

Web开发实训

第2讲：Web开发技术

冯顺磊

18337149582 / fengshunlei@51xueweb.cn

2018.08

本讲主要内容

- 代码托管
- 开源体系
- 静态开发
- 响应式网站
- 移动Web

1.代码托管

- 代码托管主要有以下方面功能：
 - 版本控制：每一次改动是一个版本，在必要时可以迅速、准确地取出相应的版本。
 - 灵活：对于大型项目，可以根据需要从云端复制部分代码到本地，开发时间不受时间、地域的限制。
 - 备份：将代码进行托管，同时也是对代码进行备份，是项目安全的一个保障。
 - 并行开发：允许多个团队同时开发一个应用程序的多个版本，从而提高了整体的效率。

1.代码托管

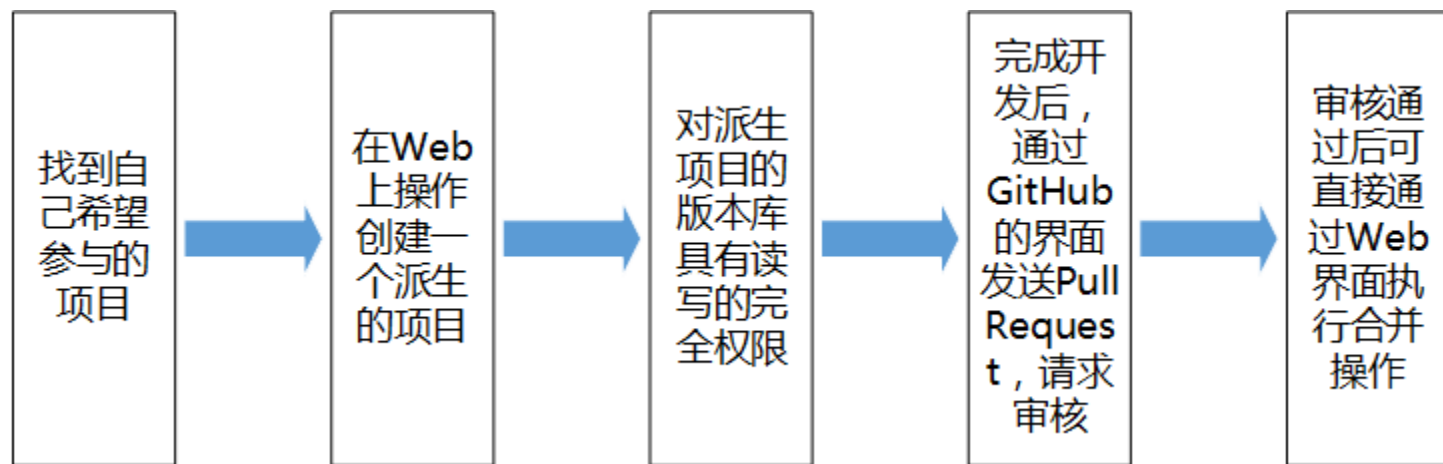
□ 简介

- GitHub是面向开源及私有软件项目的托管平台。GitHub简化了版本控制的管理和操作流程，为开发者提供了更好的交流平台。
- 使用GitHub使创建项目变得非常轻松，创建者只需在GitHub上点击一下鼠标即可创建一个新版本库。

GitHub

1.代码托管

- GitHub的操作过程为：



1.代码托管

□ 主要特点

- 对Git的完整支持：相比其他开源项目托管平台，GitHub对Git版本库提供了完整的协议支持，支持HTTP协议、Git-daemon、SSH协议。
- 在线编辑文件：GitHub提供了在线编辑文件的功能，不熟悉Git的用户也可以直接通过浏览器修改版本库里的文件。
- 社交编程：将社交网络引入项目托管平台，用户可以关注项目、关注其它用户进而了解项目和开发者动态。

1.代码托管

□ 简介

- SVN (Subversion) , 是一个开放源代码的版本控制系统, 用于团队开发中的多人文档操作的更新、处理和合并。
- SVN能够跨平台使用, 支持大多数常见的操作系统。



