<基于SDN的卫星网络仿真控制系统>

测试计划

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 20/11/2023 | <1.0> | 第一版 | 王劭、余彦、徐培凯、林彤彦、顾芷瑜 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 背景 4

1.3 范围 4

1.4项目标识 5

2. 测试需求 6

3. 测试策略 7

3.1 测试类型 7

3.1.1 功能测试 7

3.1.2 用户界面测试 8

3.1.3 性能评价 9

3.2 工具 9

4. 资源 10

4.1 角色 10

4.2 系统 11

5. 项目里程碑 12

6. 可交付工件 13

6.1 测试模型 13

6.2 测试日志 13

6.3 缺陷报告 13

7. 附录 A：项目任务 14

测试计划

# 简介

## 目的

基于SDN的卫星网络仿真控制系统的这一“测试计划”文档有助于实现以下目标：

1、确定现有项目的信息和应测试的软件构件。

2、列出推荐的测试需求（高层次）。

3、推荐可采用的测试策略，并对这些策略加以说明。

4、确定所需的资源，并对测试的工作量进行估计。

5、列出测试项目的可交付元素。

## 背景

1. SDN系统部分：

SDN系统部分是本项目的核心部分之一，负责实现卫星网络仿真的功能。其主要功能包括卫星部署仿真、卫星间路由拓扑计算、卫星实时位置计算等。该部分的特性包括实时性、准确性和集中式控制等。

1. 信息采集部分：  
    信息采集部分负责从部署SDN系统的集群收集数据，提供实时的信息支持。其主要功能包括采集每个卫星的资源消耗和网络连接、点对点的流量生成和卫星上的负载模拟等。这部分的特性包括实时性、准确性和高可视化性。
2. 拍照上传部分：

拍照上传部分分为app端和web端，负责对到达指定位置的手机持有人进行拍照上传至web端展示。其主要功能包括上传手机持有者所在的经纬度、拍照上传和web显示。这部分的特性包括实时性和准确性。

本系统的构架采用分层架构，SDN系统部分、信息采集部分和拍照上传部分分别作为核心组件嵌入其中。这种模块化的构架使得各个部分可以独立开发、测试和部署。

对于卫星网络仿真，现有的工具无法很好地管理支持大规模的卫星网络仿真，这对仿真工具提出了新的要求。本项目旨在提供一套经济、高效、准确的卫星网络仿真系统，助力完成国产“星网”网络的设计与研发。

## 范围

（1）本项目的测试将包括三个阶段：单元测试、集成测试和系统测试。

在单元测试阶段，将对系统的各个独立单元进行测试，以验证其功能是否按照设计要求正常工作。测试将主要侧重于代码的逻辑正确性和边界条件的处理。

集成测试阶段旨在验证各个独立单元之间的协同工作是否符合预期。测试将集中于模块之间的接口、数据流和错误处理机制。

系统测试将全面测试整个软件系统，确保系统的功能和性能都符合用户需求和预期。测试将覆盖功能测试、性能测试、用户界面测试三个方面。

（2）本测试计划主要针对功能测试、性能测试和用户界面测试。功能测试将确保系统的各项功能按照规格书要求正常运行，而性能测试将评估系统在各种负载条件下的性能表现。

（3）接受测试的特性和功能包括，1、SDN系统部分：卫星间拓扑计算、卫星实时位置计算。2、信息采集部分：信息采集、流量生成。3、拍照上传部分：照片拍摄与显示。

（4）不接受测试的特性和功能包括，第三方库和组件的正确性和功能。

（5）风险或意外事件包括，1、测试环境的不稳定性或不准确可能会影响测试的可靠性。2、如果用户需求在测试期间发生较大变化，可能需要重新评估和修改测试计划。

（6）约束包括，1、项目有时间限制，测试计划需要在预定时间内完成。2、有限的测试人员和测试设备可能限制测试的广度和深度。3、依赖于第三方组件或服务的可用性和稳定性，可能影响测试的进行。

## 1.4项目标识

下表列出了制定*测试计划*所用的文档，并标明了文档的可用性：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **文档 （版本/日期）** | **已创建或可用** | **已被接受或已经过复审** | **作者或来源** | **备注** |
| 需求规约（v1/2023.11.7） | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 王劭、余彦、徐培凯、林彤彦、顾芷瑜 |  |
| 项目计划 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 王劭、余彦、徐培凯、林彤彦、顾芷瑜 |  |

# 测试需求

下面列出了那些已被确定为测试对象的项目。此列表说明了测试的对象。

|  |  |  |  |
| --- | --- | --- | --- |
|  | SDN系统部分 | 信息采集部分 | 拍照上传部分 |
| 功能测试需求 | 1、验证卫星仿真，确保按照用户设定的规则正确配置。  2、验证卫星间路由拓扑，确保按照正确规则进行卫星间拓扑连接。  3、验证卫星实时位置，确保模拟仿真的正确性。 | 1、验证采集到的每个卫星的资源消耗和网络连接，确保功能能够正常运作。 | 1、验证拍照上传功能，确保能够满足需求。  2、验证web显示，确保能够正确显示照片。 |
| 性能测试需求 | 1、测试拓扑计算时间，以满足实时性的特性。  2、测试模拟部署的性能，确保在高负载情况下仍能有效部署。 | 1、测试数据抓取的性能，包括数据源数量的扩展和抓取速度。  2、评估实时信息处理的性能，包括处理引擎在高负载下的稳定性。 | 1、评估用户交互的性能，包括用户操作的响应时间。  1、测试照片上传的性能。 |
| 用户界面测试需求 | 1、验证仿真界面的展示是否清晰、易操作。 | 1. 验证实时信息的展示是否清晰、易读，并及时更新。 2. 测试用户对实时信息的筛选和排序功能。 | 1. 测试拍照功能，确保拍照操作正常，图像能够正确上传。 2. 确认图像预览功能。 |

