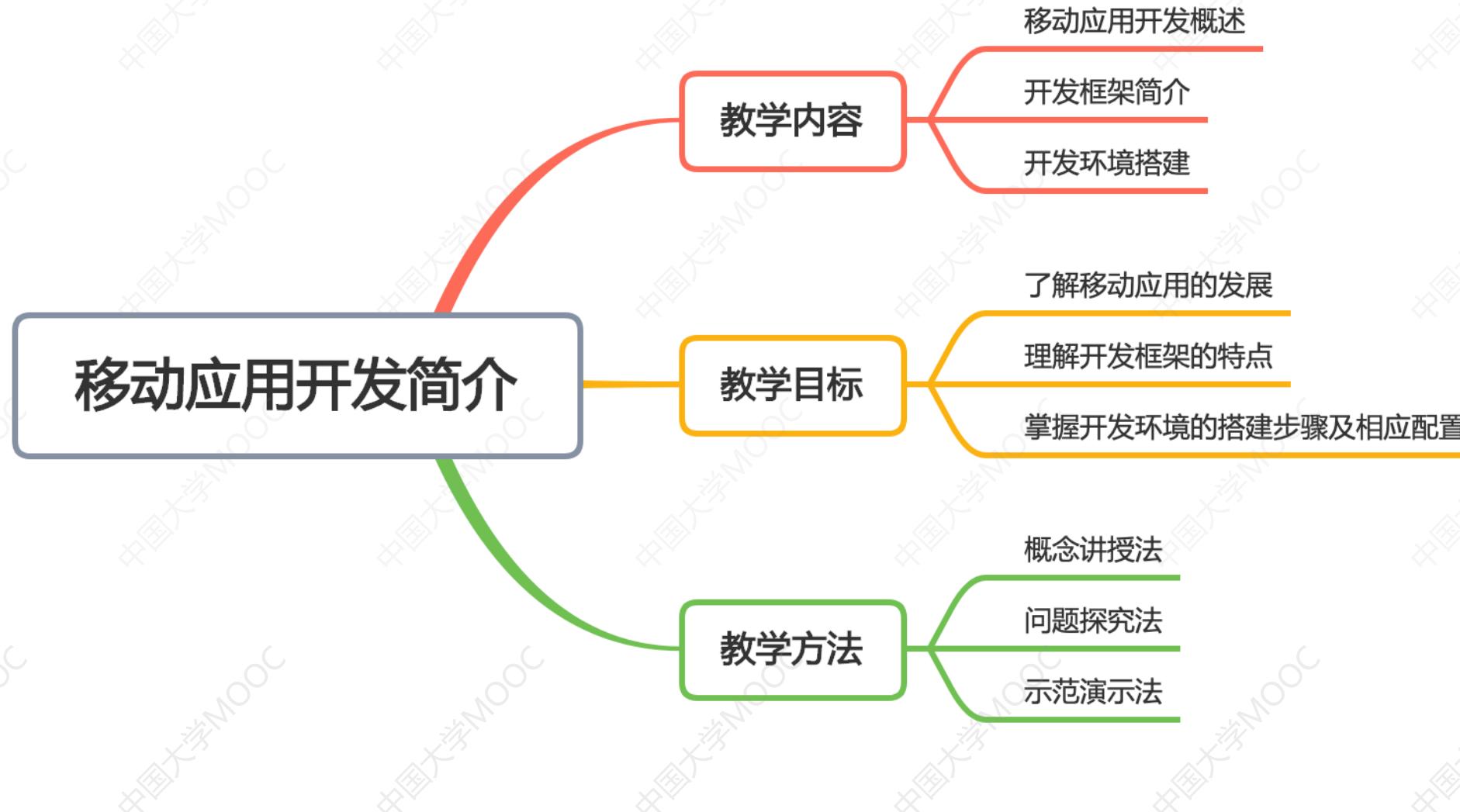




移动应用程序开发

常熟理工学院 计算机科学与工程学院

第一章 移动应用开发简介



移动应用开发概述

1. 什么是移动应用开发

移动应用开发又称为手机开发，移动互联网开发。是指以手机、PDA、UMPC等移动终端为基础，进行相应的移动终端应用程序开发。目前主要以手机端的开发为主。



移动应用开发概述

2. 移动应用的发展

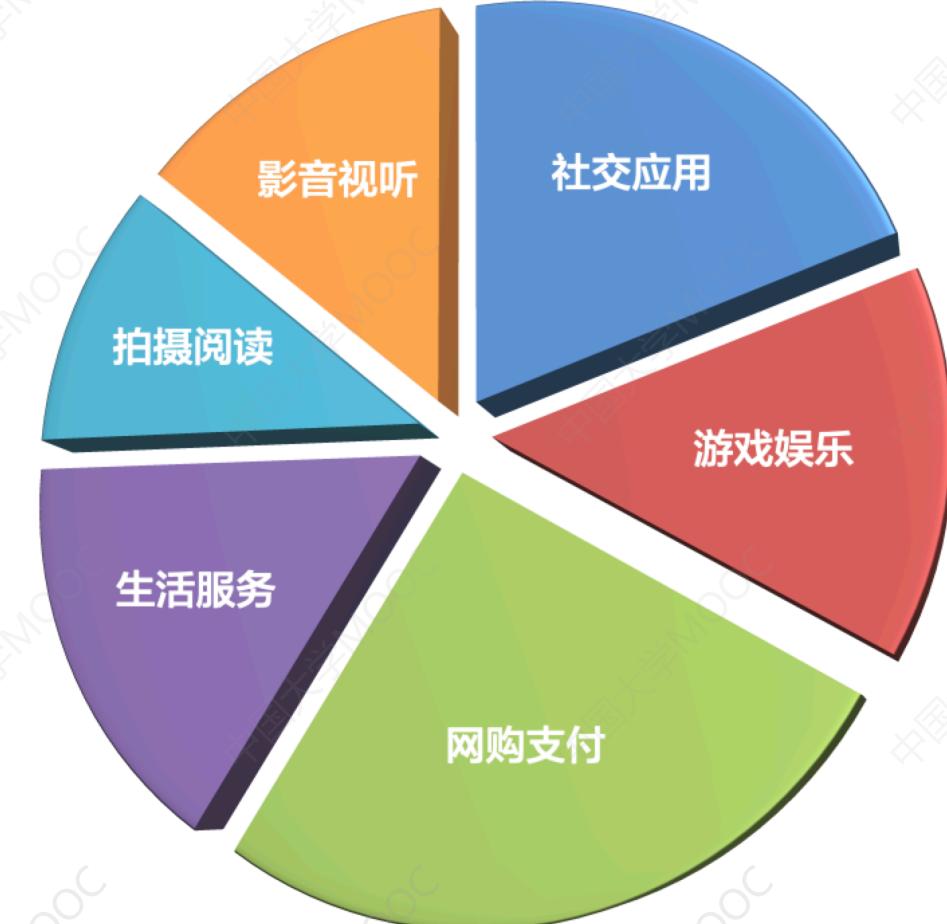
随着移动互联网的飞速发展，人们可以通过手机、平板电脑等可移动数据终端与互联网连接，从而获得互联网中的海量资讯。目前，移动应用程序已经深入到我们日常生活的衣食住行，工作，人际交流等方方面面，已经成为我们生活的一部分。



移动应用开发概述

3. 移动应用的分类

| 类型 | 主要软件 |
|------|----------------------------|
| 影音视听 | 爱奇艺、网易云音乐、腾讯视频等 |
| 社交应用 | 微信、QQ、微博、米聊等 |
| 游戏娱乐 | 棋牌、跑酷、动作、策略等 |
| 网购支付 | 淘宝、天猫、京东、美团等 |
| 生活服务 | 携程、浏览器类APP、墨迹天气、地图导航、一点资讯等 |
| 拍摄阅读 | 美图、美颜相机、QQ阅读 |

