**TMS测试计划**

**组长：张昌琳**

**组员：陈毅强、李永杰、王俊皓、谢思宇、杨璞、应雯**

| **版本** | **主要作者** | **版本描述** | **日期** |
| --- | --- | --- | --- |
| V1.0 | 张昌琳、陈毅强、李永杰、王俊皓、谢思宇、杨璞、应雯 | 初始版本 | 11.30 |

**目录**

[1.前言 1](#_Toc22144)

[1.1 文档术语 1](#_Toc31370)

[1.2 参考资料 2](#_Toc16414)

[2.测试目标 2](#_Toc22842)

[2.1 测试包含项 2](#_Toc5008)

[2.2 测试不包含项 2](#_Toc18942)

[3.测试方法 3](#_Toc21270)

[3.1 测试识别和论证 3](#_Toc23704)

[3.2 执行测试 6](#_Toc5089)

[3.2.1 工作版本可接受测试 6](#_Toc27655)

[3.2.2 功能测试 6](#_Toc15684)

[3.2.3 回归测试 7](#_Toc375)

[3.2.4 容量测试与性能测试 7](#_Toc32333)

[3.2.5 集成测试 9](#_Toc6122)

[3.2.6 用户界面测试 10](#_Toc5477)

[3.2.7 安全性和访问控制测试 10](#_Toc3564)

[3.3 测试自动化策略 11](#_Toc8611)

[3.4 缺陷管理 12](#_Toc15879)

[3.5 测试度量 12](#_Toc13298)

[3.6 测试报告 13](#_Toc15326)

[4.测试进入退出标准 13](#_Toc17986)

[4.1 测试进入标准 13](#_Toc25364)

[4.2 测试退出标准 14](#_Toc12627)

[4.3 测试暂停和重启标准 14](#_Toc3687)

[5.环境配置 14](#_Toc15846)

[5.1 系统硬件 14](#_Toc16553)

[5.2 环境软件 14](#_Toc5200)

[6.职责及培训 15](#_Toc5756)

[6.1 人员与角色 15](#_Toc9665)

[6.2 培训需要 16](#_Toc20225)

[7. 时间计划 16](#_Toc17608)

[8.风险、依赖、条件与限制 17](#_Toc27345)

# 1.前言

## 1.1 文档术语

* SRS：Software Requirements Specification，软件需求说明书
* FS：Functional Specification，功能定义文档
* TMS：Task Management System，任务管理系统，即我们即将测试的系统。
* B/S系统架构：Browser/Server，[浏览器](http://baike.so.com/doc/2920715-3082096.html" \t "http://baike.so.com/doc/_blank)/[服务器](http://baike.so.com/doc/4487696-4696885.html)模式。这种模式统一了[客户端](http://baike.so.com/doc/4889711-5107810.html)，将系统功能实现的核心部分集中到[服务器](http://baike.so.com/doc/4487696-4696885.html)上，简化了系统的开发、维护和使用。
* BAT测试：Build Acceptance Testing，工作版本可接受测试
* SQL：Structured Query Language，结构化[查询](http://baike.so.com/doc/6735594.html)语言。是一种数据库[查询](http://baike.so.com/doc/6735594.html)和程序设计语言，用于存取[数据](http://baike.so.com/doc/5387430.html)以及查询、[更新](http://baike.so.com/doc/10039327.html)和管理关系数据库[系统](http://baike.so.com/doc/5402991.html)。

## 1.2 参考资料

《Task Management Functional Specification》

《Task Management Software Requirement Specification》

《软件测试》

# 2.测试目标

## 2.1 测试包含项

* 测试设计：根据FS和SRS编写测试计划和测试方案，包括测试用例、收集测试方法和测试工具。
* 功能测试：包括流程测试、边界值测试、容错性测试等。主要是根据FS和SRS检查产品是否实现了功能。
* 部分非功能测试：包括文档测试，安装测试，兼容性测试，界面测试，安全性测试，性能测试等。主要是检测该系统是否达到了FS和SRS中提到的性能方面和交互方面的要求。

## 2.2 测试不包含项

* 单元测试：由于本小组不包含开发人员故不进行单元测试而是进行集成测试。
* 验收测试：由于本系统还未达到验收阶段故不进行验收测试。
* 升级测试：由于本系统还在进行初始阶段的测试故还未涉及升级测试。
* 回归测试：由于本系统只进行一轮测试故不包含回归测试

