
疫情管理系统
SQA Test Plan (STP)
Version: < 1.0 >
< 09/01/2021 >

Revision History

Date	Version	Description	Prepared / Revised By
2021.01.09	1.0	第一版测试计划	胡洋凡 陈川 王绍兴 蔡灿宇 刘轩铭 杨凌霄

Table of Contents

1.	介绍.....	4
1.1	文档术语和缩略词.....	4
1.2	参考及引用.....	5
2.	测试目标	6
2.1	测试归纳	6
2.2	测试排除	6
3.	测试方法	7
3.1	测试参考及依据	7
3.2	执行测试	7
3.2.1	功能测试	7
3.2.2	性能测试	7
3.2.3	系统测试	错误!未定义书签。
3.3	自动测试方法	8
3.4	漏洞管理	8
3.5	测试指标	8
3.6	测试报告	9
4.	测试开启及终止条件.....	10
4.1	测试执行开启条件.....	10
4.2	测试执行终止条件.....	10
4.3	暂停及重启条件	10
5.	测试环境搭建	11
5.1	硬件系统	11
5.2	软件要求	11
6.	职责，人员配备和培训需求	11
6.1	人员和角色	11
6.2	人员配备和培训需求	11
7.	主要项目/里程碑.....	12
8.	风险、依赖、假设和约束	13
8.1	风险	13
1)	未能在预定时间完成	13
2)	对需求存在误解	13
3)	缺乏使用测试工具的经验	13
4)	特定需求无法在短期内进行测试	13
8.2	依赖	13
8.3	假设	13
8.4	约束	13

SQA Test Plan

1. 介绍

1.1 文档术语和缩略词

文档术语/ 缩略词	解释
单元测试	单元测试，是指对软件中的最小可测试单元进行检查和验证。对于单元测试中单元的含义，一般来说，要根据实际情况去判定其具体含义，单元就是人为规定的最小的被测功能模块。单元测试是在软件开发过程中要进行的最低级别的测试活动，软件的独立单元将在与程序的其他部分相隔离的情况下进行测试。
系统测试	系统测试，也叫组装测试或联合测试。在单元测试的基础上，将所有模块按照设计要求（如根据结构图）组装成为子系统或系统，进行系统测试。
验收测试	验收测试是部署软件之前的最后一个测试操作。在软件产品完成了单元测试、系统测试和系统测试之后，产品发布之前所进行的软件测试活动。它是技术测试的最后一个阶段，也称为交付测试。验收测试的目的是确保软件准备就绪，并且可以让最终用户将其用于执行软件的既定功能和任务。
Bug	一个软件 bug 是一个导致程序出现错误的或是不可预料的结果的错误。
Build	Build 版在软件发布上主要用于区分不同时期的版本，它是编译时的版本标记，一般序号都是递增的。可用于辨别软件的版本。版本号里面的 Build 说明这个版本是第几次编译的结果，后面一般跟数字。
功能测试	功能测试是一个质量保证过程，即对产品的各功能进行验证，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。
负载测试	负载测试（Load testing），通过测试系统在资源超负荷情况下的表现，以发现设计上的错误或验证系统的负载能力。

性能测试	性能测试是通过自动化的测试工具模拟多种正常、峰值以及异常负载条件来对系统的各项性能指标进行测试。
冒烟测试	在将代码更改嵌入到产品的源树中之前对这些更改进行验证的过程。在检查了代码后，冒烟测试是确定和修复软件缺陷的最经济有效的方法。冒烟测试设计用于确认代码中的更改会按预期工作。
质量测试	为了提供足够的信任表明实体能够满足质量要求，而在质量管理体系中实施并根据需要进行证实的全部有计划和有系统的活动
测试环境	指测试运行其上的软件和硬件环境的描述，以及任何其它与被测软件交互的软件，包括驱动和桩。
测试用例	测试用例（Test Case）是为某个特殊目标而编制的一组测试输入、执行条件以及预期结果，以便测试某个程序路径或核实是否满足某个特定需求。
测试覆盖率	覆盖率是度量测试完整性的一个手段，是测试有效性的一个度量。测试覆盖是对测试完全程度的评测。测试覆盖是由测试需求和测试用例的覆盖或已执行代码的覆盖表示的。

1.2 参考及引用

Joe Timoney, Stephen Brown , Tom Lysaght, Deshi Ye. Software Testing: Principles and Practice[M]. 机械工业出版社, 2012.

Myers G J, Sandler C, Badgett T. The art of software testing[M]. John Wiley & Sons, 2011. Patton R. Software Testing[M]. Indianapolis: Sams, 2001.