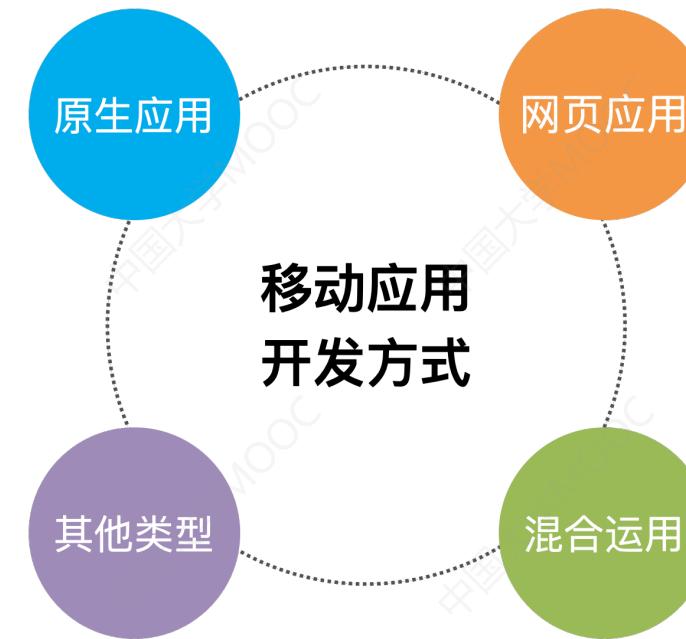


移动应用开发概述

4. 移动应用的开发方式

基于某个移动平台(如安卓)，使用平台支持的开发工具和语言(如安卓平台支持Eclipse和Java)开发的应用。

在现有的应用程序框架下开发的小程序(如微信小程序)，或在统一开发平台(如uni-app)下开发的程序。



使用HTML5、JavaScript和CSS技术开发的基于Web的，可以在多个设备上运行的跨平台应用。

把HTML5应用程序嵌入到一个细薄的原生容器里面，集原生应用程序和HTML5应用程序的特点于一体的应用。

移动应用开发概述

5. 移动应用的开发方式比较

| | 网页应用 | 混合应用 | 原生应用 | 其他类型 |
|------|------|------|------|------|
| 开发成本 | 低 | 中 | 高 | 低 |
| 维护更新 | 简单 | 简单 | 复杂 | 简单 |
| 体验 | 差 | 中 | 优 | 中 |
| 商店认可 | 不认可 | 认可 | 认可 | 不认可 |
| 安装 | 不需要 | 需要 | 需要 | 不需要 |
| 跨平台 | 优 | 优 | 差 | 中 |

网页应用开发方式具有开发成本低、维护更新简单、无需安装可跨平台等优点，在用户体验和性能上稍差，随着手机硬件设备的完善、移动Web浏览器对新技术的支持日益加大，用户体验和网站性能也会逐步得到改善。

开发框架简介

1. 基础技术



HTML5：互联网的下一代标准，是构建以及呈现互联网内容的一种语言方式。被认为是互联网的核心技术之一。用于构建页面内容。



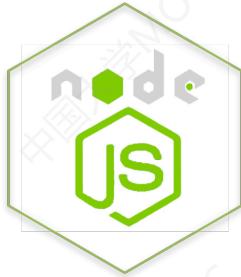
CSS3：是CSS（层叠样式表）技术的升级版本，用于控制页面的样式。



JavaScript：是一种属于网络的高级脚本语言，用于为页面添加各式各样的动态功能，为用户提供流畅美观的浏览效果。

开发框架简介

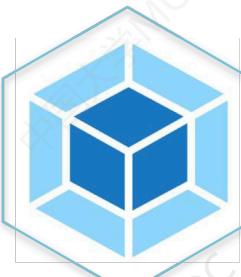
2. 框架技术



Node.js：一个基于 Chrome V8 引擎的 JavaScript 运行环境，用于方便地搭建响应速度快、易于扩展的网络应用。



Vue.js：一套用于构建用户界面的渐进式JavaScript框架。简单小巧的核心，渐进式技术栈，足以应付任何规模的应用。



Webpack：是一个用于现代 JavaScript 应用程序的静态模块打包工具，用于应用程序的封装。

开发框架简介

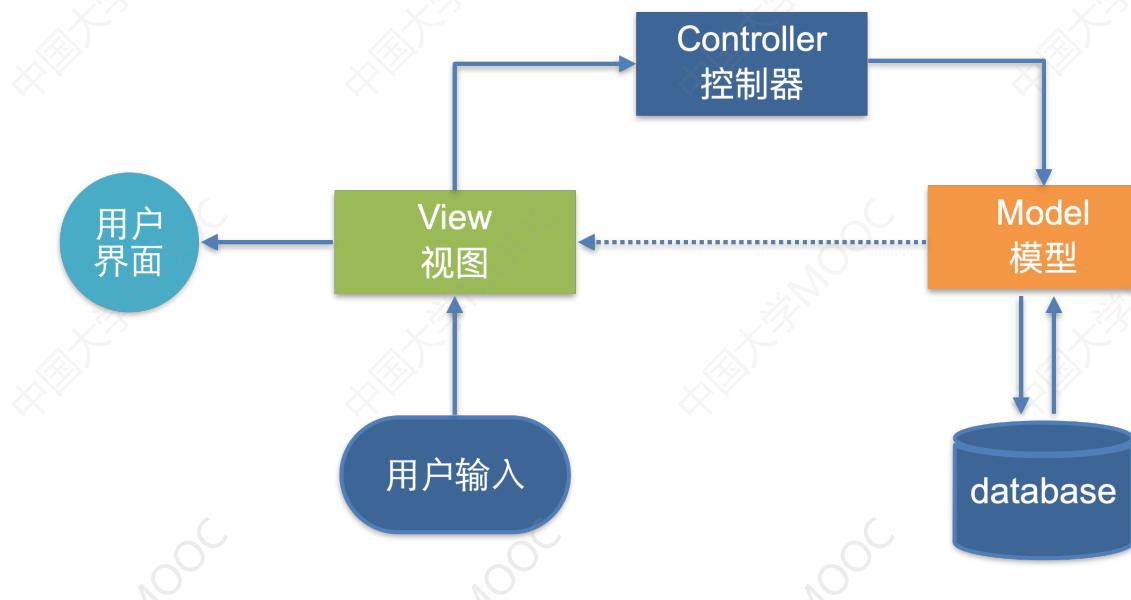
3. 开发模式比较（MVC模式）

MVC开发模式，其结构如下。

M指的是model（业务逻辑，接口数据）

V指的是view（用户可视页面）

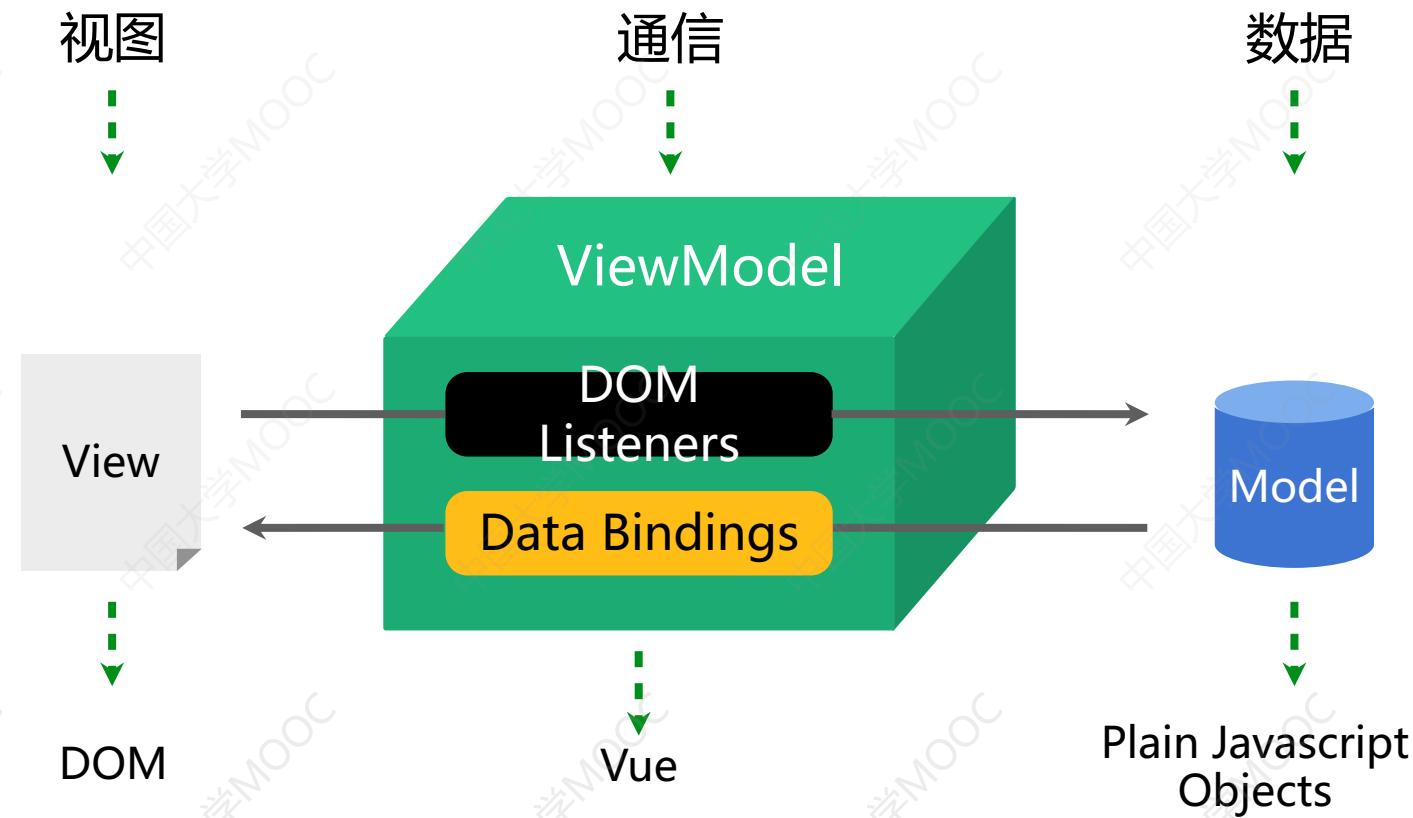
C指的是controller（用户界面和后台接口之间的控制层，主要是应用逻辑处理）



