



廣東工業大學

本科毕业设计（论文）

基于 WEB 的虚拟生物实验室

学 院 物理与光电工程学院

专 业 电子科学与技术

年级班别 2013 级(3) 班

学 号 3113008339

学生姓名 郭桐汕

指导教师 陈国鼎

2017 年 6 月

基于 W E B 的虚拟生物实验室

郭桐汕

物理与光电工程学院

摘 要

WEB 虚拟生物实验室是基于网页浏览器的模拟大概生物实验的一种移动 WEB 应用程序。随着移动互联网的兴起，手机、平板电脑等移动电子设备的功能越来越先进，硬件功能越来越强大，甚至超越普通的台式电脑的配置，而且普及程度远远大于台式电脑。正因此，将虚拟生物实验室迁移到移动端是一个非常不错的选择，而随着 HTML5^[7]、CSS3、ECMAScript6 等网页标准的实施，移动 WEB 的功能越来越强大。

WEB 虚拟生物实验室利用当前的 H5、C3、ES6 标准构建页面结构和样式，利用 Vue 框架实现单页面应用程序，利用 Echarts 生成图表，利用 webpack 框架作为开发依赖，其它的依赖模块在 package 文件中有详细说明。

相对于原生的 Android 或者 IOS 应用，移动 WEB 更具有优势：支持设备广泛、跨平台、较低的开发成本、自动更新，简而言之就是开发一套应用之后，部署到服务器上，可以以浏览网页的形式在 PC，移动等浏览器直接访问。当然，对于 HTML 应用对于大规模数据支持不够友好的缺点，在当前 WEB 虚拟生物实验室里面并不突出，因为我们本身的数据就相对较少。

虚拟生物实验室能够在联网状态下，获取到已经部署在服务器上面的虚拟实验数据和页面，通过操作得到实验的大概的生物现象，图形曲线，而不是昂贵的实验设备得到简单的实验结果。移动 WEB 页面添加新的功能或者修改更具有优势，相对其他的程序更为简洁。

关键词：移动 WEB，Vue，Echarts，虚拟实验室，生物实验室

注：本设计（论文）来源于教师的国家级（或部级、省级、厅级、校级、企业）科研项目，项目编号为：_____

Abstract

WEB virtual biology laboratory is a kind of mobile WEB application based on Web browser simulation. With the rise of mobile Internet, mobile electronic devices, mobile phone function more and more advanced tablet computer hardware, more powerful, even beyond the ordinary desktop computer configuration, and popularity is far greater than the desktop computer. Therefore, it is a very good choice to migrate virtual laboratory to the mobile terminal, and with the implementation of HTML5, CSS3, ECMAScript6 and other web standards, the function of mobile WEB is becoming more and more powerful.

WEB virtual biological laboratory building the page structure and style to use the H5, C3, ES6 standard, single page application using the Vue framework, using Echarts to generate charts, using the webpack framework as a development dependent, rely on other modules are specified in the package file.

Compared with the original Android or IOS application, mobile WEB has more advantages: support equipment widely, cross platform, low development cost, automatic updates, is to develop a set of application in short, deployed to the server, which can browse the web in the form of PC, direct access to mobile browser. Of course, for HTML applications for large-scale data support is not friendly enough shortcomings in the current WEB virtual laboratory is not prominent, because our own data is relatively small.

Virtual Biological Laboratory in the network, access to the server has been deployed in the above virtual experiment data and pages are probably biological phenomena, the experiment through the operation curve, instead of expensive experimental equipment to obtain experimental results of simple. Adding new features or modifications to the mobile WEB page also has an advantage over other programs.

Key words: Mobile WEB, Vue, Echarts, Virtual, Biological Laboratory

目 录

1 绪 论.....	7
1.1 生物教学发展现状.....	7
1.2 虚拟生物实验室.....	7
1.3 基于 WEB 的虚拟生物实验室.....	8
1.4 基于 WEB 的虚拟实验室的未来.....	9
2 题目研究方法和工具.....	10
2.1 研究方法.....	10
2.2 使用工具.....	10
2.2.1 Webstorm 编辑器.....	10
2.2.2 Sublime Text 3.....	10
2.2.3 HTML5.....	11
2.2.4 CSS3.....	12
2.2.5 JavaScript.....	12
2.2.6 Vue.js.....	13
2.2.7 Echarts.js.....	15
2.2.8 jQuery.....	15
2.2.9 Webpack.....	16
2.2.10 Mint-ui.....	16
3 应用系统分析.....	17
3.1 系统分析.....	17
3.2 虚拟实验室设计目标.....	17
3.3 虚拟实验室可行性分析.....	17
3.4 虚拟实验室设计特点.....	18
4 系统主要架构及开发模式.....	19
4.1 基于 B/S 的体系结构.....	19
4.2 基于 MVC 结构.....	19
5 系统设计与流程分析.....	20

5.1 虚拟实验室模块设计分析.....	20
5.2 虚拟实验室代码结构.....	20
5.3 虚拟实验室主要页面展示.....	21
6 生物学原理.....	24
6.1 刺激强度和频率对肌肉收缩的影响.....	24
6.2 实验目的.....	24
6.3 实验原理.....	24
6.4 实验结果.....	25
6.4.1 从小递增模式.....	25
6.4.2 电压刺激模式.....	25
6.4.3 频率刺激模式.....	27
结 论.....	30
参考文献.....	31
致 谢.....	32
附录 A 实验室 HTML 入口文件.....	33
附录 B Vue 主要入口文件.....	34
附录 C 前端访问路由配置文件.....	37
附录 D 虚拟实验室主要配置文件.....	39
附录 E 虚拟实验室肌肉刺激实验结构文件.....	40

1 绪 论

1.1 生物教学发展现状

在生物教学过程中，特别是初高中的生物教学，生物实验教学在整个过程中占据了相当大的部分。初级的生物学实验不像化学实验、物理实验，虽然对于场地和实验材料，实验设备都有所要求，但是很多生物学都需要占用更多的时间。生物学实验中的对比实验，时间跨度从几小时到几天，甚至几个月，多的需要几年，这样的时间对于学校教学来说是不允许的。所以在平时的生物教学时，主要是以教材为主，还有教材配置的图片展示，在这种方式上，在学生的接受程度上并不高。对于实验现象比较明显的，区别比价大的，还能够从图文描述中获取到重点，而图文描述差异比较小的时候，对于学生来说就相对难以理解了。而一些生物对比实验在现象上区别不够明显，虽然在文字描述的时候能够叙说比较清楚，但是在接触之后可能并不能够分辨出来。

部分具有危险性的实验，比如需要液氨、液氮等低温环境保存；高温、高压灼烧，一些学校根本没有能力为学校提供安全实验条件，这也是生物教学实验在中学教学过程中处于弱势的原因之一。

随着对于教育的越发重视，特别是九年义务教育普及之后，大部分学校在扩招，师资力量有较大的缺口，对于生物教学，基本上只能是纸上谈兵。

1.2 虚拟生物实验室

虚拟实验室是针对于生物实验过程中局限于场地，设备，材料，时间，安全等限制因素做出的利用计算机模拟整个实验过程，通过实验公式或者预留的实验数据及不同的条件进行模拟拟合从而产生和真实实验过程一致的实验结果，甚至能够避免非实验因素的影响，得到更加准确的实验结果。虚拟实验室得益于计算机设备的快速发展，自身取得了长足的发展。从实验场所、实验材料、设备等方面，虚拟实验室从创立到至今都取得了长足的进步。通过虚拟实验室，减少了教学开支，提高了学生的实验理论水平，在教学过程中能够发挥非常大的作用。

1.3 基于 WEB 的虚拟生物实验室

从移动电子设备，特别是手机从功能机发展到智能机之后，随着互联网宽带的不断提速降费，移动互联网发展如火如荼，日新月异。而调查“一带一路”沿线 20 国青年评选出的中国“新四大发明”^[1]：高铁、支付宝、共享单车、网购，涉及到移动互联网的就有三个。

移动电子设备的飞速发展，WiFi 普及，4G 网络的全面覆盖让移动互联网逐渐的深入到人们的生活中，移动端流量占据数据流量中的比例越来越大，更多的 PC 项目重构整理成为移动端项目。移动互联网与人们的生活结合的越来越密切，而在教学方面依然是相对迟滞的状态。

虚拟生物实验室正是利用移动设备的便携性，移动 WEB^[2]浏览器的通用性、跨平台功能，打造一款移动的，基于 WEB 应用。虚拟实验室是尝试将移动互联网和生活中普遍存在的手机、平板电脑等移动电子设备和生物实验教学结合起来。利用移动电子设备的普遍性，虚拟实验室在能够取得重大发展的条件下，能够将实验室搬进课堂，搬进家里，甚至随身携带在身边。而传统的 PC 虚拟生物实验室和真实生物实验室局限于场地，设施，实验材料，安全措施等方面，在便捷性，安全性，使用性等方面远远比不上移动虚拟生物实验室。

