

## Web 项目开发流程及规范

开发流程及规范：**Web** 开发的分散性和交互性，决定了 **Web** 开发必须遵从一定的开发规范和技术约定。

只有每个开发人员都按照一个共同的规范去设计、沟通、开发、测试、部署，才能保证整个开发团队协调一致的工作，从而提高开发工作效率，提升工程项目质量。

### 一、项目的角色划分

如果不包括前、后期的市场推广和产品销售人员，开发团队一般可以划分为项目负责人、程序员、美工三个角色。

项目负责人在我们中国习惯称为"项目经理"，负责项目的人事协调、时间进度等安排，以及处理一些与项目相关的其它事宜。程序员主要负责项目的需求分析、策划、设计、代码编写、网站整合、测试、部署等环节的工作。美工负责网站的界面设计、版面规划，把握网站的整体风格。如果项目比较大，可以按照三种角色把人员进行分组。

角色划分是 **Web** 项目技术分散性甚至地理分散性特点的客观要求，分工的结果还可以明确工作责任，最终保证了项目的质量。分工带来的负效应就是增加了团队沟通、协调的成本，给项目带来一定的风险。所以项目经理的协调能力显得十分重要，程序开发人员和美工在项目开发的初期和后期，都必须有充分的交流，共同完成项目的规划和测试、验收。

### 二、开发工具的选取

不象 C/S 结构程序开发，可以一门语言从头到尾，你用 **Delphi**，就是 **Delphi** 程序员，你用 **VC++**，你就是 **VC** 程序员。**B/S** 结构的 **Web** 开发工作，工具的选择是一件痛苦的事情。从 **Windows** 到 **Linux**，从 **IIS** 到 **Apache**，从 **J2EE** 到 **.NET**，从 **COM** 到 **.NET** 到 **EJB** 组件.....还有 **Asp**、**Asp.net**、**Jsp**、**Php**、**Perl**、**Javascript**、**Vbscript**.....

美工也轻松不了多少，什么"网页三剑客" "新网页三剑客"、**FrontPage**、**Photoshop**、**CorelDraw**.....谁都说自己是最强大的！

我们的经验是，选用工具时最好是统一的，比如美工统一用 **DreamwaverMX** 制作网页，程序员全部用文本编辑器书写代码。统一工具的好处是可以保持同一个项目文档的一致性，便于开发人员的交流和文档的保存。

但是也不必刻意强求一致，比如美工可以使用任何自己熟悉的图形处理软件，只要最后能生成浏览器支持的图片就可以了。正是 **Web** 开发工具的多样性，才成就了今天互联网多姿多彩的局面。

只要程序员的纯 **Html** 和 **Javascript** 代码的功夫足够过硬，就能胜任最后的网站整合工作。

### 三、项目开发流程

如果项目真正谈下来了，就需要正式确定前阶段的需求分析，该补充的步骤必须补上。然后进行详细的总体设计，其实也基本是前阶段工作的重复和完善。

产生各栏目文件夹的结构图（一些公共文件夹如 **images**、**scripts**、**styles** 等需要固定存放，共同调用）。

然后由美工根据内容表现的需要，设计静态网页和其它动态页面界面框架，该切分的图片要根据尺寸切割开来。给需要程序动态实现的页面预留页面空间。制定字体、字号、超级链接等 **CSS** 样式等。  
在美工设计页面的同时，程序员着手开发后台程序代码，做一些必要的测试。美工界面完成后，由程序员添加程序代码，整合网站。由项目组共同联调测试，发现 **bug**，完善一些具体的细节。

制作帮助文档、用户操作手册。向用户交付必要的产品设计文档。然后进行网站部署、客户培训。

最后进入网站维护阶段。这一阶段也可以不包括在该项目中，而作为公司的服务内容。

以上的每一部都会产生一些阶段性成果，项目经理需要及时监督、审核，发现问题及时纠正。为了控制项目的进度，应当实施填写"项目进度表"制度，即每天填写工作日志，记录当天的工作细目和工作量，以及需要解决和已经解决的问题。

### 四、一些技术规则

#### 1， 数据库命名约定(参考了"匈牙利命名法")

数据库(Database): 格式 [db]\_[ desc]。

表(Table): 格式 [tab]\_[desc]。表名长度不能超过 30 个字符，单词首写字母大写，多个单词间不用连接符号。

字段(Field or Column): 格式 f\_[type]\_[desc]。f: 表明这是一个字段名称; type: 可选，表明字段类型，字符型为 **c**，整型为 **i**，逻辑型为 **b**，货币类型为 **m**，浮点型为 **f**，日期型为 **d**，时间型为 **t**，二进制为 **bl**。如果类型为字符型，可以省略。**desc**: 对字段属性的有意义的描述，可以用英语单词、单词缩写、汉语拼音、字段实际含义的拼音缩写等，单词之间可以用单词首字母大写软分割（推荐），也可以用"\_"隔开。