开发框架简介

3. 开发模式比较（MVVM模式）

根据问题导向和需求导向，我们对控制层进行瘦身，并将控制器的功能移动到View上，由此产生了MVVM模式。



4. 开发框架简介

Vue.js 可以说是MVVM 架构的最佳实践，是一个JavaScript MVVM 库，是一套构建用户界面的渐进式框架。专注于 MVVM 中的 ViewModel，不仅做到了数据双向绑定，而且也是一款相对比较轻量级的JS 库，API 简洁特点：

1. 轻量级 JavaScript 框架
2. 数据驱动，简化 DOM 操作
3. 组件化的前端开发
4. 视图数据结构分离



开发框架简介

5. 开发架构（前后端分离总架构图）

访问层：移动设备，pc电脑

前端ui：web应用使用element，iview，jquery

移动app的ui有android/IOS原生ui

微信小程序类ui等

交互层：vue，vuex，vue-router

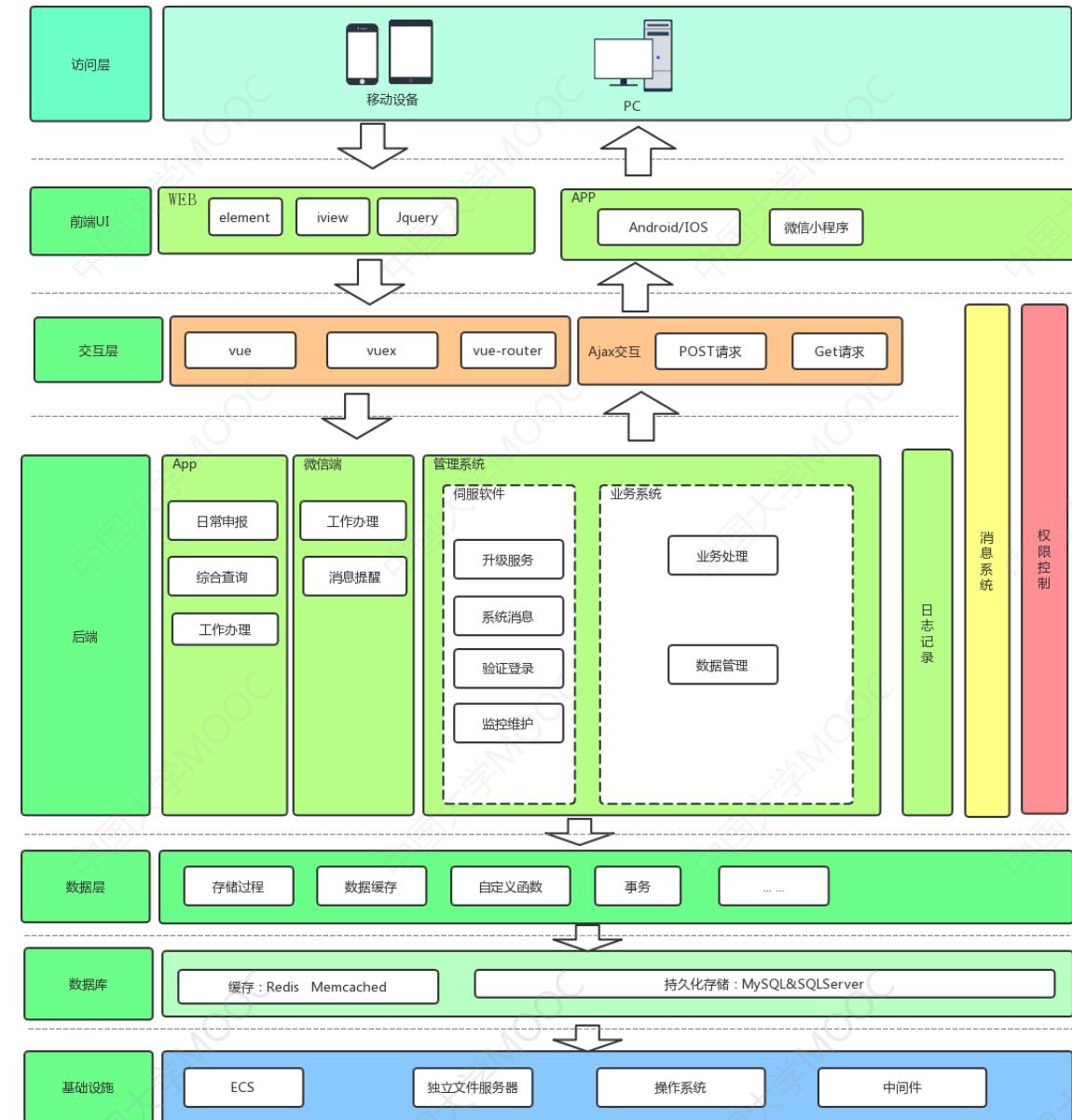
ajax交互（post请求，get请求）

后 端：App日常申报，查询及工作办理

微信端工作办理，消息提醒

管理系统主要用于业务管理，数据管理

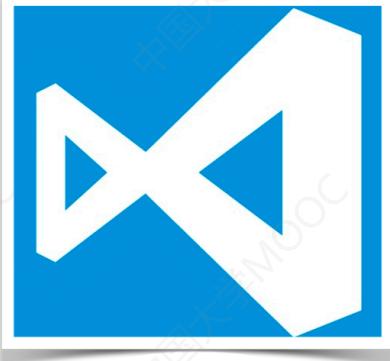
数据层：数据库，基础设置基本类似



开发环境搭建

1. 开发工具

基础学习



1. 代码编辑工具

Visual Studio Code

下载地址：

<https://code.visualstudio.com/>



2. 前端页面测试，调试工具

Chorme

项目开发



1. JavaScript运行时环境

node.js

下载地址：

<http://nodejs.cn/download/>



2. 快速构建vue项目结构工具

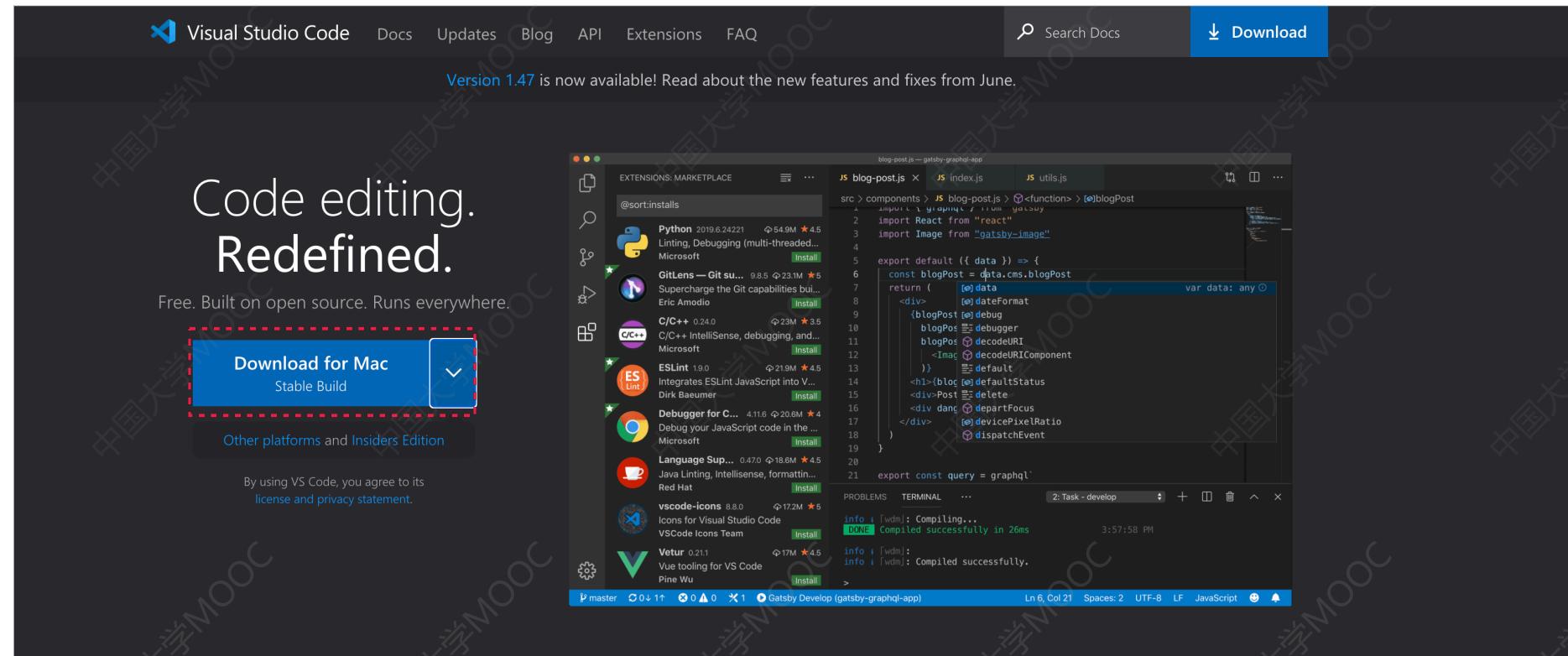
