|  |  |  |
| --- | --- | --- |
| **文件状态**  **[ ] 草稿**  **[ ] 讨论稿**  **[ ] 正式发布** | **文档类型** | **xxx** |
| **文件标识** |  |
| **版 本** | **2.2** |
| **作 者** | xxx |
| **完成日期** | **2016.5.xx** |

文档名称：xxxxx

烫烫烫烫烫小组

文档修订

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **更改人** | **描述（注明修改的条款或页）** |
| 1.0 | 2016.3.05 | 王韬懿 | 初始化配置管理计划 |
| 1.1 | 2016.3.13 | 安剑锋 | 完善初始配置管理计划 |
| 1.2 | 2016.3.14 | 安剑锋 | 增加配置标识内容 |
| 1.3 | 2016.3.15 | 安剑锋 | 编写转包商/供应商控制 |
| 1.4 | 2016.3.16 | 安剑锋 | 修改配置标识 |
| 2.0 | 2016.3.29 | 安剑锋 | 修改配置控制变更等部分 |
| 2.1 | 2016.3.30 | 安剑锋 | 完善配置项 |
| 2.2 | 2016.4.4 | 安剑锋 | 更改配置状态审计等内容 |

批准人签字

|  |  |  |
| --- | --- | --- |
| **职务** | **姓名** | **日期** |
| 项目经理 | 王韬懿 | 2016年3月5日 |
| 项目经理 | 王韬懿 | 2016年3月13日 |
| 项目经理 | 王韬懿 | 2016年3月14日 |
| 项目经理 | 王韬懿 | 2016年3月15日 |
| 项目经理 | 王韬懿 | 2016年3月16日 |
| 项目经理 | 王韬懿 | 2016年3月29日 |
| 项目经理 | 王韬懿 | 2016年3月30日 |
| 项目经理 | 王韬懿 | 2016年4月4日 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目 录**

[一测试内容及方法 4](#_Toc449444516)

[1.1测试方法 4](#_Toc449444517)

[1.2测试内容 4](#_Toc449444518)

[1.2.1个人中心模块 4](#_Toc449444519)

[1.2.2数据分类模块 5](#_Toc449444520)

[1.2.3回收站模块 5](#_Toc449444521)

[1.2.4权限模块 5](#_Toc449444522)

[2.1安全可靠性测试 5](#_Toc449444523)

[2.2易用性测试 5](#_Toc449444524)

[2.3速度测试 6](#_Toc449444525)

[2.4可扩展性测试 6](#_Toc449444526)

[2.5回归测试 6](#_Toc449444527)

[2.6兼容性测试 6](#_Toc449444528)

[2.7数据库测试 6](#_Toc449444529)

[2.8文档检查 7](#_Toc449444530)

[2.9代码评审 7](#_Toc449444531)

# 一测试内容及方法

## 1.1测试方法

本次测试方式以黑盒测试为主，个别位置采用少量的白盒测试，对项目的功能完成情况、流程规范以及其软件质量进行测试。

本次的测试对象为内容管理系统，采用的测试技术为：压力测试、功能测试、回归测试、易用性测试、性能测试

**压力测试：**压力测试是对系统不断施加压力的测试,是通过确定一个系 统的瓶颈或者不能接收的性能点,来获得系统能提供的最大服务级别的 测试。本次测试主要测试该软件在大量的信息请求负荷下,何时系统的的影响会退化或者失败。

**性能测试：**性能测试是对[软件性能](https://zh.wikipedia.org/w/index.php?title=%E8%BD%AF%E4%BB%B6%E6%80%A7%E8%83%BD&action=edit&redlink=1)的评价。简单的说，[软件性能](https://zh.wikipedia.org/w/index.php?title=%E8%BD%AF%E4%BB%B6%E6%80%A7%E8%83%BD&action=edit&redlink=1)衡量的是软件具有的响应及时度能力。因此，[性能测试](https://zh.wikipedia.org/w/index.php?title=%E6%80%A7%E8%83%BD%E6%B5%8B%E8%AF%95&action=edit&redlink=1)是采用测试手段对软件的响应及时性进行评价的一种方式。根据软件的不同类型，性能测试的侧重点也不同。

