

# 网站功能与性能测试方法初探

钟元生 (江西财经大学 信息学院 江西 南昌 330013)  
(江西财经大学 软件学院 江西 南昌 330013)  
徐 娟 (江西财经大学 信息学院 江西 南昌 330013)  
陈 媚 (江西财经大学 软件学院 江西 南昌 330013)

**摘 要:**网站测试越来越受人们重视,但网站实际测试时,人们常觉无章可循。在简要分析网站测试内容的基础上,本文介绍了基于业务流程分析的网站功能测试用例设计方法以及基于测试工具的性能测试方法。

**关键词:**业务流程分析 web 测试 性能测试 测试用例设计

## 1 引言

国内对于 web 系统的测试研究还处于起步阶段,人们在网站测试方面缺乏经验,有必须研究网站测试的方法。网站测试主要从功能、性能、用户界面等方面进行考虑。本文主要讨论基于业务流程的 web 网站测试的方法及其应用,特别是测试用例的设计方法。

## 2 网站测试的内容

Web 的系统测试与传统的软件测试不同,它不但需要检查和验证是否按照设计的要求运行,而且还要测试系统在不同用户的浏览器端的显示是否合适,即从最终用户的角度进行安全性和可用性测试。由于 web 程序综合了大量的新技术,诸如 HTML、JAVA、Javascript、VBScript 等,同时它还依赖很多其它的因素,比如 Link、Database、Network 等,使得 WEB 应用程序测试变得非常复杂。其中,网站的功能测试和性能测试是网站测试的两个重点<sup>[2]</sup>。此外,Internet 和 Web 媒体的不可预见性使测试基于 Web 的系统变得困难。因此,我们必须为测试和评估复杂的基于 Web 的系统研究新的方法和技术。

### 2.1 网站功能测试

网站功能测试主要包括链接测试和表单测试。

(1) 链接测试主要测试其各页面链接是否正确,是否存在死链接,保证 web 应用系统中没有孤立的页面。

(2) 表单测试是通过模拟用户提交,验证表单功

能是否完善,如注册信息。要测试表单功能的完善性,需要验证服务器能否正确保存这些数据,后台运行的程序能否正确解释和使用这些信息,并结合易用性要求,对数据正确性进行验证及异常处理。

### 2.2 性能测试

性能测试包括负载测试和压力测试。性能是 web 网站是否完善的一个重要指标之一,也是在 web 应用投入使用之前人们最为关心的问题。因此,对 web 网站的性能测试尤为重要。

负载测试中负载级别可以是某个时刻同时访问 Web 系统的用户数量,也可以是在线数据处理的数量。应分别测试 Web 应用系统能允许多少个用户同时在线? 如果超过了这个数量,会出现什么现象? Web 应用系统能否处理大量用户对同一个页面的请求?

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。压力测试的区域包括表单、登陆和其他信息传输页面等。

Web 性能测试通常是在客户端进行场景模拟,模拟大量并发用户执行不同业务操作,达到实施负载压力的目的。要模拟真实的负载压力场景,必须创建方案,即如何模拟现实生活中用户使用的方式。方案包含如何模拟实际用户的信息,即建立虚拟用户组 (Vuser), Vuser 将运行的测试脚本及用于运行脚本的负载生成器计算机。

本文主要以对广州立白集团网站的测试为例,阐

述如何针对以上几种测试方法分别进行用例设计。

### 3 基于业务流程分析的 Web 测试用例设计方法

#### 3.1 测试用例设计概述

测试用例为软件测试工作的指导性文件,测试用例的优劣直接影响软件测试的质量。如何以最少的人力、资源投入,在最短的时间内完成测试,发现软件系统的缺陷,保证软件的优良品质,是软件公司探索和追求的目标。尤其对第三方测试机构来说,软件测试中的测试用例设计及复用更为重要。

