

同濟大學

通识课之课程设计

课程名称	网页制作教程 (10019601)		
设计名称	学生的信息		
学院(系)	电子信息工程学院		
专 业	通信工程		
学生姓名	斯提凡	学号	1656038

2018-2019 学年 第 2 学期

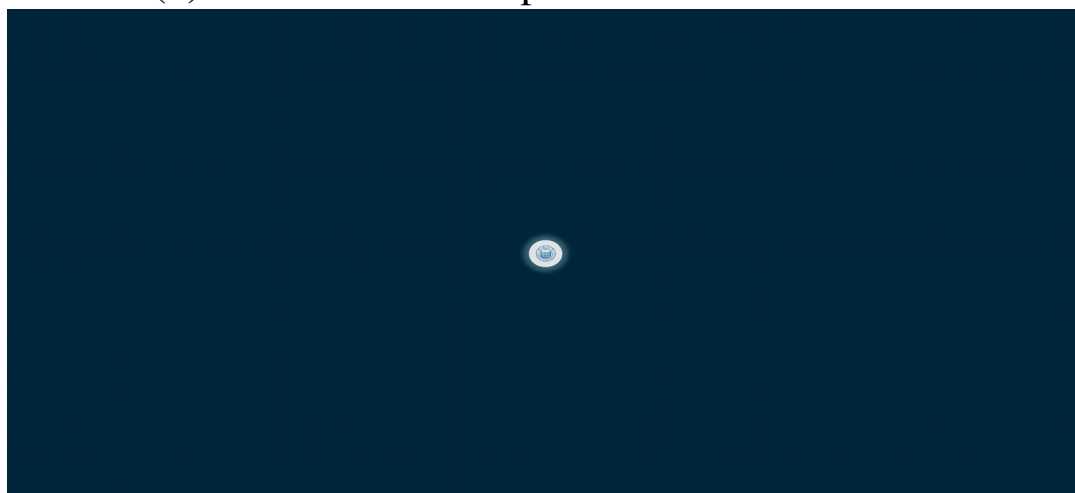
目录

页面效果截屏.....	3
HTML.....	3
JAVASCRIPT.....	3
用 SCSS 就是一个 CSS 的编译器.....	4
VIEWS-FILES.....	4
显示主页面.....	6
软件功能.....	6
我创建了一个静态数组，其中包含所有学生的信息.....	8
设计思想.....	10
整体程序框架.....	10
逻辑结构与物理结构.....	10
开发平台.....	10
系统的运行结果分析说明.....	12
调试遇到的问题一.....	12
VUE 设计.....	13
VUE ROUTER.....	13
项目 ROUTER.....	14
攻击了所有的 VUE 包.....	15
关于 VUE 的更多信息.....	16
生命周期图.....	16
继续：.....	17
结论.....	18
参考文献.....	19

页面效果截屏

1.一 给定一个图，完成：

- (1) 网站设计的第一个图像，显示 40 秒的动画
- (2) 对于第二张图片我们有页面效果截屏
- (3) 对于第三个图像，我们有静态轮播动画
- (4) 在 Vue JavaScript 方法中设置超时



HTML

```
<template>
  <div>
    <div class="img">
      
    </div>
  </div>
</template>
```

JavaScript

```
<script>
export default {
  name: "load",
  data() {
    return {
      user: {
        name: "load"
      }
    };
  },
  mounted() {
    setTimeout(() => {
      this.$router.push("/init");
    }, 3000);
  }
};
</script>
```

用 SCSS 就是一个 CSS 的编译器

```
<style lang="scss" scoped>
.img {
  position: fixed;
  width: 100%;
  height: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  img {
    width: 60px;
    height: 60px;
    border-radius: 50px;
    box-shadow: 0 0 10px 10px rgba(#ccffff, 0.3);
    animation-name: img;
    animation-duration: 3s;
    animation-iteration-count: infinite;
    @keyframes img {
      from {
        opacity: 0.6;
      }
      50% {
        opacity: 1;
      }
      to {
        opacity: 0.6;
      }
    }
  }
}
}
</style>
```

我正在用 Laravel 和 Vue 制作一个购物车系统。当我将一个项目添加到购物篮时，我通过切换由 setTimeout 监视的 Vue 变量来显示确认消息：这很好，首页出现。但是，我希望这段消息在一段时间后再次消失，比如几秒钟。

VIEWS-files

```
<style lang="scss" scoped>
.unix {
  background-color: #00263c;
  min-height: 100vh;
}
</style>
<script>
import load from "../components/load";
```