1.代码托管

□ 主要特点

- 统一的版本号：任何一次提交都会对所有文件增加到同一个新版本号，即使是提交并不涉及的文件。
- 原子提交：进行一系列相关的更改后，可以选择要提交的内容。
- 一致的数据操作：Subversion用二进制差异算法描述文件的变化，对于文本（可读）和二进制（不可读）文件其操作方式是一致的。
- 高效的分支和标签操作：在Subversion中，分支与标签操作的开销与工程的大小无关。Subversion的分支和标签操作的作用只是一种类似于硬链接的机制拷贝整个工程。

2. 静态开发

□ 善用浏览器调试模式

The screenshot displays the Chrome Developer Tools interface for the URL `https://www.baidu.com/`. The **Elements** panel on the left shows the DOM tree structure:

```

<!DOCTYPE html>
<!--STATUS OK-->
<html>
  <head>...</head>
  <body link="#0000cc" style=
    <script>...</script>
    <div id="wrapper" style="display: block;">
      <script>...</script>
      <div id="head">
        <div class="head_wrapper">
          <div class="s_form">
            <div class="s_form_wrapper soutu-env-nomac soutu-env-index">
              <div id="lg">...</div>
              <a href="#" id="result_logo" onmousedown="return c...">...</a>
              <form id="form" name="f" action="/s" class="fm">...</form>
              <div id="m">...</div>
            </div>
          </div>
        </div>
      </div>
    </div id="u"> == $0
  </body>
</html>
  
```

The **Styles** panel on the right shows the computed styles for the selected element:

```

element.style {
}
#u {
  z-index: 301;
  position: absolute;
  right: 0;
  top: 0;
  margin: 21px 9px 5px 0;
  padding: 0;
}
#u {
  display: none;
}
div {
  user agent stylesheet
  display: block;
}
  
```

The **Console** panel at the bottom shows a warning message:

```

A Parser-blocking, cross site (i.e. different eTLD+1) script, https://ssl.bdstatic.com/5eN1big8AAUym2zgoY3K/r/ww... (index):714
w/cache/static/protocol/https/global/js/all_async_search_f5b321b.js, is invoked via document.write. The network request for
this script MAY be blocked by the browser in this or a future page load due to poor network connectivity. If blocked in this
page load, it will be confirmed in a subsequent console message. See https://www.chromestatus.com/feature/5718547946799104 for
more details.
  
```

Below the warning, there is a recruitment notice in Chinese:

一张网页，要经历怎样的过程，才能抵达用户面前？
 一位新人，要经历怎样的成长，才能站在技术之巅？
 探寻这里的秘密；
 体验这里的挑战；
 成为这里的主人；
 加入百度，加入网页搜索，你可以影响世界。

请将简历发送至 `ps_recruiter@baidu.com` (邮件标题请以“姓名-应聘职位-来自console”命名)
 职位介绍: `http://dwz.cn/hr2013`

2.静态开发

- 尽量合并小图片



2. 静态开发

- ❑ 重置浏览器样式，减少默认
- ❑ 样式可能带来的问题

```
@charset "utf-8";
/*!
 * @名称: base.css
 * @功能: 1、重置浏览器默认样式
 *        2、设置通用原子类
 */
/* 防止用户自定义背景颜色对网页的影响，添加让用户可以自定义字体 */
html {
    background: #white;
    color: #black;
}
/* 内外边距通常让各个浏览器样式的表现位置不同 */
body,div,d1,dt,dd,u1,ol,li,h1,h2,h3,h4,h5,h6,pre,code,form,fieldset,legend,input,textarea,p,blockquote,th,td,hr,br,
margin:0;
padding:0;
}
/* 要注意表单元素并不继承父级 font 的问题 */
body,button,input,select,textarea {
    font:12px \5b8b\4f53,arial,sans-serif;
    font-family:"Microsoft YaHei";
}
input,select,textarea {
    font-size:100%;
}
/* 去掉 table cell 的边距并让其边重合 */
table {
    border-collapse:collapse;
    border-spacing:0;
}
/* ie bug: th 不继承 text-align */
th {
    text-align:inherit;
}
/* 去除默认边框 */
fieldset,img {
    border:none;
}
/* ie6 7 8(q) bug 显示为行内表现 */
iframe {
    display:block;
}
```