设计测试用例即设计针对特定功能或组合功能的测试方案,并编写成文档。测试用例的选择既要有一般情况,也应有极限情况以及最大和最小的边界值情况。因为测试的目的是暴露应用软件中隐藏的缺陷,所以在设计选取测试用例和数据时要考虑那些易于发现缺陷的测试用例和数据,结合复杂的运行环境,在所有可能的输入条件和输出条件中确定测试数据,来检查应用软件是否都能产生正确的输出。

我们可以采用黑盒测试中的等价类划分法来实行对网站功能的划分,进而设计测试用例。等价类划分法是黑盒测试用例设计中一种重要的、常用的设计方法,它将不能穷举的测试过程合理分类,从而保证设计出来的测试用例具有完整性和代表性。设计过程主要分为两步:

(1) 分类,即将输入域按照相同的特性或者类似功能进行分类;

(2) 抽象,即在各个子类中抽象出相同特性并用实例来表征这个特性。

#### 3.2 基于业务流程分析的功能测试方法

功能测试是为了确保测试的功能正常,如导航、数据输入、处理、检索是否正确,以及业务规则的实施是否恰当。功能测试一般都通过链接测试和表单测试分别进行来完成。这样会降低系统整个流程的衔接。我们采用的方法是基于业务流程分析的测试方法,这种方法根据业务流程的不同路径设计不同的测试用例。根据这些用例的执行判断功能是否畅通,是否有孤立的路径,验证网站的功能是否完善;所指向的地址是否正确,是否有孤立的结点;在每一个流程中对每一个表单进行测试,判断其客户端验

证是否合理,输入能否得到预期输出结果。其步骤为:(1)分析业务流程;(2)画出业务流程图;(3)根据业务流程图画业务流图;(4)分析每个业务流图的不同路径;(5)为每个业务流图的每条路径设计至少一个测试用例;(6)手工测试或自动化测试每条路径。

根据业务流图设计测试用例时采用基本路径测试的方法,保证模块中的每一条独立的路径至少执行一次。基本路径计算公式为  $V(G) = E - N + 2$ ,其中  $E$  代表业务流图中的边数, $N$  代表结点个数。

#### 3.3 实例应用

以广州立白集团网站测试为例,通过等价类方法对其进行功能划分,可将其分为:信息公布、表单提交、网页链接三大类。其中,信息公布类包括:集团介绍、产品展厅、业务往来、客服通道、新闻中心、人才招聘、联系我们模块;表单提交类包括:投诉与建议、人才招聘模块;网页链接类包括:产品展厅、客服通道、新闻中心模块。人才招聘模块的流程图如图 1 所示。

从图 1 可知,  $V(G) = E - N + 2 = 9 - 8 + 2 = 3$ ,即人才招聘模块的业务流图中有三个独立路径,它们分别是:路径 1:1-2-3-4-5-6-7;路径 2:1-2-3-4-3-4-5-6-7;路径 3:1-2-3-4-5-6-8-3-4-5-6-7

为了保证每一条路径执行一次而设计的测试用例(如表 1 至表 3 所示)

表 1 路径 1 的测试用例

|      |  |          |      |
|------|--|----------|------|
| 用例编号 | LI   |          |      |
| 功能描述 | 路径 1 的业务流程   |          |      |
| 用例目的 | 测试路径 1 的业务流程是否正确实现   |          |      |
| 前提条件 | 网站服务器运行正常,打开广州立白集团中文版中的“人才招聘”页面,进入“在线应聘”页面中  |          |      |
|      | 输入/动作  | 期望的输出/相应 | 实际情况 |
|      | 第一步,在表单中的各文本框中输入姓名:李四;性别:女;年龄:20;户口:江西;手机:13823145234;email:meide@hotmail.com;第二步,提交 | 提交成功     |      |