Vue-cli脚手架

开发环境搭建

2. 开发工具安装及配置

(1) Visual studio code 安装

打开 <https://code.visualstudio.com/>，如下图所示，根据系统选择相应版本下载，按照提示进行安装即可。

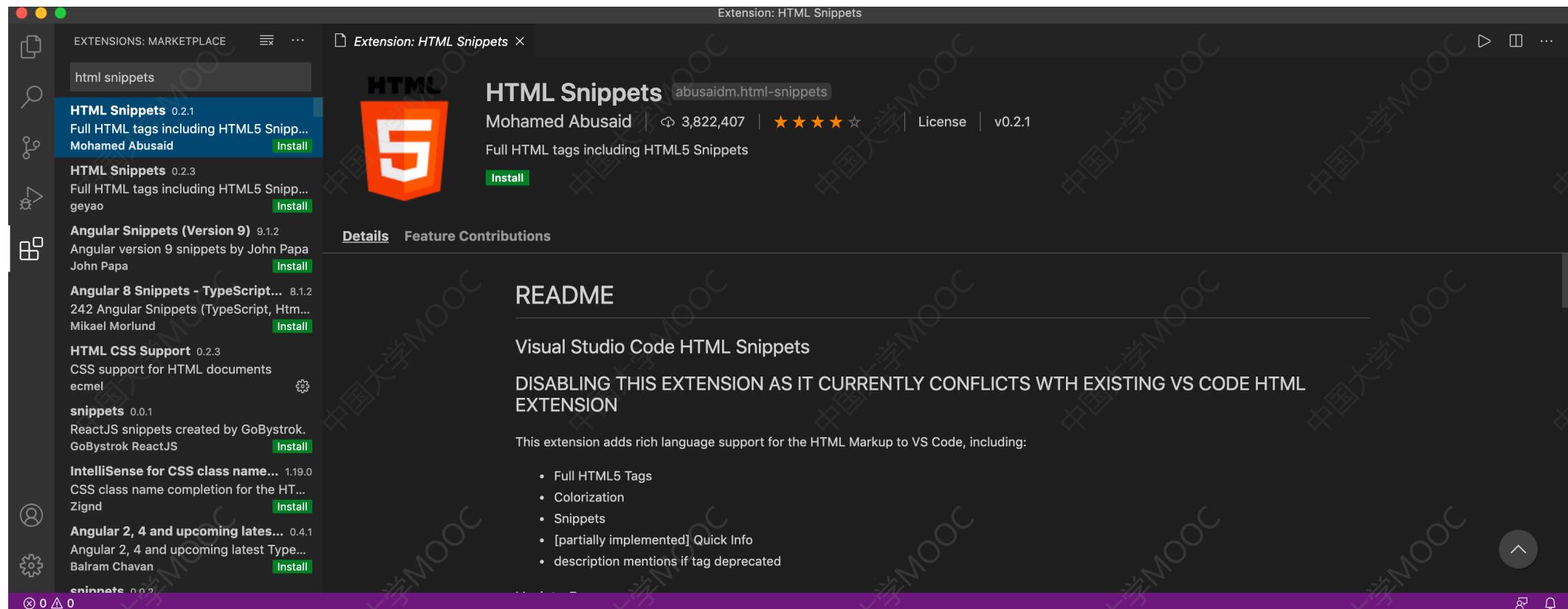


开发环境搭建

2. 开发工具安装及配置

(2) Visual studio code插件安装

选择左侧工具栏Extensions；在搜索栏中输入需安装插件的名称，在结果栏中选择对应插件；在右侧选择install进行安装即可



开发环境搭建

2. 开发工具安装及配置

(3) 需安装插件及功能介绍

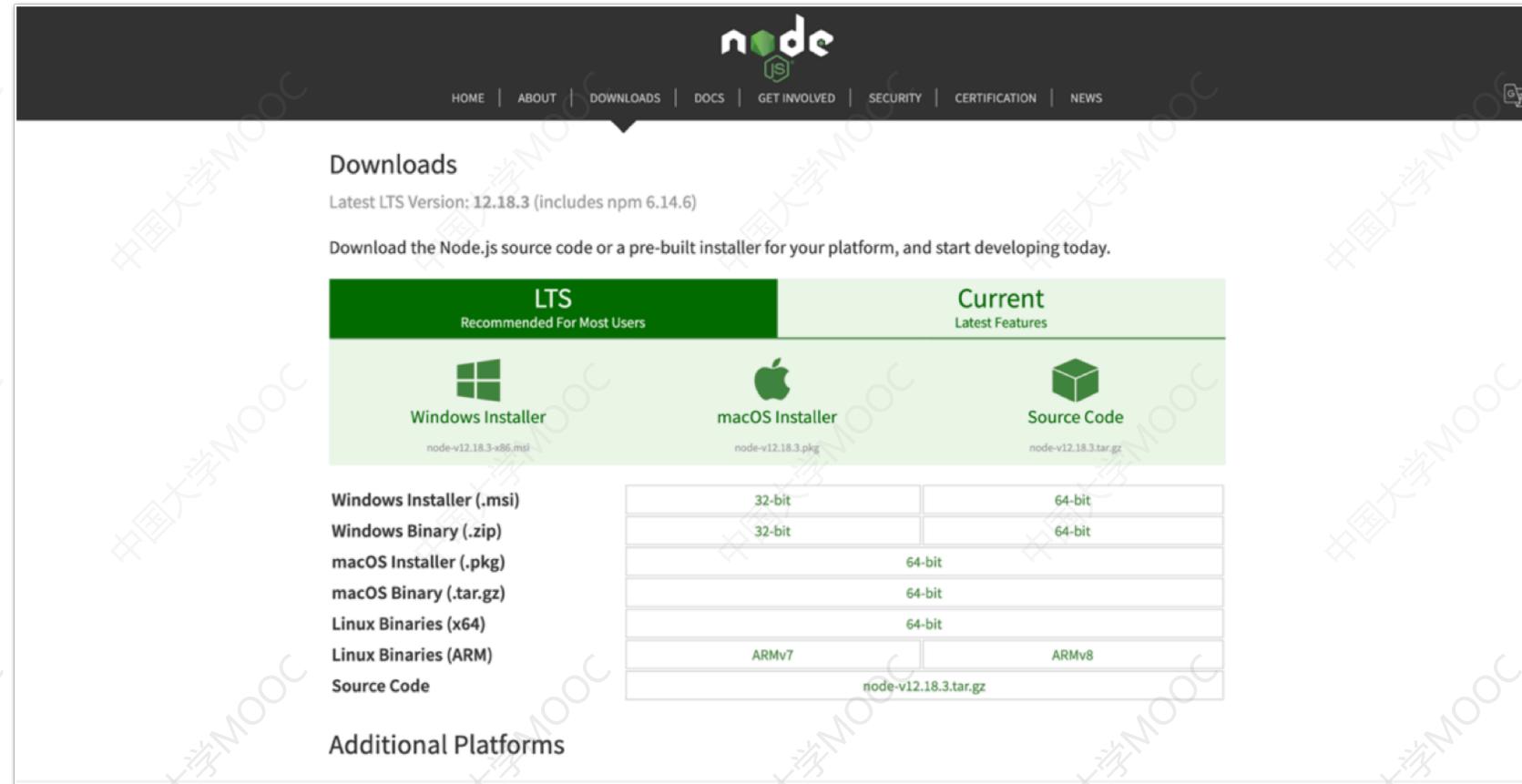
| 插件名称 | 插件功能 |
|--------------------------|------------------------------|
| HTML Snippets | H5代码片段以及提示 |
| HTML CSS Support | html 标签智能提示（当前项目所支持的class样式） |
| Debugger for Chrome | VScode 调用 Chrome 的 debug功能 |
| VScode-icons | VScode 资源树目录图标美化 |
| ESlint | JavaScript代码风格检测,代码自动格式化 |
| HTMLHint | html代码检测 |
| Atuo Rename Tag | html自动完成尾部闭合标签的同步修改 |
| Vetur | Vue 语法高亮、智能感知、Emmet等 |
| Vuehelper或Vue 2 Snippets | VS Code中vue2代码片段补齐 |
| Live Server | 开启本地服务，查看网页效果 |

