文艺爱好区。**

点击随聊区，进入随聊区的页面，此页面中有电影、音乐、学校以及其他四个模块。点击电影，跳转到电影随聊区的页面，在这个页面中，你可以就某一部电影发表自己的一些感想，在文本域内写完内容，点击发表，你所写的文字就会出现在页面的下方的区域。别人登陆的时候也可以看到你所发表感想，这样就可以互相分享了。点击音乐，跳转到音乐随聊区的页面，同上，在这个页面中你可以就某首歌曲作一些分享，或者是有关最新音乐消息的发布的通知，让大家都能够更好地了解和欣赏到一些美的音乐；点击学校，在这个页面里，你可以写出自己有关学校各个方面的一些看法，或者对于食堂饭菜的吐槽，或者是浴室喷头坏了不修等一些糟糕的问题，还可以给大家安利一些周边的美食等方面的消息；点击其他，进入到“其他”随聊区的页面，在这里，你可以畅所欲言，但是还是得注意文明的。你可以寻找志同道合的朋友，可以分享人生的感想，可以吐槽生活中让你感到糟心的事情。

点击文艺爱好区，进入文艺爱好区的页面，这个页面中一共有4个部分，分别是电影区，音乐区，绘画区和其他。点击电影区，进入到电影区的页面，在这个页面里是一些电影的分享。点击音乐区，进入到音乐区的页面，此页面中是一些音乐的分享。点击绘画区，这里是一些优秀的作品的展示。点击其他是一个书籍推送的页面。

**生活模块，此模块中有四个区域，分别是兼职区，跳搔市场区，图书借阅区和日常问答区。**

点击兼职区，进入兼职消息发布的页面，这个页面有相关兼职消息的介绍，包括兼职置位，地点，日薪以及联系方式。当然，如果你有好的兼职消息，也可以在这里分享给大家。点击页面右上角的加号，会出现添加消息的矩形框，点击添加消息，会跳转到添加兼职消息的页面。在这个页面里你输入相关的职位，日薪等的兼职消息，然后点击发布，会自动跳转到兼职消息发布的页面。刷新浏览器，你所添加的兼职消息就会出现在兼职消息发布的页面中。

点击跳搔市场区，进入跳搔市场页面，在这个页面中展示了一些可以买卖的二手商品的相关信息。

点击图书借阅区，进入图书借阅系统。图书借阅系统是一个独立的系统，这个系统相对来说比较复杂。首先是一个选择目标的页面，你可以选择“个人”这个目标，也可以选择“图书管或者书店”这个目标。选择个人，进入个人借还图书的系统，一共有4个模块。在页面下面的导航栏中，可以选择“发现”，“我的书屋”，“消息”和“联系人”任意一个模块。

点击“发现”，进入发现的页面。这个页面中有“已借图书”，“附近图书”，“附近书友”，“推荐群组”，“推荐群组”，“推荐图书”等模块。点击上方导航栏的左边的个人头像，会跳转到个人资料的页面，点击个人资料页面中的“退出登陆”按钮，会回到生活模块的主页面。点击上方导航栏右边的加号，会出现搜索结束的矩形框，显示“搜索借书”的字样，点击搜索借书进入搜索结束的页面。

点击“已借图书”，进入已借图书的页面，在此页面中显示了你所借过的书籍。点击“附近图书”，进入附近图书的页面，此页面中展示了附近的图书信息。点击附近书友，进入附近书友的页面，此页面中展示了附近的书友的相关讯息。“推荐群组”页面则是一些书籍互借群组的消息的展示。“推荐图书”页面是推荐书籍的消息的显示。

点击“我的书屋”，点击上方导航栏右侧的加号，会出现“搜索加书”的字样，点击“搜索加书”，跳转到搜索加书的页面，在搜索框里填入你需要的书籍的名称，点击搜索按钮，在“我的书屋”的页面就会出现你所添加的书籍。