**功能测试：**按照测试软件的各个功能划分进行有条理的测试，在功能测试部分要保证测试项覆盖所有功能和各种功能条件组合。

**回归测试：**回归测试是指修改了旧代码后,重新进行测试以确认修改没 有引入新的错误或导致其他代码产生错误。本次测试在使用其他方法发 现软件的问题后,尝试进入代码修改,修改后重新进行测试。

**易用性测试：**易用性测试:易用性测试是指用户使用软件该功能时是否感觉方便。该 软件针对的是广大消费者,用户面广,易用性是一个重要的测试点,本 次测试主要针对软件界面的链接方式、事件响应方式进行测试。

## 1.2测试内容

内容管理系统的功能模块包括：个人中心模块、数据分类模块、回收站模块和权限模块。本次的测试会涵盖所有模块。

本次测试分配如下：

### 1.2.1个人中心模块

压力测试：由于同时在线人数的压力测试与服务器性能也有关系，因此个人中心模块的压力测试不做太多要求，只进行文件大小的压力测试。

性能测试：主要测试系统的交互速度，对每一个操作进行性能测试，根据系统反应的迅速程度进行评估。

功能测试：根据需求进行对系统所实现的功能的测试，评估标准为系统功能的完成度，测试功能包括我的文档、草稿箱、待审核。

### 1.2.2数据分类模块

压力测试：对可分类数据的大小和数量以及子文档嵌套层数进行压力测试。

性能测试：主要测试系统的交互速度和分类速度，对每一个操作进行性能测试，根据系统反应的迅速程度进行评估，主要对移动、复制、粘贴这三个功能进行性能测试。

功能测试：根据需求进行对系统所实现的功能的测试，评估标准为系统功能的完成度，测试子文档有关功能和文件的移动、复制粘贴功能。

### 1.2.3回收站模块

压力测试：对回收站容量进行压力测试。

性能测试：对回收站的删除速度和还原速度进行性能测试，监测交互时间。

功能测试：对回收站的删除功能和还原功能进行模拟测试。

### 1.2.4权限模块

压力测试：由于管理员数量较少权限模块不做压力测试要求。

性能测试：无

功能测试：访问授权，分类授权，成员授权，用户行为，用户信息，测试以上功能的完成情况。

## 2.1安全可靠性测试

1. 用户信息安全：测试对用户信息的保密情况。

2. 用户权限限制：考察不同的用户权限限制情况。

3. 留痕功能：系统是否有操作日志,操作日志记录的操作情况的全面性和

准确性,是否在发生非正常情况下可以进行恢复或查找到责任人信息。

4. 账号安全：对于登陆错误的提示信息，防止恶意操作如SQL注入等，还有数据库的安全问题。

5. 异常情况处理：在程序运行过程中如果发生为定义操作（强制关闭）对于现场的保护以及数据恢复等。

## 2.2易用性测试

1. 用户界面的友好性：用户界面做到布局合理、简单明了，使用户能够轻松理解界面模块内容，方便操作，功能布局与需求规格说明书规定的功能需求一致。

2. 易操作性：软件使用人员对于计算机技术的掌握情况不同，因此需要操作简便友好。

3. 界面风格的一致性：用户界面的风格和操作是否保持一致。

5. 错误提示信息：对于用户的错误操作是否有合理的提示，以及对程序运行错误的报告。

## 2.3速度测试

1. 用户的注册速度。
2. 内容模块的基本的操作：新增、编辑、禁用（启用）、删除、查询、排序等操作的速度。
3. 回收站的删除速度和还原速度。
4. 分配权限的速度。
5. 用户行为的日志查询速度。
6. 用户信息的调用速度。

## 2.4可扩展性测试

1. 权限管理的可扩展性：在新增功能后权限管理是否能够应对新增功能的权限管理。
2. 软件结构的可扩展性：是否能够灵活的支持新增功能

## 2.5回归测试

对每一个阶段所作出的针对测试内容的修改是否达到了预期的水准进行测试评估，对于没有达到预期水准的部分需要重新修改。

## 2.6兼容性测试

对于当前的主流浏览器进行兼容性测试，包括IE内核的浏览器，Google Chrome，Safari，Fire Fox等。

## 2.7数据库测试

对于数据库内部结构进行测试，数据库的设计范式是数据库设计所需要满足的规范，满足这些规范的数据库是简洁的、结构明晰的。此处检测数据库内每张表满足的范式。

## 2.8文档检查

1. 规范性检查：用户文档描述是否规范，是否有版本控制和修改记录。

2. 完整性：用户文档是否齐全，内容是否完整。

3. 一致性：文档的描述和软件实际的功能是否基本一致（既不可超出，也

不可少于）。

4. 正确性：文档是否存在会影响用户阅读的语法和字词错误。

5. 易理解程度：用户手册编写逻辑是否合理，是否图文相结合，对生僻词或自己使用的一些标记词语等是否有解释。

## 2.9代码评审

静态测试技术通常在代码完成并无错误地通过编译或汇编后进行。代码评审 （Code Review）也是一种有效的测试方法。我们提倡编写功能单一、结构清晰、 接口简单的函数，因为过于复杂的函数会给我们带来很多问题：加深其他开发人 员理解代码的难度；不方便测试人员对其编写测试用例；容易隐藏错误；出现问 题难以定位等等。

此处会采用一种代码评审工具，目前尚未确定。