开发环境搭建

2. 开发工具安装及配置

(4) Node.js的安装

打开 <http://nodejs.cn/download/> 页面，选择LTS系列，根据操作系统选择相应版本下载。

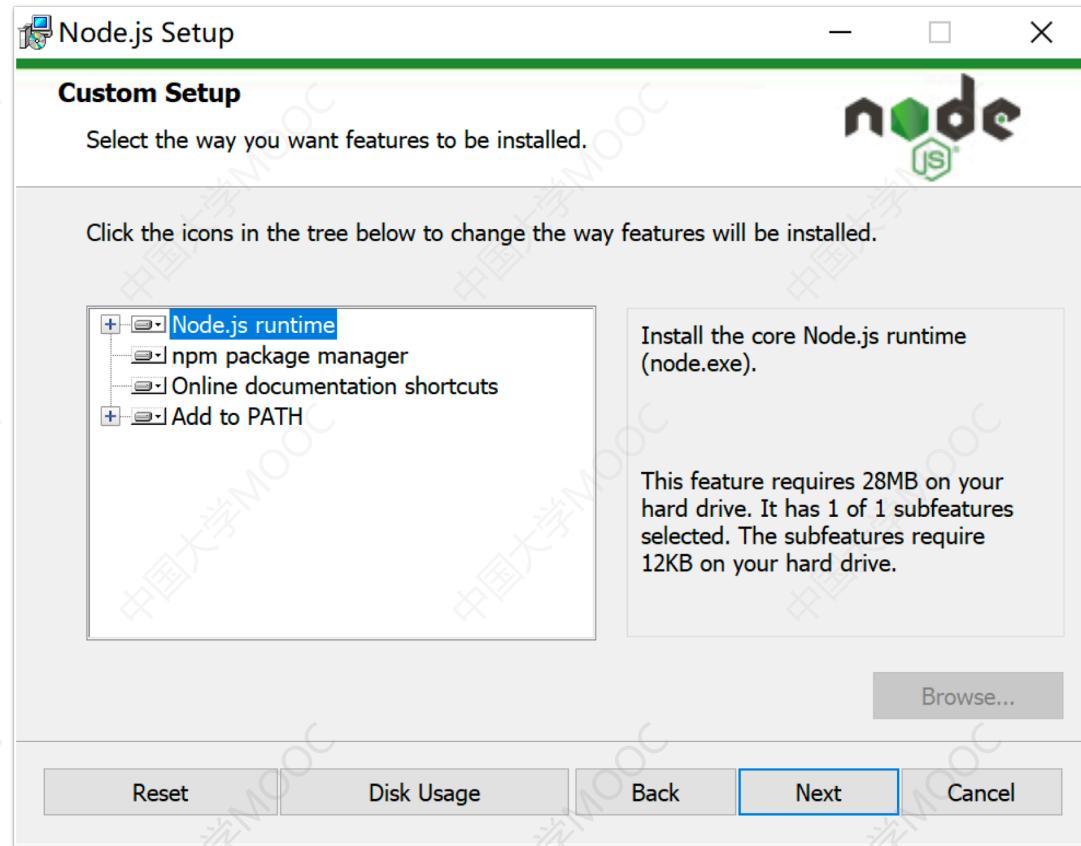


开发环境搭建

2. 开发工具安装及配置

(4) Node.js的安装

打开安装程序，按步骤进行操作，安装时这四个选项都需选中（默认）



各选项含义

Node.js runtime 表示运行环境

npm package manager 表示 npm 包管理器

online documentation shortcuts 在线文档快捷方式

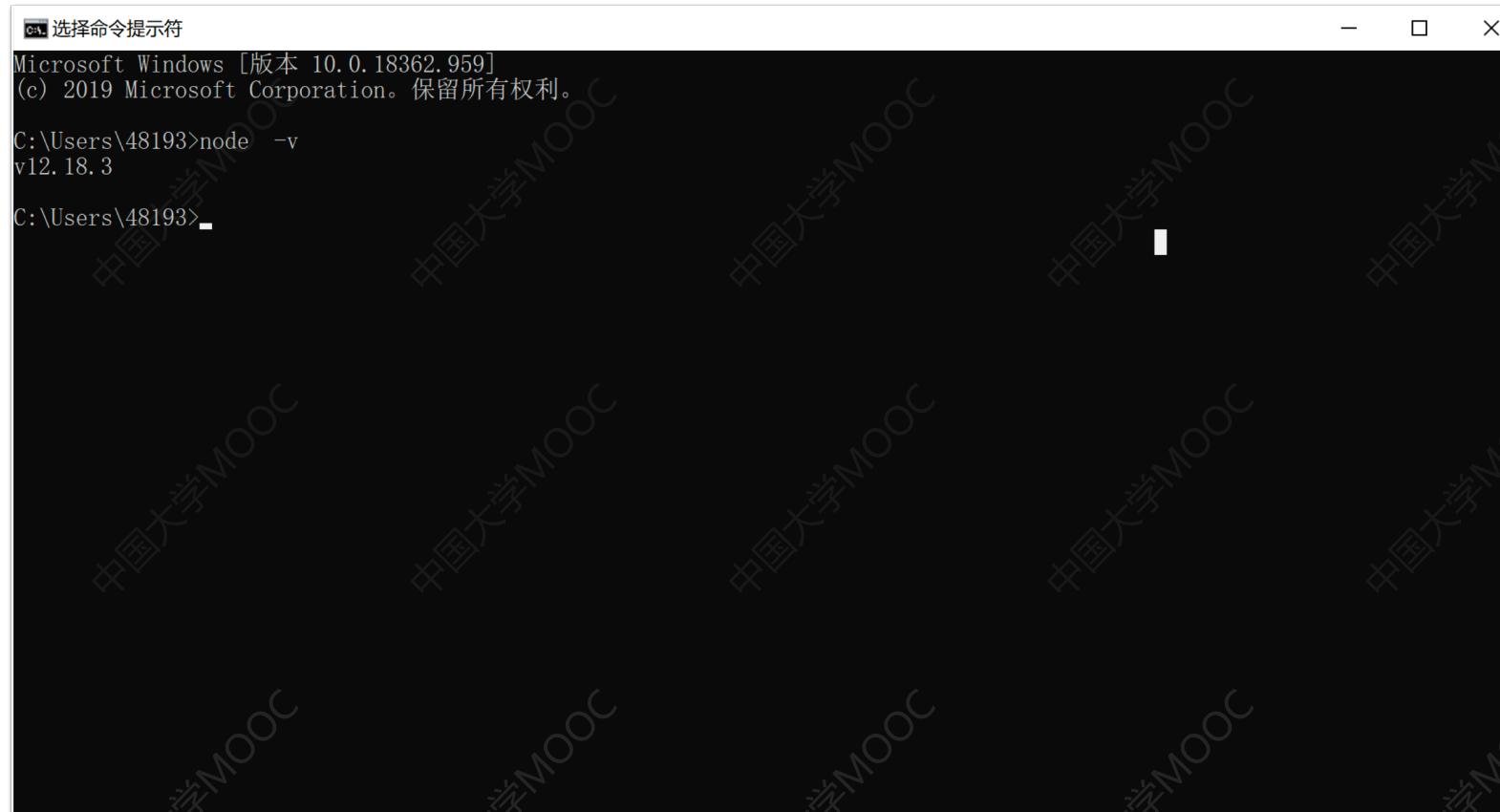
Add to PATH 添加到环境变量

开发环境搭建

2. 开发工具安装及配置

(4) Node.js的安装

安装完成后，打开CMD窗口，输入node -v，有版本信息显示即安装成功。



```
选择命令提示符
Microsoft Windows [版本 10.0.18362.959]
(c) 2019 Microsoft Corporation。保留所有权利。
C:\Users\48193>node -v
v12.18.3
C:\Users\48193>-
```