点击“消息”，进入消息的页面，在消息页面中显示了你所借的图书的情况，比如，成功借还，或者是接错了等的信息。

点击“联系人”，进入联系人的页面，这个页面中显示了一些借还书籍的好友，点击好友，会跳转到好友信息的页面。

“个人”的系统功能就是这些。“图书管或者书店”的系统功能和“个人”的类似，只是页面内的一些具体内容有所差异。

“日常问答区”和“学习”模块的“日常问答”区功能相同。

个人页面显示的是一些个人的基本信息。点击“编辑资料”，进入“编辑资料”的页面。你可以在此页面修改一些个人信息，比如“个性签名”，“性别”，“地区”等。

## 4.2 app系统功能设计

（1）注册登录

每个App都有用户登录注册的功能来进行身份识别。所以用户首先必须注册，提供可登录比对的个人验证信息，进入个人帐号之后才能进行其他操作。

（2）课程搜索

用户可根据自己的专业名称来搜索自己所学的课程，或者是根据课程名称来搜索此课程所对应的题目。

（3）修改密码

用户登录的时候，如果忘记密码，需要在忘记密码的页面重新输入自己的密码，对原有的密码进行修改操作。

（4）发布兼职消息

用户可以自行发布相关兼职消息。

（5）页面跳转

用户在使用app的时候，有点击页面跳转的功能，还有返回上一次所浏览的页面的功能。

（6）提问和回答

用户可以将自己的发布在问题页面，也可以救别人所提的问题作相应的回答。

app界面的实现：

界面实现的基础是UI设计人员设计的psd图，每个界面对应一张psd图。所以界面的内容的尺寸是由psd图中对应内容的尺寸来决定的。但是每种型号手机的屏幕大小不同，所以，psd图的尺寸不可能和手机屏幕的尺寸相同。需要根据psd的宽高和手机屏幕的宽高计算出一个合理的比例，方便在页面布局时可以根据psd图所给内容的宽度和高度来给盒子模型合适的宽度和高度，而避免出现页面比例不协调的情况。

为了让界面能够更好地适应不同尺寸的屏幕，我采用了响应式布局，在实现响应式布局的过程中我使用rem布局和百分比布局以及媒体查询等方法，提高页面的实现效果。rem的基准值，是用js来动态计算的。代码如下：

(function(doc, win){

var docEl = doc.documentElement,

resizeEvt = 'orientationchange' in window ? 'orientationchange' : 'resize',

recalc = function() {

var clientWidth = docEl.clientWidth;

if (!clientWidth) return;

console.log(clientWidth);

docEl.style.fontSize = clientWidth / 390 \* 16 + "px";

console.log(' docEl.style.fontSize'+ docEl.style.fontSize);

};

if (!doc.addEventListener) return;

win.addEventListener(resizeEvt, recalc, false);

doc.addEventListener('DOMContentLoaded', recalc, false);

})(document, window);

媒体查询是根据屏幕不同的尺寸，给出页面不同的最大的宽度和高度。而且要注意的是，每个页面的高度应该等于手机屏幕的高度减去手机导航条的高度，才是页面真实的高度。以下是页面中媒体查询的代码。

@media screen and (width:360px){

.content{

width:360px;

height:622px;

}

}

@media screen and (width:412px){

.content{

width:412px;

height:714px;

}

}

@media screen and (width:320px){

.content{

width:320px;

height:550px;

}

}

@media screen and (width:375px){

.content{

width:375px;

height:649px;

}

}

@media screen and (width:414px){

.content{

width:414px;

height:718px;

}

}

页面中的轮播图，我使用了Swiper框架来实现。

app的功能实现：

**（1）注册登录**

每个App都有用户登录注册的功能来进行身份识别。所以用户首先必须注册，提供可登录比对的个人验证信息，进入个人帐号之后才能进行其他操作。将页面中的数据发送到后台服务器和接受后台返回前端页面的信息的过程中，使用了Ajax来传递和接受相关参数。

**验证帐户是否存在**

用户登录的时候需要输入个人帐号和密码。在输入帐户之后，将用户的帐号发送到数据库进行查找，如果数据库中存在此帐号，则帐户验证通过。然后接着验证用户所输入的密码是否正确。如果数据库中不存在此帐号，则提醒用户帐号不存在。则需注册属于自己的帐号。如果密码错误，则出现“忘记密码”的提示，用户需要修改自己的密码。