在前期，虚拟实验室在使用过程中主要是基于网页浏览器的使用，技术也是基于移动 WEB 技术，在这个方面，移动 WEB 在技术和应用上都是相对简单。在长远的发展规划来看，应该结合真正的虚拟现实技术，将场地、设备、材料等转移到我们身边，真正的进行试验操作。当然，本文主要讲的是前期实现。

虚拟实验室采用的主要技术有网页编程语言 HTML^[3]，CSS^[4]，JavaScript^[5]，网页编程框架 Vue.js^[6]，百度团队开发的商业级数据图表 Echarts.js^[7]，基于服务器端 Node.js^[8]的 webpack，网页操作元素 JavaScript 库 jQuery^[9]，饿了么前端团队推出的基于 vue.js 的移动端组件库 mint-ui^[10]。并利用组件懒加载技术在没有访问的页面中不加载数据，提高访问效率和减少服务器压力。

虚拟生物实验室能够减少实验仪器，实验材料等方面的支出，能够实现简单的实验效果并应用于教学。不同于 Flash 工具的虚拟实验室，WEB 实验室使用 HTML5 自带的 canvas 标签能够实现更好的动画效果，是一种在网页上取代 Flash 的新技术。

肌肉刺激^[11]实验做为虚拟生物实验室的第一个实验，也是这个平台的第一个实验，实验主要是针对真实实验结果进行拟合，模拟真实实验条件，得出实验结果。将实验结果以图表的形式进行呈现。虽然减少了实验操作这一部分，但是对于实验理论能有较明显的提高。

1.4 基于 WEB 的虚拟实验室的未来

虚拟实验室进入到移动电子设备之后，下一里程碑将会是和虚拟现实技术结合，打造真正的虚拟实验室。

利用虚拟现实 VR 技术，能够模拟出一系列的实验场所、材料、设备，更重要的是能够提供学生动手操作实验的机会。不同于当前的虚拟实验室采用的按钮选择，仪表仪器选择，点击按钮就执行实验过程，产生实验结果，在整个实验过程中，学生只是一个旁观者，采取建议引导实验方向。结合虚拟现实技术的虚拟实验室，学生在其中更是作为一个实验参与者，需要动手的地方都是有学生亲手操作。实验过程中出现操作错误也将会影响实验结果，甚至产生安全实验事故。当然，这种安全实验事故也是虚拟的，重要的是给学生一个警示的作用。

从一个实验的旁观者到实验的参与者，学生的理论水平和动手能力都能够得到质的提高。

当前虚拟现实技术在当前科技发展中有非常重要的地位，众多科技厂商投入巨额资金研发虚拟现实技术及其配套设施，在不久的将来就能够实现。

2 题目研究方法和工具

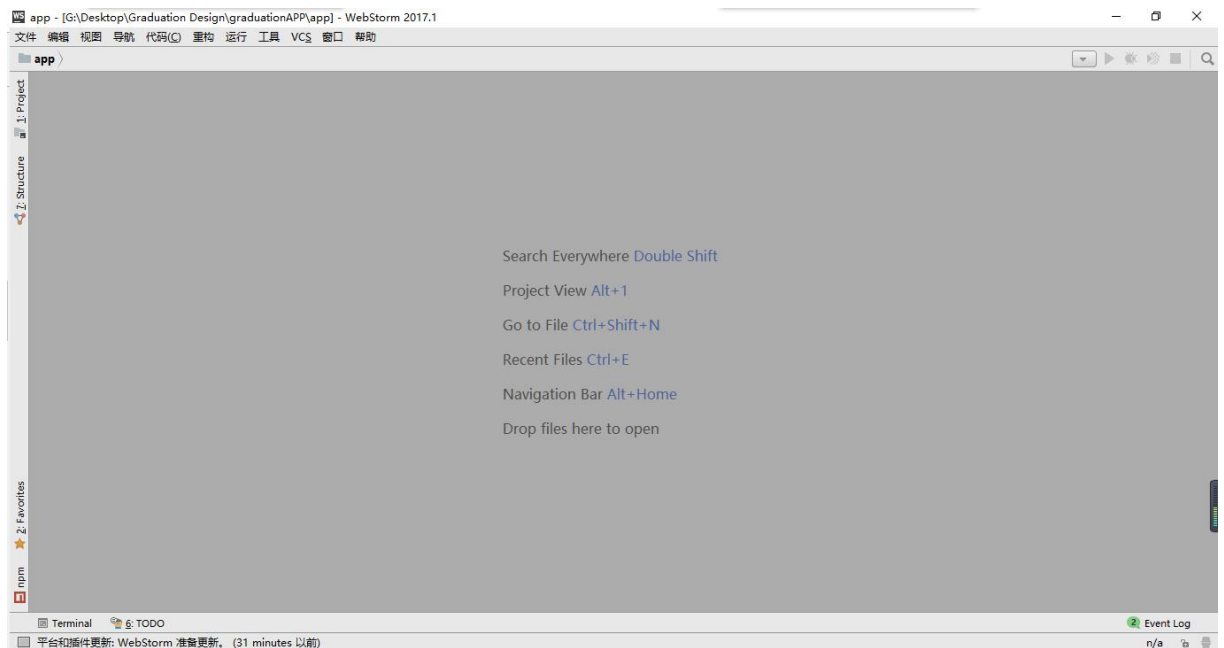
2.1 研究方法

利用计算机软件将具体实验产生的数据生成波形，将数据保存在服务器中，电子设备请求服务器请求数据，通过 echarts^[3]数据展示框架展示成具体的波形。

2.2 使用工具

2.2.1 Webstorm 编辑器

一款对于 WEB 页面开发友好的 IDE，对于页面的 HTML，CSS，JavaScript 支持相对比其它 IDE 有优势。



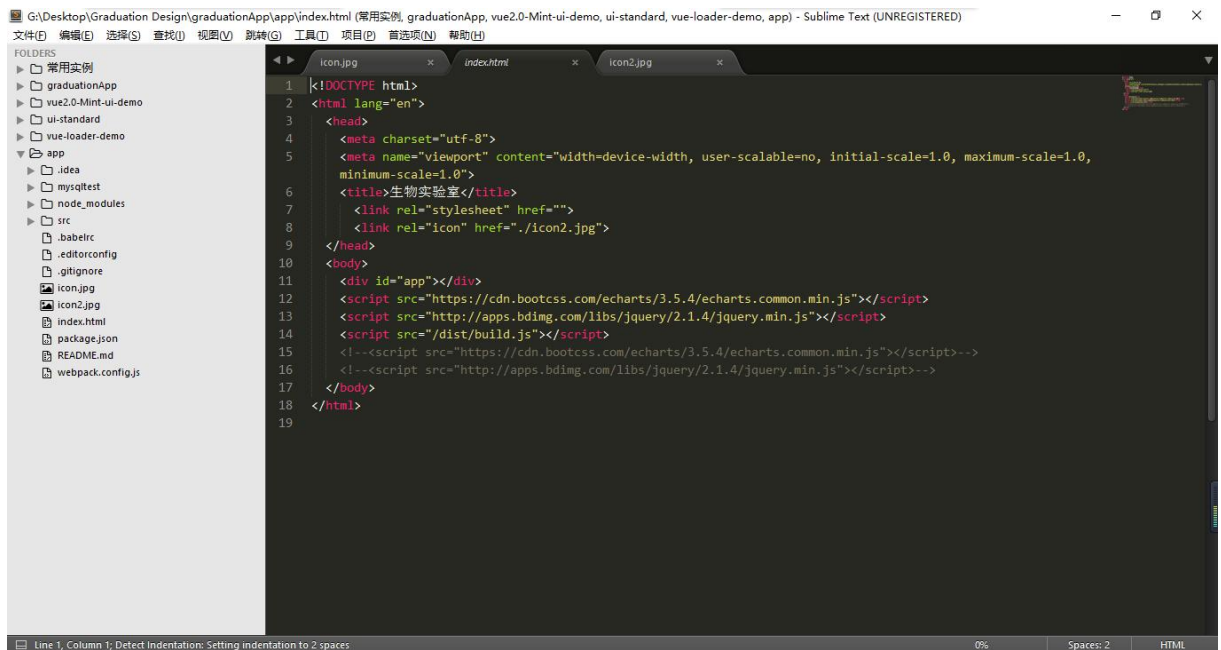
注:webstorm 的插件功能非常强大，能够提高开发效率。

图 2.1 webstorm 2017.1 编辑器主界面

2.2.2 Sublime Text 3

免费的代码编辑器，使用插件的方式添加对不同语言和文件的支持，打开速度非常

快，占用的资源少，可自定义皮肤、配色方案等。



注：sublime 占用的资源较少，对于硬件的要求比 webstorm 低。

图 2.2 sublime text 3 编辑器主界面

2.2.3 HTML5

规定了页面的结构。2014 年 10 月底发布的第五次重大修改的万维网核心语言，标准通用标记语言下的一个应用超文本标记语言，对于移动端更加支持，当前 PC 端高级浏览器及所有的手机端浏览器都支持这一语言。