2.静态开发

- 合理的标签使用、清晰的代码结构、良好的注释、语义准确的类名称与ID

```
var modeText = ['顺序播放', '单曲循环', '随机播放', '列表循环'];
var player = new MPlayer( {
    // 容器选择器名称
    containerSelector: '.mp',
    // 播放列表
    songList: mplayer_song,
    // 专辑图片错误时显示的图片
    defaultImg: 'img/mplayer_error.png',
    // 自动播放
    autoPlay: true,
    // 播放模式(0->顺序播放,1->单曲循环,2->随机播放,3->列表循环(默认))
    playMode: 0,
    playList: 0,
    playSong: 0,
    // 当前歌词距离顶部的距离
    lrcTopPos: 34,
    // 列表模板,用${变量名}$插入模板变量
    listFormat: '<tr><td>${name}$</td><td>${singer}$</td><td>${time}$</td></tr>',
    // 音量滑块改变事件名称
    volSlideEventName: 'change',
    // 初始音量
    defaultVolume: 80
}
```

2.静态开发

- 尽量重用CSS

base.css+common.css+page.css

base提供最底层的、功能和粒度最小的的通用类样式，可以被所有页面引用，力求精简和通用

common则是模块化组件，即将功能相对独立的划分为模块，重复次数多的模块划分为组件，提取到common层，这一层需要尽可能实现封装，对可能经常变化的部分提供灵活的接口

page是页面级别的样式表，提供各页面独有的样式，可以再根据需求细分如page-index.css，page-connect.css

3. 响应式网站

4.1 移动网站

- Bootstrap 是最受欢迎的 HTML、CSS 和 JS 框架，用于开发响应式布局、移动设备优先的 WEB 项目。



3. 响应式网站

4.1 移动网站

下载压缩包之后，将其解压缩到任意目录即可看到以下（压缩版的）目录结构：

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap.min.css.map
│   ├── bootstrap-theme.css
│   ├── bootstrap-theme.css.map
│   ├── bootstrap-theme.min.css
│   └── bootstrap-theme.min.css.map
├── js/
│   ├── bootstrap.js
│   └── bootstrap.min.js
└── fonts/
    ├── glyphicons-halflings-regular.eot
    ├── glyphicons-halflings-regular.svg
    ├── glyphicons-halflings-regular.ttf
    ├── glyphicons-halflings-regular.woff
    └── glyphicons-halflings-regular.woff2
```

上面展示的就是 Bootstrap 的基本文件结构：预编译文件可以直接使用到任何 web 项目中。我们提供了编译好的 CSS 和 JS (`bootstrap.*`) 文件，还有经过压缩的 CSS 和 JS (`bootstrap.min.*`) 文件。同时还提供了 CSS 源码映射表 (`bootstrap.*.map`)，可以在某些浏览器的开发工具中使用。另外还包含了字体文件 `glyphicons-halflings-regular`，在附带的 `glyphicons-glyphicons` 主题中使用到了这些图标。

3. 响应式网站

4.1 移动网站

- 栅格系统
- Bootstrap 提供了一套响应式、移动设备优先的流式栅格系统，随着屏幕或视口（viewport）尺寸的增加，系统会自动分为最多12列。

简介

栅格系统用于通过一系列的行 (row) 与列 (column) 的组合来创建页面布局, 你的内容就可以放入这些创建好的布局中。下面就介绍一下 Bootstrap 栅格系统的工作原理:

- “行 (row)”必须包含在 `.container` (固定宽度) 或 `.container-fluid` (100% 宽度) 中, 以便为其赋予合适的排列 (alignment) 和内补 (padding)。
- 通过“行 (row)”在水平方向创建一组“列 (column)”。
- 你的内容应当放置于“列 (column)”内, 并且, 只有“列 (column)”可以作为行 (row)”的直接子元素。
- 类似 `.row` 和 `.col-xs-4` 这种预定义的类, 可以用来快速创建栅格布局。Bootstrap 源码中定义的 `mixin` 也可以用来创建语义化的布局。
- 通过为“列 (column)”设置 `padding` 属性, 从而创建列与列之间的间隔 (gutter)。通过为 `.row` 元素设置负值 `margin` 从而抵消掉为 `.container` 元素设置的 `padding`, 也就间接为“行 (row)”所包含的“列 (column)”抵消掉了 `padding`。
- 负值的 `margin`就是下面的示例为什么是向外突出的原因。在栅格列中的内容排成一行。
- 栅格系统中的列是通过指定1到12的值来表示其跨越的范围。例如, 三个等宽的列可以使用三个 `.col-xs-4` 来创建。
- 如果一“行 (row)”中包含的“列 (column)”大于 12, 多余的“列 (column)”所在的元素将被作为一个整体另起一行排列。
- 栅格类适用于与屏幕宽度大于或等于分界点大小的设备, 并且针对小屏幕设备覆盖栅格类。因此, 在元素上应用任何 `.col-md-*` 栅格类适用于与屏幕宽度大于或等于分界点大小的设备, 并且针对小屏幕设备覆盖栅格类。因此, 在元素上应用任何 `.col-lg-*` 不存在, 也影响大屏幕设备。

媒体查询

在栅格系统中，我们在 Less 文件中使用以下媒体查询（media query）来创建关键的分界点阈值。

