|  |  |  |
| --- | --- | --- |
| **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 引言 4](#_Toc449297870)

[1.1 简介 4](#_Toc449297871)

[1.2 测试 4](#_Toc449297872)

[2.1 SQL注入简介 4](#_Toc449297873)

三、测试计划

5.测试规范性

测试流程规范性

任何阶段的测试都要遵循以下流程: 测试计划、测试设计、测试用例、执行测试、测试评审。

测试计划规范性

“5W1H“原则,明确测试内容和过程,对测试形成实质性的指导作用。

(1) What明确测试的范围和内容

(2) Why明确测试的目的

(3) When明确测试开始和结束日期

(4) Where明确给出测试文档和相关资料的位置

(5) Who明确测试人员的任务分配

(6) How明确指出测试的方法和测试工具

测试用例规范性

测试用例设计方法

(1) 等价类划分法

(2) 路径覆盖法

(3) 边界值分析法

(4) 流程分析法

(5) 场景法

(6) 其他设计方法

缺陷管理规范性

本测试项目使用专门的缺陷管理工具----禅道

测试评审规范性

6.缺陷定义

软件缺陷有四种级别，分别为：致命的(Fatal)，严重的(Critical)，一般的(Major)，微小的(Minor)。

A类—致命的软件缺陷(Fatal): 造成系统或应用程序崩溃、死机、系统挂起，或造成数据丢失，主要功能完全丧失，导致本模块以及相关模块异常等问题。如代码错误，死循环，数据库发生死锁、与数据库连接错误或数据通讯错误，未考虑异常操作，功能错误等

B类—严重错误的软件缺陷（critical）：系统的主要功能部分丧失、数据不能保存，系统的次要功能完全丧失。问题局限在本模块，导致模块功能失效或异常退出。如致命的错误声明，程序接口错误，数据库的表、业务规则、缺省值未加完整性等约束条件

C类—一般错误的软件缺陷（major）：次要功能没有完全实现但不影响使用。如提示信息不太准确，或用户界面差，操作时间长，模块功能部分失效等，打印内容、格式错误，删除操作未给出提示，数据库表中有过多的空字段等

D类—较小错误的软件缺陷（Minor），使操作者不方便或遇到麻烦，但它不影响功能过的操作和执行，如错别字、界面不规范（字体大小不统一，文字排列不整齐，可输入区域和只读区域没有明显的区分标志），辅助说明描述不清楚

E类- 建议问题的软件缺陷（Enhancemental）：由问题提出人对测试对象的改进意见或测试人员提出的建议、质疑。

7.风险管理

软件测试的风险是指软件测试过程出现的或潜在的困难或问题，软件测试风险的发生可能导致软件产品的测试不够充分或者测试结果的不准确。

软件测试的风险是不可避免的，它存在于整个软件测试过程之中，所以对软件测试风险的管理非常重要，必须尽力降低软件测试中所存在的风险， 最大程度地保证质量和满足用户的需求。

风险管理，一般可以分成5个步骤

1.风险识别: 用系统化的方法来确定威胁项目计划的因素。就是要知道风险是什么，导致风险的原因，在什么地方最易出现风险以及风险显示出的特征。

2.风险分析: 量化风险。

3.风险计划: 风险标识、风 险描述、风险影响程度、责任、资源、时间、活动、应 对措施、结果、负责人。

4.风险控制: 主要方法有风险避免、风险弱化、 风险转移和风险承担。

5.风险跟踪: 在风险受到控制以后，要及时进行跟踪，做好风险跟踪。­­

测试过程中常见的风险及应对策略

1.时间进度风险

2.产品认识风险

3.人的风险

4.质量目标风险

5.测试环境和依赖风险

6.测试充分性风险

7.工具的风险

8.变更管理

变更来源及应对办法

项目计划的变更，

第一种办法是调整测试计划中的测试策略和测试范围

其他应对办法有减少进入测试的阻力，例如降低测试计划中系统测试准入准则； 分步提交测试，例如改成迭代方式增量测试；减少回归测试的要求，例如开发人员实时修改，在测试计划中对缺陷修复响应时间和过程 进行约定：简化配置管理，跳过正式发布环节；缺陷进行局部回归而 不是重新全部测试等等。

需求的变更；

面临一个需求动态的项目，必须在计划中对需求 变更造成的测试(设计)方式变化进行说明，

测试产品版本的变更：

测试资源的变更：

参考文献：

［1］论软件测试计划的成功制定 曹耀辉，王刚

# 一 引言

## 1.1 简介

随着互联网的发展，人们在享受互联网带来的便捷的服务的时候，也面临着个人的隐私泄漏的问题。小到一个拥有用户系统的小型论坛，大到各个大型的银行机构，互联网安全问题都显得格外重要。而这些网站的背后，则是支撑整个服务的核心数据库。可以说数据库就是这些服务的命脉，没有数据库，也就无从谈起这些服务了。

## 1.2 测试

随着互联网的发展，人们在享受互联网带来的便捷的服务的时候，也面临着个人的隐私泄漏的问题。小到一个拥有用户系统的小型论坛，大到各个大型的银行机构，互联网安全问题都显得格外重要。而这些网站的背后，则是支撑整个服务的核心数据库。可以说数据库就是这些服务的命脉，没有数据库，也就无从谈起这些服务了。

二 SQL注入

## 2.1 SQL注入简介

随着互联网的发展，人们在享受互联网带来的便捷的服务的时候，也面临着个人的隐私泄漏的问题。小到一个拥有用户系统的小型论坛，大到各个大型的银行机构，互联网安全问题都显得格外重要。而这些网站的背后，则是支撑整个服务的核心数据库。可以说数据库就是这些服务的命脉，没有数据库，也就无从谈起这些服务了。