注：HTML5 对于移动端的支持更加友好。

图 2.3 HTML5 主要标签及部分语法

2.2.4 CSS3

规定了页面的样式。

使用 CSS 语言给页面的结构添加不同的样式，使页面的呈现方式更为丰富多彩。

CSS3 使用模块化技术，不同的模块对应不同的样式，更加的灵活。

```
/*根字体大小,以1rem=100px做为标准*/
@media screen and (min-width: 320px) {html{font-size:100px;}}
@media screen and (min-width: 360px) {html{font-size:101.25px;}}
@media screen and (min-width: 375px) {html{font-size:105.46065px;}}
@media screen and (min-width: 400px) {html{font-size:112.5px;}}
@media screen and (min-width: 414px) {html{font-size:116.4375px;}}
@media screen and (min-width: 440px) {html{font-size:123.75px;}}
@media screen and (min-width: 480px) {html{font-size:135px;}}
@media screen and (min-width: 520px) {html{font-size:146.25px;}}
@media screen and (min-width: 560px) {html{font-size:157.5px;}}
@media screen and (min-width: 600px) {html{font-size:168.75px;}}
@media screen and (min-width: 640px) {html{font-size:180px;}}
@media screen and (min-width: 680px) {html{font-size:192.25px;}}
@media screen and (min-width: 720px) {html{font-size:202.5px;}}
@media screen and (min-width: 760px) {html{font-size:213.75px;}}
@media screen and (min-width: 800px) {html{font-size:225px;}}
@media screen and (min-width: 960px) {html{font-size:270px;}}
/*css样式初始化*/
body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,code,form,fieldset,legend,input,button,textarea,p,blockquote,th,td { margin:0; padding:0; }
body { background:#fff; color:#555; font-size:0.14rem;line-height:1.6; font-family: Verdana, Arial, Helvetica, sans-serif;width:100%;overflow-x:hidden}
td,th,caption { font-size:0.14rem; }
h1, h2, h3, h4, h5, h6 { font-weight:normal; font-size:100%; }
address, caption, cite, code, dfn, em, strong, th, var { font-style:normal; font-weight:normal;}
a { color:#555; text-decoration:none; -webkit-tap-highlight-color: rgba(0, 0, 0, 0);}
a:hover { text-decoration:none;}
a:active{text-decoration: none;}
a,img { border:none; }
ol,ul,li { list-style:none; }
input, textarea, select, button { font:0.14rem Verdana,Helvetica,Arial,sans-serif;outline: none }
```

注：CSS3 的发布，减少了前台页面动画对 javascript 的依赖。

图 2.4 CSS 层叠样式表

2.2.5 JavaScript

规定了页面的行为。

JavaScript^[5]的出现使得网页从单纯的信息展示向页面交互、信息收集等方向发展，使得页面对于用户更加的友好。不同的 JavaScript 代码定义不同的行为，让页面动起来。

作为一门轻量级的语言，简单、易学、扩展性强，JavaScript 的使用范围越来越广泛，Node.js 的出现更是让其从前台走向后台，功能越来越强大。

```

// var echarts = require('echarts');
// var $ =require('jquery');
var $ =jQuery;
var Xarray=createX(.02,2.0);
var voltage={
    name:"voltage",
    title:"刺激电压与肌肉收缩特征",
    xData:Xarray,
    tip:['收缩力'],
    color:"rgb(219,50,51)",
    chart_end:100,
    yData:createY(Xarray,[ '1.2', '1.3', 15, '1.3', '1.2'],1.38)
},
frequency={
    name:"frequency",
    title:"刺激频率与肌肉收缩特征",
    xData:[1,2,3,4,5,6,7,8,9,10,11,12],
    tip:['收缩力'],
    color:"rgb(0,136,212)",
    chart_end:100,
    yData:[1, 1.5, 8, 10, 11, 11.5, 11.7, 11.8, 11.2, 6, 2, 1]
},
single_xtemp=[.2,.4,.6,.8,1],
single_ytemp=['.05', '.1', 15, '.1', '.05'];

/**
 * 根据频率的值产生相应的数据,每次只产生一组数据,每组数据为100,获取多次的时候需要拼接数组
 * y轴数据需要根据频率手动产生...
 * @param value
 */

```

注：javascript 是弱类型的语言，对于初学者来说相对友好。

图 2.5 javascript 部分语法使用

2.2.6 Vue.js

一个利用高级浏览器的特性，实现页面数据绑定，易用、灵活、性能强大的前端框架，在当前的 WEB 框架中占有很大的比例。

Vuejs^[2]的核心功能是双向数据绑定及组件化。

双向数据绑定，利用高级浏览器的一个新的特性将数据和页面之间进行绑定，数据变化，页面随之变化；页面变化，数据也随着变化。这一核心功能能够实现页面的无刷新更新，同时省略了使用 AJAX 技术之后将获取的数据手动渲染到页面这一过程。只要使用 AJAX 或者 websocket 技术将服务端的数据更新到浏览器的 javascript 解析器中即能够实现页面的刷新。

组件化，组件化开发是提高编程人员工作效率的一种极为有效的手段。将特定功能

封装成组件，在使用的过程中直接引入组件，就能够形成一个成品。组件就像砌房子的砖，程序员通过使用不同型号的砖砌成一个完整的建筑——web 应用程序。

Vuejs 能够利用 HTML 编写模板，同时将 CSS 和 javascript 集中在一个文件中，通过 Node 服务器和 webpack 等打包工具动态的将模板解析拼接成为浏览器能够正常解析的文件。

```
<template>
  <div id="app">
    <div class="indexRouterTop"></div>
    <router-view :panelClick="panelClick">

    </router-view>

    <div class="indexRouterBottom"></div>
    <mt-tabbar v-model="selected" fixed>
      <!-- 底部导航 -->
      <router-link v-for="item in tabbarItem" :to="item.href">
        <mt-tab-item id="item.tabbarMsg">
          
            {{item.tabbarMsg}}
          </mt-tab-item>
        </router-link>
      </mt-tabbar>
    </div>
  </template>

<script>

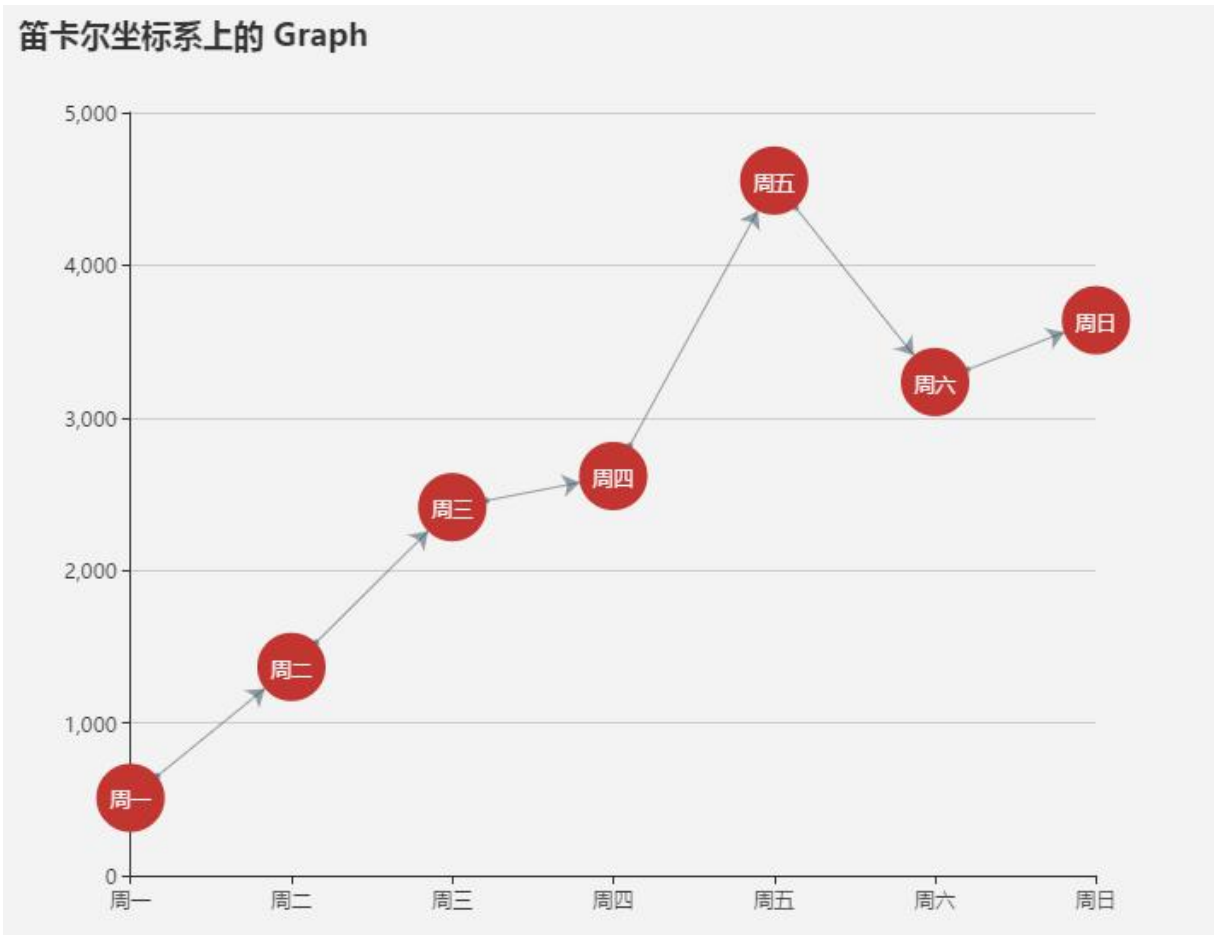
export default {
  name: 'app',
  data () {
    return {
      panelClick:{count:0,title:"虚拟生物实验室"},
      selected:"0",
      tabbarItem:[
        {
          href:"/index",
```

注：vue 的核心功能是组件和双向数据绑定，组件可以创造使用 HTML 本身没有的标签。

图 2.6 vuejs 语法

2.2.7 Echarts.js

Echarts^[7]是百度技术团队开发的一款商业级数据展示框架，能够将数据在网页上以不同的类型展示出来。



注：echarts 数据展示模式多样，K 线图、饼图、柱状图等都是常用功能。

图 2.7 百度 echarts 官网实例

2.2.8 jQuery

jQuery 是一个封装的大部分常用方法的开源 JavaScript 库，能够让开发人员在开发过程中节约大量的时间和减少代码编写量。

正如 jQuery 的口号：The Write Less, Do More.