```
/* 超小屏幕（手机，小于 768px） */  
/* 没有任何媒体查询相关的代码，因为这在 Bootstrap 中是默认的（还记得 Bootstrap 是移动设备优先的吗？） */  
  
/* 小屏幕（平板，大于等于 768px） */  
@media (min-width: @screen-sm-min) { ... }  
  
/* 中等屏幕（桌面显示器，大于等于 992px） */  
@media (min-width: @screen-md-min) { ... }  
  
/* 大屏幕（大桌面显示器，大于等于 1200px） */  
@media (min-width: @screen-lg-min) { ... }
```

我们偶尔也会在媒体查询代码中包含 `max-width` 从而将 CSS 的影响限制在更小范围的屏幕大小之内。

```
@media (max-width: @screen-xs-max) { ... }  
@media (min-width: @screen-sm-min) and (max-width: @screen-sm-max) { ... }  
@media (min-width: @screen-md-min) and (max-width: @screen-md-max) { ... }  
@media (min-width: @screen-lg-min) { ... }
```

栅格参数

通过下表可以详细查看 Bootstrap 的栅格系统是如何在多种屏幕设备上工作的。

	超小屏幕 手机 ($<768\text{px}$)	小屏幕 平板 ($\geq 768\text{px}$)	中等屏幕 桌面显示器 ($\geq 992\text{px}$)	大屏幕 大桌面显示器 ($\geq 1200\text{px}$)
栅格系统行为	总是水平排列	开始是堆叠在一起的, 当大于这些阈值时将变为水平排列C		
<code>.container</code> 最大宽度	None (自动)	750px	970px	1170px
类前缀	<code>.col-xs-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>
列 (column) 数	12			
最大列 (column) 宽	自动	~62px	~81px	~97px
槽 (gutter) 宽	30px (每列左右均有 15px)			
可嵌套	是			
偏移 (Offsets)	是			
列排序	是			

响应式工具

3.1

为了加快对移动设备友好的页面开发工作，利用媒体查询功能并使用这些工具类可以方便的针对不同设备展示或隐藏页面内容。另外还包含了针对打印机显示或隐藏内容的工具类。

有针对性的使用这类工具类，从而避免为同一个网站创建完全不同的版本。相反，通过使用这些工具类可以在不同设备上提供不同的展现形式。

可用的类

通过单独或联合使用以下列出的类，可以针对不同屏幕尺寸隐藏或显示页面内容。

	超小屏幕 手机 (<768px)	小屏幕 平板 (≥768px)	中等屏幕 桌面 (≥992px)	大屏幕 桌面 (≥1200px)
<code>.visible-xs-*</code>	可见	隐藏	隐藏	隐藏
<code>.visible-sm-*</code>	隐藏	可见	隐藏	隐藏
<code>.visible-md-*</code>	隐藏	隐藏	可见	隐藏
<code>.visible-lg-*</code>	隐藏	隐藏	隐藏	可见
<code>.hidden-xs</code>	隐藏	可见	可见	可见
<code>.hidden-sm</code>	可见	隐藏	可见	可见
<code>.hidden-md</code>	可见	可见	隐藏	可见
<code>.hidden-lg</code>	可见	可见	可见	隐藏

4.移动Web

4.1移动网站

- 可以按照表现方式的不同将移动Web分为移动网页和移动APP两种。



4.移动Web

4.1移动网站

- 移动网页：在移动端表现良好的网页。
- 移动端的网页开发与PC端的网页开发有着什么样的区别？

4.移动Web

4.1移动网站

- 移动端的浏览器兼容性问题少之又少。
- 移动端有着各种各样的手势操作。
- 移动端存在着不同手机的适配问题。

4.移动Web

4.1移动网站

□ 移动APP

- 先了解下三种移动端的APP

- **Native App原生app手机应用程序**

使用原生的语言开发的手机应用，Android系统用的是java或者是新的Kotlin，iOS系统用的是object-C或者是新的Swift。

- **Hybrid App 混合型app手机应用程序**

也有另外的一个名字叫做【套壳开发】，混合使用原生的程序和html5页面开发的手机应用。

- **Web App 基于Web的app手机应用程序**

完全使用html5页面加前端JS框架开发的手机应用。

4.移动Web

4.1移动网站

- Web APP
 - apple-mobile-web-app-capable

