

河南中医药大学信息技术学院（智能医疗行业学院）智能医学工程专业《互联网医疗服务开发》课程

第01章：Web概述

冯顺磊

河南中医药大学信息技术学院（智能医疗行业学院）
河南中医药大学信息技术学院智能医疗教研室
<https://aitcm.hactcm.edu.cn>
2025/2/20

本章概要

- Web基础
- 工作方式
- 设计工具
- 开发工具
- 调试工具
- 扩展1：Web最佳实践
- 扩展2：什么是原型设计？



[illegible]

The diagram illustrates the evolution of network architectures across three stages:

- Web 1.0:** Shows a central 'CONTENT' box. It is connected to two 'Consumers' (top) and two 'Producers' (bottom). The connections are unidirectional: Producers send content to the central box, and Consumers receive content from the central box.
- Web 2.0:** Shows a central 'CONTENT' box. It is connected to two 'Consumers' (top) and two 'Producers' (bottom). The connections are bidirectional, allowing both Consumers and Producers to interact with the central content.
- Web 3.0:** Shows a central 'CONTENT' box. It is connected to six 'Consumers' (top) and six 'Producers' (bottom). The connections are bidirectional, and the network is more complex, with additional connections between the Consumers and Producers themselves, suggesting a decentralized or peer-to-peer structure.

1. Web基础： 1.2 发展阶段



Web1.0

用户基本是被动地接受互联网中的内容，很少能参与到互联网建设中

Web2.0

用户可以自主创建互联网中的内容，但流量入口与利益分配等却被各个互联网巨头公司把控，且隐私与安全问题难以保障

Web3.0

将会是一个更加开放，公平和安全的网络

1. Web基础： 1.3 基本组成

● 客户端

- 客户端主要是指用户使用的浏览器，如Chrome、Firefox、Safari等。浏览器的作用是向服务器发送请求，获取网页资源（HTML文件、CSS文件、JavaScript文件等），然后解析这些资源，将其渲染成用户可以看到的网页界面。

● 服务器端

- 服务器是托管网页资源的计算机。当客户端（浏览器）发送请求时，服务器会根据请求的内容返回相应的网页资源。服务器端通常会运行一些服务软件，如Apache、Nginx等，并且可能结合后端编程语言（如PHP、Python、Java、C#等）来处理动态内容。



1. Web基础：1.3 基本组成

- 通信协议：HTTP

- HTTP (HyperText Transfer Protocol, 超文本传输协议)，详细规定了浏览器和万维网服务器之间互相通信的规则，通过因特网传送万维网文档的数据传送协议。HTTP是互联网上应用最为广泛的一种网络协议。所有的Web文件都必须遵守这个标准。设计HTTP最初的目的是提供一种发布和接收HTML页面的方法。
- HTTP的发展是万维网协会 (World Wide Web Consortium) 和Internet工作小组 (Internet Engineering Task Force) 合作的结果，万维网协会和Internet工作小组最终发布了一系列的RFC，其中RFC 2616定义了HTTP协议的普遍使用的一个版本——HTTP 1.1。
- 访问网页时，使用的就是HTTP协议。例如访问一个网站，在浏览器中输入网址后，浏览器会自动在地址前添加“http://”，这就说明访问该网址使用的是HTTP协议。



1. Web基础：1.3 基本组成

- 通信协议：HTTPS

- HTTPS (HyperText Transfer Protocol over Secure Socket Layer)，是以安全为目标的HTTP通道，是安全的HTTP。在HTTP传输中加入SSL层，HTTPS的安全基础是SSL，因此加密详细内容就需要SSL，用于安全的HTTP数据传输。
- https://URL表明它使用了HTTP，但HTTPS存在不同于HTTP的默认端口及一个加密/身份验证层（在HTTP与TCP之间）。这个系统的最初研发由网景公司 (Netscape) 进行，并内置于其浏览器 Netscape Navigator中，提供了身份验证与加密通讯方法。现在它被广泛用于万维网上安全敏感的通讯。

1. Web基础： 1.3 基本组成

- 通信协议：FTP

- FTP (File Transfer Protocol, 文件传输协议)，是 TCP/IP 协议簇中的协议之一，是Internet网络上两台计算机传送文件的协议，也是在TCP/IP网络和Internet上最早使用的协议之一。FTP协议属于网络协议组的应用层。FTP客户机可以给服务器发出命令来下载文件、上传文件、创建或改变服务器上的目录等。
- FTP协议包括两个组成部分：FTP服务器、FTP客户端。FTP服务器用来存储文件，用户可以使用FTP客户端通过FTP协议访问位于FTP服务器上的资源。在开发网站的时候，通常利用FTP协议把网页或程序传到Web服务器上，以实现网站发布和网站更新。

1. Web基础： 1.4 学习内容

- HTML

- HTML（超文本标记语言）是一个网页的骨架，无论是静态网页还是动态网页，最终返回到浏览器端的都是HTML，浏览器将HTML代码解释渲染后呈现给用户。
- HTML并不是一种编程语言，它是一种标记语言，是用来识别和描述一个文件中各个组件的系统，比如标题、段落和列表，标记表示文档的底层结构。

```
<h1>这是一个顶级标题</h1>

<p>这是一个文本段落</p>


```

1. Web基础： 1.4 学习内容

- CSS

- CSS（层叠样式表）用来描述网页内容如何显示，是能够真正做到网页表现与内容分离的一种样式设计语言。页面的外观称为“展示”，字体、色彩、背景图片、行间距、页面布局等展示都是由CSS来控制的。

```
p {  
  color: red;  
}
```