举例:

f\_name (姓名)

f\_c\_UserInfo 或 f\_c\_User\_Info

f\_xm (姓名)

f\_grp\_id (组标识)

索引(Index): 格式 [idx]\_[desc]。

视图(View): 格式 [View]\_[表 A]\_[表 B]\_[表 C]..., 其中 View 表示"视图"。这个视图由几个表产生就用连字符"\_"连接几个表的名, 如果表过多可以将表名适当简化。

存储过程: 格式 [sp]\_[表名]\_[存取过程名(缩写)], 比如 sp\_User\_Delete。

触发器(Trigger): 格式 [trg]\_[d][i][u]\_[desc]。trg 代表触发器; d, i, u 表明触发器类型>Delete, Insert, Update)定义, 书写顺序为 d、i、u; desc 是表的名称, 表明触发器所在的表。

数据库设备(Database Device): 格式 [dev]\_[desc]。

约束(Constraint): 格式 [cns]\_[desc]。

## 2, SQL 语句书写规范

SQL 语句中, SQL 关键字全部大写, 其它的遵照"数据库命名约定"。

例如:

```
SELECT * FROM tabNewsInfo WHERE f_UserName=" ORDER BY  
f_i_autoid
```

## 3, 文件夹命名约定

公共文件夹:

/images 公共图片

/styles 样式表

/scripts 脚本

/ftps 下载

/doc 网站相关素材、文档

/readme.txt 网站说明文档

/helps.htm 网站帮助文档

/mylogs.txt 网站维护记录

其它栏目的命名, 可以用拼音首字母简称, 也可以用英文单词。全部文件夹的含义在 readme.txt 文件中说明。

4, 对象及变量命名约定: 每个变量名必须先定义, 再使用。

在 ASP 文件的最开头添加语句<%Option Explicit%>可以强制变量定义。代码块必须采用缩进格式。每个函数前必须标明函数的功能、输入参数、返回值的相关信息。

变量类型 缩写前缀

String str 或 s

Integer Int

Date Dt

Object obj 或 o

Boolean bol 或 b

Byte Byt

Double Dbl

Error Err

Long Lng

Single Sng

#### 5, 图形对象约定

图片的格式: 最后生成 jpg, gif, png, swf 格式的图形文件。

图片的字节大小: 最大不能超过 30k。

图片的尺寸: 根据需要确定, 最好使用小图片, 大的图片必须切割成小图片使用。

图片的留白: 图片的边界不能留白, 图片只包含有效的色彩元素。

#### 6, 媒体对象约定

流媒体的格式: asf, wmv, wma, rm, 不建议使用 avi 格式的动画文件。

#### 7, 页面布局的基本约定

中文段落必须有 2 个汉字的缩进。字间距采用默认大小。行间距为 16pt~20pt。文字布局必须留有"天""地""左""右", 不能把版面占满。

页面布局必须保持色彩平衡。

注意上下、左右的呼应。注意页面的整体协调。提倡画面和文字的融合, 而不是画面和文字的明显分离。要按照设计广告的要求来设计网页页面, 特别是一些产品展示性的页面。

### 五、一些经验和教训

1, 能用静态网页表现的内容, 尽量不用程序代码动态实现。

2, 设计阶段, 必须和用户进行充分的交流, 完全、准确的了解用户的需求。

既不能歪曲用户的意思, 也不能一味迎合用户的非正当需求, 也不能对自己没有把握的技术甚至不可能实现的技术夸下海口。需求分析是一个沟通、交流、引导、教育、斗争、妥协的过程。需求分析结果要有文字资料存档。

3, 技术参数必须了解准确。

比如用户的软件平台是 linux 系列，那你的系统就要考虑用 Java 或者 Php 加 MySQL 开发了，这时候你的 ASP.NET 技术就用不上了。

4，最好让用户对已经确定的需求内容签字，盖章。

5，任何交流，必须有书面记录。

对一些喜欢"健忘"-实际上是懒惰的开发人员，要求他必须每天花 10 分钟写工作日志。

6，每个项目的有关文档，全部、统一集中归档。