2.2.9 Webpack

结合 Node.js，让前端模块化开发的一个前端开发工具。能够利用不同的插件启用不同的功能。

2.2.10 Mint-ui

Mint UI^[10]由饿了么前端技术团队针对 Vue.js 开发的一个移动组件库，可以提高前端代码的开发效率。

打个比方，组件好比砌墙的砖头，在使用的过程中能够直接使用，只要对砖头的状态进行调整，就能够砌出一扇墙。而组件就是在开发过程中能够直接使用，而样式，行为等按照需要进行自定义。



快速开始

注：饿了么前端团队推出多种基于 Vue 的框架，element 针对 PC，Mint 针对移动端。

图 2.9 Mint UI 官网界面

3 应用系统分析

3.1 系统分析

虚拟生物实验室的开发应当遵循易开发，易添加，易支持，易使用的原则进行设计。

易开发，实验室的开发不能过于复杂和臃肿，能够简单快速的开发，因此使用 mint-ui 框架进行开发，而 mint-ui 是基于 vuejs 的。因此是需要使用 vuejs 的。

易添加，利用 vue 的组件化开发，能够极大的提高开发效率，而添加功能也是通过添加组件的方式进行。

易支持虚拟实验室系统通过 nodejs 作为服务器端语言，相对于其它后台语言，nodejs 创建服务器的步骤极其简单，而当前 windows 设备都能够支持 nodejs。

易使用，虚拟实验室基于 web 设计，能够通过访问网页的方式在电子设备的浏览器上面运行。而当前的电脑、手机、平板电脑等电子设备已经普及，在使用上基本没有限制。特别是手机、平板电脑等移动电子设备，普及率相当高。

基于以上的原则设计此系统。

3.2 虚拟实验室设计目标

虚拟生物实验室能够按照实验文档需求，提供可选择的实验方法和可执行的实验方式、虚拟材料、虚拟试剂，按照真实实验结果模拟实验过程，减少实验时间，实验仪器损耗，实验材料浪费，快速得到实验结果，减少实验开支，避免实验事故。

3.3 虚拟实验室可行性分析

生物实验在生物教学过程中一直以来都相对处于弱势或者被忽略的地位，很多地方局限于实验场所，实验仪器，实验材料，更在于生物实验持续时间长的原因而无法进行。因此，长期以来，生物实验处于一种在书面上或教学视频中的一种困境。

而 WEB 虚拟生物实验室按照现实实验过程和实验结果进行模拟仿真，在时间上可调节，无需实验仪器，实验材料，实验场所，只需要一台移动电子设备或者电脑即可使用。在很多方面对于传统的生物实验都更具有优势。虚拟仿真能够实现条件无法满足的

情况下，进行各种实验，得到实验结论，在各个领域都有运用。

3.4 虚拟实验室设计特点

采用移动 WEB 设计思想，能在当前普及的移动电子设备上很好的使用，简单方便。不限制实验场所，实验器材，实验时间，实验条件，能够随时随地的展开实验；得出实验结果方便快捷；摒弃了电脑软件安装的方式，大大提高了虚拟实验室的易用性，减少了传统虚拟实验室依赖电脑，安装软件的方式；通过网页访问，对于设备要求不高，关闭网页即能关闭虚拟实验室，在电子设备内部无残留垃圾文件；部分依赖文件通过百度服务器 CDN 方式引入，对于服务器电脑要求降低，带宽要求降低，提高了访问效率。

4 系统主要架构及开发模式

4.1 基于 B/S 的体系结构

在这种结构中，将虚拟实验室部署到 WEB 服务器上，连接数据库，学生通过浏览器访问服务器，服务器将页面结构文件（*.html）、样式文件（*.css）、行为文件（*.js）和从数据库中提取数据返回到浏览器，浏览器内核解析文件和数据，渲染到浏览器中，呈现在浏览器中。



注:请求之间遵循 TCP/IP、HTTP 等网络协议

图 4.1 B/S 结构示意图

4.2 基于 MVC 结构

随着技术的发展，特别是从 2015 年以来，前端技术发展如火如荼，日新月异，新技术层出不穷。MVC 思想也从后台转化到前端开发中，并产生了非常多的 MVC 框架，衍生出 MVVM 等 MV* 框架，前端开发变得规范有序。比如当前使用的 Vue.js 就是一门非常具有代表性的前端 MVVM 框架。

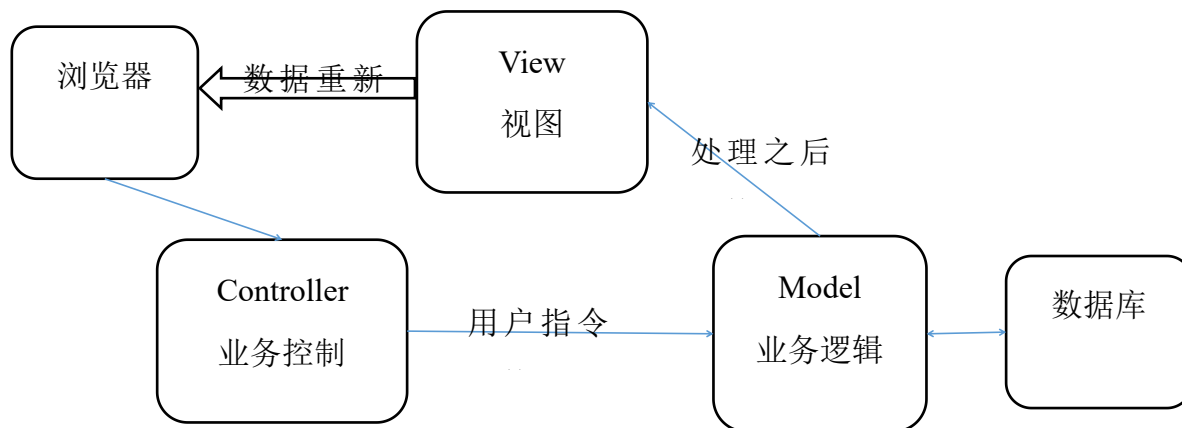


图 4.2 MVVM 框架示意图

5 系统设计与流程分析

5.1 虚拟实验室模块设计分析

虚拟生物实验室具有多种模块，首页的新闻版块可以打通网络信息服务商的咨询内容，进行动态的生成新闻信息；动植物、人类实验版块则是选择具有代表性的实验版块，能够针对需求进行灵活调整。

表 5.1 虚拟实验室页面结构

首页新闻版块	动物实验版块	植物实验版块	人类实验版块
动物新闻	动物实验 1	植物实验 1	人类实验 1
	动物实验 2	植物实验 2	人类实验 2
植物新闻	动物实验 3	植物实验 3	人类实验 3
	动物实验 4	植物实验 4	人类实验 4
人类新闻	动物实验 5	植物实验 5	人类实验 5
	动物实验...	植物实验...	人类实验...

