

Web前端开发

第2章：开发工具

阮晓龙

13938213680 / rxl@hactcm.edu.cn
<http://web.book.51xueweb.cn>
<http://www.51xueweb.cn>

河南中医药大学管理科学与工程学科

2018.5

本章主要内容

- 开发工具综述
- 原型设计工具
- 开发工具
- 调试工具
- 代码托管工具
- 项目管理系统



1. 开发工具综述

- Web前端开发工具根据开发的阶段和用途不同，可分为：
 - 原型设计工具
 - 开发工具
 - 调试工具
 - 代码托管工具



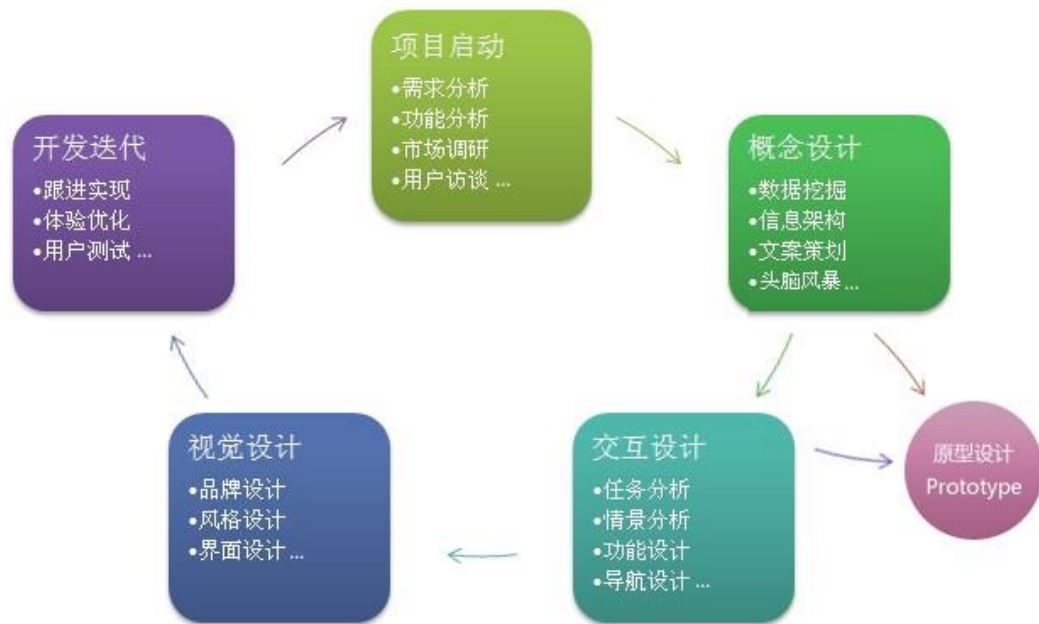
表 2-1 Web 前端开发常用工具表

开发阶段	使用工具
原型设计	Axure RP
	Microsoft Office Visio
技术开发	Adobe Dreamweaver
	Oracle Netbeans
	Microsoft Visual Studio Code
Web 调试	FireFox
	Google Chrome
	Internet Explorer
	Microsoft Edge
代码托管	GitHub
	SVN

2.原型设计工具

2.1什么是原型设计

- 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，将功能具体、生动地进行表达。
- 原型可划分为以下3种类型：
 - **纸原型**：画在文档纸、白板上的设计原型、示意图。特点为：便于修改和绘制，不便于保存和展示。
 - **低保真原型**：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。特点为：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。
 - **高保真原型**：产品演示Demo或概念设计展示。特点为：视觉上与实际产品等效，体验上与真实产品接近，但需要较多的投入，成本高。



原型设计的传统流程

2.原型设计工具

2.2原型设计工具-Axure RP

□ 简介

- Axure是一款专业的快速原型设计工具，可以通过组件的方式帮助网站或软件设计师快速建立带有注释的原型（流程图、线框图），并凭借自定义的可重用元件、动态面板以及丰富的脚本建立能够展示基本功能或页面逻辑的动态演示文件。
- Axure为用户提供了丰富的组件样式，通过该工具能够创建低保真、高保真甚至接近于实际效果的界面。



2.原型设计工具

2.2原型设计工具-Axure RP

□ 主要特点

- 快速建构原型
 - 拥有全套web控件库，直接拖拽即可快捷而简便的制作产品原型。
- 交互效果逼真，便于需求验证
 - 丰富的动态面板可以用来模拟各种复杂的交互效果，导出HTML后可以更加准确的传达信息架构和页面跳转。