1. Web基础： 1.4 学习内容

- JavaScript

- JavaScript是应用最为广泛的脚本语言，可以在网页中添加互动和行为，例如验证表单输入，以确保输入正确有效；更换一个元素或整个站点的风格；使浏览器记住有关用户的资料，方便下一次访问；构建界面窗口等。
- JavaScript是最常用的操作网页元素或者某些浏览器窗口功能的语言。使用JavaScript可以访问并控制的网页的网页元素的标准列表。例如通过JavaScript可以增加HTML标记、隐藏HTML区块等。

```
let pElem = document.querySelector("p");  
pElem.textContent = "我们改变了文本！";
```

1. Web基础： 1.4 学习内容

- TypeScript
 - TypeScript是由微软开发的一种开源编程语言，是JavaScript的一个超集，添加了静态类型系统和其他增强功能。TypeScript的主要目标是帮助开发者更高效地编写大型JavaScript应用程序，同时提供更好的代码可维护性和工具支持。

```
interface Student {
  id: number; // 学生ID
  name: string; // 学生姓名
  age: number; // 学生年龄
  scores: number[]; // 学生成绩数组
  getAverageScore(): number; // 计算平均成绩的方法
}
```

1. Web基础： 1.4 学习内容

- 程序编程语言
 - 目前大多数网站是动态网站，虽然Web前端工程师不需要进行大量的动态网站程序开发，但是却需要和程序开发人员进行配合和业务衔接，因此掌握一定的动态网站开发语言对于工作来讲非常必要。
 - Web前端工程师应该掌握一门动态网站开发语言，例如PHP、GO、Python、Java、C#等，能够具备一定的开发能力，能够理解动态网站开发语言的工作原理。

```
import com.sun.net.httpserver.HttpExchange;
import com.sun.net.httpserver.HttpHandler;
import com.sun.net.httpserver.HttpServer;

import java.io.IOException;
import java.io.OutputStream;
import java.net.InetSocketAddress;

// 定义一个处理 GET 请求的处理器
class MyHandler implements HttpHandler {
    @Override
    public void handle(HttpExchange exchange) throws IOException {
        // 当前请求方法为 GET
        if ("GET".equals(exchange.getRequestMethod())) {
            // 定义响应内容
            String response = "这是一个简单的 HTTP GET 请求响应!";
            // 设置响应头，状态码为 200 表示成功
            exchange.sendResponseHeaders(200, response.length());
            // 获取输出流用于发送响应内容
            OutputStream os = exchange.getResponseBody();
            // 将响应内容写入输出流
            os.write(response.getBytes());
            // 关闭输出流
            os.close();
        }
    }
}

public class SimpleHttpServer {
    public static void main(String[] args) throws IOException {
        // 定义一个 HttpServer 对象 (地址为本机 8080 端口)
        HttpServer server = HttpServer.create(new InetSocketAddress(8080), 0);
        // 注册处理器
        server.createContext("/", new MyHandler());
        // 设置线程执行器
        server.setExecutor(null);
        // 启动服务器
        server.start();
        System.out.println("服务器已启动，监听端口 8080...");
    }
}
```

1. Web基础：1.5 Web标准

- Web标准定义网页主要由三部分组成：结构（Structure）、表现（Presentation）和行为（Behavior）。
- 对应的标准也分三方面：
 - 结构化标准语言标准，主要包括HTML；
 - 表现标准语言标准，主要包括CSS；
 - 行为标准，主要包括对象模型（如W3C DOM）、ECMAScript等。
- 这些标准大部分由万维网联盟（W3C）起草和发布，也有一些是其他标准组织制订的标准，比如ECMA（European Computer Manufacturers Association）制定的ECMAScript标准。



官网地址: <https://www.w3.org/>
中文版本: <https://www.chinaw3c.org/>



标准地址: <https://tc39.es/ecma262/>
中文版本: <https://ecma262.com/2024/>

1. Web基础：1.5 Web标准

- 结构标准
 - 超文本标记语言（HTML） 是用于构建网页和Web应用程序的标准标记语言。HTML标准由W3C（万维网联盟）和WHATWG（Web超文本应用技术工作组）共同维护和更新。

版本	发布时间	说明
HTML 1.0	1993	最早的HTML标准，非常基础，主要用于简单的文档结构。
HTML 2.0	1995	引入了表单（<form>）、表格（<table>）等基本元素，支持简单的交互功能。
HTML 3.2	1997	引入了更多布局和样式功能，如框架（<frame>）、样式表（<style>）等。
HTML 4.01	1999	引入了CSS支持、JavaScript脚本、表单增强等，成为当时Web开发的主流标准。
XHTML 1.0	2000	基于HTML 4.01，要求使用XML的严格语法，所有标签必须闭合，属性值必须用引号包围。
HTML5	2014	引入了大量新特性，包括语义化标签（如<header>、<footer>、<article>等）、多媒体支持（<audio>、<video>）、表单增强、Canvas绘图、Web存储、Web Workers等。现代Web开发的主流标准，支持更丰富的交互和动态功能。
HTML Living Standard	-	持续更新的“活文档标准”，不断引入新特性，如Web组件、Shadow DOM、Intersection Observer等。现代Web开发中，HTML Living Standard是实际开发中遵循的主要标准。