实验内容也能够针对需求添加。每个实验暂时都是相对独立的，包括场景生成，实验模式选择，在远期的设计里面，则会将场景，仪器模块，材料模块单独按需选择，动态生成实验。

5.2 虚拟实验室代码结构

在软件的逻辑代码中，良好的代码结构能够给开发和后期维护带来便利。本文中的代码结构遵循模块化开发的思想，除了不同文件类型分类区分之外，逻辑功能代码按照访问路由、组件功能、业务逻辑等功能模块进行区分。

表 5.2 代码结构

Dist	分块开发后打包目录	build.js	整体打包之后生成的整体文件	
Src 开发模块 总文件夹	components	组件文件夹	Animal.vue	动物面板组件
			AnimalExper.vue	动物实验组件
			AnimalExper0.vue	肌肉刺激实验组
			Human.vue	人类面板组件
			HumanExper.vue	人类实验组件
			Index.vue	首页组件
			IndexNews.vue	首页新闻组件
			NewsPanel.vue	新闻面板组件
			Plant.vue	植物组件
			PlantExper.vue	植物实验组件
	App.vue		主组件	
	css		样式文件夹	
	images		图片文件夹	
	components.js		组件引入文件	
	main.js		开发主文件	
	style.min.css		Mint-ui 样式文件	
	node_modules		项目依赖文件夹	
	router-config.js		路由配置文件	
package.j	项目开发配置、依赖文件			
webpack.	Webpack 配置文件			
icon.jpg	虚拟实验室收藏夹图标			
index.ht	网页入口文件			

本文的代码结果主要由 vue 脚手架工具 vue-cli 自动生成，在使用过程中按照个人需要进行调整。

5.3 虚拟实验室主要页面展示

虚拟生物实验室的主要板块的内容在远期规划过程中能够允许用户自定义展示的内容和展示顺序。采用上传数据库的方式或者利用浏览器的本地存储功能将用户自定义数据进行存储。



(a) 首页新闻板块展示 (b) 动物实验板块展示

注: 板块之间的切换是同通过改变 hash 值, 触发页面 body 标签的 onhashchange 事件进行改变的。

图 5.1 实验室四大板块展示(1)

新闻版块在前期是展示开发人员提供的数据, 在远期规划中可以在后台添加用户自定义新闻或者引入网络信息提供商资讯内容。

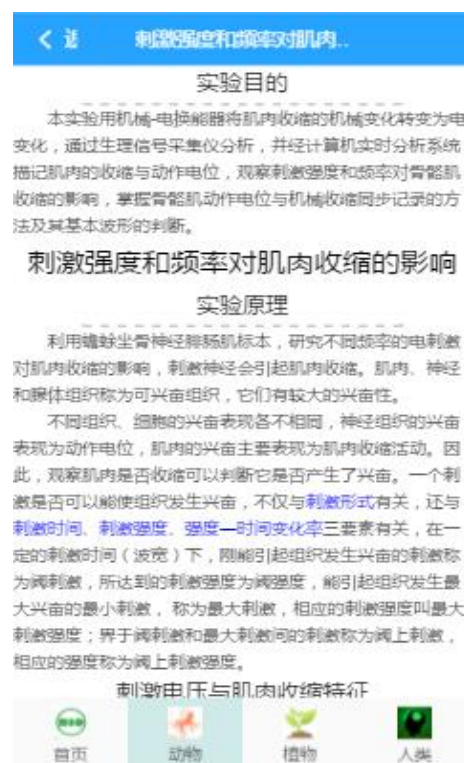
动物实验展示动物相关实验, 在前期, 实验之间是相互独立的。

植物实验和人类实验也是类似的。

在远期规划中的将能够将设备, 材料等组件化, 在用户使用的时候进行自定义的实验内容将能够使用。这样将不会局限于实验模式和实验类型。

主要板块相当于一级菜单, 而板块内部即是具体功能。在新闻展示页面中, 主要由新闻标题, 发布时间, 新闻内容, 暂时没有添加其它较为复杂的引用、分享等功能。而实验中, 首先进入的是实验文档页面, 在进行实验之前, 先了解实验目的、实验原理等实验理论, 再进行实验。

实验文档和实验过程版块通过按钮控制模块的显示隐藏, 达到前进后退的页面效果。



(a) 首页新闻详情 (b) 实验文档 (c) 实验参数选择 (d) 实验结果展示

注: 模块之间的切换通过 vuejs 的 v-if 或者 v-show 状态进行切换。

图 5.2 新闻板块和实验板块展示

6 生物学原理

6.1 刺激强度和频率对肌肉收缩的影响

本次实验通过在实际实验中采集到数据之后，将数据导入到数据文件中，利用百度团队开发的 echarts 商业级网页图表插件进行波形生成，在以后的实验中，遵循实际实验的数据，生成拟合的波形，这样在移动端得到实验结果。

利用这种方式，减少了材料的开支，并利用了常人普遍拥有的手机或者平板电脑等移动设备，能够简洁明了的向学生展示不同频率或者电压对于肌肉的刺激的影响。

6.2 实验目的

本实验用机械-电换能器将肌肉收缩的机械变化转变为电变化，通过生理信号采集仪分析，并经计算机实时分析系统描记肌肉的收缩与动作电位，观察刺激强度和频率对骨骼肌收缩的影响，掌握骨骼肌动作电位与机械收缩同步记录的方法及其基本波形的判断。

6.3 实验原理

利用蟾蜍坐骨神经腓肠肌标本，研究不同频率的电刺激对肌肉收缩的影响，刺激神经会引起肌肉收缩。肌肉、神经和腺体组织称为可兴奋组织，它们有较大的兴奋性。

不同组织、细胞的兴奋表现各不相同，神经组织的兴奋表现为动作电位，肌肉的兴奋主要表现为肌肉收缩活动。因此，观察肌肉是否收缩可以判断它是否产生了兴奋。一个刺激是否可以能使组织发生兴奋，不仅与刺激形式有关，还与刺激时间、刺激强度、强度—时间变化率三要素有关，在一定的刺激时间（波宽）下，刚能引起组织发生兴奋的刺激称为阈刺激，所达到的刺激强度为阈强度，能引起组织发生最大兴奋的最小刺激，称为最大刺激，相应的刺激强度叫最大刺激强度；介于阈刺激和最大刺激间的刺激称为阈上刺激，相应的强度称为阈上刺激强度。

6.4 实验结果

因为是虚拟实验室，省略了实验过程这一步骤，直接通过输入不同的参数，显示不同的实验结果。

肌肉刺激实验主要由电压和频率两个因素影响肌肉的收缩。因此，虚拟实验室也主要由这两个因素作为实验参数进行实验。进行频率实验的时默认电压处于最大阈值刺激，电压实验时默认频率处于最大阈值刺激。

6.4.1 从小递增模式



(a) 电压实验开启从小递增 (b) 频率实验开启从小递增

注：开启从小递增时将忽略强度选择，此时的强度默认从 0 到最大阈值刺激强度。

图 6.1 新闻板块和实验板块展示

6.4.2 电压刺激模式

电压刺激主要分为单刺激和自动强度刺激两种模式。采用单刺激模式时，每次刺激只产生一个波形，波形不累加，此时可以调整不同的刺激强度。因为 echarts 能够根据

数据动态的调整图像，因此，在电压使用单刺激的时候，波形并没有区别，而坐标的值却改变了。

从比较小的强度开始，此时随着电压强度的增加，肌肉达到最大收缩程度，并且继续增加电压，肌肉的收缩程度依然保持，达到最大收缩的最小刺激强度称为最大刺激强度。而能引起肌肉收缩的刺激强度称为阈强度。阈强度和最大刺激强度之间称为阈上刺激强度。



(a) 电压处于 0.10V (b) 电压处于 1.48V

注：收缩力表示在不同强度下的结果，0.6 是单个刺激过程所用时间的比值。

图 6.2 电压单刺激实验

自动强度刺激模式下，每次产生一个波形并累加，此时在图表中能看到多个波形叠加在一起，可以使用横坐标轴上的图标显示区域选择查看的图像区域及范围。



(a) 不改变电压强度 (b) 动态调整电压强度

注：在自动化刺激时候动态调整强度。

图 6.3 电压自动强度刺激实验

6.4.3 频率刺激模式

频率刺激模式分为单刺激和串刺激两种模式。单刺激和电压实验单刺激方式一致，结果有所不同，在频率单刺激时，会根据强度的变化而变化波形。肌肉在短时间内接收连续兴奋，刺激时间间隔短，前后刺激产生交叉，后一个刺激落在前一刺激收缩尚未结束前，肌肉无法得到完全宽息所需要的时间，将一直维持在收缩状态。肌肉的这种收缩状态称之为强直刺激，简称强直。

在给肌肉适当的频率强度刺激，多次单收缩相继累加，呈现出持续的收缩状态。频率低时是锯齿状的，频率增加，收缩频率增加，曲线变得光滑，变成完全强直收缩。但是达到一定频率之后，即使再次增加频率，后刺激落在前刺激引起的不应期中，也是无效的，收缩不再增加。产生这种效果时的收缩叫做最大强直收缩。在图像中，肌肉处于完全强直收缩之后，收缩不会分开，肌肉维持在稳定状态。