$('#password').blur(function(){

var msg1='';

var count=$('#count').val();

var password=$('#password').val();

$.get('/checkname',{count:count,password:password},function(msg){

$('.word2 a').text(msg.data1);

msg1=msg.data2;

if(msg1==0){

window.location.href='../shouye';

}else{

$('#login a').attr('href','javascript:void(0)');

}

})

})

router.get('/checkname',function(req,res){

var obj={};

people.findOne({count:req.query.count},function(err,data){

if(err){

console.log(err);

}else{

if(data==null){

obj.data1='账户不存在';

obj.data2=1;

}else{

if(req.query.password==data.password1){

obj.data1='';

obj.data2=0;

}else{

obj.data1='忘记密码?';

obj.data2=1;

}

}

}

res.json(obj);

})

});

**帐户注册功能**

点击“注册”跳转到注册页面。注册页面中会验证用户帐号的格式是否符合手机号或者邮箱的规则，并且会在数据库中查找，验证用户输入的此帐号是否被其它人注册占用过。用户密码要求是由6-20位的数字或者字母组成，并且要验证两次输入的密码是否相同。当这些验证都通过，然后勾选“我同意本校园帮用户协议”方可完成个人帐户的注册。这时候，所注册的帐号和密码，会被更新到数据库。如果帐户和密码都验证通过，则跳转用户体验的页面。

//手机号的正则表达式验证

var pattern1=/^1[3|4|5|8][0-9]\d{4,8}$/;

//邮箱的正则表达式验证

var pattern2=/^([a-zA-Z0-9\_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/;

//密码的正则表达式验证

var pattern3=/^[0-9A-Za-z]{6,20}$/;

$('#count').blur(function(){

var username=$('#count').val();

if(pattern1.test(username)||pattern2.test(username)){

$('#count').css('borderColor','green');

$('#one').hide();

//将所要注册的用户名用Ajax发送到服务器，在数据库里进行查找和比对，检查是否有人已经用此用户名注册过

$.get('/checkcount',{count:username},function(msg){

if(msg==1){

$('#count').val('改用户名已经被占用');

}

})

}else{

$('#count').css('borderColor','red');

$('#one').show();

$('#one').text('请输入邮箱或者手机号');

}

})

var password='';

$('#pas1').blur(function(){

var password=$('#pas1').val();

if(pattern3.test(password)){

$('#pas1').css('borderColor','green');

$('#two').hide();

}else{

$('#pas1').css('borderColor','red');

$('#two').text('请输入6-20的数字或者字母');

$('#two').show();

}

})

$('#pas2').blur(function(){

var pas2=$('#pas2').val();

var password=$('#pas1').val();

if(pas2==password){

$('#pas2').css('borderColor','green');

$('#three').hide();

}else{

$('#pas2').css('borderColor','red');

$('#three').show();

$('#three').text('请与上面输入的密码保持一致');

}

})

//后台服务器接受到用户名，在数据库中进行信息查找，从而比对和验证。

router.get('/checkcount',function(req,res){

people.findOne({count:req.query.count},function(err,data){

if(err){

console.log(err);

}else{

if(data==null){

res.json('0');

}else{

res.json('1');

}

}

})

});

//将用户的帐号和密码信息更新到数据库中

router.post('/doLogin',function(req,res){

people.create(req.body,function(err,data){

if(err){

console.log(err);

}else{

res.redirect('/');

}

})

});

**修改密码**

如果帐号正确，密码错误，用户可能忘记了自己的密码。会出现“忘记密码”的提示。点击“忘记密码”，进入到修改密码的页面，你的帐号信息会直接出现在页面中，你只需要修改你的密码就可以了。修改完成的密码会被更新到数据库中。让这个页面的信息出现在另一个页面中，先用Ajax将数据信息发送到服务器上，然后，通过ejs模板引擎的用路由实现页面跳转同时可以替换页面信息的功能来实现。

<input type="text" name="count" value="" placeholder='<%= count%>' id="count">

//完成页面跳转和页面信息”count”的替换

router.get('/login/update',function(req,res){

res.render('login/update',{count:Data});

});

//更新用户的密码

router.post('/reLogin',function(req,res){

people.update({count:Data},{$set:{"password1":req.body.password1,"password2":req.body.password2}},function(err,data){

if(err){

console.log(err);