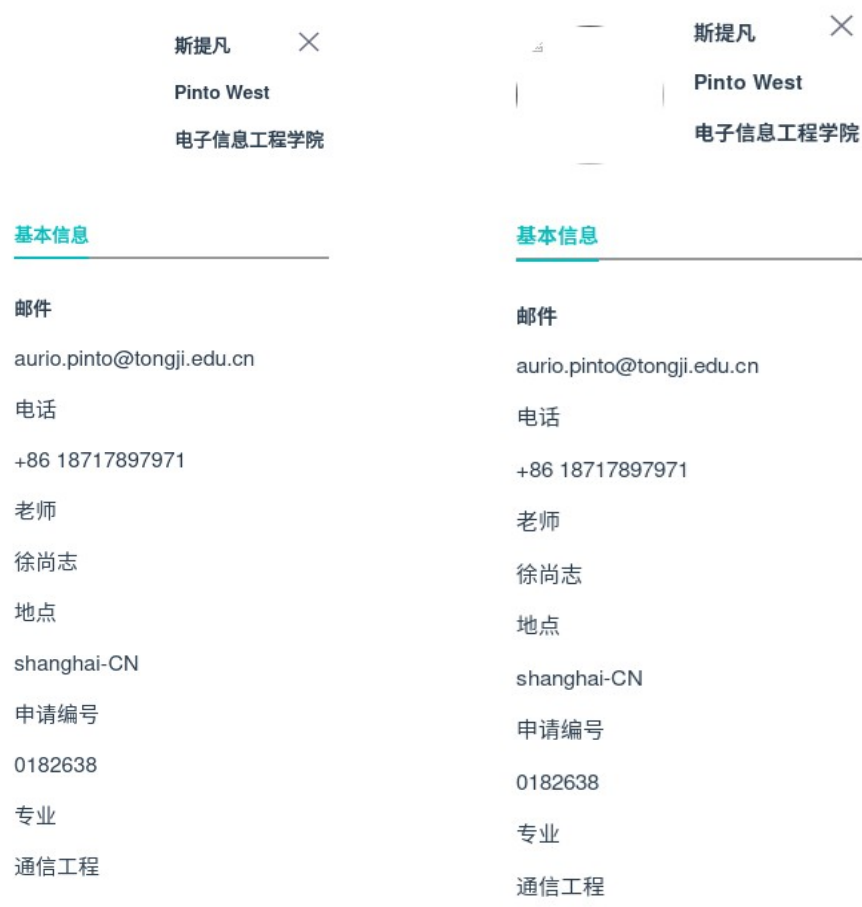
```
export default {
  name: "load2",
  components: { load },
  data: function() {
    return {};
  }
};
</script>
```



显示主页面

下面的两张图片代表了我的网页设计的首页。 我使用 Bootstrap（JS-Framework）开发了旋转木马动画，静态展示了电子教员最新活动的一些图像

软件功能



此移动应用程序将显示学生在其部门系统中保存的所有信息

2.一 代码

```
<div class="step-step">
  <div class="search-1">
    <input class="search" type="text" placeholder="Search">
    
  </div>
  <div class="box-box">
    <div class="profile-pp" v-show="onOff">
```

```

        
        <div class="profile"> 
<div class="paragh">
<P>{{currentPeople.cnName}}</P>
<P>{{currentPeople.enName}}</P>
<P>电子信息工程学院</P>
</div>
</div>
<div class="first-buttom">
<p class="lol">
<span>基本信息</span>
</p>
</div>
<div class="text-box">
<p class="bold">邮件</p>
<p>{{currentPeople.邮件}}</p>
</div>
<div>
<p class="bold">电话</p>
<p>{{currentPeople.电话}}</p>
</div>
<div>
<p class="bold">老师</p>
<p>{{currentPeople.老师}}</p>
</div>
<div>
<p class="bold">地点</p>
<p>shanghai-CN</p>
</div>
<div>
<p class="bold">申请编号</p>
<p>{{currentPeople.申请编号}}</p>
</div>
<div>
<!-- <p class="bold">毕业院校</p> -->
<!-- <p>{{currentPeople.毕业院校}}</p> -->
</div>
<div>
<p class="bold">专业</p>
<p>{{currentPeople.专业}}</p>
</div>
</div>
<div class="box-flow">
<div class="big-box">
<p>电子信息工程学院 1049 人</p>
</div>
<div class="the-box">

```

```

<div class="avatar" v-for="(people , $index) in person"
@click="showInfo(people)">  <p class="first-
p">{{people.cnName}}</p>
<p class="second-p">{{people.enName}}</p>
</div>
</div>
</div>
</div>
</div>