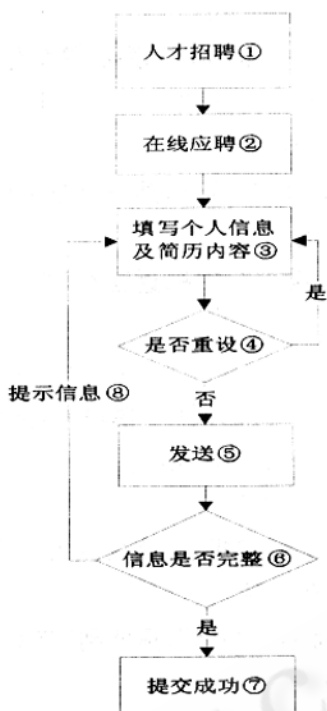


图 1 人才招聘模块业务流程图

表 2 路径 2 的测试用例

|      |  |          |      |
|------|--|----------|------|
| 用例编号 | L2   |          |      |
| 功能描述 | 路径 2 的业务流程   |          |      |
| 用例目的 | 测试路径 2 的业务流程是否正确实现   |          |      |
| 前提条件 | 网站服务器运行正常, 打开广州立白集团中文版中的“人才招聘”页面, 然后进入“在线应聘”页面中  |          |      |
|      | 输入/动作  | 期望的输出/相应 | 实际情况 |
|      | 第一步, 在表单中的各文本框中输入姓名: 李四; 第二步, 重填一次, 输入姓名: 李四; 性别: 女; 年龄: 20; 户口: 江西; 手机: 13823145234; email: meicle@hotmail.com; 第三步, 提交 | 提交成功     |      |

## 4 基于测试工具的网站性能测试用例设计

### 4.1 基于测试工具的性能测试方法

性能测试主要通过使用负载测试工具来进行, 测试工具是软件开发工具的重要组成部分, 在产品开发及其管理过程中起着非常重要的作用<sup>[1]</sup>。性能测试的关键是模拟实际的用户使用环境。如果不利用测试工具, 软件性能测试根本就无法完成。性能测试的基本步骤为:

(1) 测试环境配置。性能测试通常采用多台客户机进行模拟测试, 其中一台为监控机, 其余为代理机。

表 3 路径 3 的测试用例

|      |  |                    |      |
|------|--|--------------------|------|
| 用例编号 | L3   |                    |      |
| 功能描述 | 路径 3 的业务流程   |                    |      |
| 用例目的 | 测试路径 3 的业务流程是否正确实现   |                    |      |
| 前提条件 | 网站服务器运行正常, 打开广州立白集团中文版中的“人才招聘”页面, 然后进入“在线应聘”页面中  |                    |      |
|      | 输入/动作  | 期望的输出/相应           | 实际情况 |
|      | 第一步, 在表单中的各文本框中输入姓名: 小风; 第二步, 提交; 第三步, 弹出表单没有填写完整; 第四步, 再重新填一次, 输入姓名: 李四; 性别: 女; 年龄: 20; 户口: 江西; 手机: 13823145234; email: meicle@hotmail.com; 第五步, 提交成功 | 先弹出相关提示信息, 最后是提交成功 |      |

(2) 场景设计。场景设计即性能测试用例的设计。主要是通过模拟用户实际操作以测试系统使用情况, 测试系统能否负载多用户的并发操作等。可以分为以下几种情况进行场景设计:

① 多用户并发操作。多个用户同时登录 web 应用系统的同一模块, 同时进行并发操作。此时可以逐渐增加用户人数, 以测试系统所能承受的负载压力。如多用户同时登录“人才招聘模块”模块, 同时点击“提交”。用户人数可以逐渐增加, 查看系统的承受能力。

② 用户每秒登录若干个。用户登录同一模块, 设定一个总的人数限制, 然后每秒登录若干用户, 并执行同一操作。每秒登录的用户数可以逐渐增加。

(3) 使用性能测试工具实施场景。此时开始使用测试工具, 首先创建要执行的脚本, 脚本是一种测试工具执行的指令集合, 可以直接用脚本语言编写也可以通过录制测试操作产生。后者可以减少脚本编程的工作量。创建好要使用的脚本后还要进行编辑修改以检查脚本是否有误; 接着使用已经设计好的场景进行脚本的运行; 最后分析测试结果。