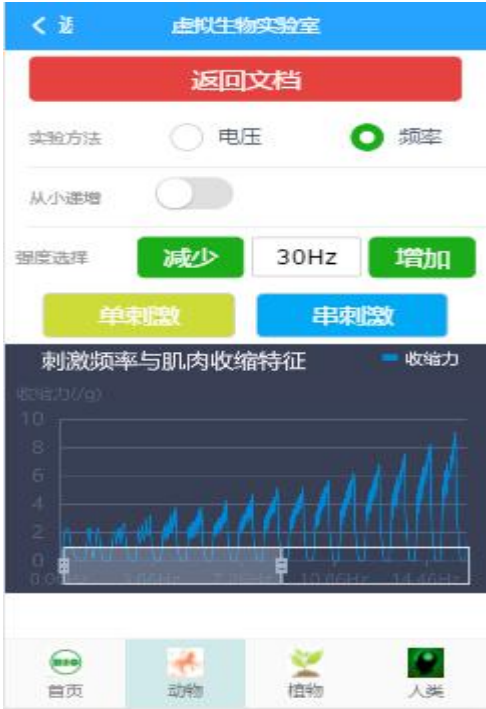
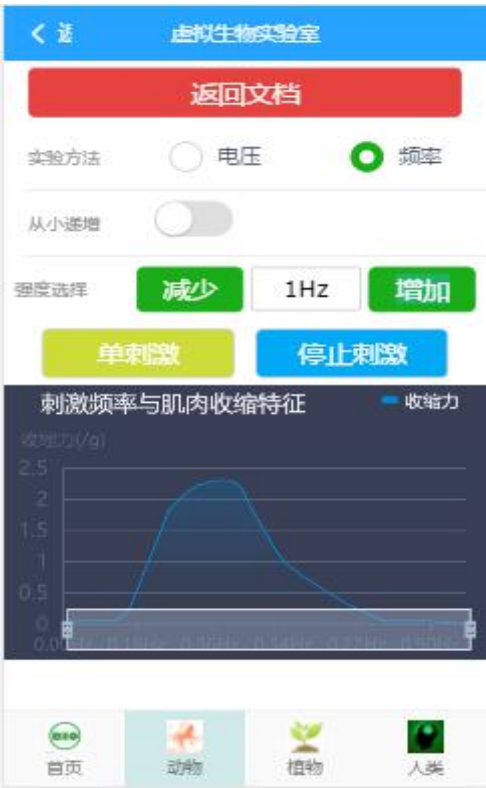


(a) 单刺激—1Hz 波形 (b) 单刺激—2Hz 波形 (c) 单刺激—3Hz 波形 (d) 单刺激—30Hz 波形

注：多频率下，前后刺激兴奋交叉，肌肉处于持续兴奋状态。

图 6.4 频率实验在不同强度单刺激时的收缩状态

频率实验串刺激和电压实验自动强度刺激类似，波形累加，不同之处在于每次累加的波形的强度都会比前一个波形的强度高 1Hz，相当于比较精细的从小递增模式。



(a) 串刺激初始波形 (b) 串刺激前期波形 (c) 串刺激中期波形 (d) 串刺激末期波形
注：串刺激时，与电压刺激的自动连续刺激有所不同，实验强度将会递增。

图 6.5 频率实验在串刺激时的收缩状态

结 论

移动 WEB 虚拟生物实验室能够契合当前移动互联网发展势头，利用快速发展的移动 WEB 技术，结合传统的生物学实验知识和实际实验过程，减少实验材料，设备，场所，时间等方面的开支。能够替代传统的虚拟实验室技术，在生物教学过程中发挥作用，提高实验人员的理论水平。

虚拟实验室采用移动 WEB 技术构建，摒弃了传统软件的安装模式，降低了使用的系统要求，配合当前最新的 Node 服务器技术，在部署上减低了难度，利用百度 CDN 公共资源库，极大的提高虚拟实验室的访问速率。

同时，因为 WEB 开发比传统软件开发上更加简单，更有效率，能够快速的对实验室进行功能模块添加，实验板块添加，大大提高了虚拟实验室的可扩展性。当实验功能不断添加完善，对于生物实验教学更有帮助。

肌肉刺激实验作为运动生理学中较为简单的实验，但是在生物教学过程中依然会收到各种条件的限制，而虚拟生物实验室可以减少这些限制，提高实验参与者的实验理论水平。

在远期能够结合虚拟显示技术再次进行开发，则学生能够参与到实验过程中去，从实验的旁观者到实验的参与者的角色转变。届时，就能够像参与真实的生物实验那样，不但实验理论水平能够提高，实验动手能力都能够相应提高。终极目标则是实验虚拟即现实，现实即虚拟。

参考文献

- [1] 百度百科. 新四大发明[EB/OL] . 2017[2017-5]. <http://baike.baidu.com/item/新四大发明>.
- [2] 唐俊开. HTML5 移动 Web 开发指南[M] . 北京:电子工业出版社, 2012:12-152.
- [3] 菜鸟教程. W3CSchool 教程[EB/OL] . <http://www.runoob.com/>.
- [4] 屈超, 周志. CSS3 实用指南[M] . 北京:人民邮电出版社, 2012: 76-188.
- [5] (美) (Nicholas, C, Zakas) 扎卡斯. JavaScript 高级程序设计[M] . 北京:人民邮电出版社, 2012: 96-255.
- [6] 尤雨溪. Vue.js[EB/OL]. <https://cn.vuejs.org/v2/guide/>.
- [7] 百度 Echarts 团队. Echarts3 教程[EB/OL] . <http://echarts.baidu.com/index.html>.
- [8] nodejs 官网. nodejs 技术文档[EB/OL] . 2017[2017-5]. <https://nodejs.org/en/docs/>.
- [9] 单东林, 张晓菲, 魏然. 锋利的 jQuery[M] . 北京:人民邮电出版社, 2009: 35-65.
- [10] 饿了么前端团队. Mint UI 使用文档[EB/OL] . [2017]. <http://mint-ui.github.io/docs/#/zh-cn2>.
- [11] 王瑞元, 苏全生. 运动生理学[M] . 北京:人民体育出版社, 2012:12-35.
- [12] Schmitt A O, Speech T, Beckmann G. Exhaustivemining of EST libraries for genes differentially ex-pressed in normal and tumour tissues[J]. NucleicAcids Res, 1999:13-25.

致 谢

本设计(论文)是在我的指导老师陈国鼎的亲切关怀和悉心指导下完成的。他严肃的科学态度，严谨的治学精神，精益求精的工作作风，深深地感染和激励着我。从题目的选择到最终完成，陈老师都始终给予我细心的指导和不懈的支持。

感谢学院的领导、老师和同学四年来对我的支持和帮助！

感谢大学四年来对我的栽培！

附录 A 实验室 HTML 入口文件

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, user-scalable=no,
initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

    <title>生物实验室</title>

    <link rel="stylesheet" href="">

    <link rel="icon" href="/icon2.jpg">

  </head>

  <body>

    <div id="app"></div>

    <script src="https://cdn.bootcss.com/echarts/3.5.4/echarts.common.min.js"></script>

    <script src="http://apps.bdimg.com/libs/jquery/2.1.4/jquery.min.js"></script>

    <script src="/dist/build.js"></script>

    <!--<script
src="https://cdn.bootcss.com/echarts/3.5.4/echarts.common.min.js"></script>-->

    <!--<script src="http://apps.bdimg.com/libs/jquery/2.1.4/jquery.min.js"></script>-->

  </body>

</html>
```

附录 B Vue 主要入口文件

```
<template>

  <div id="app">

    <div class="indexRouterTop"></div>

    <router-view :panelClick="panelClick">

    </router-view>

    <div class="indexRouterBottom"></div>

    <mt-tabbar v-model="selected" fixed>

    <!-- 底部导航 -->

      <router-link v-for="item in tabbarItem" :to="item.href">

        <mt-tab-item id="item.tabbarMsg">

            {{item.tabbarMsg}}

          </mt-tab-item>

        </router-link>

      </mt-tabbar>

    </div>

  </template>

<script>

export default {

  name: 'app',

  data () {

    return {

      panelClick: {count:0,title:"虚拟生物实验室"},

      selected:"0",

      tabbarItem:[

        {

          href:"/index",
```

```
        imgSrc:"/src/images/nav_index.jpg",
        tabbarMsg:"首页"},
    {
        href:{path:"/animal",params:{count:0}},
        imgSrc:"/src/images/nav_animal.jpg",
        tabbarMsg:"动物"},
    {
        href:{path:"/plant",params:{count:0}},
        imgSrc:"/src/images/nav_plant.jpg",
        tabbarMsg:"植物"},
    {
        href:{path:"/human",params:{count:0}},
        imgSrc:"/src/images/nav_human.jpg",
        tabbarMsg:"人类"}
    ]
}
},
methods:{
}
}
</script>
<style>
    .mint-header{
        /*background-color: lightblue;*/
    }
    .mint-tabbar{
        display: flex;
    }
    .mint-tabbar a{
```

```
        flex:1;
    }
    .indexRouterTop{margin-top:40px;}
    .indexRouterBottom{margin-top:55px;}
    .fade-enter-active, .fade-leave-active {
        transition: all .5s ease-out;
    }
    .fade-enter, .fade-leave-active {
        opacity: 0;
        transform: translate3d(300px, 0, 0);
    }
    #app {
        font-family: 'Avenir', Helvetica, Arial, sans-serif;
        -webkit-font-smoothing: antialiased;
        -moz-osx-font-smoothing: grayscale;
        text-align: center;
        color: #2c3e50;
    }
</style>
```

附录 C 前端访问路由配置文件