现场演示：

- 在Windows 7系统平台上安装Axure RP 8。
- Axure RP软件界面和基本使用方法。
- 使用Axure RP实现百度登录页（或其他页面）的原型设计。

3. 开发工具

3.1 开发工具的作用

- 开发工具的作用：
 - 编写HTML、CSS和JavaScript程序。
 - 提高开发效率。



3.开发工具

3.2网站开发工具- Adobe Dreamweaver

□ 简介

- Adobe Dreamweaver是第一个针对专业网页设计师特别开发的视觉化网页开发工具，可用于设计并部署极具吸引力的网站和Web应用程序，利用Dreamweaver可以轻而易举地制作出跨越平台限制和跨越浏览器限制的网页，并提供强大的编码环境及基于标准的WYSIWYG（所见即所得）设计界面。



3.开发工具

3.2网站开发工具- Adobe Dreamweaver

□ 主要特点

- **最佳的制作效率**：Dreamweaver可以用最快速的方式将Fireworks, FreeHand或Photoshop等文件移至网页上。对菜单、快捷键与格式控制，只要一个简单步骤即可完成。
- **网站管理**：快速制作网站雏形、设计、更新和重组网页。
- **无可比拟的控制能力**：该软件是唯一提供Roundtrip HTML、视觉化编辑与原始码编辑同步的设计工具。
- **所见即所得**：整合动态式出版视觉编辑及电子商务功能，用户不需要透过浏览器就能预览网页。
- **样板和XML**：将内容与设计分开，应用于快速网页更新和团队合作网页编辑。



现场演示：

- 在Windows 7系统平台上的安装Adobe Dreamweaver CS6。
- Adobe Dreamweaver CS6的软件界面和基本操作方法。
- 创建站点。
- 创建网页。

3.开发工具

3.3网站开发工具- Oracle Netbeans

□ 简介

- NetBeans是一个为软件开发者设计，**自由、开放**的IDE（集成开发环境），可建立桌面应用、企业级应用、Web开发、Java移动应用程序开发、C/C++等。
- NetBeans帮助开发人员编写、编译、调试和部署应用，并将版本控制融入其中。
- NetBeans可以实现跨平台应用，包括Windows、Linux、Mac OS和Solaris等操作系统。



3.开发工具

3.3网站开发工具- Oracle Netbeans

□ 主要特点

- **功能全面的Web应用开发环境**：开发者可通过页面检查、CSS 样式编辑器和JavaScript 编辑器、调试器等工具来提升开发效率。
- **NetBeansHTTP监视器**：开发者可以通过HTTP监视器来监视请求、HTTP Header、cookies、会话、servlet上下文及客户端/服务器端参数，并将其输出到一个日志中进行查看。
- **代码自动完成**：NetBeans只需键入几个字符，即可进行代码补全提醒。

3.开发工具

3.4网站开发工具- Microsoft Visual Studio

□ 简介

- Visual Studio CodeMicrosoft Visual Studio Code是微软公司发布的运行于Mac OS X、Windows和Linux 环境，针对编写现代Web和云应用的跨平台源代码编辑器。



3.开发工具

3.4网站开发工具- Microsoft Visual Studio

□ 主要特点

- **跨平台**：Visual Studio Code支持Mac OS X、Windows和 Linux。
- **海量语言支持**：支持30多种语言的代码编辑器，可以编辑C#，VB，JavaScript，HTML，CSS，TypeScript，Ruby，Objective-C，PHP，JSON，Less，Sass，Markdown等。
- **并排编辑**：可以支持三个同步的文件编辑，每个人都可以发射命令提示符。
- **智能感应**：确保基于语言特性能够获得友好的提示，同时在编写代码时结合上下文智能感应。
- **调试**：可设置断点，查看调用堆栈或运行时的变量，暂停或逐步执行代码。



现场演示：

- 在Windows 7系统平台上安装Microsoft Visual Studio Code。
- Microsoft VS Code的软件界面与基本操作。
- 创建网站。
- 创建网页。

4.调试工具

4.1什么是Web调试

□ 什么是Web调试：

- 在Web应用开发过程中，开发人员需要借助工具了解程序的执行情况，从而修正语法错误和逻辑错误，以确定程序的正确性、安全性和稳定性等。Web调试的步骤：
 - 错误定位
 - 修改设计和代码
 - 排除错误，防止新错误产生