# 3.测试方法

本章节描述了在测试过程中采用的整体策略和测试方法，测试团队的成员需要按照这个策略和方法去对TMS系统所需要进行的具体测试进行分析、设计、实现和执行。此次测试主要采用黑盒测试方法，以及手工测试为主，自动化测试为辅的策略。测试小组人员通过分析文档所描述的需求设计测试用例，测试软件的功能项。由于不清楚软件内部代码，采用黑盒测试的方式，模拟用户实际业务操作，检查每个功能是否正常，并检查异常操作。

## 3.1 测试识别和论证

根据TMS系统的特点和用户特点，考虑如下的一些测试类型：

表格 3-1：TMS系统测试类型

|  |  |  |
| --- | --- | --- |
| **测试类型** | **是否采用** | **说明** |
| 功能测试 | 采用 | 根据系统的功能定义文档（Task Management Functional Specification）和需求规格说明书（Task Management SRS），检查产品是否正确实现了功能。 |
| 流程测试 | 采用 | 按操作流程进行的测试，主要有业务流程、数据流程、逻辑流程、正反流程，检查TMS系统在按流程操作时是否能够正确执行。 |
| 边界值测试 | 采用 | 选择边界数据进行测试，确保系统功能正常，程序无异常。 |
| 容错性测试 | 采用 | 检查系统的容错能力，错误的数据输入不会对功能和系统产生非正常的影响，且程序对错误的输入有正确的提示信息。 |
| 异常测试 | 采用 | 检查系统能否处理异常 |
| 启动停止测试 | 采用 | 检查每个模块能否正常启动停止、异常停止后能否正常启动 |
| 安装测试 | 采用 | 检查系统能否正确安装、配置 |
| 易用性测试 | 采用 | 检查系统是否易用友好 |
| 界面测试 | 采用 | 检查界面是否美观合理 |
| 接口测试 | 不采用 | 检查系统能否与外部接口正常工作，由于本系统没有外部接口故不采用 |
| 配置测试 | 采用 | 检查系统配置是否合理、配置是否正常 |
| 安全性和访问控制测试 | 采用 | 该系统使用的用户有着明确的权限划分。因此需要对系统的访问控制和安全性进行测试。主要有两种安全性测试。  应用程序级别的安全性：检查用户是否只能访问其所属用户类型已被授权访问的那些功能或数据。  系统级别的安全性：检查只有具备系统和应用程序访问权限的管理员才能访问系统和应用程序。 |
| 性能测试 | 采用 | 提取系统性能数据，检查系统是否满足在需求中所规定达到的性能。 |
| 兼容性测试 | 采用 | 该系统是典型的B/S系统架构，需要考虑用户端浏览器的版本。 |
| 割接/升级测试 | 不采用 | 进行专门的割接测试或升级测试，提供工程升级割接方案。本系统已经是比较成熟的系统，该测试不适合当前系统。 |
| 文挡测试 | 采用 | 检查文档是否足够、描述是否合理。同时需要检查文档前后描述是否一致，是否可以验证。 |
| 回归测试 | 不采用 | 检查程序修改后有没有引起新的错误、是否能够正常工作以及能否满足系统的需求。由于本系统只进行一轮测试故不采用。 |

根据不同的测试方法技术，考虑如下几种测试：

表格 3-2：TMS系统测试方法技术

|  |  |  |
| --- | --- | --- |
| **测试技术** | **是否采用** | **说明** |
| 里程碑技术 | 采用 | 里程碑的达成标准及验收方法在测试完后制订 |
| 自动测试技术 | 采用 | 核心业务流程采用自动测试技术 |
| 审评测试 | 采用 | 对软件产品功能说明文档（Functional Specification）和需求规格说明书（SRS）进行检查，在需求与设计阶段进行 |
| 编写测试用例 | 采用 | 根据FS文档和SRS文档编写合适的测试用例 |
| 单元测试 | 不采用 | 由开发人员进行，但是本系统已经没有开发人员的参与，故不采用。 |
| 集成测试 | 采用 | 检测系统是否达到需求，对业务流程以及数据流程的处理是否符合规范，检测系统对业务流处理是否存在逻辑不严谨以及错误，检测是否存在不合理的标准以及要求，此阶段的测试基于功能测试的完成。 |
| 确认测试 | 采用 | 在产品发布前，对照feature list 进行基本需求的确认，确认产品是否正确实现了功能。 |
| 系统测试 | 采用 | 包括性能测试、压力测试和回归测试 |
| 验收测试 | 不采用 | 由工程实施人员进行 |
| 压力测试 | 采用 | 检查系统能否承受大压力，测试产品应该能够在高强度条件下（同时并发很多用户使用）正常运行，不会出现任何错误。 |

