Web测试要点

**一、功能测试**

1、链接测试　　  
（1）、测试所有链接是否按指示的那样确实链接到了该链接的页面；    
（2）、测试所链接的页面是否存在；    
（3）、保证Web应用系统上没有孤立的页面(所谓孤立页面是指没有链接指向该页面，只有知道正确的URL地址才能访问)。    
  
2、表单测试  
（1）、注册、登陆、信息提交等，必须测试提交操作的完整性，以校验提交给服务器的信息的正确性；  
（2）、用户填写的出生日期与职业是否恰当，填写的所属省份与所在城市是否匹配等；    
（3）、检验默认值的正确性；  
（4）、如表单只能接受指定的某些值，测试时跳过这些字符，看系统是否会报错。

3、Cookies测试(session测试同)  
（1）、Cookies是否起作用；    
（2）、Cookies是否按预定的时间进行保存；  
（3）、刷新对Cookies有什么影响。

4、设计语言测试  
（1）、使用哪种版本的HTML；  
（2）、验证不同的脚本语言。例如Java、Javascrīpt、 ActiveX、VBscrīpt或Perl等。

5、数据库测试  
（1）、数据一致性错误：主要是由于用户提交的表单信息不正确而造成的；  
（2）、输出错误：主要是由于网络速度或程序设计问题等引起的。

**二、性能测试**

1、连接速度测试  
（1）、Web系统响应时间；  
（2）、超时的限制。  
  
2、负载测试  
（1）、某个时刻同时访问Web系统的用户数量；  
（2）、也可以是在线数据处理的数量。

3、压力测试  
（1）、压力测试是测试系统的限制和故障恢复能力，也就是测试Web应用系统会不会崩溃，在什么情况下会崩溃。黑客常常提供错误的数据负载，直到Web应用系统崩溃，接着当系统重新启动时获得存取权。  
（2）、压力测试的区域包括表单、登陆和其他信息传输页面等。

**三、可用性测试**

1、导航测试  
（1）、导航是否直观  
（2）、Web系统的主要部分是否可通过主页存取    
（3）、系统是否需要站点地图、搜索引擎或其他的导航帮助    
（4）、Web应用系统的页面结构、导航、菜单、连接的风格是否一致    
（5）、Web应用系统导航帮助要尽可能地准确。Web应用系统的层次一旦决定，就要着手测试用户导航功能。

2、图形测试  
一个Web应用系统的图形可以包括图片、动画、边框、颜色、字体、背景、按钮等。图形测试的内容有：   
（1）、要确保图形有明确的用途，图片或动画不要胡乱地堆在一起，以免浪费传输时间；  
（2）、Web应用系统的图片尺寸要尽量地小，并且要能清楚地说明某件事情，一般都链接到某个具体的页面；  
（3）、验证所有页面字体的风格是否一致；  
（4）、背景颜色应该与字体颜色和前景颜色相搭配；  
（5）、图片的大小和质量也是一个很重要的因素，一般采用JPG或GIF压缩。  
  
3、内容测试  
检验Web应用系统提供信息的正确性、准确性和相关性。  
信息的正确性是指信息是可靠的还是误传的 。  
  
4、整体界面测试  
整体界面是指整个Web应用系统的页面结构设计，是给用户的一个整体感。例如：当用户浏览Web应用系统时是否感到舒适，是否凭直觉就知道要找的信息在什么地方？整个Web应用系统的设计风格是否一致？当然，对界面的整体测试并不能单靠个人直觉来评定；每个人的审美观、专业角度、系统面向的行业及用户 、甚至性别与年龄等等，都是可能导致对界面作出不同评价的因素。所以要明白在对整体界面的测试过程中，其实是一个对最终用户进行调查的过程。一般Web应用系统采取在主页上做一个调查问卷的形式，来得到最终用户的反馈信息。

**四、兼容性测试**

1、平台测试  
在Web系统发布之前，需要在各种操作系统下对Web系统进行兼容性测试。

2、浏览器测试  
（1）、浏览器是Web客户端最核心的构件，来自不同厂商的浏览器对Java，、Javascrīpt、ActiveX、 plug-ins或不同的HTML规格有不同的支持。例如，ActiveX是Microsoft的产品，是为Internet Explorer而设计的，Javascrīpt是Netscape的产品，Java是Sun的产品等等。另外，框架和层次结构风格在不同的浏览器中也有不同的显示，甚至根本不显示。不同的浏览器对安全性和Java的设置也不一样。  
（2）、测试浏览器兼容性的一个方法是创建一个兼容性矩阵。在这个矩阵中，测试不同厂商、不同版本的浏览器对某些构件和设置的适应性。

**五、安全性测试**（1）、现在的Web应用系统基本采用先注册，后登陆的方式。因此，必须测试有效和无效的用户名和密码，要注意到是否大小写敏感，可以试多少次的限制，是否可以不登陆而直接浏览某个页面等；  
（2）、Web应用系统是否有超时的限制，用户登陆后在一定时间内（例如15分钟）没有点击任何页面，是否需要重新登陆才能正常使用；  
（3）、为了保证Web应用系统的安全性，需要测试相关信息是否写进了日志文件、是否可追踪；  
（4）、当使用了安全套接字时，还要测试加密是否正确，检查信息的完整性；    
（5）、服务器端的脚本常常构成安全漏洞，这些漏洞又常常被黑客利用。所以，还要测试没有经过授权，就不能在服务器端放置和编辑脚本的问题。  
（6）、通过模拟攻击的形式拷贝Web应用程序的某个功能点的url地址，然后打开新的页面输入该url地址看其是否能跨过系统的登录模块直接进入该功能点。  
（7）、服务器端IIS是否设置了默认文档功能。  
（8）、IIS服务器的主目录应该与操作系统的安装路径设置在不同的盘符下。