```
<!DOCTYPE html>
<html lang="zh_CN" data-dpr="2.625" style="font-size: 57.2222px;">
  <head>
    <meta charset="utf-8">
    <title>小米商城-小米官方网站, 小米手机、红米手机正品专卖</title>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,initial-scale=1,minimum-scale=1,maximum-scale=1,user-scalable=0,minim
    <meta data-hid="description" data-vue-meta="true" name="description" content="小米商城直营小米公司旗下所有产品, 囊括小米手机
    及售后支持。">
    <meta data-hid="keywords" data-vue-meta="true" name="keywords" content="小米,小米官网,小米手机,小米网首页,小米商城">
    <link rel="manifest" href="/static/manifest.json">
    <meta name="apple-mobile-web-app-capable" content="yes" == $0
    <meta name="apple-mobile-web-app-status-bar-style" content="black">
    <meta name="apple-mobile-web-app-title" content="m-mi-com">
    <link rel="apple-touch-icon" href="/static/img/icons/apple-touch-icon-152x152.png">
    <meta name="msapplication-TileImage" content="/static/img/icons/msapplication-icon-144x144.png">
    <meta name="msapplication-TileColor" content="#000000">
```

4.移动Web

4.1移动网站

□ 什么是viewport?

- 手机浏览器是把页面放在一个虚拟的“窗口”（viewport）中，通常这个虚拟的“窗口”（viewport）比屏幕宽，这样就不用把每个网页挤到很小的窗口中（这样会破坏没有针对手机浏览器优化的网页的布局），用户可以通过平移和缩放来看网页的不同部分。

- 在这个里边有三个层次的viewport

layout viewport visual viewport ideal viewport

- `<meta name="viewport">` 设置的是layout viewport

- <https://www.cnblogs.com/exhausted/p/5838090.html>



4.移动Web

4.1移动网站

□ Viewport 的设置?

- 一个常用的针对移动网页优化过的页面的 viewport meta 标签大致如下:

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">
```

- width: 控制 viewport 的大小, 可以指定的一个值, 如果 600, 或者特殊的值, 如 device-width 为设备的宽度 (单位为缩放为 100% 时的 CSS 的像素)。
- height: 和 width 相对应, 指定高度。
- initial-scale: 初始缩放比例, 也即是当页面第一次 load 的时候缩放比例。
- maximum-scale: 允许用户缩放到的最大比例。
- minimum-scale: 允许用户缩放到的最小比例。
- user-scalable: 用户是否可以手动缩放。

4.移动Web

4.1移动网站

□ 理解px、em、rem三种前端常用尺寸

- px: 将显示器分成非常细小的方格, 每个方格就是一像素。
 - em: 继承父级元素的字体大小
 - rem: 继承根元素的字体大小, 即为HTML的字体大小, 为CSS3新增的属性。
-
- http://tgideas.qq.com/webplat/info/news_version3/804/7104/7106/m5723/201509/376281.shtml

4.移动Web

4.1移动网站

- 理解@media{}
 - 使用 @media 查询，你可以针对不同的媒体类型定义不同的样式。
 - @media 可以针对不同的屏幕尺寸设置不同的样式，特别是如果你需要设置设计响应式的页面，@media 是非常有用的。
 - 当你重置浏览器大小的过程中，页面也会根据浏览器的宽度和高度重新渲染页面。

4.移动Web

4.1移动网站

- 移动前端的布局
 - 基于rem实现的布局

```
<script>  
  var html = document.getElementsByTagName( 'html' )[0];  
  var pageWidth = html.getBoundingClientRect().width;  
  html.style.fontSize = pageWidth / 50 + 'px';  
</script>
```