```

我创建了一个静态数组，其中包含所有学生的信息

```

<script>
export default {
  name: "navbar",
  data() {
    return {
      user: {
        name: "navbar"
      },
      person: [
        {
          avatar: "../assets/images/west-bj.jpg",
          cnName: "斯提凡",
          enName: "Pinto West",
          邮件: "aurio.pinto@tongji.edu.cn",
          电话: "+86 18717897971",
          专业: "通信工程",
          老师: "徐尚志",
          // 毕业院校: "Tongji University",
          申请编号: "0182638"
        },
        {
          avatar: "../assets/images/img-1.jpg",
          cnName: "文蓉蓉",
          enName: "Charlie Torphy",
          邮件: "xiaoxin.fang@tongji.edu.cn",
          电话: "+86 138363293",
          专业: "自动化",
          老师: "徐尚志",
          // 毕业院校: "Fudan University",
          申请编号: "01874638"
        },
        {
          avatar: "../assets/images/img-3.jpg",
          cnName: "廖咪",
          enName: "Raquel Haag",
          邮件: "raquel.haag@tongji.edu.cn",
          电话: "+86 127978293",

```



```

        专业: "计算机科学与技术",
        老师: "徐尚志",
        // 毕业院校: "Beijing University",
        申请编号: "01289138"
    },
    {
        avatar: "../assets/images/img-9.jpg",
        cnName: "李童童",
        enName: "Simeon Gislason",
        邮件: "simeon.gislason@tongji.edu.cn",
        电话: "+86 122739293",
        专业: "自动化",
        老师: "徐尚志",
        // 毕业院校: "Mexico University",
        申请编号: "01289138"
    },
    {
        avatar: "../assets/images/img-1.jpg",
        cnName: "文蓉蓉",
        enName: "Charlie Torphy",
        邮件: "xiaoxin.fang@nio.com",
        电话: "+86 138363293",
        专业: "计算机科学与技术",
        老师: "徐尚志",
        // 毕业院校: "Fudan University",
        申请编号: "01874638"
    }
],
currentPeople: {},
onOff: false
};

},
methods: {
    showInfo(person) {
        console.log(person);
        this.currentPeople = person;
        this.onOff = true;
    }
}
};
</script>

```

设计思想

整体程序框架

到了学期，我们已经深入了解了网页设计和应用程序的 JavaScript 和 NodeJs 和 PHP 和 HTML 与 CSS, 顺便说一句，我没有爱上 JavaScript，我深入了解它是什么，从我知道 反应和 Vue 这是最流行的 JavaScript 框架

Vue: 是用于构建用户界面的渐进式框架。与其他单片框架不同，Vue 是从头开始设计的，可逐步采用。核心库仅专注于视图层，易于获取并与其他库或现有项目集成。另一方面，当与现代工具和支持库结合使用时，Vue 也完全能够为复杂的单页应用程序提供动力。