注

nodejs内置了npm。npm是Node Package(包)Manager(管理器)的缩写，当我们开发需要引用相应的包或者插件时，就可以利用npm命令来添加，加快我们的开发。

开发环境搭建

2. 开发工具安装及配置

(5) chrome插件安装 (Vue.js devtools调试插件)

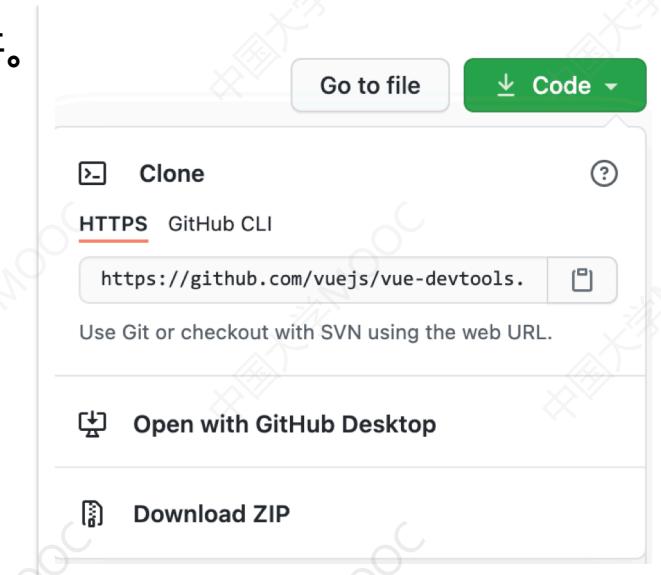
- 安装cnpm：由于npm的仓库源布置在国外，资源传输速度较慢，一般使用淘宝镜像源的cnpm，在命令提示符中输入如下命令。

```
npm install -g cnpm --registry=https://registry.npm.taobao.org
```

- 下载Vue : Devtools访问如下网址，选择code，download zip 下载文件。

<https://github.com/vuejs/vue-devtools/tree/master>

- 解压文件，并在命令提示符下进入文件目录vue-devtools-master
输入如下命令：cnpm install



开发环境搭建

2. 开发工具安装及配置

(5) chrome插件安装 (Vue.js devtools 调试插件)

- 找到shells/chrome/manifest.json 文件 将文件中的persistent:false修改 为true

```
"background": {  
    "scripts": [  
        "build/background.js"  
    ],  
    "persistent":false ➔ true  
},
```

| Asset | Size | Chunks | Chunk Names |
|------------------------|-----------|--------|-----------------|
| devtools.js | 878 kB | 0 | [emitted] [big] |
| backend.js | 34.7 kB | 1 | [emitted] |
| hook.js | 2.34 kB | 2 | [emitted] |
| detector.js | 3.61 kB | 3 | [emitted] |
| proxy.js | 878 bytes | 4 | [emitted] |
| devtools-background.js | 1.78 kB | 5 | [emitted] |
| background.js | 2.21 kB | 6 | [emitted] |