4.移动Web

4.1移动网站

- 移动前端的布局
 - 基于@media{}的布局

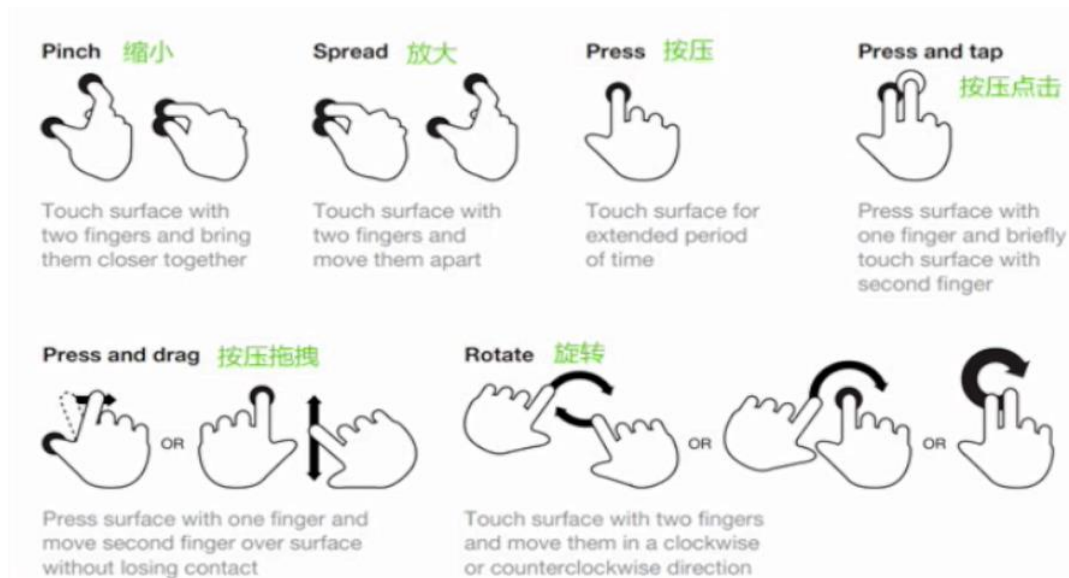
```
@media (max-width:960px){  
  .menu li{  
    width:48%;  
    margin:10px 1%;  
  }  
}  
  
@media (max-width:760px){  
  .menu li{  
    width:99%;  
    margin:10px 1%;  
  }  
}
```

<https://www.w3cplus.com/mobile/lib-flexible-for-html5-layout.html>

4.移动Web

4.1移动网站

□ 手势



4.移动Web

4.1移动网站

- 移动端的手势基于JS的Touch事件实现
 - TouchEvent
 - 代表当触摸行为在平面上变化的时候发生的事件。
 - touchstart : 触摸开始 (手指放在触摸屏上)
 - touchmove : 拖动 (手指在触摸屏上移动)
 - touchend : 触摸结束 (手指从触摸屏上移开)
 - touchenter : 移动的手指进入一个dom元素。
 - Touch
 - 代表用户手指与触摸平面间的一个接触点。
 - TouchList
 - 代表一系列的Touch; 一般在用户多个手指同时接触触控平面时使用这个接口。
 - DocumentTouch
 - 包含了一些创建 Touch对象与TouchList对象的便捷方法。

4.移动Web

4.1移动网站

- Hammer.js
- 官网: <http://hammerjs.github.io/>.

- Swiper
- 中文网: <http://www.swiper.com.cn/>

4.移动Web

4.1移动网站



4.移动Web

4.2微信小程序

- 微信小程序是一种全新的连接用户与服务的方式，它可以在微信内被便捷地获取和传播，同时具有出色的使用体验。



小程序

一种新的开放能力，可以在微信内被便捷地获取和传播，同时具有出色的使用体验。

4.移动Web

4.2微信小程序

- 微信小程序演示
 - 账号申请
 - 小程序的平台
 - 小程序的目录结构
 - 小程序的框架介绍
 - 小程序的组件介绍
 - 小程序开发演示
 - 新建Page
 - 数据绑定
 - 列表
 - 条件
 - tabBar
 - setData
 - navigateTo

Thanks.