1. Web基础：1.5 Web标准

- 表现标准
 - W3C创建CSS标准的目的是以CSS取代HTML表格式布局、帧和其他表现的语言。纯CSS布局与结构式HTML相结合能帮助设计师分离外观与结构，使站点的访问及维护更加容易。

版本	发布时间	说明
CSS1	1996	最初版本，引入了基本的样式规则，如字体、颜色、边距、填充等。主要用于简单的网页排版和样式设置。
CSS2	1998	引入了更复杂的布局控制，如定位（position）、浮动（float）、分页媒体（@page）、表格布局等。支持更复杂的页面布局和打印样式。
CSS2.1	2004	对CSS2进行了修正和澄清，解决了CSS2中的一些问题，增强了稳定性和兼容性。广泛应用于现代Web开发，是目前浏览器支持最稳定的CSS版本。
CSS3	2001年开始分模块发布（仍在持续更新）	引入了许多新特性，如： 选择器：伪类（:nth-child、:last-child等）、伪元素（::before、::after等）。 布局：Flexbox（弹性布局）、Grid（网格布局）。 视觉效果：渐变（linear-gradient）、阴影（box-shadow）、动画（@keyframes）、过渡（transition）。 响应式设计：媒体查询（@media）。 现代Web开发中广泛使用，支持更丰富的视觉效果和灵活的布局。

1. Web基础：1.5 Web标准

- 行为标准-文档对象模型（Document Object Model, DOM）
 - 根据W3C DOM规范（<http://www.w3.org/DOM/>）所定义，DOM是一种与浏览器与平台的语言接口，使得可以访问页面其他的标准组件。

版本	发布时间	说明
DOM Level 0	-	DOM Level 0 不是W3C规范，而是对早期浏览器（如Netscape Navigator 3.0和Microsoft Internet Explorer 3.0）中功能的描述。它提供了对HTML文档中某些元素（如表单和图像）的基本访问。
DOM Level 1	1998	定义了基本的文档结构和操作接口，包括元素、属性、文本等节点的操作。它是DOM标准的基础。
DOM Level 2	2000	增加了事件处理、样式操作、范围选择等功能。DOM Level 2进一步扩展了DOM的功能，使其能够更好地支持动态网页。
DOM Level 3	2004	进一步扩展了DOM的功能，包括XPath、键盘事件、序列化等。

2. 工作方式： 2.1 网页

- 网页是构成网站的基本元素，是承载各种网站应用的平台。
- 网页是一个文件，可以存放在某一台计算机中，是万维网中的一“页”。
- 网页经由网址（URL）来识别与存取，再通过浏览器解释，最后显示到用户眼前。
- 文字和图片是构成一个网页的最基本的元素。
- 文字是网页的内容，图片使网页表现美观，除此之外，网页的元素还包括动画、音视频、程序等。

```
<!DOCTYPE html>
<html><head>

<title>河南中医药大学</title><meta name="pageType" content="1">
<meta name="pageTitle" content="河南中医药大学">
<meta name="keywords" content="河南中医药大学" />

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta name="renderer" content="webkit">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="Description" content="">
<meta name="robots" content="all">
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0, user-scalable=no">
<meta name="format-detection" content="telephone=no">
<meta name="apple-mobile-web-app-capable" content="yes">
<meta name="apple-mobile-web-app-status-bar-style" content="black">
<meta name="Author" content="www.hactcm.edu.cn">
<meta name="Copyright" content="www.hactcm.edu.cn">
<link href="/favicon.ico" rel="icon" type="image/x-icon">
<link href="/css/mian.css" rel="stylesheet" type="text/css">
<link href="/css/socialize.css" rel="stylesheet" type="text/css">
<link href="/css/index.css" rel="stylesheet" type="text/css">
<link href="/style/blue.221018.css?v=3" rel="stylesheet" type="text/css">
<link href="/style/index_v.20241208.css?v=5" rel="stylesheet" type="text/css">
<link href="/style/center_main.css?v=1" rel="stylesheet" type="text/css">
<link href="/style/stopgap.css?v=6" rel="stylesheet" type="text/css">
</style>
</head>
<body>
<div class="navlist">
<div class="title">
<div class="close">
</div>
</div>
</div>
</body>
</html>
```

网页实际上只是一个纯文本文件。网页通过各式各样的标记对页面上的文字、图片、表格、音视频等元素进行描述，而浏览器通过对这些标记进行解释并生成页面，把网页通过一定的格式展现出来。在网页文件中存放的是图片的链接位置，而图片文件、网页文件是互相独立存放的，甚至可以不在同一台计算机上，因此在查看源文件时是无法看到图片、音视频、动画等内容。

河南中医药大学信息技术学院（智能医疗行业学院）智能医疗教

2. 工作方式： 2.1 网页

- 静态网页
 - 在网站设计中，HTML格式的网页通常被称为静态网页，静态网页是标准的HTML文件，它的文件扩展名是.htm、.html，可以包含文本、图像、声音、动画、客户端脚本及程序等。静态网页是网站建设的基础。静态网页是指没有后台数据库、不含开发程序且不可交互的网页。静态网页制作完成后，页面的内容和显示效果就确定了，除非修改页面代码，因此静态网页相对更新起来比较麻烦，适用于一般更新较少的展示型网站。
- 动态网页
 - 动态网页一般以数据库技术为基础，可以与后台数据库进行交互与数据传递，可以大大降低网站维护的工作量。采用动态网页技术的网站可以实现更多的功能，如用户注册、用户登录、在线调查、用户管理、订单管理等等。动态网页实际上并不是独立存在于服务器上的网页文件，只有当用户请求时服务器才返回一个完整的网页。