逻辑结构与物理结构

逻辑结构：邻接表非连续工具

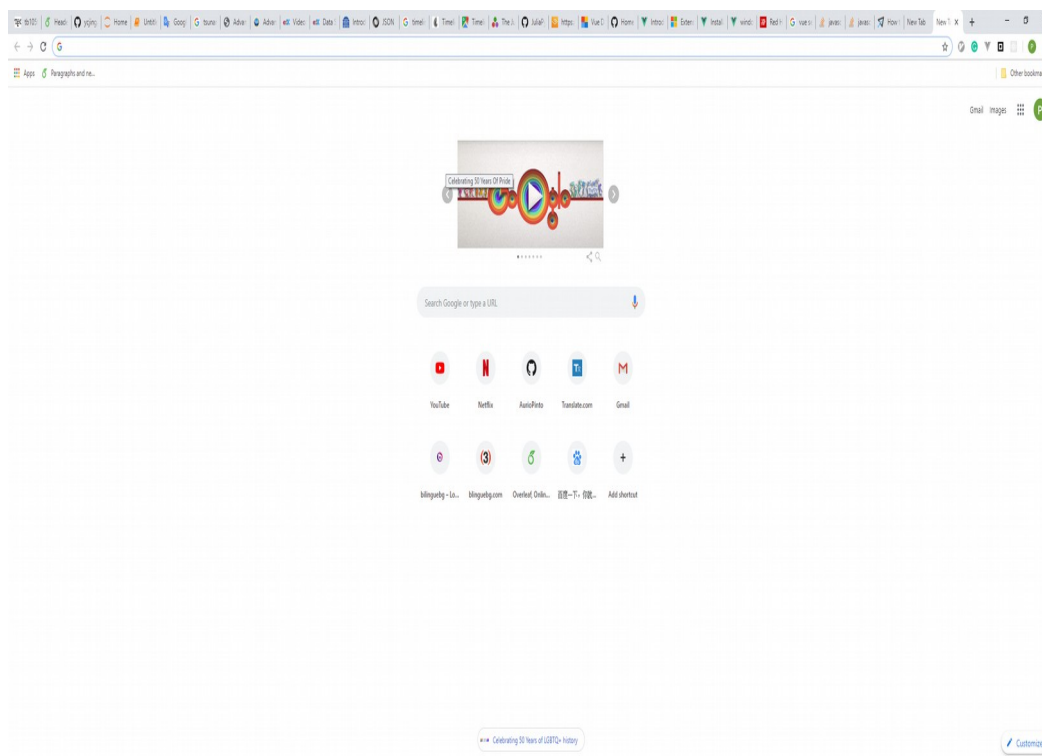
开发平台

程序开发平台：
Webstorm 2019.1



Chrome

Google Chrome: 69.0.3497.81 (Official Build) (64-bit) (cohort: Stable Installs Only)
Revision: 032b3ca19e9af20182f9bd03deefc0faf4695558-refs/branch-heads/3497@{#869}
OS: Windows
JavaScript: V8 6.9.427.19
Flash: 30.0.0.154 C:\Users\JiaHL\AppData\Local\Google\Chrome\User Data\PepperFlash\30.0.0.154\pepflashplayer.dll
User Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.81 Safari/537.36
Command Line: "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" --flag-switches-begin --flag-switches-end

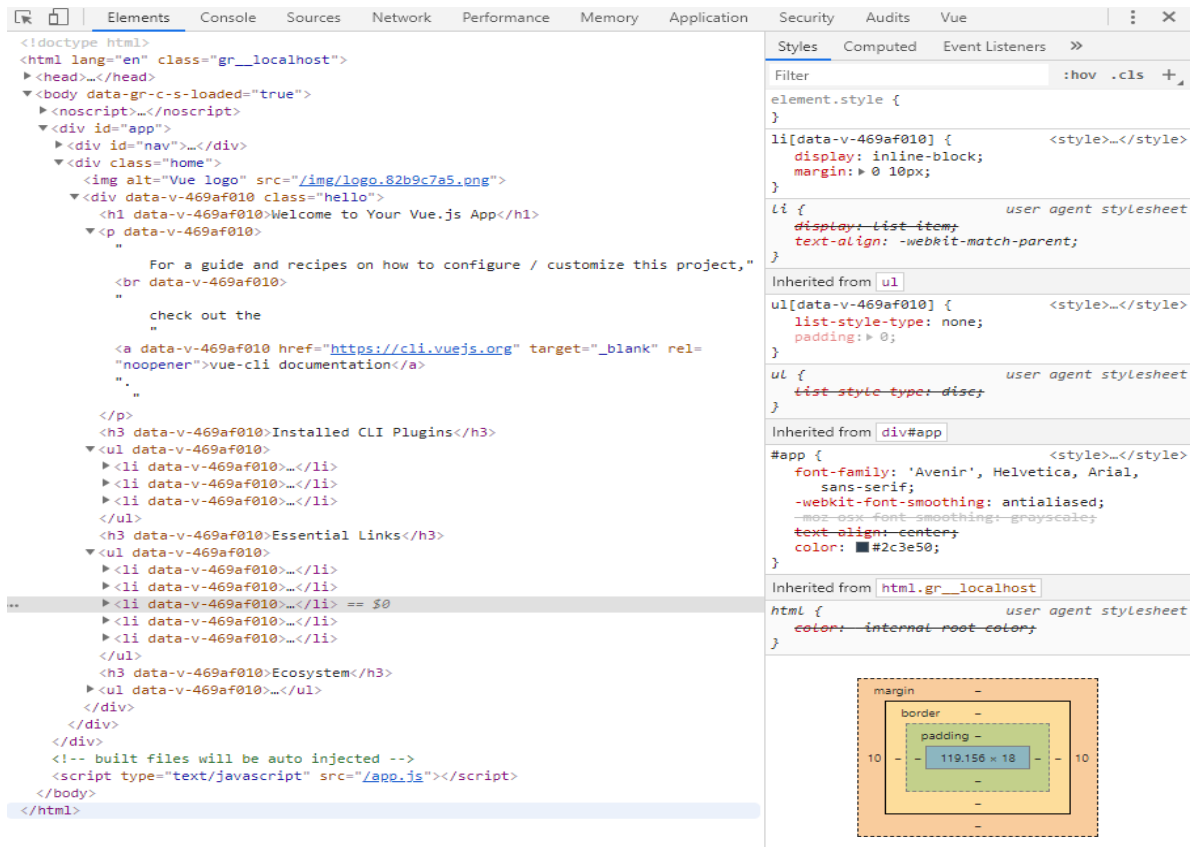


Chrome:

系统的运行结果分析说明

调试遇到的问题一

JavaScript 运行 html 时 Chrome 无法兼容信息，直到现在仍没解决



Vue 设计

运行 Vue 服务器:

- npm run serve

如果你的 vue 配置上的所有内容都运行良好，那么无论何时运行以下命令：npm run serve 将按照上图显示图像，说编译器已成功

```
DONE Compiled successfully in 2832ms 10:37:22 PM

App running at:
- Local: http://localhost:8081/
- Network: http://172.18.0.177:8081/

Note that the development build is not optimized.
To create a production build, run npm run build.
```

Vue Router

Vue Router 是 Vue.js 的官方路由器。它与 Vue.js 核心深度集成，使用 Vue.js 构建单页应用程序变得轻而易举。

功能包括：

- 嵌套路由/视图映射
- 模块化，基于组件的路由器配置
- 路线参数，查询，通配符
- 查看由 Vue.js 过渡系统提供支持的过渡效果
- 细粒度的导航控制
- 与自动活动 CSS 类的链接
- HTML5 历史模式或散列模式，在 IE9 中具有自动回退功能
- 可自定义的滚动行为

项目 router

```
import Vue from 'vue'
import Router from 'vue-router'
import Home from './views/Home.vue'
import load from './views/load.vue'
import init from './views/init'
import navbar from './views/navbar'
```

```
Vue.use(Router)
```

```
export default new Router({
  mode: 'history',
  base: process.env.BASE_URL,
  routes: [{
    path: '/',
```

```

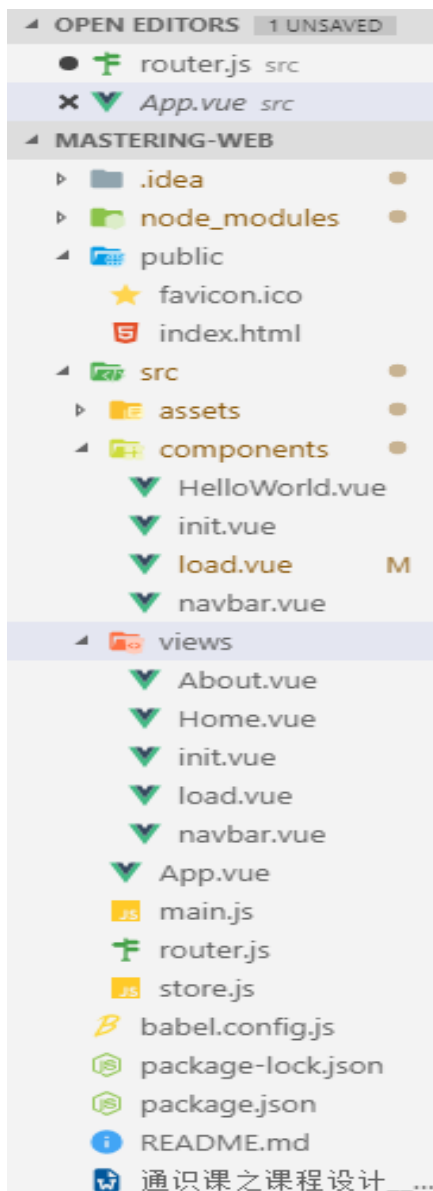
        name: 'home',
        component: Home
      },
      {
        path: '/load',
        name: 'load',
        component: load
      },
      {
        path: '/init',
        name: 'init',
        component: init
      },
      {
        path: '/navbar',
        name: 'navbar',
        component: navbar
      },
      {
        path: '/about',
        name: 'about',
        // route level code-splitting

```

```

        // this generates a separate
        chunk (about.[hash].js) for
        thisroute
        // which is lazy-loaded when
        the route is visited.
        component: () =>
          import(/*
            webpackChunkName: "about" */ './
            views/About.vue')
      }
    ]
  })

```



攻击了所有的 Vue 包

Vue API:

Vue.config 是一个包含 Vue 全局配置的对象。您可以在引导应用程序之前修改下面列出的属性：

Vue.filter(id, [definition])

- Arguments:

- {string} id
- {Function} [definition]

- Usage:

Register or retrieve a global filter.

```
// register
Vue.filter('my-filter', function (value) {
  // return processed value
})

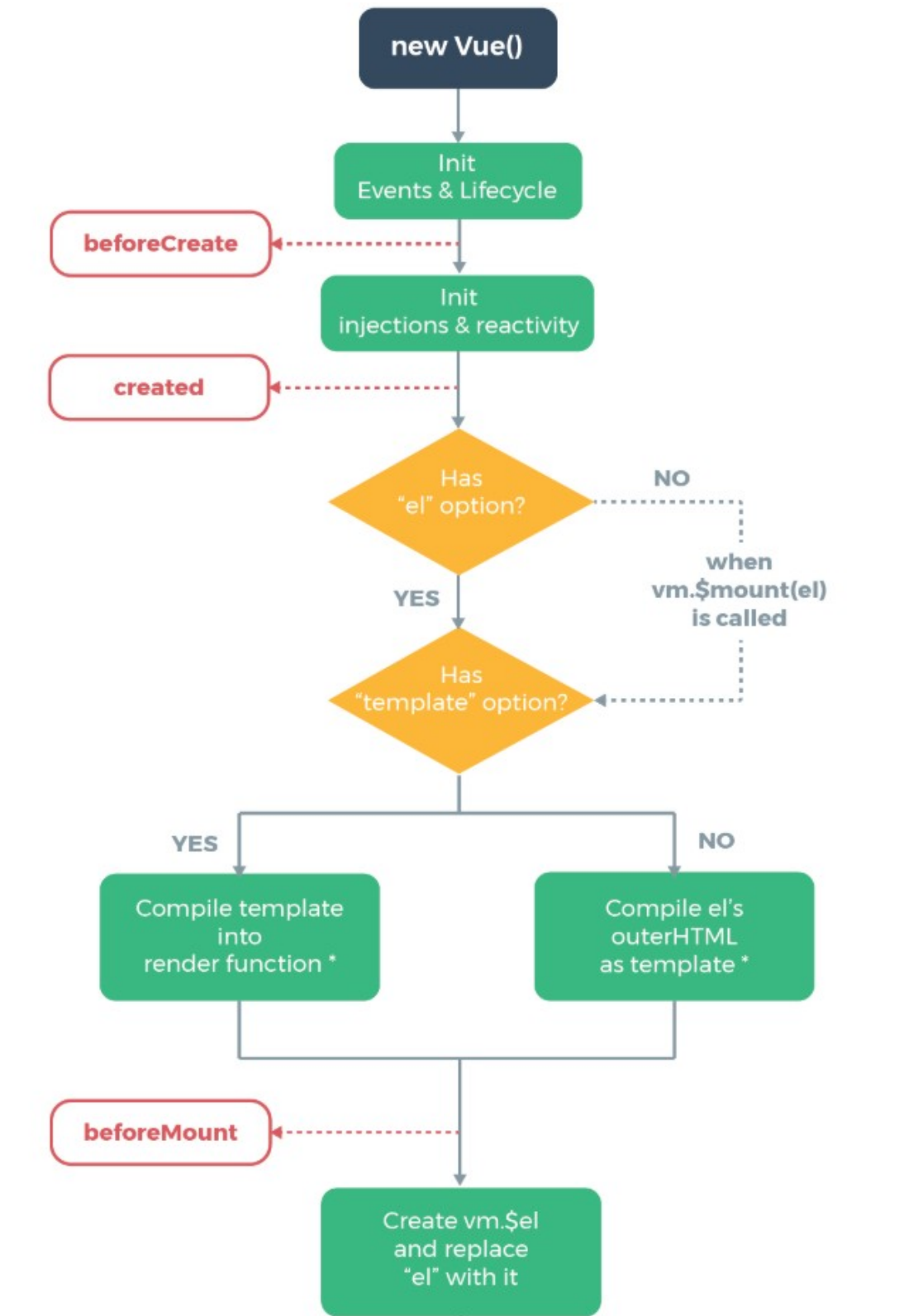
// getter, return the filter if registered
var myFilter = Vue.filter('my-filter')
```

- See also: [Filters](#)