## 3.2 执行测试

### 3.2.1 工作版本可接受测试

表格 3-3：TMS工作版本可接受测试

|  |  |
| --- | --- |
| 测试目标： | 测试软件的基本功能和内容是否正确完整，具有可测试性 |
| 测试范围： | 系统安装是否正确，系统功能是否正确完整 |
| 技术： |  |
| 开始标准： | 开发人员向测试小组提交和传递了测试用的build版本系统 |
| 完成标准： | 确定系统功能完整或者不完整，正确或者不正确，确实是否具有可测试性。 |
| 测试重点和优先级： | 每个模块的功能，根据每个功能模块的优先级进行测试优先级的确定 |
| 需考虑的特殊事项： | 对于不同的环境，可能会得出不同的结论，因而测试小组进行测试的环境应该保持一致。 |

### 3.2.2 功能测试

表格 3-4：TMS功能测试

|  |  |
| --- | --- |
| 测试目标 | 确保测试的每个模块的功能正常，其中包括导航，数据输入，处理和检索等功能。 |
| 测试范围： | 各个模块的功能 |
| 技术： | 利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。 |
| 开始标准： | 完成了BAT测试，证明了该系统具有可测试性，并且组员根据文档完成了测试用例的编写 |
| 完成标准： | 测试覆盖到了每一个功能点，并且得出该功能是否正确运行的结论 |
| 测试重点和优先级： |  |
| 需考虑的特殊事项： | 确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的） |

### 3.2.3 回归测试

将不实施该测试。该测试在本测试项目中不适用。

### 3.2.4 容量测试与性能测试

1. **容量测试**

表格 3-5：容量测试

|  |  |
| --- | --- |
| 测试目标 | 核实测试对象在以下高容量条件下能否正常运行：  连接或模拟了最大（实际或实际允许）数量的客户机，所有客户机在长时间内执行相同的、且情况（性能）最坏的业务功能。  已达到最大的数据库大小（实际的或按比例缩放的），而且同时执行多个查询或报表事务。 |
| 测试范围： |  |
| 技术： | 使用为性能评测或负载测试制定的测试。  应该使用多台客户机来运行相同的测试或互补的测试，以便在长时间内产生最繁重的事务量或最差的事务组合。  创建最大的数据库大小（实际的、按比例缩放的、或填充了代表性数据的数据库），并使用多台客户机在长时间内同时运行查询和报表事务。 |
| 开始标准： |  |
| 完成标准： | 所计划的测试已全部执行，而且达到或超出指定的系统限制时没有出现任何软件故障。 |
| 测试重点和优先级： |  |
| 需考虑的特殊事项： | 对于上述的高容量条件，需要考虑被测试系统的环境情况 |

**b.性能测试**

表格 3-6：TMS性能测试

|  |  |
| --- | --- |
| 测试目标 | 核实所指定的事务或业务功能在以下情况下的性能行为：  正常的预期工作量  预期的最繁重工作量 |
| 测试范围： |  |
| 技术： | 使用为功能或业务周期测试制定的测试过程。  通过修改数据文件来增加事务数量，或通过修改脚本来增加每项事务的迭代数量。  脚本应该在一台计算机上运行（最好是以单个用户、单个事务为基准），并在多个客户机上重复。 |
| 开始标准： |  |
| 完成标准： | 单个事务或单个用户：在每个事务所预期时间范围内成功地完成测试脚本，没有发生任何故障。  多个事务或多个用户：在可接受的时间范围内成功地完成测试脚本，没有发生任何故障。 |
| 测试重点和优先级： |  |
| 需考虑的特殊事项： | 综合的性能测试还包括在服务器上添加后台工作量。  可采用多种方法来执行此操作，其中包括：  直接将“事务强行分配到”服务器上，这通常以“结构化语言”（SQL）调用的形式来实现。  通过创建“虚拟的”用户负载来模拟许多个（通常为数百个）客户机。此负载可通过“远程终端仿真（Remote Terminal Emulation）工具来实现。此技术还可用于在网络中加载“流量”。  使用多台实际客户机（每台客户机都运行测试脚本）在系统上添加负载。  性能测试应该在专用的计算机上或在专用的机时内执行，以便实现完全的控制和精确的评测。  性能测试所用的数据库应该是实际大小或相同缩放比例的数据库。 |