- 在命令提示符下 (文件目录为vue-devtools-master) 输入命令 : npm run build

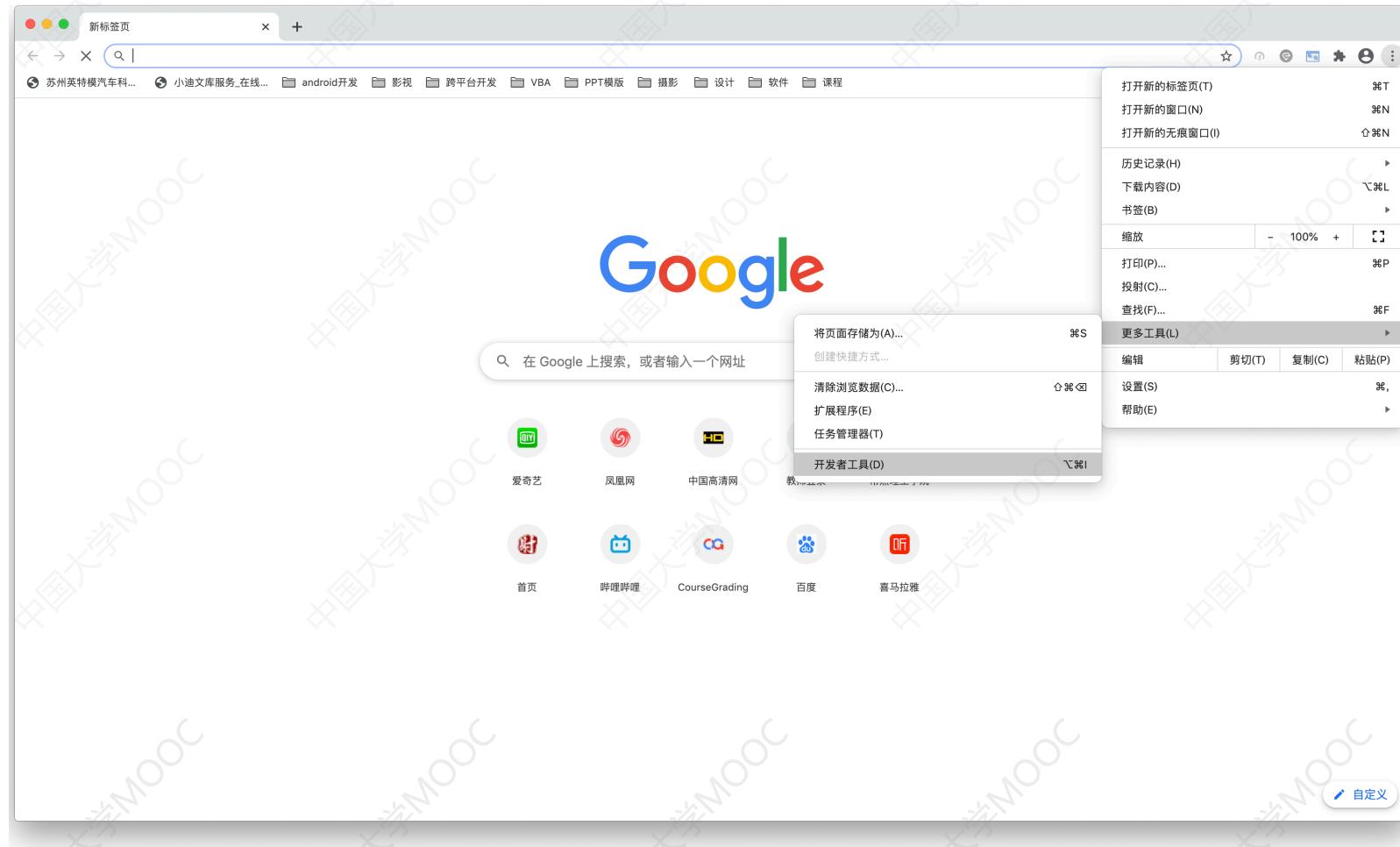
出现上图所示 , 即代表编译成功

- 找到谷歌浏览器的扩展程序功能 , 勾选开发者模式 , 然后我们将插件文件夹里的 shells>chorme文件夹直接拖到页面中 , 完成安装。

开发环境搭建

2. 开发工具安装及配置

(6) chrome调试模式开启

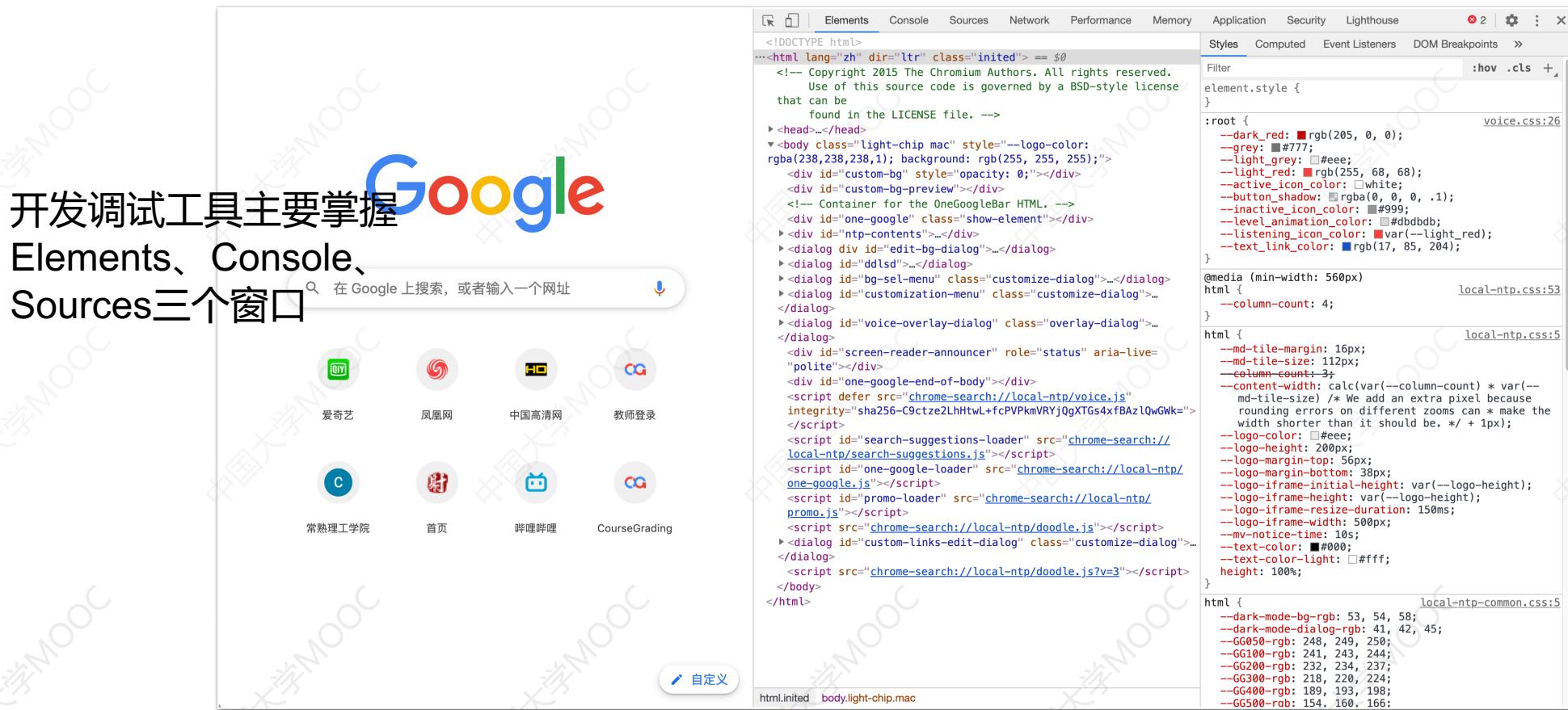


开发环境搭建

2. 开发工具安装及配置

(7) 调试模式窗口介绍

窗口主要分为两大部分，左侧为页面效果预览，右侧是开发调试工具



开发环境搭建

2. 开发工具安装及配置

(7) 移动调试模式窗口介绍

点击该按钮可切换到移动调试模式



The screenshot shows the mobile debugging mode interface. At the top, it says "iPhone X" with a dropdown menu, "375 x 812", "100% ▾", "Mobile ▾", and a three-dot menu. In the center, there is a preview window showing a Google search results page on an iPhone X screen. To the right of the preview is a "Responsive" sidebar with a list of devices. The "iPhone X" option is checked. Other listed devices include Moto G4, Galaxy S5, Pixel 2, Pixel 2 XL, iPhone 5/SE, iPhone 6/7/8, iPhone 6/7/8 Plus, iPad, iPad Pro, Surface Duo, and Galaxy Fold. At the bottom of the sidebar are "Edit..." and "语音" buttons.

iPhone X ▾ 375 x 812 100% ▾ Mobile ▾

Responsive

- Moto G4
- Galaxy S5
- Pixel 2
- Pixel 2 XL
- iPhone 5/SE
- iPhone 6/7/8
- iPhone 6/7/8 Plus
- ✓ iPhone X
- iPad
- iPad Pro
- Surface Duo
- Galaxy Fold