静态网页：



动态网页：



河南中医药大学信息技术学院（智能医疗行业学院）智能医疗教研室 / <https://aitcm.hactcm.edu.cn>

2025年2月20日星期四 第 20 页

2. 工作方式： 2.2 网站

- 定义一：网站（Website）是指在因特网上根据一定的规则，使用HTML等工具制作的用于展示特定内容相关网页的集合。
- 定义二：因特网上一块固定的面向全世界发布消息的地方，由域名（网站地址）和网站空间构成，通常包括主页和其他具有超链接文件的页面。
- 定义三：网站是一个逻辑上的概念，是由一系列的内容组合而成的。网站包含的内容有：网站的域名、提供网站服务的服务器或者网站空间、网页、网页内容所涉及的图片视频等文件、网页之间的关系。
- 通过上述三种定义不难看出，从不同的观测角度网站的定义也不同。



2. 工作方式： 2.3 网站与网页的关系

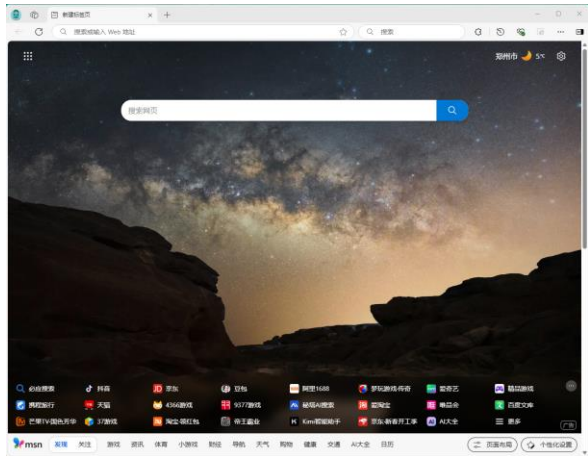
- 网站是一个整体，网页是一个个体，一个网站是由多个网页构建而成。即网站是由网页集合而成的，通过浏览器所看到的页面就是网页。
- 具体来说，网页是一个html文件，浏览器是用来解析这份文件的，网站是由许多html文件集合而成。
- 网站是有域名、网站存放空间的内容集合，网站所包含的内容有网页、程序、图片、视频、音频等内容和内容之间的链接关系，一个网站可能有很多网页，也可以只有一个网页。网页是网站内容的重要组成部分。

WEBPAGE VERSUS WEBSITE

WEBPAGE	WEBSITE
A single document on the internet	A collection of many related web pages linked together under one domain
Has content about a single entity	Has content about several entities
Adress of the URL of a webpage depends on the website	URL of the website does not depend on a webpage
Development requires a minimum amount of time	Development takes a long time
	Visit www.PEDIAA.com

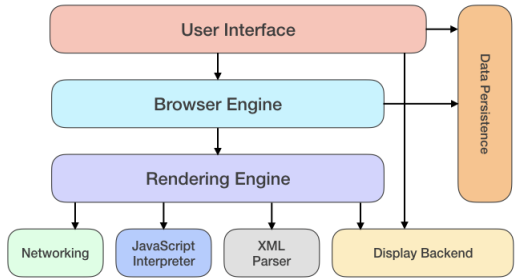
2. 工作方式： 2.4 浏览器的工作原理

- 浏览器的主要功能是将用户选择的Web资源呈现出来，它需要从服务器请求资源，并将其显示在浏览器窗口中，资源的格式通常是HTML，也包括PDF、image及其他格式。用户用URI（Uniform Resource Identifier统一资源标识符）来指定所请求资源的位置。



2. 工作方式： 2.4 浏览器的工作原理

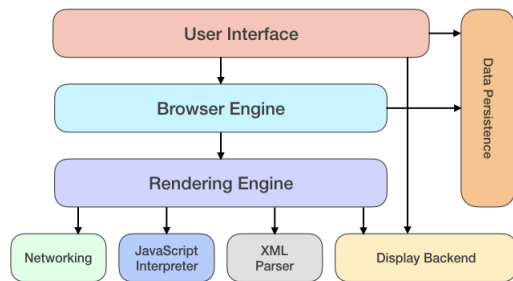
- 浏览器的主要组件包括7个方面。
 - 用户界面（User Interface）
 - 用户界面主要包括工具栏、地址栏、前进/后退按钮、书签菜单、可视化页面加载进度、智能下载处理、首选项、打印等。除了浏览器主窗口显示请求的页面之外，其他显示的部分都属于用户界面。
 - 用户界面还可与桌面环境集成，以提供浏览器会话管理或其他桌面应用程序的通信。
 - 浏览器引擎（Browser Engine）
 - 浏览器引擎是一个可嵌入的组件，其为渲染引擎提供高级接口。
 - 浏览器引擎可以加载一个给定的URI，并支持诸如：前进/后退/重新加载等浏览操作。
 - 浏览器引擎提供查看浏览会话的各个方面的挂钩，例如：当前页面加载进度、JavaScript alert。
 - 浏览器引擎还允许查询/修改渲染引擎设置。
 - 渲染引擎（Rendering Engine）
 - 渲染引擎为指定的URI生成可视化的表示。
 - 渲染引擎能够显示HTML和XML文档，可选择CSS样式，以及嵌入式内容（如图片）。
 - 渲染引擎能够准确计算页面布局，可使用“回流”算法逐步调整页面元素的位置。
 - 渲染引擎内部包含HTML解析器。