```
import AnimalExper from './components/AnimalExper.vue'
import AnimalExper0 from './components/AnimalExper0.vue'
import PlantExper from './components/PlantExper.vue'
import HumanExper from './components/HumanExper.vue'
import IndexNews1 from './components/IndexNews1.vue'
import IndexNews2 from './components/IndexNews2.vue'
import IndexNews3 from './components/IndexNews3.vue'
import IndexNews4 from './components/IndexNews4.vue'
const Index = resolve => require(['./components/Index.vue'], resolve)
const Human = resolve => require(['./components/Human.vue'], resolve)
const Plant = resolve => require(['./components/Plant.vue'], resolve)
const Animal = resolve => require(['./components/Animal.vue'], resolve)
export default {
  routes:[
    {path:'/animal', component:Animal,
      children:[
        {path:'exper/ciji',component:AnimalExper0},
        {path:'exper/:index',component:AnimalExper}
      ]
    },
    {path:'/plant', component:Plant,
      children:[
        {path:'exper/:index',component:PlantExper}
      ]
    },
    {path:'/human', component:Human,
      children:[ {path:'exper/:index',component:HumanExper}
```

```
    ]
  },
  {path:'/index', component:Index,
    children:[
      {path:'news/1',component:IndexNews1},
      {path:'news/2',component:IndexNews2},
      {path:'news/4',component:IndexNews4},
      {path:'news/3',component:IndexNews3}
    ]
  },
  {path:'*',redirect:'/index'}
]
}
```

附录 D 虚拟实验室主要配置文件

```
import Vue from 'vue'
import VueRouter from 'vue-router'
import VueResource from 'vue-resource'
import Mint from 'mint-ui'
import 'mint-ui/lib/style.css'
import './css/basestyle.css'
import './css/index.css'
import './css/panel.css'
import App from './App.vue'
import routerConfig from './router-config'
import newsPanel from './components/NewsPanel.vue'
Vue.component("my-panel", newsPanel);
Vue.use(VueResource)
Vue.http.options.xhr = { withCredentials: true }
Vue.use(Mint)
Vue.use(VueRouter);
const router=new VueRouter(routerConfig);
var vm=new Vue({
  router,
  el: '#app',
  data: {
    land: ""
  },
  methods: {
  },
  render: h => h(App)
})
```

附录 E 虚拟实验室肌肉刺激实验结构文件

```
<template>
  <div>
    <div class="text-content" v-show="!show">
      <p class="sub_title">实验目的</p>
      <div class="exper">
        </div>
      <button class="primary" id="created" @click="show=!show;">进入实验
    </button>
  </div>
  <div class="chart" v-show="show">
    <button class="warn" @click="show=!show;return_text()">返回文档
  </button>
  <mt-radio @change="experMethod()"
    title="实验方法"
    v-model="stimulatioStyle"
    :options="experStyle">
  </mt-radio>
  <div class="cell cell_switch">
    <div class="cell__bd">从小递增</div>
    <div class="cell__ft">
      <input class="switch" type="checkbox" v-model='addAuto'
        @change='isAddUp($event)'>
    </div>
  </div>
  <div class="cell cell_switch">
    <div class="cell__bd">强度选择</div>
```



```

<div class="cell__ft powerCtrl">
  <!--电压实验+-->
  <button v-if="stimulatioStyle=='1'" id="powerSub" class='primary'
@click.prevent="onValuesChange(-1)" >减少</button>
  <input
v-if="stimulatioStyle=='1'"
type="text" :value='powerChoose.values[powerChoose.select]+"V"' readonly>
  <button v-if="stimulatioStyle=='1'" id=powerAdd class='primary'
@click.prevent="onValuesChange(1)" >增加</button>
  <!--频率实验-->
  <button v-if="stimulatioStyle=='2'" id="powerSub2" class='primary'
@click.prevent="onValuesChange(-1)" >减少</button>
  <input
v-if="stimulatioStyle=='2'"
type="text" :value='powerChoose2.values[powerChoose2.select]+"Hz"' readonly>
  <button v-if="stimulatioStyle=='2'" id=powerAdd2 class='primary'
@click.prevent="onValuesChange(1)" >增加</button>
</div>
</div>
<!--电压实验-->
<button
v-if="stimulatioStyle=='1'"
class="btn_single"
@click="single_ciji(0);">单刺激</button>
<button v-if="stimulatioStyle=='1'" class="btn_series" @click="series_ciji();"
v-text='this.powerChoose.series_text'></button>
<!--频率实验-->
<button v-if="stimulatioStyle=='2'||stimulatioStyle=='3'" class="btn_single"
@click="vue_charts(0)">单刺激</button>
<button v-if="stimulatioStyle=='2'||stimulatioStyle=='3'" class="btn_series"
@click="vue_charts();" v-text='this.powerChoose2.series_text'></button>
<!--<mt-button
v-if="stimulatioStyle=='2'||stimulatioStyle=='3'"

```

```
class="primary" @click="vue_charts();">启动</mt-button>-->
```

```
</div>
```

```
<div id="exper">
```

```
</div>
```

```
</div>
```

```
</template>
```

```
<script src="../../js/AnimalExper0.js"></script>
```

附录 F 肌肉刺激实验逻辑控制文件

```
var $ =jQuery;
```

```
var Xarray=createX(.02,2.0);
```

```
var voltage={
```

```
    name:"voltage",
```

```
    title:"刺激电压与肌肉收缩特征",
```

```
    xData:Xarray,
```

```
    tip:["收缩力"],
```

```
    color:"rgb(219,50,51)",
```

```
    chart_end:100,
```

```
    yData:createY(Xarray,[ '1.2', '1.3', 15,  '1.3', '1.2'],1.38)
```

```
},
```

```
frequency={
```

```
    name:"frequency",
```

```
    title:"刺激频率与肌肉收缩特征",
```

```
    xData:[1,2,3,4,5,6,7,8,9,10,11,12],
```

```
    tip:["收缩力"],
```

```
    color:"rgb(0,136,212)",
```

```
    chart_end:100,
```

```
    yData:[1, 1.5, 8, 10, 11, 11.5, 11.7, 11.8, 11.2, 6, 2, 1]
```

```

    },
    single_xtemp=[.2,.4,.6,.8,1],
    single_ytemp=['.05', '.1', 15,  '1', '.05'];

/**
 * 根据频率的值产生相应的数据,每次只产生一组数据,每组数据为 100,获取多次的时
候需要拼接数组
 * y 轴数据需要根据频率手动产生...
 * @param value
 */
function frequency_data(value) {
    var xData=[],
        i=0,
        j=1;
    for(;i<10;i++){
        j=0;
        for(;j<10;j++){
            xData.push(value-1+"."+i+j+'Hz')
        }
    }
    var yData;
    return {
        xData,
        yData,
    }
}

function newArr(arr) {
    var arr2=arr,newA=[];

```

```

        for(var i=19;i>0;i--){
            newA=arr2.map((j)=>Number((j/1.1).toFixed(2)))
            arr2=newA;
        }
        return newA
    }
}

/**
 * 从小递增模板
 * @param aArr
 * @param bArr
 * @param percentArr
 * @returns {{Num: Array, Str: Array}}
 */
function tempX(aArr,bArr,percentArr) {
    var arr=[],brr=[];
    for(var i=0;i<aArr.length;i++){
        for(var j=0;j<bArr.length;j++){
            arr[arr.length]=typeof(bArr[j])=='number'?bArr[j]*percentArr[i]:bArr[j]
            brr[brr.length]=aArr[i]+j*aArr[i]/bArr.length+'Hz'
        }
    }

    return{
        Num:arr,
        Str:brr
    }
}

```

```

/**
 * 实验强度选择数组,电压从 0.1--1.48V 频率 1-30Hz
 * type 1 表示是频率实验,无表示电压实验
 * @param type
 * @returns {Array}
 */
function powerArr(type){
    var a=[],b=0.1;
    if(type==1){
        for(var i=1;i<=30;i++){
            a.push(i)
        }
    }else{
        for(b;b<=1.5;b+=0.02){
            a[a.length]=b.toFixed(2)
        }
    }
    return a;
}

function transform_ydata(array,k) {
    return array.map(function (item) {
//        return typeof (item)=="number"?item*k:item
        return (item*k).toFixed(2)
    })
}

function transform_xdata(array,count) {
    return array.map(function (item) {

```

```

        return (parseInt(Math.round(item*count*100)))/100
    })
}

```

```

$(function(){
    $("#exper").html('<div id="main"></div>').show()
    voltage.yData=[]
    chart('main',voltage)
})

```

```

/**
 * 根据传入的强度增量,X 轴最值,最大刺激强度生成一个 X 轴坐标数组
 * @param step 强度增量 Number
 * @param target X 轴最值 Number
 * @param max 最大刺激强度 Number
 * @returns {Array}
 */

```

```

function createX(step,target) {
    var arr=[];arr[0]=.1
    for(var i=1;arr[i-1]<target;i++){
        var a;
        if(arr[i-1]<target/3){
            a=arr[i-1]+step/2
        }else if(arr[i-1]<target/2){
            a=arr[i-1]+step
        }else{
            a=arr[i-1]+step*2
        }
        arr[i]=parseFloat(a.toFixed(2))
    }
}

```

