

功能测试就是对产品的各功能进行 `php?name=%D1%E9%D6%A4">`验证, 根据功能测试用例, 逐项测试, 检查产品是否达到用户要求的功能。针对 web 系统的常用测试方法如下:

1. 页面链接检查: 每一个链接是否都有对应的页面, 并且页面之间切换正确。可以使用一些工具, 如 LinkBotPro、File-AIDCS、HTML Link Validator、Xenu 等工具。LinkBotPro 不支持中文, 中文字符显示为乱码;HTML Link Validator 只能测试以 Html 或者 htm 结尾的网页链接;Xenu 无需安装, 支持 asp、do、jsp 等结尾的网页, xenu 测试链接包括内部链接和外部链接, 在使用的时候应该注意, 同时能够生成 html 格式的测试报告。如果系统用 QTP 进行自动化测试, 也可以使用 QTP 的页面检查点检查链接。

2. 相关性检查: 功能相关性: 删除/增加一项会不会对其他项产生影响, 如果产生影响, 这些影响是否都正确, 常见的情况是, 增加某个数据记录以后, 如果该数据记录某个字段内容较长, 可能会在查询的时候让数据列表变形。

数据相关性: 下来列表默认值检查, 下来列表值检查, 如果某个列表的数据项依赖于其他模块中的数据, 同样需要检查, 比如, 某个数据如果被禁用了, 可能在引用该数据项的列表中不可见。

3. 检查按钮的功能是否正确: 如新建、编辑、删除、关闭、返回、保存、导入, 上一页, 下一页, 页面跳转, 重置等功能是否正确。常见的错误会出现在重置按钮上, 表现为功能失效。

4. 字符串长度检查: 输入超出需求所说明的字符串长度的内容, 看系统是否检查字符串长度。还要检查需求规定的字符串长度是否是正确的, 有时候会出现, 需求规定的字符串长度太短而无法输入业务数据。

5. 字符类型检查: 在应该输入指定类型的内容的地方输入其他类型的内容(如在应该输入整型的地方输入其他字符类型),看系统是否检查字符类型。

6. 标点符号检查: 输入内容包括各种标点符号,特别是空格,各种引号,回车键。看系统处理是否正确。常见的错误是系统对空格的处理, 可能添加的时候, 将空格当作一个字符, 而在查询的时候空格被屏蔽, 导致无法查询到添加的内容。

7.特殊字符检查: 输入特殊符号, 如@、#、\$、%、!等, 看系统处理是否正确。常见的错误是出现在% ‘ ” 这几个特殊字符

8. 中文字符处理: 在可以输入中、英文的系统输入中文,看会否出现乱码或出错。

9. 检查信息的完整性: 在查看信息和更新信息时,查看所填写的信息是不是全部更新,更新信息和添加信息是否一致。要注意检查的时候每个字段都应该检查, 有时候, 会出现部分字段更新了而个别字段没有更新的情况。

10. 信息重复: 在一些需要命名,且名字应该唯一的信息输入重复的名字或 ID,看系统有没有处理,会否报错,重名包括是否区分大小写,以及在输入内容的前后输入空格,系统是否作出正确处理。

11. 检查删除功能: 在一些可以一次删除多个信息的地方,不选择任何信息,按“delete”,看系统如何处理,会否出错;然后选择一个和多个信息,进行删除,看是否正确处理。如果有多页,翻页选,看系统是否都正确删除,并且要注意,删除的时候是否有提示,让用户能够更正错误,不误删除。

12. 检查添加和修改是否一致: 检查添加和修改信息的要求是否一致,例如添加要求必填的项,修改也应该必填;添加规定为整型的项,修改也必须为整型。

13. 检查修改重名: 修改时把不能重名的项改为已存在的内容,看会否处理,报错。同时,也要注意,会不会报和自己重名的错。

14. 重复提交表单: 一条已经成功提交的纪录,返回后再提交,看看系统是否做了处理。对于 Web 系统来说,可以通过浏览器返回键或者系统提供的返回功能。

15. 检查多次使用返回键的情况: 在有返回键的地方,返回到原来页面,重复多次,看会否出错。

16. 搜索检查: 有搜索功能的地方输入系统存在和不存在的内容,看搜索结果是否正确。如果可以输入多个搜索条件,可以同时添加合理和不合理的条件,看系统处理是否正确,搜索的时候同样要注意特殊字符,某些系统会在输入特殊字符的时候,将系统中所有的信息都搜索到。