### 3.2.5 集成测试

表格 3-7：TMS集成测试

|  |  |
| --- | --- |
| 测试目标 | 检测需求中业务流程，数据流的正确性 |
| 测试范围： | 需求中明确的业务流程，或组合不同功能模块而形成一个大的功能。 |
| 技术： | 利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。 |
| 开始标准： | 单独的功能测试已经完成 |
| 完成标准： | 所计划的测试已全部执行。  所发现的缺陷已全部解决。 |
| 测试重点和优先级： | 测试重点指在业务流程本身以及数据流的过程，优先级根据文档确定 |
| 需考虑的特殊事项： | 确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的） |

### 3.2.6 用户界面测试

表格 3-8：TMS用户界面测试

|  |  |
| --- | --- |
| 测试目标 | 核实以下内容：  通过测试进行的浏览可正确反映业务的功能和需求，这种浏览包括窗口与窗口之间、字段与字段之间的浏览，以及各种访问方法（Tab键、鼠标移动、和快捷键）的使用  窗口的对象和特征（例如，菜单、大小、位置、状态和中心）都符合标准。 |
| 测试范围： |  |
| 技术： | 为每个窗口创建或修改测试，以核实各个应用程序窗口和对象都可正确地进行浏览，并处于正常的对象状态。 |
| 开始标准： |  |
| 完成标准： | 成功地核实出各个窗口都与基准版本保持一致，或符合可接受标准 |
| 测试重点和优先级： |  |
| 需考虑的特殊事项： |  |

### 3.2.7 安全性和访问控制测试

表格 3-9：TMS安全性和访问控制测试

|  |  |
| --- | --- |
| 测试目标 | 应用程序级别的安全性：核实Actor只能访问其所属用户类型已被授权访问的那些功能或数据。  系统级别的安全性：核实只有具备系统和应用程序访问权限的Actor才能访问系统和应用程序。 |
| 测试范围： |  |
| 技术： | 应用程序级别的安全性：确定并列出各用户类型及其被授权访问的功能或数据。  为各用户类型创建测试，并通过创建各用户类型所特有的事务来核实其权限。  修改用户类型并为相同的用户重新运行测试。对于每种用户类型，确保正确地提供或拒绝了这些附加的功能或数据。 |
| 开始标准： |  |
| 完成标准： | 各种已知的Actor类型都可访问相应的功能或数据，而且所有事务都按照预期的方式运行，并在先前的应用程序功能测试中运行了所有的事务。 |
| 测试重点和优先级： |  |
| 需考虑的特殊事项： | 必须与相应的网络或系统管理员一直对系统访问权进行检查和讨论。由于此测试可能是网络管理可系统管理的职能，可能会不需要执行此测试。 |

## 3.3 测试自动化策略

**自动化测试**

本次测试部分采用自动化测试技术和工具进行测试管理、控制测试实现、辅助测试分析。

**覆盖率**

由于测试使用黑盒测试方式，要求功能覆盖率达到99%以上，并对需求文档中说明的需求点，都要被测试到。

**测试工具**

表格 3-10：TMS测试工具

|  |  |  |  |
| --- | --- | --- | --- |
| 用途 | 工具 | 生产厂商/自产 | 版本 |
| 缺陷管理 | Bugzilla | Terry Weissman |  |
| 功能测试 | Selenium |  |  |
| 性能测试 | LoadRunner |  |  |

## 3.4 缺陷管理

对于软件的错误和缺陷，根据其严重程度分为五个级别：

1. **致命性错误**

数据丢失，数据计算错误、数据传递错误、对数据库造成破坏、造成操作系统或者其他支撑系统崩溃、非正常关闭和非正常死机。

1. **严重性错误**

应用系统崩溃、非正常关闭和无响应，但没有造成数据丢失。系统的主要功能不能正常实现或不完整。

1. **一般性错误**

规定的非主要功能没有实现或者不完整、影响系统运行。设计不合理造成性能低下。

1. **告警性错误**

不影响业务运行的功能问题

1. **建议**

软件设计和功能实现等不完全合理之处提出建议。

从测试结果中得到的系统缺陷要记录在缺陷报告中，并标出明确的优先级和进行排序。

采用bugzilla进行缺陷的自动化管理。

## 3.5 测试度量

**测试度量采集**

表格 3-11：测试度量采集

|  |  |  |  |
| --- | --- | --- | --- |
| 原始数据 | 来源 | 采集频率 | 负责人 |
| 测试用例数 | 测试用例文档 | 全程 |  |
| 测试缺陷数 | Bugzilla | 全程 |  |