```

    }

    return arr;
}

/**
 * 根据传入的 X 轴坐标数组和 Y 轴模板数组,通过不同的放大倍数,将模板数组进行处理,并组成数组返回
 * @param Xarr X 轴坐标数组 Array
 * @param Yarr Y 轴模板数组 Array 对于数字将会进行处理,字符串不处理
 * @returns {Array}
 */
function createY(Xarr,Yarr,max) {
    var newArr=[],aaa="";
    for(var i=0,e=0;i<Xarr.length;i++){
        if(typeof(Yarr[e])=='number'){
            aaa=Xarr[i]>max?max:Xarr[i];
            var a=Yarr[e]*aaa;
            newArr[i]=parseFloat(a.toFixed(2))
        }else {
            newArr[i]=Yarr[e]
        }
        e++
        if(e==Yarr.length){
            e=0
        }
    }
    return newArr
}

```

```

/**
 * 绘制图标波形
 * @param obj_id
 * @param obj
 */
function chart(obj_id,obj){
    }
export default{
    name:"animalExper0",
    data(){
        return{
            myChart:",
            show:false,//实验显示
            addUp:false,//增加
            addAuto:false,//是否从小递增开关控制
            powerInput:"",//强度选则显示
            isSeries:false,//是否自动刺激|串刺激
            stimulatioStyle:"1",//实验方法 1 电压,2 频率
            stimulatioImpu:'1',
            //实验方法
            experStyle:[
                {
                    label: '电压',
                    value: '1',
                },
                {
                    label: '频率',
                    value: '2'
                }
            ]
        }
    }
}

```



```
],
//实验刺激输入
experImpu1:[
    {
        label: '单刺激',
        value: '1'
    },
    {
        label: '自动强度刺激',
        value: '2'
    }
],
experImpu2:[
    {
        label:'串刺激',
        value:'3'
    }
],
winsHeight:window.screen.height,
single_ydata:[],//电压刺激数据
single_xdata:[],//电压刺激数据
single_count:0,//电压刺激计数
//电压实验强度控制
powerChoose: { //电压实验时强度选择
    values: powerArr(),
    select:0,
    series_text:"自动强度刺激",
    intervalNum:""
},
```

```

//频率实验强度控制
powerChoose2: { //频率实验时强度选择
    values: powerArr(1),
    select: 0,
    series_text: "串刺激",
    intervalNum: "",
    xData: [],
    yData: []
},
},
mounted() {
    //开启长按增加
    this.longPress()
},
methods: {
    /**
     * 实验强度长按增加
     */
    longPress() {
        var
timer="",btn=$("#powerAdd,#powerSub,#powerAdd2,#powerSub2"),_this=this,type=-1;
        btn.on('touchstart',function (e) {
            var delay=100;
            if(this.stimulatioStyle==2)delay=200
            type=e.target.id.indexOf("A")>0?1:-1;
            timer=setInterval(function () {
                _this.onValuesChange(type)
            },delay)

```

```

    })
    btn.on("touchend",function () {
        clearInterval(timer)
    })
    btn.bind('contextmenu', function(e) {
        e.preventDefault();
    })
},
/**
 * 实验强度变化引起强度显示变化
 * @param type
 */
onValuesChange(type){
    var obj=this.powerChoose;
    if(this.stimulatioStyle=="2")obj=this.powerChoose2;
    if(type>0){
        if(obj.select>obj.values.length-2){
            obj.select=obj.values.length-1
        }else{
            obj.select+=1
        }
    }else{
        if(obj.select<1){
            obj.select=0
        }else{
            obj.select-=1
        }
    }
},

```

```

/**
 * 电压刺激实验,单刺激
 * @param type
 */
single_ciji(type){
    var obj=this.powerChoose;
    this.addAuto=false;
    if(type==0){
        if(this.isSeries){
            this.isSeries=false
            obj.series_text='自动强度刺激'
            this.single_count=0
            this.single_ydata=[]
            clearInterval(obj.intervalNum)
        }
    }
    $("#exper").html('<div id="main"></div>').show()
    if(this.single_count==0){
        this.single_xdata=[];
    }
    //电压单刺激的时候,每次只有一个波形,并且波形不累加
    if(type==0&&this.stimulatioStyle=="1"){
        this.single_count=1
        this.single_ydata=transform_ydata(single_ytemp,obj.values[obj.select])
        // this.single_ydata=[0,5,10,15,20,25,30]
        this.single_xdata=single_xtemp
    }else{//自动刺激的时候,波形累加
        this.single_count+=1;
    }
}

```

```
this.single_ydata=this.single_ydata.concat(transform_ydata(single_ytemp,obj.values[obj.select]))
```

```
        this.single_xdata=this.single_xdata.concat(single_xtemp)
    }
    voltage.yData=this.single_ydata
    voltage.xData=this.single_xdata
    chart('main',voltage)
},
/**
```

```
 * 电压刺激实验,自动强度刺激
```

```
 */
```

```
series_ciji(){
    voltage.yData=[]
    voltage.xData=[]
    this.isSeries=true;
    var obj=this.powerChoose,_this=this;
    if(obj.series_text=="自动强度刺激"){
        _this.single_ciji(1)
        obj.intervalNum=setInterval(function () {
            if(_this.isSeries==true){
                _this.single_ciji(1)
            }else{
                obj.series_text='自动强度刺激'
                this.single_count==0
                clearInterval(obj.intervalNum)
            }
        },1400)
        obj.series_text='停止刺激'
    }else{
```

```

        obj.series_text='自动强度刺激'
        this.single_count==0
        clearInterval(obj.intervalNum)
    }
},
/**
 * 实验方法选择控制,停止正在进行的自动刺激,串刺激,从小递增
 */
experMethod(){
    this.stimulatioImpu=this.stimulatioStyle=='1'?1:'3';
    this.addAuto=false;
    this.isSeries=false;
    this.powerChoose.series_text='自动强度刺激'
    this.powerChoose2.series_text='串刺激'
    clearInterval(this.powerChoose.intervalNum)
    clearInterval(this.powerChoose2.intervalNum)
},
/**
 * 从小递增方法
 */
isAddUp(){
    var obj=this.powerChoose,
        obj2=this.powerChoose2;
    if(this.addAuto){
        obj.series_text='自动强度刺激'
        obj2.series_text='串刺激'
        this.single_count=0
        this.single_ydata=[]
        clearInterval(obj.intervalNum)
    }
}

```

```

clearInterval(obj2.intervalNum)

if(this.stimulatioStyle=='1'){
    $("#exper").html('<div id="main2"></div>').show()
    this.single_count=0
    this.single_ydata=[]
    voltage.xData=createX(.02,2.0)
    voltage.yData=createY(voltage.xData,[ '1.2', '1.3', 15, '1.3',
'1.2'],1.38)

    chart('main2',voltage)
}else{
    var str,idName;
    str='<div id="main2"></div>';
    idName='main2';
    var
d=tempX([1,2,3,4,6,12,24],[1,'1.2',10,12,13.2,11,7,'1.2','1'],[.1,.2,.3,.4,.8,2.2,2.5])
    frequency.xData=d.Str
    frequency.yData=d.Num
    $("#exper").html(str).show()
//    this.main1=true
    chart(idName,frequency)
}
}else{
}
},
/**
 * 频率刺激实验方法
 * @param type 0 表示单刺激,否则为串刺激
 */
vue_charts(type){

```

```

var str,idName,d;

var obj=this.powerChoose2,_this=this;

this.addAuto=false;

str='<div id="main2"></div>';

idName='main2';

if(type==0){

    this.isSeries=false;

    d=frequency_data(obj.values[obj.select])

    frequency.xData=d.xData

    frequency.yData=d.yData

    frequency.chart_end=100

    clearInterval(obj.intervalNum)

} else{

    obj.xData=[];

    obj.yData=[];

    obj.select=0;

    this.isSeries=true;

    if(obj.series_text=="串刺激"){

        obj.series_text='停止刺激'

        d=frequency_data(obj.values[obj.select])

        obj.xData=obj.xData.concat(d.xData)

        obj.yData=obj.yData.concat(d.yData)

        frequency.xData=obj.xData

        frequency.yData=obj.yData

        // chart(idName,frequency)

        // _this.vue_charts(0)

        obj.intervalNum=setInterval(function () {

            obj.chart_end=101;

            _this.onValuesChange(1)

```



```

        if(obj.select==29||_this.isSeries==false||_this.addAuto==true){
            clearInterval(obj.intervalNum)
            obj.series_text='串刺激'
        }
        d=frequency_data(obj.values[obj.select])
        obj.xData=obj.xData.concat(d.xData)
        obj.yData=obj.yData.concat(d.yData)
        frequency.xData=obj.xData
        frequency.yData=obj.yData
        chart(idName,frequency)
    },1400)
    }else{
        obj.series_text='串刺激'
        this.single_count==0
        clearInterval(obj.intervalNum)
    }
}
$("#exper").html(str).show()
//        this.main1=true
        chart(idName,frequency)
    },
    /**
     * 返回实验文档
     */
    return_text(){
        $("#exper").hide()
    },
}
}
}

```