《软件设计文档国家标准》

《软件工程项目开发文档范例》

《Software Requirements edition2》Karl E. Wiegers

《软件需求》刘伟琴、刘洪涛 译

2. 测试目标

通过对该系统整体和局部的测试，不同角色的测试来验证该系统能否作为一个疫情管理系统完成相应的指标，从而进入正常使用阶段。

2.1 测试归纳

此次测试包括以下项目：

功能测试：

根据疫情管理系统的设计，整个系统可以分为 5 个模块，每个模块都有一些用例。功能性测试时基于这些用例进行测试，看实际功能是否满足了要求。同时要根据数据字典，数据库定义以及范围进行边界测试。

性能测试：

性能测试主要集中在三个方面：响应时间，错误信息，压力下的性能。

UI 测试：

需要考虑整体风格，并且测试系统对用户输入的反馈，和其对不同浏览器的支持情况。

安全性测试：

采用几种常见的攻击方式来测试系统是否可以避免被攻破，如 SQL 注入和 XSS。此外，我们还测试了需求分析报告中注明的，系统如何控制账户登录、权限等。

系统测试：

在这个测试中，此系统需要被作为整体进行测试，在功能性测试中做过的测试要再次进行，看在整体环境下是否工作正常，并且观察用户、管理员系统是否能够在操作后保持一致，前端、后端和数据库是否能互相很好配合。

2.2 测试排除

此次测试不包括以下项目：

SQL 深层安全测试：

考虑到学习层次及设计重心，仅对 SQL 做简单的安全测试，不进行更深一步地 SQL 安全性测试。

3. 测试方法

3.1 测试参考及依据

本测试计划所测试的工程项目系统在开发过程中书写过《软件需求分析报告》和《软件设计报告》，并且测试人员可以得到并参考该《用户手册》以进行测试。因此，测试人员将按照以上设计模式中对该系统的功能、性能、安全性需求以及优先级顺序的描述和要求来设计测试和执行测试。

3.2 执行测试

3.2.1 功能测试

测试范围	所有在《软件需求分析报告》描述和要求过的功能
测试目标	检测系统的实际功能是否满足了《软件需求分析报告》中的要求和描述
测试技术	黑盒测试(EP, BVA, CI, RID, EG)
自动化测试工具	NA
测试开始时间	测试用例开始设计
测试结束时间	95%的测试用例已经通过并且测试过程中发现的大部分问题得到了解决

3.2.2 性能测试

测试范围	响应时间、错误时间和压力下的性能表现
测试目标	检测系统的实际性能是否满足了《软件需求分析报告》中的要求和描述
测试技术	自动化测试
自动化测试工具	Web bench

测试开始时间	工程初版开始设计
测试结束时间	系统性能满足《软件需求分析报告》中的要求和描述

3.2.3 系统测试

测试范围	《软件需求分析报告》中的功能性和非功能性需求
测试目标	检测系统的所有模块整合起来后的整体性能
测试技术	黑盒测试 (EP, BVA, CI, RID, EG)
自动化测试工具	NA
测试开始时间	所有单个模块通过了测试并被证明能正常的运行
测试结束时间	系统整体性能满足《软件需求分析报告》中的要求和描述

3.3 自动测试方法

压力测试中，我们将使用 Jmeter 和 Badboy 作为测试和分析系统性能的测试工具。通过这个工具制造同时并发的大规模网址访问行为，并记录下网址的响应以用于测试。

3.4 漏洞管理

我们将使用 bugzilla 来管理的所有发现的缺陷。我们可以用它来搜索之前发现的缺陷和记录新发现的缺陷。同时,使用 bugzilla 可以使缺陷的的严重程度更加明显,这样我们就可以知道哪些缺陷应该先被考虑。

3.5 测试指标

1. 需求的总数量
2. 系统中发现的总缺陷。

3. 每个模块中发现的缺陷的数量。

4. 每天发现的缺陷的数量。

5. 覆盖率,这显示了测试如何覆盖需求

公式:覆盖率=需求覆盖的测试用例的数量/需求的总数

6. 执行速度,显示了测试设计的实际上的执行程度

公式:执行速度=数量的测试用例设计/测试用例总数

7. 通过率,它显示了系统在测试下的整体性能

公式:通过测试的测试用例/测试用例的总数

3.6 测试报告

在测试过程中,我们将记录所有我们发现的缺陷,以及如何找到他们的方法。然后,我们将分析数据来再出最后的结论。

测试报告主要有四个部分:系统测试的结论,测试过程,测试的数据和分析建议。测试报告主要通过分析工具和适当的图来显示测试结果,以便报告更容易理解。

4. 测试开启及终止条件

4.1 测试执行开启条件

- 所有代码都必须进行单元测试。必须完成单元和连接测试并由开发团队验收。
- 应当明确功能性需求和商业需求，并且团队对此达成一致。
- 团队须对测试计划和测试用例做很好的检查反思，并且达成一致。
- 测试环境应当准备就绪。
- 测试数据应当处于可用状态。
- QA 或测试工程师应当学习应用软件的关键知识。
- 各种资源应当准备就绪。