17. 输入信息位置: 注意在光标停留的地方输入信息时,光标和所输入的信息会否跳到别的地方。

18. 上传下载文件检查: 上传下载文件的功能是否实现,上传文件是否能打开。对上传文件的格式有何规定,系统是否有解释信息,并检查系统是否能够做到。下载文件能否打开或者保存,下载的文件是否有格式要求,如需要特殊工具才可以打开等。上传文件测试同时应该测试,如果将不能上传的文件后缀名修改为可以上传文件的后缀名,看是否能够上传成功,并且,上传文件后,重新修改,看上传的文件是否存在。

19. 必填项检查: 应该填写的项没有填写时系统是否都做了处理,对必填项是否有提示信息,如在必填项前加“*”;对必填项提示返回后,焦点是否会自动定位到必填项。

20. 快捷键检查: 是否支持常用快捷键,如 Ctrl+C、Ctrl+V、Backspace 等,对一些不允许输入信息的字段,如选人,选日期对快捷方式是否也做了限制。

21. 回车键检查: 在输入结束后直接按回车键,看系统处理如何,会否报错。这个地方很有可能会出现错误。

22.刷新键检查：在 Web 系统中，使用浏览器的刷新键，看系统处理如何，会否报错。

23.回退键检查：在 Web 系统中，使用浏览器的回退键，看系统处理如何，会否报错。对于需要用户验证的系统，在退出登录后，使用回退键，看系统处理如何；多次使用回退键，多次使用前进键，看系统如何处理。

24.直接 URL 链接检查：在 Web 系统中，直接输入各功能页面的 URL 地址，看系统如何处理，对于需要用户验证的系统更为重要。如果系统安全性设计的不好，直接输入各功能页面的 URL 地址，很有可能会正常打开页面。

25.空格检查：在输入信息项中，输入一个或连串空格，查看系统如何处理。如对于要求输入整型、符点型变量的项中，输入空格，既不是空值，又不是标准输入。

26.输入法半角全角检查：在输入信息项中，输入半角或全角的信息，查看系统如何处理。如对于要求输入符点型数据的项中，输入全角的小数点(“。”或“.”，如 4.5)；输入全角的空格等。

27.密码检查：一些系统的加密方法采用对字符 Ascii 码移位的方式，处理密码加密相对较为简单，且安全性较高，对于局域网系统来说，此种方式完全可以起到加密的作用，但同时，会造成一些问题，即大于 128 的 Ascii 对应的字符在解密时无法解析，尝试使用“uvwxyz”等一些码值较大的字符作为密码，同时，密码尽可能的长，如 17 位密码等，造成加密后的密码出现无法解析的字符。

28.用户检查：任何一个系统，都有各类不同的用户，同样具有一个或多个管理员用户，检查各个管理员之间是否可以相互管理，编辑、删除管理员用户。同时，对于一般用户，尝试删除，并重建同名的用户，检查该用户其它信息是否重现。同样，提供注销功能的系统，此用户再次注册时，是否作为一个新的用户。而且还要检查该用户的有效日期，过了有效日期的用户是不能登录系统的。容易出现错误的情况是，可能有用户管理权限的非超级管理员，能够修改超级管理员的权限。

29.系统数据检查：这是功能测试最重要的，如果系统数据计算不正确，那么功能测试肯定是通不过的。数据检查根据不同的系统，方法不同对于业务管理平台，数据随业务过程、状态的变化保持正确，不能因为某个过程出现垃圾数据，也不能因为某个过程而丢失数据。

30.系统可恢复性检查：以各种方式把系统搞瘫，测试系统是否可正常迅速恢复。

31.确认提示检查：系统中的更新、删除操作，是否提示用户确认更新或删除，操作是否可以回退(即是否可以取消操作)，提示信息是否准确。事前或事后提示，对于 Update 或 Delete 操作，要求进行事前提示。

32.数据注入检查：数据注入主要是对数据库的注入，通过输入一些特殊的字符，如“'”，“/”，“-”等或字符组合，完成对 SQL 语句的破坏，造成系统查询、插入、删除操作的 SQL 因为这些字符而改变原来的意图。如 `select * from table where id = ' ' and name = ' '`，通过在 id 输入框中输入“12' -”，会造成查询语句把 name 条件注释掉，而只查询 id=12 的

记录。同样，对于 update 和 delete 的操作，可能会造成误删除数据。当然还有其它一些 SQL 注入方法，具体可以参考《SQL 应用高级 SQL 注入.doc》，很多程序都是基于页面对输入字符进行控制的，可以尝试跳过界面直接向数据库中插入数据，比如用 Jmeter，来完成数据注入检查。

