**3、测试标准**

**3.1 UI验收**

1.风格、样式、颜色是否协调

　　2. 界面布局是否整齐、协调（保证全部显示出来的，尽量不要使用滚动条

　　3. 界面操作、标题描述是否恰当（描述有歧义、注意是否有错别字）。

　　4. 操作是否符合人们的常规习惯（有没有把相似的功能的控件放在一起，方便操作）

　　5. 提示界面是否符合规范（不应该显示英文的cancel、ok，应该显示中文的确定等）

　　6. 界面中各个控件是否对齐。

　　7. 日期控件是否可编辑。

　　8. 日期控件的长度是否合理，以修改时可以把时间全部显示出来为准。

　　9. 查询结果列表列宽是否合理、标签描述是否合理。

　　10. 查询结果列表太宽没有横向滚动提示。

　　11. 对于信息比较长的文本，文本框有没有提供自动竖直滚动条。

　　12. 数据录入控件是否方便。

　　13. 有没有支持Tab键，键的顺序要有条理，不乱跳。

　　14. 有没有提供相关的热键。

　　15. 控件的提示语描述是否正确。

　　16. 模块调用是否统一，相同的模块是否调用同一个界面。

　　17. 用滚动条移动页面时，页面的控件是否显示正常。

　　18. 页面是否有多余按钮或标签。

　　19. 窗口标题或图标是否与菜单栏的统一。

　　20. 窗口的最大化、最小化是否能正确切换。

　　21. 对于正常的功能，用户可以不必阅读用户手册就能使用。

　　22. 执行风险操作时，有确认、删除等提示吗。

　　23. 操作顺序是否合理。

　　24. 正确性检查：检查页面上的各类按钮，图片，标题，提示信息，还有其他文字拼写，句子的语法等是否正确。

　　25. 系统应该在用户执行错误的操作之前提出警告，提示信息.

　　26. 页面分辨率检查，在各种分辨率浏览系统检查系统界面友好性。

　　27. 合理性检查：做删除, 刷新, 添加, 取消, 返回等操作后，查看信息回到的页面是否合理。

28. 检查本地化是否通过：英文版不应该有中文信息，英文翻译准确，专业。

29.背景灰度冻结

**3.2 容错机制**

1.输入系统不允许的数据作为输入。

2.把某个相关模块或者子系统停掉，验证对当前系统的影响。

3.配置文件删除或者配置错误。

4.数据库注入错误数据。

**3.3 常规性**

1.系统不间断运行（7\*24），验证是否内存泄露、系统其他资源是否存在泄露

2.跑一晚上或者周末跑两天。

2.一般压力很大的情况下，数据库连接数问题、内存泄露问题会曝露的比较快但是死锁可能不能体现。

**3.4 性能测试**

**3.4.1.连接速度测试**

　　用户连接到Web应用系统的速度根据上网方式的变化而变化，他们或许是电话拨号，或是宽带上网。当下载一个程序时，用户可以等较长的时间，但如果仅仅访问一个页面就不会这样。如果Web系统响应时间太长（例如超过5秒钟），用户就会因没有耐心等待而离开。另外，有些页面有超时的限制，如果响应速度太慢，用户可能还没来得及浏览内容，就需要重新登陆了。而且，连接速度太慢，还可能引起数据丢失，使用户得不到真实的页面。

**3.4.2.负载测试**

　　负载测试是为了测量Web系统在某一负载级别上的性能，以保证Web系统在需求范围内能正常工作。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。例如：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？

**3.4.3.压力测试**

负载测试安排在Web系统发布以后，在实际的网络环境中进行测试。因为一个企业内部员工，特别是项目组人员总是有限的，而一个Web系统能同时处理的请求数量将远远超出这个限度，所以，只有放在Internet上，接受负载测试，其结果才是正确可信的。

　　进行[**压力测试**](https://link.zhihu.com/?target=http%3A//www.51testing.com/html/97/category-catid-97.html)是指实际破坏一个Web应用系统，测试系统的反映。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。

　　压力测试的区域包括表单、登陆和其他信息传输页面

**3.5 交互体验**

1.系统界面的控件是否可以通过tab键遍历，并且顺序合理

　　2.主要功能的入口和操作是否易于理解

　　3.界面是否布局合理，功能是否易于查找和使用

　　4.操作步骤

　　5.操作习惯

　　6.有足够的提示信息，且信息文字描述准确

**3.6 兼容性**

兼容性测试不只是指界面在不同操作系统或浏览器下的兼容，有些功能方面的测试，也要考虑到兼容性，包括操作系统兼容和应用软件兼容，可能还包括硬件兼容，比如涉及到ajax、jquery、javascript等技术的，都要考虑到不同浏览器下的兼容性问题。除了上面所说的这些测试以外，还有算法测试、配置测试、安全性测试等等，在工作中不断总结和分析，形成自己的功能测试框架。

**3.7 功能完整性**

1.使用所有默认值进行测试

　　2.根据所有产品文档、帮助文档中描述的内容要进行遍历测试

　　3.输入判断

　　4.所有界面出现是和否的逻辑，要测试

　　5.异常处理

　　6.敏感词

　　7.根据需求文档的流程图遍历所有流程图路径

　　8.根据程序内容，遍历if elif else switch的逻辑点要遍历

9.界面各种控件测试

**3.8业务流程测试**

业务流程，一般会涉及到多个模块的数据，所以在对业务流程测试时，首先要保证单个模块功能的正确性，其次就要对各个模块间传递的数据进行测试，这往往是容易出现问题的地方，测试时一定要设计不同的数据进行测试。

　　如某一功能模块具有最基本的增删改查功能，则需要进行以下测试：

　　1.单项[**功能测试**](https://link.zhihu.com/?target=http%3A//www.51testing.com/html/92/category-catid-92.html)（增加、修改、查询、删除）

　　2.增加——>增加——>增加 （连续增加测试）

　　3.增加——>删除

　　4.增加——>删除——>增加 （新增加的内容与删除内容一致）

　　5.增加——>修改——>删除

　　6.修改——>修改——>修改 （连续修改测试）

　　7.修改——>增加（新增加的内容与修改前内容一致）

　　8.修改——>删除

　　9.修改——>删除——>增加 （新增加的内容与删除内容一致）

　　10.删除——>删除——>删除 （连续删除测试）