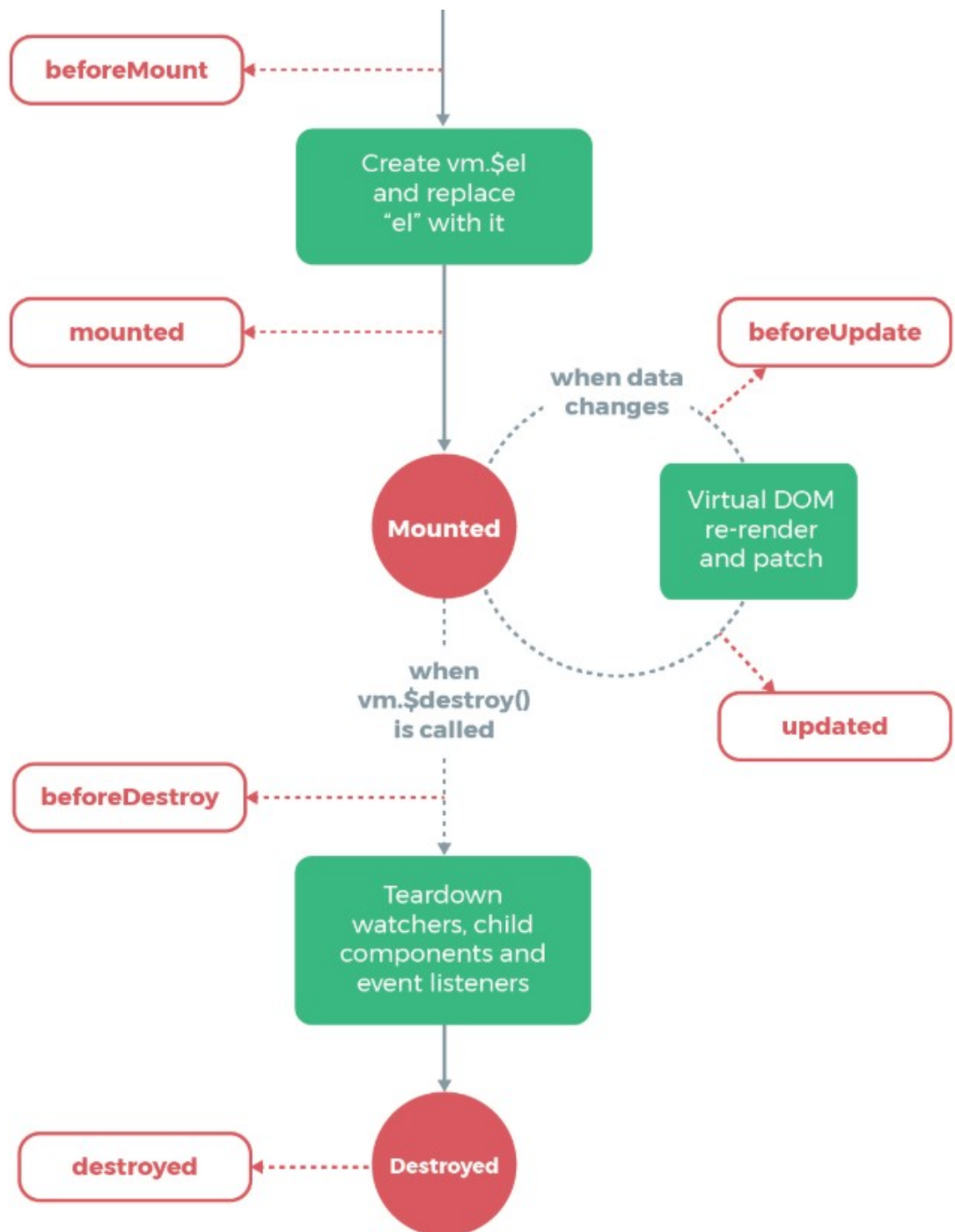
关于 Vue 的更多信息

生命周期图

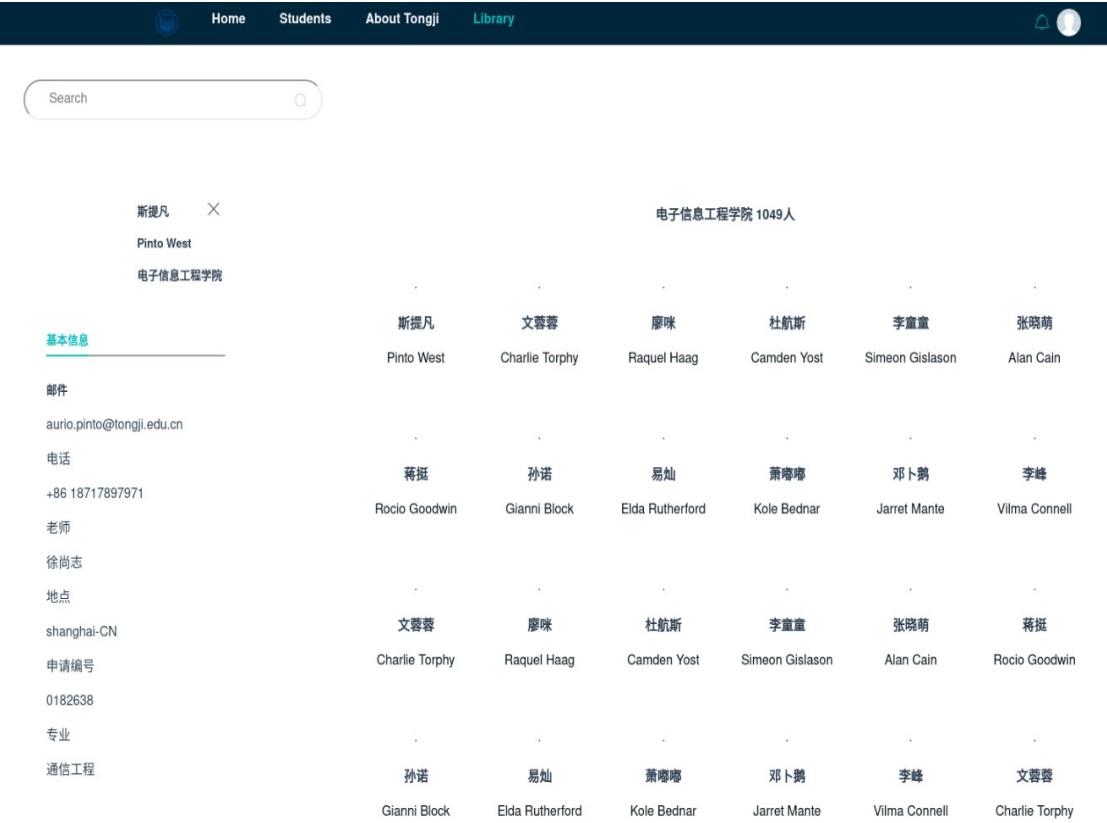
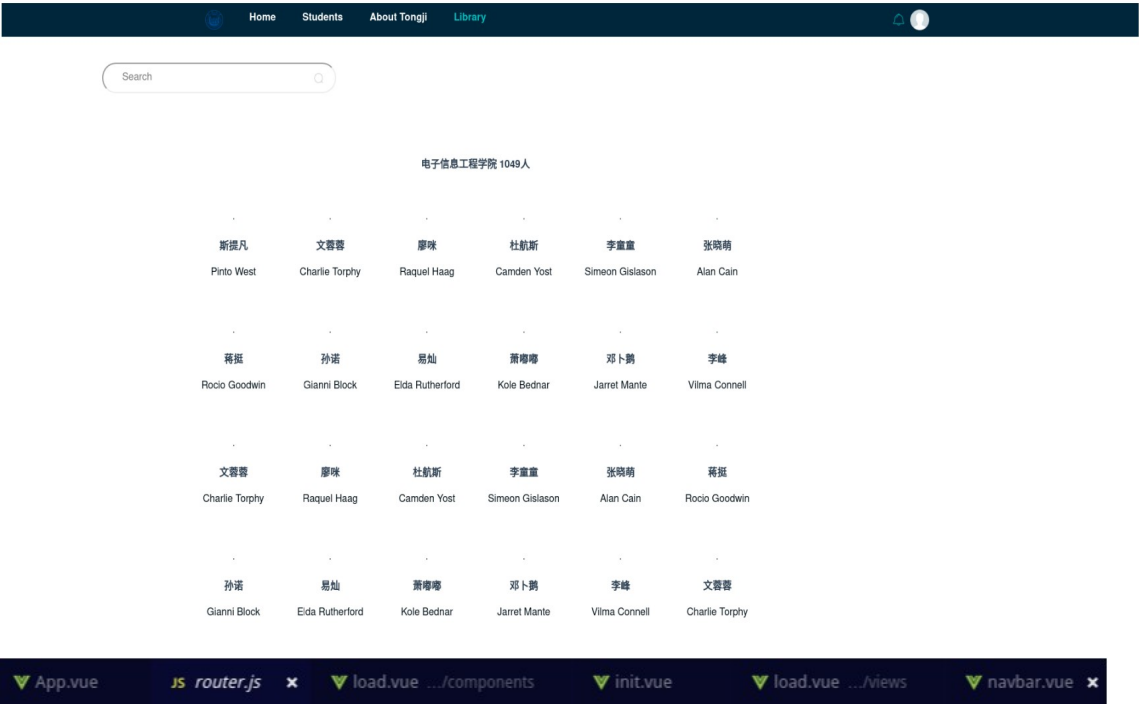
下面是实例生命周期的图表。你不需要完全理解现在正在发生的一切，但是当你学习和构建更多时，它将是一个有用的参考。



继续:



结论



参考文献

- [1] 廖雪峰. javascript 教学[EB/OL].
<https://www.liaoxuefeng.com/wiki/001434446689867b27157e896e74d51a89c25cc8b43bdb3000>.
- [2] google
- [3] <https://chrome.google.com/webstore/category/extensions>
- [4] <https://vuejs.org/>
- [5] <https://getbootstrap.com/docs/4.3/layout/overview/>
- [6] 学习 HTML 标签以及各个前端类及其属性
- [7] 为了美化前端，选用朴素简单上手的 CSS 库
- [8] 参考学生的社交关系网络开源代码，高级用 JavaScript 编写，我学到了经验，加入了我的想法并实现了我的代码。

注：所有代码、文档以及学习笔记均以 push 到我的 github 上：
<https://github.com/AurioPinto/Mastering-Web>