}else{

res.redirect('/');

}

})

});

**（2）课程搜索**

在input框中输入你的专业名称，点击搜索按钮。此处是用Ajax将所输入专业名称发送到服务器，然后在数据库中进行信息查找返回此专业所对应的课程数据给前台页面。前台页面将接收到的数据信息进行循环遍历然后显示在搜索框的下方。(此处应该有页面截图)

$('.searchbtn').click(function(){

var value=$('input').val();

$('input').val(null);

$.get('/sousuo',{title:value},function(data){

var msg=data.split(" ");

for(var i=0;i<msg.length;i++){

var str='';

str+='<p class="kecheng" onclick="getname()"><a href="#">'+msg[i]+'</a></p>';

$('.searchresult').append(str);

}

})

})

$('.topimg').click(function(){

window.history.back();

})

用户需要选择自己复习的课程，点击课程名称，实现本课程的页面跳转。因为页面中的课程内容是通过Ajax接受到后端数据添加到页面的，因为Ajax异步传输的原因，所以不能直接用元素的点击事件，即$(‘p’).click(callback()),的方法，这样是不能给元素绑定上点击事件的。在Ajax还没有执行完的过程中，它会先去执行点击事件，获取不到标签元素的情况，就绑定不了事件。通过在标签里给元素绑定点击事件。同时需要获取点击元素的内容，应用浏览器的离线缓存功能localStorage.setItem(‘name’,neirong),将其存储在浏览中，以便在跳转到此项课程的页面的时候，能够获取到页面标题，也就是离线缓存中的内容。

Localstorge是HTML5提供的在客户端存储数据的方法之一，该方法存储的数据没有时间限制。

代码如下：

var click=true;

function getname(){

if(click){

$('p').bind('click',function(){

localStorage.setItem('subject',$(this).text());

$('p a').attr('href','/finalExam');

})

click=false;

}else{

click=true;

}

}

在课程页面中获取课程名称：

<div class="topword1"></div>

var subject=localStorage.getItem('subject');

$('.topword1').text(subject);

点击平常练习或者复习资料，将离线存储的课程名称传递到后台服务器，在数据库中搜索此课程对应的学习资料和章节，同理，点击章节的时候，会从后台数据库返回此章节所对应的内容。每次都会将“平常练习”和“复习资料”页面关闭时的页面内容用离线缓存功能存储在浏览器中。在打开“继续上次”的页面的时候，都会将其缓存的内容显示在页面中。在这里，离线存储就是一个可以存临时数据的地方。

$('#chapter a').click(function(){

localStorage.setItem('chapter',$(this).text());

var text=$(this).text();

$.post('/content',{content:text},function(msg){

console.log(msg);

})

})

router.get('/study/lianxi',function(req,res){

weiji.find(function(err,data){

if(err){

console.log(err);

}else{

// console.log(data);

res.render('study/lianxi',{data:data});

}

})

})

**提问和回答**

**提问**

在我要提问的页面的文本域中写出自己的问题，写完自己的问题，点击发送，会获取文本域中的内容。通过时间对象，来获取提问的时间点。然后将提问的内容和时间数据信息发送到服务器，存储在数据库中。文本域内容清空，并返回到展示问题的页面。此页面的数据是后台数据库的数据信息，用ejs模板引擎进行页面信息替换。

$('.tiright').click(function(){

//获取文本域内容

var info=$('textarea').val();

//获取时间对象

var time=new Date();

var year=time.getFullYear();

var month=time.getMonth();

var day=time.getDate();

var hour=time.getHours();

var minute=time.getMinutes();

var second=time.getSeconds();

if(info==''){

return false;

}else{

//将文本域的内容和时间数据信息发送到后台服务器 $.post('/tiwen',{wenti:info,year:year,month:month,day:day,hour:hour,minute:mi nute,second:second},function(msg){

console.log(msg);

})

window.history.back();

}

})

点击“最新提问”，让问题页面显示，让回答问题的页面隐藏；点击“最新回答”，让回答问题的页面显示，让问题页面隐藏。

$('#tiwen').click(function(){

$('#tiwens').css('display','block');