下面以 LoadRunner 测试工具为例, 使用 LoadRunner 8.0 进行性能测试, 整个测试过程如下:

(1) 启动 Virtual User Generator 创建脚本。选择系统通讯的协议; 输入要测试的站点地址, 然后开始录

制脚本;录制完成后进行脚本的完善,对脚本进行编译,并检查脚本是否有误。

(2) 设计场景。

(3) 使用中央控制器 (Controller) 来调度虚拟用户。根据前面提出的几种场景分别创建 Scenario, 选择脚本;设置机器虚拟用户数;设置 Schedule;如果模拟多机测试,则设置 Ip Spoofer。

(4) 运行脚本。脚本执行完毕后,LoadRunner 会自动分析结果,得出测试报告。

在性能测试过程中,测试人员要注意了解以下一些情况:(1)随着虚拟用户的增加,服务器在规定时间内所能处理的总点击数;(2)随着虚拟用户的增加,服务器在规定时间内所能处理的每秒请求数;(3)随着模拟访问时间的延续,出现错误的数量;(4)随着虚拟用户的增加,出现错误的数量;(5)针对虚拟用户数的应用性能分布情况,包括服务器的内存、CPU 使用情况等;(6)随着虚拟用户的变化,应用性能的变化等。因此在整个性能测试过程中,合理的场景设计是性能测试成功的关键。

## 4.2 测试用例设计

LoadRunner 是一种适用于各种体系架构的自动负载测试工具,它能预测系统行为并优化系统性能。LoadRunner 的测试对象是整个企业的系统,它通过模拟实际用户的操作行为和实行实时性能监测,来帮助更快的查找和发现问题。下面用 LoadRunner 对广州立白集团的网站进行测试,通过设计 5 种不同场景来模拟用户的不同操作行为来查看其性能,及其负载能力。

在此针对该网站的用途并估计其访问量,设计了 5 种不同场景来模拟用户的不同操作行为来查看其性能,及其负载能力。

测试用例 1:用户并发操作

场景描述一:

(1) 用户登录的“产品展厅”模块,总共登录 24 个用户,所有用户都同时并发操作。

(2) 用户点击“立白”。

(3) 进入“立白系列”模块,然后点击下面的页面数字键查看该系列的所有产品。

(4) 点击“产品展厅”按钮,返回到“产品展厅”模块。

场景描述二:

(1) 用户登录的“产品展厅”模块,总共登录 48 个用户,所有用户都同时并发操作。

(2) 用户点击“立白”

(3) 进入“立白系列”模块,然后点击下面的页面数字键查看该系列的产品。

(4) 点击“产品展厅”按钮,返回到“产品展厅”模块。

场景描述三:

(1) 用户登录的“产品展厅”模块,总共登录 100 个用户,所有用户都同时并发操作。

(2) 用户点击“立白”。

(3) 进入“立白系列”模块,然后点击下面的页面数字键查看该系列的产品。

(4) 点击“产品展厅”按钮,返回到“产品展厅”模块。

测试用例 2:用户每秒登录若干个

场景描述四:

(1) 用户登录的“产品展厅”模块,总共登录 48 个用户,每 1 秒登录 1 个用户。

(2) 用户点击“立白”。

(3) 进入“立白系列”模块,然后点击下面的页面数字键查看该系列的产品。

(4) 点击“产品展厅”按钮,返回到“产品展厅”模块。

场景描述五:

(1) 用户登录的“产品展厅”模块,总共登录 48 个用户,每秒同时登录 10 个用户。

(2) 用户点击“立白”。

(3) 进入“立白系列”模块,然后点击下面的页面数字键查看该系列的所有产品。

(4) 点击“产品展厅”按钮,返回到“产品展厅”模块。

## 4.3 测试结果分析

根据前述方法运用 loadrunner,可以得到相关的结果。

持续时间:5 分 8 秒

统计信息摘要如下:

最大运行 vuser 数:24

总吞吐量(字节):21,512,646

平均吞吐量(字节/秒):69,620  
 总点击次数:1,598  
 平均每秒点击次数:5.172

表 4 Http 响应摘要

| HTTP Responses | Total | Per second |
|----------------|-------|------------|
| HTTP 200       | 1,598 | 5.172      |

表 5 事务摘要

| 事务名                    | 最小值    | 平均值     | 最大值     | 标准偏差   | 90%     | 通过 | 失败 | 停止 |
|------------------------|--------|---------|---------|--------|---------|----|----|----|
| Action transaction     | 86.027 | 137.407 | 246.791 | 37.617 | 164.204 | 21 | 3  | 0  |
| Vuser end transaction  | 0      | 0       | 0       | 0      | 0       | 24 | 0  | 0  |
| Vuser init transaction | 0      | 0.002   | 0.005   | 0.002  | 0       | 24 | 0  | 0  |

根据 5 个不同的场景设计,配置脚本后进行测试,对得到的结果进行比较,分析如下:

当 24 个 vusers 并发操作时,有 3 个事务失败,通过率是:95.8%,总吞吐量(字节):39,105,740,说明这种情况下网站负载得起;

当 48 个 vusers 每秒登录 1 个时,失败的事务有 15 个,通过率是:89.58%,总吞吐量(字节):39,105,740,说明一个事务的响应时间有点过长,这种情况下事务响应超时;

当 48 个 vusers 并发操作时,有 29 个事务失败,通过率是:79.86%,总吞吐量(字节):32,183,597,这种情况下网站反应比较慢,说明有点超过网站的负载能力。

当 48 个 vusers 每秒登录 10 个时,失败事务总数为 22 个,通过率是:84.72%,总吞吐量(字节):32,889,219,这种情况还是属于网站响应比较慢;

当 100 个 vusers 并发操作时,有 54 个失败事务,通过率是:82%,总吞吐量(字节):68,029,664,这种情况网站响应时间也是不够理想。

事务失败的主要原因是:由于网站中的图片过大,加载时间超时,随着对网站访问的人数增加,导致总的页面响应超时,最终导致事务失败。据该网站的目前情况来看,并发访问的人数控制在 24 人左右,网站基本上可以正常工作。

## 5 结论

本文提出了一种基于业务流程的 web 网站测试方法,实现了网站的功能和性能测试用例的设计。当

前的 web 测试主要采用手动测试与自动化测试相结合的技术。使用自动化测试可以极大地提高测试效率,尤其是当测试用例不需大的改动,而需多次执行时,其优势比手动测试明显。但当前的自动化测试系统还不十分完善,对一些小概率软件缺陷不易发现。因此,在具体实施 web 测试中,应将手动测试与自动化测试相结合。

## 参考文献

- 1 李年、王诚,基于 Java 的 Web 应用全球化自动测试设计[J],计算机工程与设计,2005,26(12):3341-3343.
- 2 黄海隆,Web 测试技术研究,大众科技,2006 年第 4 期.
- 3 翟长远、郑明,自动化测试在软件黑盒测试中的应用,EIC Vol.13 2006 No.4 81.
- 4 许蕾、徐宝文,Web 应用测试框架研究[J],东南大学学报(自然科学版),2004,34(6):751-755.
- 5 Xu Lei, Xu Baowen, Chen Zhenqiang. Survey of Web testing [J]. Computer Science, 2003, 30(3):100-104.
- 6 Kallepalli C, Tian J. Measuring and modeling usage and reliability for statistical Web testing [J]. IEEE Trans Software Engineering, 2001, 27(11):1023-1036.
- 7 Liu C H. A formal object-oriented test model for testing Web applications [D]. Arlington: Graduate School of the University of Texas, 2002.