**测试度量分析计划**

表格 3-12：测试度量分析计划

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 度量名称 | 说明 | 类别 | 基线值 | 上限值 | 下限值 |
| 缺陷严重等级比例（致命性错误为例） | 缺陷按照严重级别分布情况 |  | 5% | 10% | 0% |
| 缺陷修复率 | 已修复的缺陷占全部缺陷的比例 |  | 95% | 100% | 90% |
| 缺陷分布 | 所有缺陷在各个模块中的分布情况 |  |  |  |  |
| 测试返工率 | 缺陷重复出现导致的误工率 |  | 5% | 10% | 0% |
| 测试用例覆盖率 | 执行测试用例发现的缺陷占所有缺陷的比率 |  | 95% | 100% | 90% |

## 3.6 测试报告

**测试参考文档**

Task Management ER Diagram

Task Management Functional Specification

Task Management SRS

**测试提交文档**

SQA Test Plan

Test Report

Test Cases

# 4.测试进入退出标准

## 4.1 测试进入标准

* 测试用例、测试计划准备完全
* 开发人员的本次开发基本完毕
* 开发已经可以进入修改漏洞周期
* 已通过冒烟测试

## 4.2 测试退出标准

* 测试完所有测试用例
* 测试得出结论
* 测试达到项目最终的时间节点

## 4.3 测试暂停和重启标准

测试出重大的漏洞导致测试无需继续则可暂停测试，待重大漏洞修复后再重启测试。

# 5.环境配置

## 5.1 系统硬件

CPU：≥2.0GHz

内存：≥2.0GB

键盘：能用即可

鼠标：能用即可

显示器：能用即可

硬盘：≥100GB

硬盘转速：≥7200rpm

网线：具有良好数据传输能力

网卡：100M

## 5.2 环境软件

操作系统：Win7/8/8.1/10，MacOS系列，Linux系列

数据库平台：MySQL

Javascript版本：Javascript5.0

web服务器：Apache

浏览器：IE10及以上，Chrome45及以上

# 6.职责及培训

## 6.1 人员与角色

表格 6-1：人员与角色表

|  |  |  |
| --- | --- | --- |
| 姓名 | 角色 | 负责工作 |
| 张昌琳 | 组长 | 管理整个小组，负责SRS中5.3-5.7章描述功能的测试用例的设计、文档内容的测试、功能性测试与安全测试。 |
| 谢思宇 | 组员 | 负责SRS中5.8&5.13-5.23章描述功能的测试用例的设计、文档内容的测试、功能性测试与安全测试。 |
| 陈毅强 | 组员 | 负责SRS中5.8&5.13-5.23章描述功能的测试用例的设计、文档内容的测试、功能性测试与安全测试。 |
| 应雯 | 组员 | 负责SRS中5.9-5.12 & 5.24-5.32 章描述功能的测试用例的设计、文档内容的测试、功能性测试与安全测试。 |
| 李永杰 | 组员 | 负责SRS中5.9-5.12 & 5.24-5.32 章描述功能的测试用例的设计、文档内容的测试、功能性测试与安全测试。 |
| 杨璞 | 组员 | 负责测试计划的编写与部分非功能测试包括压力测试、兼容性测试等等 |
| 王俊皓 | 组员 | 负责测试计划的编写与部分非功能测试包括性能测试、兼容性测试等等 |

## 6.2 培训需要

* 测试用例的编写规范
* 测试执行的注意事项
* 测试工具的统一
* 性能测试等其他非功能测试的方法

# 时间计划

表格 7-1：TMS测试时间计划表

| **里程碑** | **计划开始时间** | **实际开始时间** | **计划结束时间** | **实际结束时间** |
| --- | --- | --- | --- | --- |
| 项目开始 | 11.23 |  |  |  |
| 测试计划编写 | 11.23 |  | 11.30 |  |
| 需求文档测试 | 11.23 |  | 11.30 |  |
| 测试用例编写 | 11.30 |  | 12.7 |  |
| QA-测试用例执行 | 12.7 |  | 12.18 |  |
| 测试用例回顾和测试结果总结 | 12.18 |  | 12.21 |  |
| 测试报告编写 | 12.21 |  | 12.26 |  |
| 项目结束 | 12.28 |  |  |  |

# 8.风险、依赖、条件与限制

由于本测试小组首次接触完整的测试流程，故在设计和实现方面可能都存在着一些纰漏。有些优较高难度的测试可能得不到实现，但一定会确保准备执行的测试尽善尽美实现。

另外，由于本小组不包含开发人员故不能对测试测得的漏洞进行修复，故我们只进行一轮测试，只提交测试得到的漏洞，但并不解决漏洞。