4.调试工具

4.2网站调试工具-FireFox

□ Mozilla Firefox

- Mozilla Firefox，中文名为“火狐”，是一个开源网页浏览器，使用Gecko引擎。Firefox由Mozilla基金会与数百个志愿者所开发，原名“Phoenix”（凤凰），之后改名“Mozilla Firebird”（火鸟），再改为现在的名字。
- 使用Firefox可以在浏览器实时运行HTML、CSS等代码。
- Firefox内置有强大的JavaScript调试工具，可以随时暂停JS动画，观察静态细节，还可以使用JS分析器来分析校准，找到问题原因。



4.调试工具

4.3网站调试工具-Google Chrome

□ Google Chrome

- Google浏览器，是由Google公司开发的开放源代码的网页浏览器。
- 该浏览器是**基于其他开源软件**所撰写，包括WebKit和Mozilla，目标是提升稳定性、速度和安全性，并创造出简单且有效率的使用者界面。
- Chrome对于HTML5和CSS3提供比较完善的支持。此外，Chrome还能够模拟手机调试。



4.调试工具

4.4网站调试工具-Internet Explorer

□ Internet Explorer

- Internet Explorer是微软推出的一款随所有新版本的Windows操作系统内置的默认网页浏览器，同时也是微软Windows操作系统的组成部分。
- Internet Explorer可以在浏览器中交互式地突出显示被选择的网页元素，查看style元素，定位div等，用户能够直接在浏览器窗口中浏览、传输和更新HTML DOM。



4.调试工具

4.4网站调试工具-Internet Explorer

□ Microsoft Edge

- Microsoft Edge是微软在2015年发布的，为了取代Internet Exploer，设计注重实用和极简主义，渲染引擎为EdgeHTML。
- Microsoft Edge加强了源代码的导航能力，支持JavaScript和TypeScript跳转至定义；对控制台进行改进，标出某个日志消息来自于源代码的哪一部分，使用Esc键打开控制台，消息中的链接能够进行点击；DOM浏览器的集成，支持元素高亮，从时间线中检阅某个元素。





现场演示:

- 使用Firefox审查网页页面的元素。
- 使用Firefox进行网页调试。

5.代码托管工具

5.1为什么要进行代码托管

□ 场景一：

某公司进行方案设计，最初提供方案A，讨论后更改成为B，进一步讨论仍然选择A，此时发现A没有备份。

□ 场景二：

一个项目需要多个工程师合作完成，工程师每天对修订的文件进行合并无法统一进度，团队协作困难。

□ 场景三：

项目开发中，开发服务器硬盘故障，而程序无备份。

5.代码托管工具

5.2代码托管的基本功能

- 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。

5.代码托管工具

5.3代码托管工具-GitHub

□ 简介

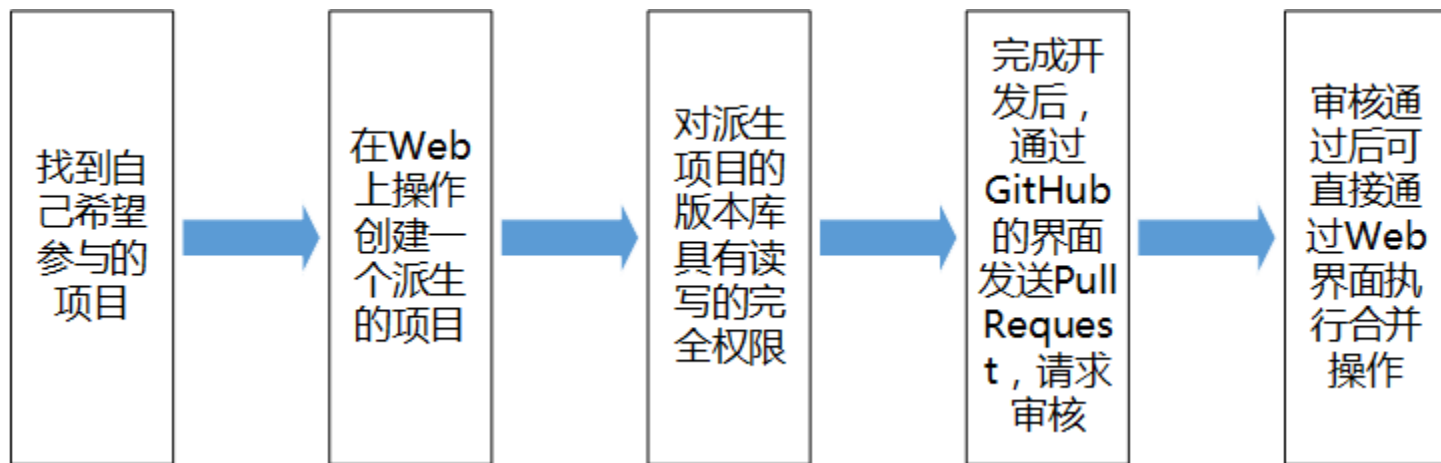
- GitHub是面向开源及私有软件项目的托管平台。GitHub简化了版本控制的管理和操作流程，为开发者提供了更好的交流平台。
- 使用GitHub使创建项目变得非常轻松，创建者只需在GitHub上点击一下鼠标即可创建一个新版本库。