2. 工作方式： 2.4 浏览器的工作原理

• 浏览器的主要组件包括7个方面。

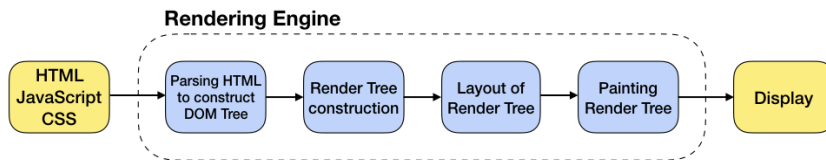
- 网络 (Networking)
 - 网络系统实现HTTP和FTP等文件传输协议。网络系统可以在不同的字符集之间进行转换，为文件解析MIME媒体类型。网络系统可以实现最近检索资源的缓存功能。
- JavaScript解释器 (JavaScript Interpreter)
 - JavaScript解释器能够解释并执行嵌入在网页中的JavaScript (又称ECMAScript) 代码。为了安全起见，浏览器引擎或渲染引擎可能会禁用某些JavaScript功能，如弹出窗口的打开。
- XML解析器 (XML Parser)
 - XML解析器可以将XML文档解析成文档对象模型 (Document Object Model, DOM) 树。XML解析器是浏览器架构中复用最多的子系统之一，几乎所有的浏览器实现都利用现有的XML解析器，而不是从头开始创建自己的XML解析器。
- 显示后端 (Display Backend)
 - 显示后端提供绘图和窗口原语，包括：用户界面控件集合、字体集合。
- 数据持久层 (Data Persistence)
 - 数据持久层将与浏览会话相关联的各种数据存储在硬盘上。这些数据可能是诸如：书签、工具栏设置等这样的高级数据，也可能是诸如：Cookie，安全证书、缓存等这样的低级数据。



2. 工作方式： 2.4 浏览器的工作原理

• 渲染引擎的工作原理

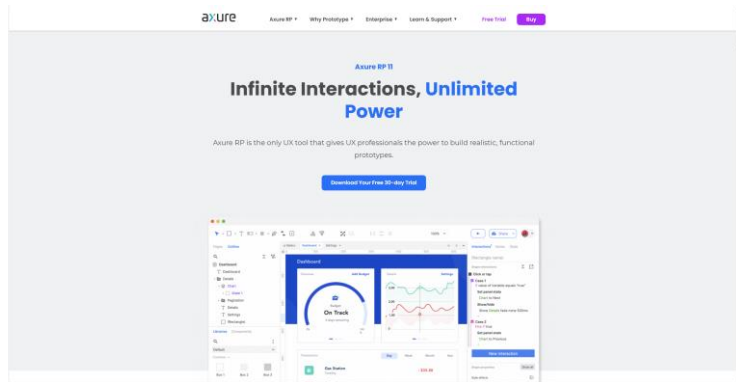
- Parsing HTML to Construct DOM Tree
 - 渲染引擎使用HTML解析器（调用XML解析器）解析HTML (XML) 文档，将各个HTML (XML) 元素逐个转化成DOM节点，从而生成DOM树。
 - 同时，渲染引擎使用CSS解析器解析外部CSS文件以及HTML (XML) 元素中的样式规则。元素中带有视觉指令的样式规则将用于下一步，以创建另一个树结构：渲染树。
- Render Tree construction
 - 渲染引擎使用第1步CSS解析器解析得到的样式规则，将其附着到DOM树上，从而构成渲染树。
 - 渲染树包含多个带有视觉属性（如颜色和尺寸）的矩形。这些矩形的排列顺序就是它们将在屏幕上显示的顺序。
- Layout of Render Tree
 - 渲染树构建完毕之后，进入本阶段进行“布局”，也就是为每个节点分配一个应出现在屏幕上的确切坐标。
- Painting Render Tree
 - 渲染引擎将遍历渲染树，并调用显示后端将每个节点绘制出来。



3. 设计工具： 3.1 Axure RP

● Axure RP

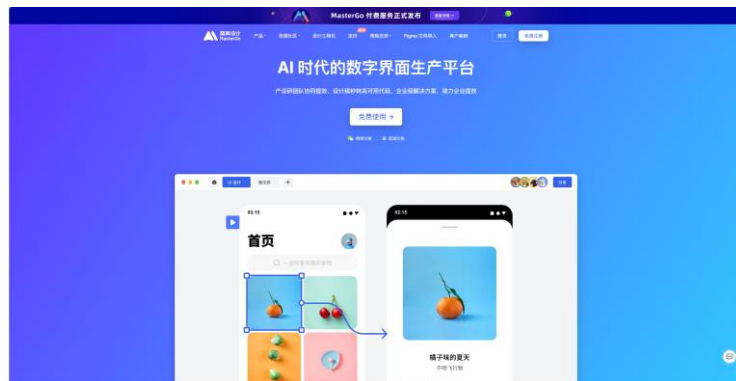
- Axure RP (Axure Rapid Prototyping) 是一款专业的原型设计工具，广泛应用于网页应用、移动应用和桌面软件的产品设计中。它帮助团队降低沟通成本，提高产品设计效率，是产品设计和开发流程中不可或缺的工具之一。Axure RP支持快速构建交互式原型并简化设计流程，能够在产品真正投入开发之前，对设计进行反复的测试和优化，从而减少开发过程中的返工和资源浪费。
- 官网地址：<https://www.axure.com/>