$('#huidas').css('display','none');

})

$('#huida').click(function(){

$('#tiwens').css('display','none');

$('#huidas').css('display','block');

})

通过ejs模板引擎的数据信息替换将数据库中的数据信息在前台页面显示。

<div class="middle22section" id="tiwens" style="display:none" >

<% data.forEach(function(value,key){ %>//数据信息的遍历

<div class="middle22">

<div>

<p><a href="/study/huida" style="color:#000;"><%= value.wenti %></a></p>

</div>

<div class="middle223">

<span><%= value.year %>-<%= value.month %>-<%= value.day %></span><span><%= value.hour %>:<%= value.minute %>:<%= value.second %></span><span>sweet</span>

</div>

</div>

<% }) %>

</div>

**回答**

点击你所回答的问题，会进入回答问题的页面。页面上次显示你所点击的问题内容。在页面下方输入问题的答案，点击“回复“，同时获取了页面中的问题和对应的回答的内容。将问题和答案发送到服务器，在数据库中查找此问题所在的那条数据，将问题的答案更新到数据库中。

返回到日常问答的页面并刷新页面，点击“最新回答“，就会出现所回答的问题及其对应的答案。

var word2\_bottom=document.querySelector('.word2\_bottom');

$('.huifu').click(function(){

var info1=$('textarea').val();//获取回答

var info2=$('.word1\_bottom').text();//获取问题

word2\_bottom.innerHTML=info1;

$('textarea').val('');

//将问题和答案发送到后台

$.get('/addhuida',{wenti:info2,huida:info1},function(msg){

console.log(msg);

})

})

router.get('/addhuida',function(req,res){//更新数据库数据信息

qanda.update({wenti:req.query.wenti},{$set:{"huida":req.query.huida}},function(err,data){

if(err){

console.log(err);

}else{

res.json(data);

}

})

});

用ejs模板引擎将服务器端的数据传输到前台页面，遍历数据并在页面显示。

<div class="middle22section" id="huidas" style="display:none" >

<% data.forEach(function(value,key){ %>

<% if(value.huida==null){ %>

<% return false; %>

<% }else{ %>

<div class="middle22">

<div>

<p><a href="/study/huida" style="color:#000;"><%= value.wenti %></a></p>

</div>

<div>

<p><a href="/study/huida" style="color:pink;"><%= value.huida %></a></p>

</div>

<div class="middle223">

<% } %>

<span><%= value.year %>-<%= value.month %>-<%= value.day %></span><span><%= value.hour %>:<%= value.minute %>:<%= value.second %></span><span>sweet</span>

</div>

</div>

<% }) %>

</div>

发布兼职消息（需要页面图来说明）

在发布兼职页面填写各项兼职消息数据后，点击发布，获取页面的数据信息，并用Ajax将数据发送到服务器，然后添加到数据库中。返回到显示所有兼职消息的页面，同时进行页面刷新。同样用ejs模板引擎传替换数据信息的方法将数据库中的兼职信息显示在页面中。

$('.right').click(function(){

//获取页面的兼职消息的数据

var jobname=$('#jobname input').val();

var zhiwei=$('#zhiwei input').val();

var place=$('#place input').val();

var date=$('#date input').val();

var pay=$('#pay input').val();

var phone=$('#phone input').val();

//将数据发送到服务器

$.post('/addjobinfo',{jobname:jobname,zhiwei:zhiwei,place:place,date:date,pay:pay,phone:phone},function(msg){

console.log(msg);

})

//清除页面兼职消息的数据

var jobname=$('#jobname input').val('');

var zhiwei=$('#zhiwei input').val('');

var place=$('#place input').val('');

var date=$('#date input').val('');

var pay=$('#pay input').val('');

var phone=$('#phone input').val('');

})

//接收前台页面的数据信息，并添加到数据库中。

router.post('/addjobinfo',function(req,res){

jobinfo.create(req.body,function(err,data){

if(err){

console.log(err);

}else{

//添加成功返回消息“did it“

res.json('did it');

}

})

});

页面跳转：

实现页面跳转主要应用的是express模板的路由。少部分应用的是window.location.href的方法。