GitHub

5.代码托管工具

5.3代码托管工具-GitHub

- GitHub的操作过程为：



5.代码托管工具

5.3代码托管工具-GitHub

□ 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。

5.代码托管工具

5.4代码托管工具-SVN

□ 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。



5.代码托管工具

5.4代码托管工具-SVN

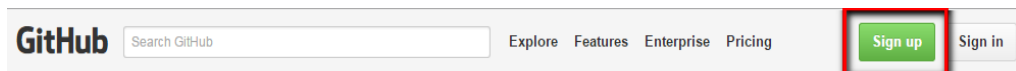
□ 主要特点

- **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。

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

- 使用GitHub开源平台实现网站代码托管的过程：
 - 创建GitHub账号，填写注册信息



5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

■ 创建项目

Owner: RuanHactm / Repository name: Demo ✓

Great repository names are short and memorable. Need inspiration? How about **creepy-rutabaga**.

Description (optional): This is my first project

Public
Anyone can see this repository. You choose who can commit.

Private
You choose who can see and commit to this repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

- 指定用户名和电子邮件

```
$ git config --global user.name "RuanHactcm" //指定用户名  
$ git config --global user.email "rxl@hactcm.edu.cn" //指定电子邮件
```

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

- GitHub选择的默认通信方式是SSH，所以要先在Git中生成公钥文件。

```
$ ssh-keygen -t rsa -C "rxl@hactcm.edu.cn"
Generating public/private rsa key pair.
//指定密钥文件存放位置
Enter file in which to save the key (/c:/Users/rxl/.ssh/id_rsa):
//为添加远程文件时创建密码
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c:/Users/rxl/.ssh/id_rsa.
Your public key has been saved in /c:/Users/rxl/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:hh3TIRcYveC6gVf25ToZRyPpZ/7oVqJLbgphG/tnMZs rxl@hactcm.edu.cn
The key's randomart image is:
+--[RSA 2048]----+
|           0=.. |
|            ..+ 0 |
|           0 .. 0.. |
|          0 0  +0..0|
|         .S. +...+. |
|          0.=+ 0.+0+ |
|          +. 0.*+B |
|           0.oE =0 |
|           0=0+0.0 |
+----[SHA256]-----+
```

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

- 检测是否能够通过SSH连接Github

```
$ ssh -T git@github.com
//输入之前创建的密码
Enter passphrase for key '/c/Users/rxl/.ssh/id_rsa':
Hi RuanHactcm! You've successfully authenticated, but GitHub does not provide shell
access.
```

- 在本地建立一个目录

```
$ mkdir Demo
//创建 Demo 目录
$ cd Demo
//进入 Demo 目录
$ pwd
//显示当前目录的路径
/e/Demo
```

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管

- 把创建的目录变成Git可以管理的仓库

```
$ git init
//把 Demo 变成 Git 可以管理的仓库
Initialized empty Git repository in E:/Demo/.git/
```

- 创建文件，并将文件导入到仓库

```
$ git add README.txt
//将 README.txt 文件添加到 Git 仓库
$ git commit -m 'README for this project.'
[master (root-commit) 2a6e019] README for this project.
1 file changed, 1 insertion(+)
create mode 100644 README.txt
```

5.代码托管工具

5.5案例：使用GitHub开源平台实现网站代码托管


- 将版本库添加到远程版本库

```
$ git remote add origin https://git@github.com:RuanHactcm/Demo.git
```

- 上传到GitHub


```
$ git push -u origin master
//输入之前创建的密码
Enter passphrase for key '/c/Users/rx/.ssh/id_rsa':
Counting objects: 3, done.
Writing objects: 100% (3/3), 255 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:RuanHactcm/Demo.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

 RuanHactcm / Demo

 Unwatch ▾ 1

This is my first project — Edit

 2 commits

 1 branch

 0 releases

 0 contributors




Branch: master ▾

Demo / +



README for this project.

RuanHactcm authored just now

latest commit 4ecbf690f3 

 README.txt

README for this project.

just now

 README.txt

这是我的第一个GitHub项目

查看上传文件



现场演示：

- 使用OSChina的GibHub服务，实现项目代码托管与版本控制。
- 使用SVN服务，实现项目版本管理和团队合作开发。

Thanks.