3. 设计工具： 3.2 MasterGo

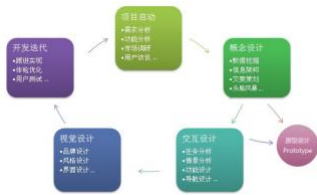
● MasterGo

- MasterGo 是一款面向团队的专业UI/UX设计工具，专为提升设计效率和团队协作而打造。作为AI时代的企业级产品设计平台，MasterGo贯穿产品设计研发的全链条，提供在线产品设计、原型图制作、网页开发设计、产品交互设计等功能。它支持多人实时协作，设计稿秒转高可用代码，帮助设计师、产品经理和工程师实现更简单灵活的工作模式。
- 官网地址：<https://mastergo.com/>



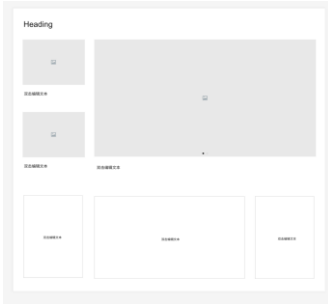
扩展：什么是原型设计？

- 在进行需求分析时，用户的口头描述和想法有时并不一致，在实际工作中用户对于图形化的沟通交流更容易理解。
- 随着互联网产品迭代的加快和用户需求复杂性的提高，通过原型的方式进行沟通，能够快速提高产品的设计效率，提高交流的准确性和便捷性。
- 原型设计是将页面的模块、元素、人机、交互的形式，利用线框描述的方法，将功能更加具体、生动地进行表达。
- 原型设计是交互设计师与客户、产品经理、网站开发工程师沟通的工具。
- 原型设计是定位于概念设计或整个设计流程初期的一个过程。随着项目大小、时间周期等，开发人员往往会根据需求确定纸原型、低保真原型、高保真原型等不同质量的内容作为输出。



扩展：什么是原型设计？

- 纸原型：顾名思义，就是画在文档纸、白板上的设计原型、示意图。便于修改和绘制，不便于保存和展示。因此想有效地利用纸原型，需要注意纸原型的承载。
- 低保真原型：通常是基于现有的界面或系统，通过电脑进行一些加工后的设计稿，示意更加明确，能够包含设计的交互和反馈，但在美观、效果等方面欠佳。可以理解为介于纸面原型和高保真原型之间输出的统称，往往也可以作为需求设计稿输出。
- 高保真原型：属于原型设计的终极武器。包括产品演示Demo或概念设计展示。视觉上与实际产品等效，体验上也与真实产品接近。





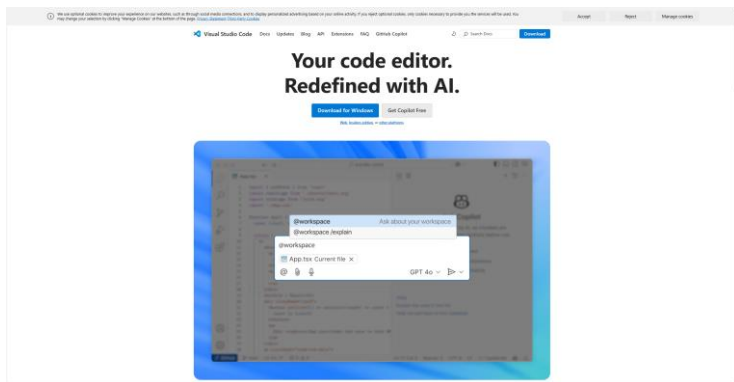
案例演示

- 访问MasterGo
- 绘制页面原型

4. 开发工具：4.1 Visual Studio Code

• Visual Studio Code

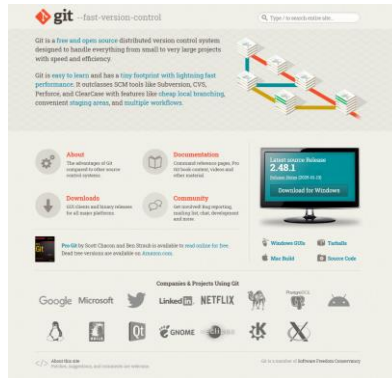
- Visual Studio Code（简称VS Code）是由微软开发的一款免费、开源的代码编辑器，适用于Windows、macOS和Linux操作系统。VS Code以其轻量级、高性能、丰富的功能和强大的插件生态系统而广受欢迎，适用于多种编程语言和开发场景，是现代开发者最常用的代码编辑器之一。
- 官网地址：<https://code.visualstudio.com/>



4. 开发工具： 4.2 Git

- Git

- Git 是一个开源的分布式版本控制系统，由Linux内核开发者Linus Torvalds于2005年创建。Git旨在提供高效、灵活且可靠的版本控制功能，帮助开发者更好地管理代码的变更历史、协作开发和项目管理。它是目前全球广泛使用的版本控制系统之一，广泛应用于软件开发、Web开发、数据科学等多个领域。
- 官网地址：<https://git-scm.com/>



5. 调试工具： 5.1 Microsoft Edge/Chrome

- 网站调试工具-Microsoft Edge/Chrome

- Microsoft Edge的网站调试工具旨在帮助开发者高效地调试、优化和测试网页及应用。Edge DevTools提供了从基础的元素检查到高级的性能分析和内存调试等多种功能，适用于前端开发和全栈开发。