4.2 测试执行终止条件

- Deadline（应用上线 deadline，测试 deadline 等）。
- 带有一定通过率的测试用例。
- 测试预算耗尽。
- 所有 bug 都被修补或者关闭。
- 团队验收测试总结报告。
- 所有测试用例都被执行并且通过。
- Beta 或 alpha 版本的测试阶段结束。
- 项目风险在可接受范围之内。

4.3 暂停及重启条件

暂停测试的条件：

- 如果有任何网络问题或硬件问题，可以暂停测试。
- 发现了高危漏洞和高优先级漏洞。
- 接到了更多的来自开发团队的测试需求。
- 客户合理地变更需求。

重启测试的条件：

- 上述导致暂停的事物被解决。

5. 测试环境搭建

5.1 硬件系统

操作系统: Windows 10 64 位 浏览器: Chrome, Safari, Firefox

5.2 软件要求

WAMP (Windows+flask+Mysql+HTML)

6. 职责, 人员配备和培训需求

6.1 人员和角色

人员	角色
胡洋凡	病例监测浏览部分的压力测试以及 Api 测试, 汇总、分析测试数据
杨凌霄	主页登录部分的 test case 设计及功能测试和系统测试, 汇总、分析测试数据
刘轩铭	"新闻咨询"模块的 test case 设计及功能测试和系统测试
蔡灿宇	"同乘交通"模块的 test case 设计及功能测试和系统测试, 测试计划的撰写
王绍兴	"复工人口"模块的 test case 设计及功能测试和系统测试
陈川	"物资申领"模块的 test case 设计及功能测试和系统测试

6.2 人员配备和培训需求

我们的团队有六名成员, 合作进行整体测试任务。

为了确保测试过程继续下去顺利, 团队成员需要受到以下培训:

- 1.系统地阅读文档, 特别是用户手册, 要透彻理解其功能和系统的要求进行测试。
- 2.学习 UML 来理解插图和能够传达他们的想法与合适的图表的基本知识。

- 3.使用 bugzilla 记录和追踪发现的 bug。
- 4.学习自动化测试工具，如 JMeter 的运用。
- 5.了解常见的安全问题和方式来攻击系统的基本概念

7. 主要项目/里程碑

里程碑	Planned Start Date	Actual Start Date	Planned End Date	Actual End Date
项目/阶段开始	2020/12/26	2020/12/26	2020/12/26	2020/12/26
SQA 测试计划通过	2020/12/27	2020/12/27	2020/12/27	2020/12/27
测试资源审核	2020/12/28	2020/12/28	2020/12/28	2020/12/28
测试环境搭建	2020/12/28	2020/12/28	2020/12/28	2020/12/28
测试小组训练完成	2020/12/30	2020/12/30	2020/12/30	2020/12/30
功能性测试	2021/1/2	2021/1/2	2021/1/3	2021/1/3
系统测试	2021/1/3	2021/1/3	2021/1/3	2021/1/3
表现/失误测试	2021/1/4	2021/1/4	2021/1/4	2021/1/4
登录安全测试	2021/1/5	2021/1/5	2021/1/5	2021/1/5
项目状态评估回顾	2021/1/6	2021/1/6	2021/1/8	2021/1/8
项目/阶段结束	2021/1/8	2021/1/8	2021/1/8	2021/1/8

8. 风险、依赖、假设和约束

8.1 风险

在这个测试项目中，由于项目实际大小不大，实现的功能并不复杂，风险可以认为是较低的。但是，考虑到我们的测试小组存在经验不足等问题，所以还是需要时刻关注避免以下几点：

1) 未能在预定时间完成

应对对策：测试人员应该合理有效地合作，及时完成项目。

2) 对需求存在误解

应对对策：首先所有人都应该确保对需求的理解整个小组保持一致，详细阅读文档，根据清晰的用例来制定合理的测试计划。

3) 缺乏使用测试工具的经验

建立一个技术开发小组来搭建测试环境并且指导每位测试人员使用。

4) 特定需求无法在短期内进行测试

考虑到时间限制，高优先级的测试用例应该先进行测试。

8.2 依赖

基线测试应该根据需求分析报告等文档来设置。

测试过程应该基于功能规范来理解系统的功能描述和操作。

8.3 假设

此项目只是一个“软件质量保证与测试”课程下的实践，因此只需要记录 bug，在此项目中并不要求修复 bug。

测试中出现的 bug 都是相对独立的，没有复杂关联。一个 bug 并不会影响到随后的测试过程。

8.4 约束

因为测试人员与开发人员相对隔离，客户并不会清楚的了解整个项目的情况，开发的时间轴和测试的时间轴相差很大，所以项目测试必须根据软件需求分析报告等文档严格实施，并自行判断做出假

设。

测试项目的时间紧张，但项目的复杂性相对较低，所以可在在适当的情况下减少测试的数量来保证功能性测试的正确进行。

由于开发时间较长和开发环境未统一等问题，开发人员的环境和测试人员的环境会可能有所不同。