Edit... ▶

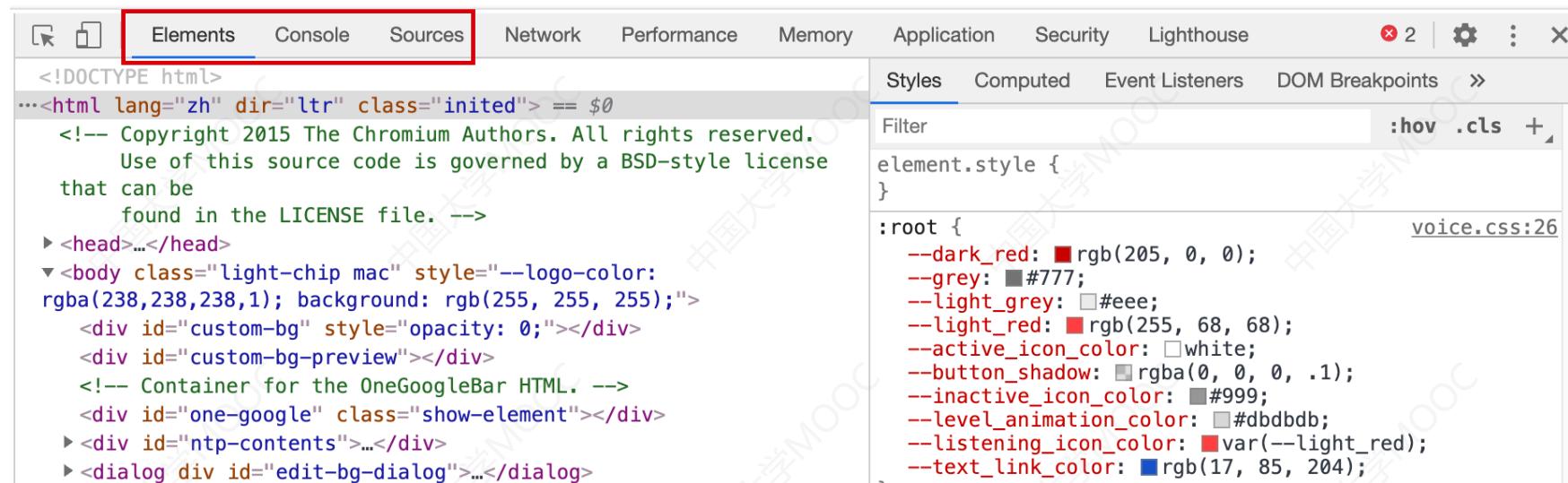
语音

开发环境搭建

2. 开发工具安装及配置

(8) 调试窗口

开发调试工具主要掌握Elements、Console、Sources三个窗口

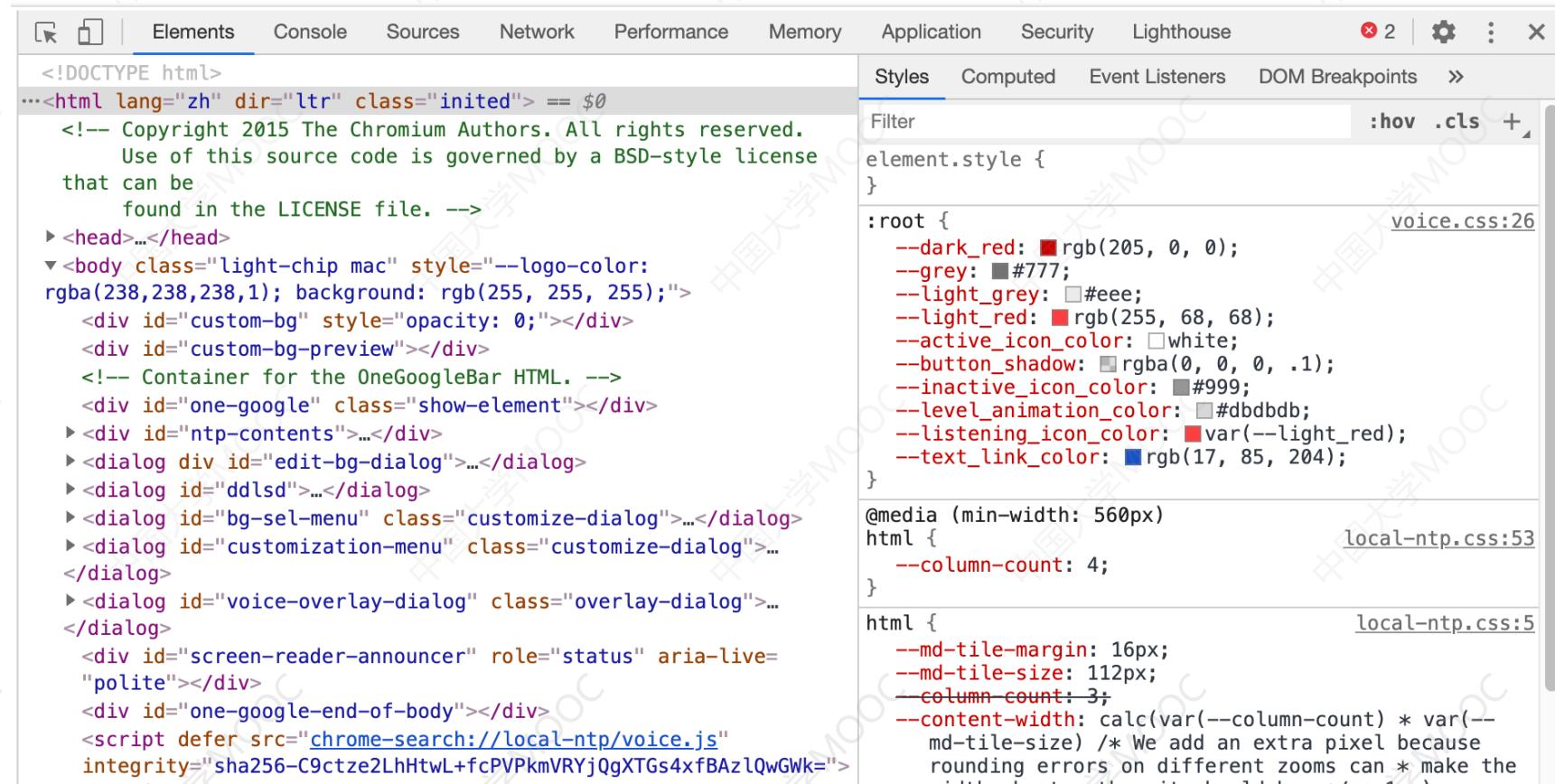


开发环境搭建

2. 开发工具安装及配置

(8) 调试窗口

Elements窗口：查看页面的HTML，CSS代码，并进行相应的调试及测试操作

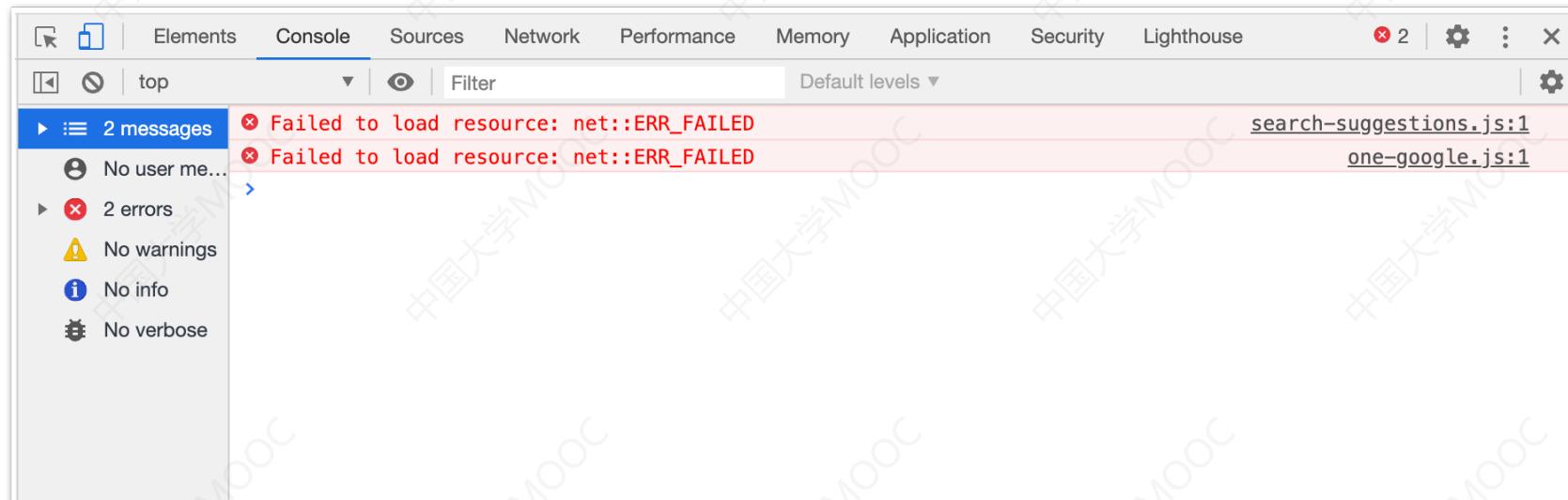


开发环境搭建

2. 开发工具安装及配置

(8) 调试窗口

Console窗口：显示当前页输出的日志信息等



开发环境搭建

2. 开发工具安装及配置

(8) 调试窗口

Sources窗口：Web站点的资源列表及javascript代码的Debug。分为资源列表，脚本代码和调试工具三个部分。

The screenshot shows the Chrome DevTools Sources tab. On the left, there's a tree view of files under 'local-ntp'. The 'assert.js' file is selected. The main area displays the source code for 'assert.js', which includes comments and several functions. A red dashed box highlights the first few lines of the code. To the right, a green sidebar contains various debugging tools: Watch (empty), Call Stack (labeled 'Not paused'), Scope (labeled 'Not paused'), Breakpoints (labeled 'No breakpoints'), XHR/fetch Breakpoints, DOM Breakpoints, Global Listeners, and Event Listener Breakpoints.

```
// Copyright (c) 2013 The Chromium Authors. All rights reserved.  
// Use of this source code is governed by a BSD-style license.  
// See LICENSE file.  
  
/**  
 * @fileoverview Assertion support.  
 */  
  
/**  
 * Verify |condition| is truthy and return |condition| if true.  
 * @template T  
 * @param {T} condition A condition to check for truthiness.  
 *   This may be used to test whether a value is defined or  
 *   to force a cast to Boolean.  
 * @param {string=} opt_message A message to show on failure.  
 * @return {T} A non-null |condition|. If |condition| is falsy,  
 *   this closure primitive asserts that it is truthy.  
 * @suppress {reportUnknownTypes} because T is not sufficient.  
 */  
/* #export */ function assert(condition, opt_message) {  
  if (!condition) {  
    let message = 'Assertion failed';  
    if (opt_message) {  
      message = message + ': ' + opt_message;  
    }  
    const error = new Error(message);  
    const global = function() {  
      const thisOrSelf = this || self;  
      /* @type {boolean} */  
      thisOrSelf.traceAssertionsForTesting;  
      return thisOrSelf;  
    };  
  }  
}
```

开发环境搭建

3. Vue的使用

(1) script引用

直接访问如下网址：<https://cn.vuejs.org/v2/guide/installation.html>

选择相应Vue.js版本下载（开发选择开发版本）

开发版本

包含完整的警告和调试模式

生产版本

删除了警告， 33.30KB min+gzip

将下载的Vue.js文件放置于项目相应文件夹内，按照如下格式引用

```
<script src=“文件路径/vue.js” ></script>
```

开发环境搭建

3. Vue的使用

(2) CDN引用

使用 CDN 的方式引入Vue无须下载，更加方便、快捷。需注意选择就近的节点，否则会产生实时性不好，服务器端和缓存器可能会存在不同步的情况。如下为几个较稳定的CDN。

<https://cdn.staticfile.org/vue/2.2.2/vue.min.js>

<https://unpkg.com/vue/dist/vue.js>

<https://cdnjs.cloudflare.com/ajax/libs/vue/2.1.8/vue.min.js>

<https://cdn.jsdelivr.net/npm/vue/dist/vue.js>

将CDN地址做为script标签中src的属性即可

`<script src= "CDN地址" ></script>`

小结

通过本章节的学习，我们能够正确搭建需要的开发环境，保证后续学习的顺利进行。本节需要掌握的内容如下。

- 1.了解什么是移动应用开发
- 2.了解移动应用开发技术的分类及各种技术的优缺点。
- 3.熟悉本课程所采用的开发技术。
- 4.代码编辑工具VS code及相应插件的安装。
- 5.Node.js的安装及npm命令的使用。
- 6.Chrome（浏览器）插件的安装及调试。
- 7.Vue.js框架的引用。