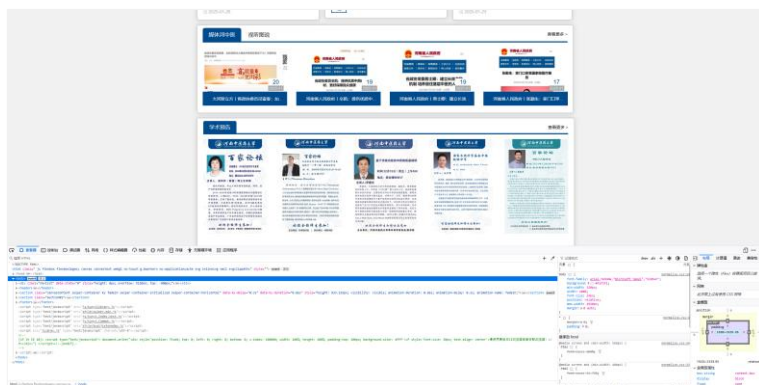


案例演示

- 打开调试工具
- 使用调试工具查看网页

5. 调试工具：5.1 Firefox

- 网站调试工具-Firefox
 - Firefox Developer Tools 是Mozilla Firefox浏览器内置的一套功能强大的开发者工具，旨在帮助开发者高效地调试、优化和测试网页及Web应用。与Chrome DevTools类似，Firefox Developer Tools提供了丰富的调试和开发功能，支持HTML、CSS、JavaScript调试，性能分析，网络监控等。

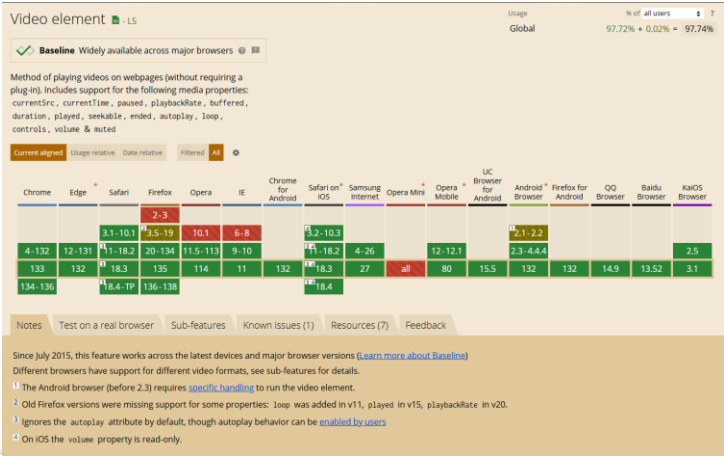


扩展：Web最佳实践

- 在 Web 开发过程中，存在诸多不确定性，主要源于实际情况中无法确切知晓每位用户会以何种技术条件访问网站。
 - 用户1 可能正通过屏幕较窄的手机浏览网站；
 - 用户2 也许使用连接宽屏显示器的电脑查看网站；
 - 用户3 可能是视障人士，依靠屏幕阅读器来“听”网页内容；
 - 用户4 可能还在使用老旧台式机，无法运行现代浏览器。
- 鉴于无法完全掌握用户的使用环境，就需要进行防御性设计，让网站具备尽可能高的灵活性，确保上述各类用户都能正常使用，即便他们获得的体验可能不尽相同。简言之，Web开发的最佳实践在于让 Web 服务于所有人。

扩展：Web最佳实践

- 跨浏览器兼容性（Cross - browser compatibility）
 - 是一种确保网页能在尽可能多设备上正常运行的实践方法。具体涵盖使用所有浏览器都支持的技术，为支持特定功能的浏览器提供更优质体验（渐进增强），以及编写代码，使老旧浏览器能回退到更简单但依然可用的体验（平稳降级）。此外，还需进行大量测试，排查某些浏览器可能出现的故障，并进一步修复。



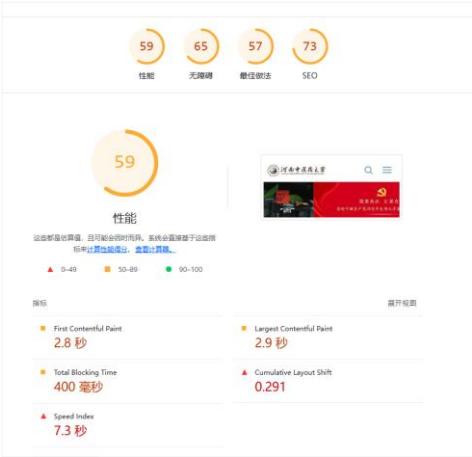
扩展：Web最佳实践

- 响应式 Web 设计（Responsive Web design）
 - 是一种让网站功能和布局具备灵活性的实践，使其能自动适配不同浏览器。典型的例子是，网站在桌面端宽屏浏览器中采用一种布局方式，而在手机浏览器中则切换为更紧凑的单列布局。你可以尝试调整浏览器窗口宽度，观察页面变化。