33.刷新检查：web 系统中的 WebForm 控件实时刷新功能，在系统应用中有利有弊，给系统的性能带来较大的影响。测试过程中检测刷新功能对系统或应用造成的影响(白屏)，检查控件是否回归默认初始值，检查是否对系统的性能产生较大影响(如每次刷新都连接数据库查询等)。

34.事务检查：对于事务性操作，断开网络或关闭程序来中断操作，事务是否回滚。

35.时间日期检查：时间、日期验证是每个系统都必须的，如 2006-2-29、2006-6-31 等错误日期，同时，对于管理、财务类系统，每年的 1 月与前一年的 12 月(同理，每年的第 1 季度与前一年的第 4 季度)。另外，对于日期、时间格式的验证，如 2006 年 2 月 28 日、2006-2-28、20060228 等。日期检查还要检查日期范围是否符合实际的业务，对于不符合时间业务的日期，系统是否会有提示或者有限制。

36.多浏览器验证：越来越多的各类浏览器的出现，用户访问 Web 程序不再单单依赖于 Microsoft Internet Explorer，而是有了更多的选择：Maxthon、Firefox、Tencent Traveler 等，考虑使用多种浏览器访问系统，验证效果。

37.安装测试：对于 C/S 架构的系统，安装程序的测试是一个重要方面，安装程序自动化程度、安装选项和设置(验证各种方案是否都能正常安装)、安装过程中断测试、安装顺序测试(分布式系统)、修复安装及卸载测试。

38.文档测试：主要是对用户使用手册、产品手册进行测试，校验是否描述正确、完整，是否与当前系统版本对照，是否易理解，是否二义性等。

39.测试数据检查：事实告诉我们，测试数据比代码更有可能是错的，因此，当测试结果显示有错误发生的时候，怀疑代码错误前要先对测试数据检查一遍。

40.请让我的机器来运行：在某些项目中，出现一个病态的问题：系统没有问题呀，它在我的机器上是能够通过的。这就说明了其中存在着和环境相关的 BUG。“是否所有的一切都受到了版本控制工具的管理?”、“本机的开发环境和服务器的环境是否一样?”、“这里是否存在一个真正的 BUG，只不过是其他的机器里偶然出现?”。所有的测试必须在所有系统要求的机器上运行通过，否则的话，代码就可能存在问题。

41.Ajax 技术的应用：Ajax 有很多优点，但也有很多缺点，如果利用优点、避免缺点，是我们对新的 Web2.0 应用的一个挑战。而 Ajax 的应用最直接的问题就是用户体验，用户体验的效果直接关系到是否使用 Ajax 技术。“会做，并不意味着应该做、必须做”，这就是对 Ajax 技术的很重要的注解。

42.Ajax 技术的应用：Ajax 采用异步调用的机制实现页面的部分刷新功能，异步调用存

在异常中断的可能，尝试各种方法异常中断异步的数据调用，查看是否出现问题。在这里遇到的一个问题就是对日期控件的操作，已经如果页面数据较多时候的刷新。

43.脚本错误：随着 Ajax、IFrame 等异步调用技术的发展，JavaScript 技术也越来越受到开发人员的重视，但 JavaScript 存在调试困难、各浏览器存在可能不兼容等问题，因此在 Web 系统中

=====

- 1、页面链接是否正确;
- 2、关联性，一个功能是否会对其他功能造成影响;
- 3、按钮功能测试，删除不选、多选、翻页选择;
- 4、字符串长度、类型、符号、特殊符号、中英文、空格、半角全角;
- 5、信息输出完整性;
- 6、信息提交重复处理;
- 7、添加修改，修改重名;
- 8、重复提交、重复删除、多用户并发操作;
- 9、索引检查;
- 10、输入信息、光标位置、快捷键使用;
- 11、上传下载文件;
- 12、必选项测试;
- 13、回车、刷新、浏览键回退检查(主要是需要验证的地方);
- 14、直接 URL 链接检查;
- 15、密码检查、长度、半角全角;
- 16、不同用户权限检查;
- 17、系统数据计算检查;
- 18、系统健壮性检查;

- 19、确认提示检查;
- 20、数据注入检查，一般程序是屏蔽掉特殊字符或者敏感字符;
- 21、刷新检查，主要是实时刷新功能;
- 22、事物检查，失败异常回滚;
- 23、时间格式检查;
- 24、浏览器兼容性检查;
- 25、安装、文档测试;
- 26、测试数据检查，即对自己测试提供数据进行检查;