扩展：Web最佳实践

- 性能（Performance）
 - 不仅要求网站加载速度快，还需确保操作直观、使用便捷，避免用户因体验不佳而离开。



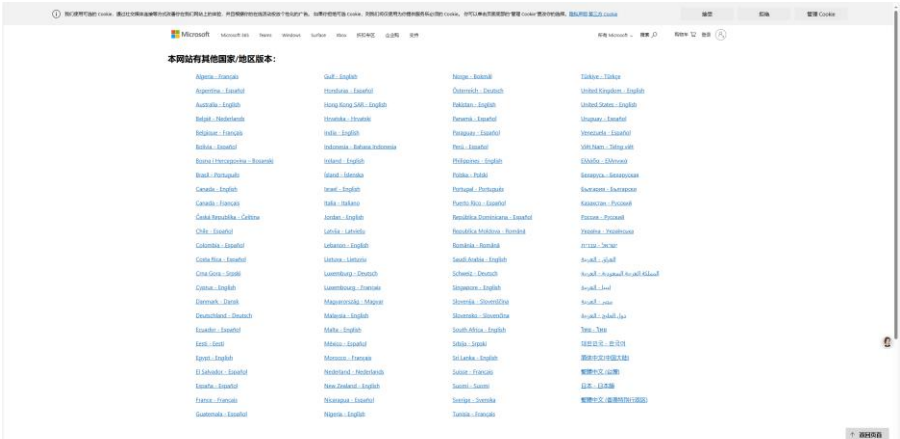
扩展：Web最佳实践

- 无障碍 (Accessibility)
 - 旨在让尽可能多不同类型的人都能使用网站 (与之相关的概念是多样性和包容性, 以及包容性设计)。其中包括视障、听障、认知障碍或肢体障碍人群。而且, 受益人群不仅限于残障人士, 还涵盖年轻人、老年人、不同文化背景的人、使用移动设备的人, 以及网络连接不稳定或速度较慢的用户。



扩展：Web最佳实践

- 国际化 (Internationalization)
 - 意味着让来自不同文化背景、使用不同语言的人都能使用网站。这既涉及技术手段 (比如, 调整布局, 使其适用于从右到左甚至垂直书写的语言), 也包含人为因素 (例如, 使用简洁、无俚语的表述, 以便将你的语言作为第二或第三语言的人更易理解文本内容)。



扩展：Web最佳实践

- 隐私与安全 (Privacy & Security)
 - 这两个概念紧密相关却又有所不同。隐私是指保障人们能私下进行活动，不被监视，也不收集非必要的的数据。安全则是指以安全的方式构建网站，防止恶意用户窃取你或用户的信息。



重点回顾

- Web基础
 - Web定义：Web是互联网上的一种服务，通过超文本和超媒体技术链接全球信息资源，为用户提供信息浏览和交互体验。
 - 发展阶段：
 - Web 1.0：以静态网页为主，用户被动浏览信息。
 - Web 2.0：用户可以与网页交互，如发布评论、上传内容，社交网络和博客兴起。
 - Web 3.0：强调去中心化、智能化和语义化，利用区块链和人工智能技术。
 - 基本组成：
 - 客户端：浏览器如Chrome、Firefox、Safari等。
 - 服务器端：托管网页资源的计算机，运行服务软件如Apache、Nginx。
 - 通信协议：HTTP、HTTPS、FTP等。
 - 学习内容：HTML、CSS、JavaScript、TypeScript、程序编程语言。
 - Web标准：结构、表现、行为三部分组成的标准集合。



重点回顾

- 工作方式
 - 网页：构成网站的基本元素，通过浏览器解释显示。
 - 网站：由多个网页构成，包含域名、服务器空间、网页内容等。
 - 网页与网站关系：网站由多个网页集合而成，网页是网站内容的重要组成部分。
 - 浏览器工作原理：包括用户界面、浏览器引擎、渲染引擎、网络、JavaScript解释器、XML解析器、显示后端、数据持久层等组件。
- 设计工具
 - Axure RP：专业的原型设计工具，支持快速构建交互式原型。
 - MasterGo：面向团队的专业UI/UX设计工具，支持多人实时协作。
- 开发工具
 - Visual Studio Code：免费、开源的代码编辑器，适用于多种编程语言和开发场景。
 - Git：开源的分布式版本控制系统，支持高效、灵活的版本控制。



重点回顾

- 调试工具
 - Microsoft Edge/Chrome：提供元素检查、控制台、网络、性能、内存、应用、安全、扩展、Lighthouse等调试功能。
 - Firefox：提供元素检查器、控制台、JavaScript调试器、网络监视器、性能工具、存储检查器、响应式设计模式、访问性检查、内存工具、应用面板等调试功能。
- Web最佳实践
 - 跨浏览器兼容性：确保网页能在多种设备上正常运行。
 - 响应式Web设计：网站功能和布局具备灵活性，自动适配不同浏览器。
 - 性能：网站加载速度快，操作直观、使用便捷。
 - 无障碍：让不同类型的用户都能使用网站。
 - 国际化：让不同文化背景、使用不同语言的用户都能使用网站。
 - 隐私与安全：保障用户隐私，构建安全的网站。



重点回顾

- 原型设计

- 原型设计概念：利用线框描述方法，将功能具体生动表达。
- 原型分类：
 - 纸原型：手绘设计原型，便于修改，不便于保存和展示。
 - 低保真原型：基于现有界面加工的设计稿，示意明确，但美观效果欠佳。
 - 高保真原型：视觉与实际产品等效，体验接近真实产品。

信创智能医疗系统研发课程体系

河南中医药大学信息技术学院（智能医疗行业学院）



河南中医药大学信息技术学院（智能医疗行业学院）智能医疗教研室

河南中医药大学医疗健康信息工程技术研究所