# 测试策略

## 测试类型

### 功能测试

|  |  |
| --- | --- |
| 测试目标： | 确认功能完整性： 确保系统的所有功能都按照规格书要求正常工作。  发现和修复缺陷： 识别并报告系统中的任何功能性缺陷，确保它们被及时修复。  验证系统需求： 验证每个功能是否满足用户和系统的需求。  确保功能的一致性： 确保相似功能在不同环境和条件下的一致性。  评估系统的稳定性： 测试功能对系统整体稳定性的影响，防止因某个功能故障导致系统崩溃。 |
| 方法： | 黑盒测试： 使用用户的视角进行测试，验证系统的外部行为是否符合预期。  边界值分析： 测试功能的边界条件，包括输入边界、输出边界和处理边界。  正常流程测试： 针对正常使用场景测试每个功能，确保其在正常操作下的稳定性。  异常流程测试： 模拟用户错误输入、不合理操作等异常情况，验证系统的容错性。 |
| 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。 |
| 需考虑的特殊事项： | 在不同环境、浏览器和操作系统下进行测试，确保系统在各种条件下正常工作。 |

### 用户界面测试

|  |  |
| --- | --- |
| 测试目标： | 核实以下内容：  1、通过浏览测试对象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，以及鼠标的使用  2、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。  3、评估用户界面的整体可用性，包括易理解性、学习曲线和效率。 |
| 方法： | 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。 |
| 完成标准： | 证实各个窗口都与基准版本保持一致，或符合可接受标准 |
| 需考虑的特殊事项： | 并不是所有定制或第三方对象的特征都可访问。 |

### 性能评价

|  |  |
| --- | --- |
| 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：  1、正常的预期工作量  2、预期的最繁重工作量 |
| 方法： | 使用为功能或业务周期测试制定的测试过程。  逐渐增加并发用户或请求，观察系统的性能变化。  将负载逐渐增加到系统峰值并持续一段时间，观察系统的性能变化。 |
| 完成标准： | 在每个事务所预期或要求的时间范围内成功地完成测试脚本，没有发生任何故障。 |
| 需考虑的特殊事项： | 1、确保使用真实和准确的负载模拟，以反映实际用户行为。  2、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。 |

## 工具

此项目将使用以下工具：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 工具 | 厂商/自行研制 | 版本 |
| 用于功能性测试的 ASQ 工具 | JUnit、GoTest |  | Go 1.14 JUnit 5 |
| 用于性能测试的 ASQ 工具 | LoadRunner |  | LoadRunner 2020 |
| 项目管理 | GitHub |  |  |
| DBMS 工具 | MySQL |  | MySQL 5.7 |

# 资源

## 角色

下表列出了在此项目的人员配备方面所作的各种假定。

|  |  |  |
| --- | --- | --- |
| 人力资源 | | |
| 角色 | 推荐的最少资源  （所分配的专职角色数量） | 具体职责或注释 |
| 测试设计员 | 2 | 确定测试用例、确定测试用例的优先级并实施测试用例。  职责：   * 生成测试计划 * 生成测试模型 * 评估测试工作的有效性 |
| 测试员 | 3 | 执行测试。  职责：   * 执行测试 * 记录结果 * 从错误中恢复 * 记录变更请求 |

## 系统

下表列出了测试项目所需的系统资源。

|  |  |
| --- | --- |
| 系统资源 | |
| 资源 | 名称/类型 |
| 客户端测试 PC |  |
| —包括特殊的配置需求 | CPU： Intel Core i7或同等性能的处理器，以确保处理高负载和复杂操作时的性能。 |

# 项目里程碑

|  |  |  |  |
| --- | --- | --- | --- |
| **里程碑任务** | **工作量** | **开始日期** | **结束日期** |
| 制定测试计划 | 适中 | 10.8 | 10.15 |
| 设计测试 | 适中 | 10.16 | 10.31 |
| 实施测试 | 适中 | 11.1 | 11.30 |
| 执行测试 | 适中 | 12.1 | 12.28 |
| 评估测试 | 适中 | 12.29 | 1.3 |

# 可交付工件

## 测试模型

测试模型是指在软件测试过程中采用的方法、框架或规范，用于组织、规划和执行测试活动。将要通过测试模型创建并分发的报告包括，测试计划报告、测试设计报告、测试执行报告和测试评估报告等。它有助于团队按照结构化的方式管理测试流程，确保全面、有组织地覆盖软件的各个方面。测试模型中的工件应该用 ASQ 工具来创建或引用。

## 测试日志

测试日志是在测试执行过程中记录关键信息、观察和发现的问题的文档。测试日志通常由测试人员手动创建，记录测试用例的执行情况、观察到的行为、测试环境的配置以及可能的问题描述。它是测试过程中的一个详细记录，有助于追踪和排查问题。

用来记录和报告测试结果和测试状态的方法包括： 手动测试结果记录、自动化测试脚本输出。

用来记录和报告测试结果和测试状态的工具包括：JUnit、TestNG。

## 缺陷报告

缺陷报告是指在测试过程中发现的软件缺陷的记录，其中包括缺陷的描述、影响程度、重现步骤、发现者等信息。缺陷报告通常由测试人员创建并提交给开发团队，用于修复和跟踪缺陷的解决进度。

用来记录、跟踪和报告测试中发生的意外情况及其状态的方法包括：手动缺陷跟踪、自动化测试脚本输出。

用来记录、跟踪和报告测试中发生的意外情况及其状态的工具包括：Redmine。

# 附录 A：项目任务

以下是一些与测试有关的任务：

• 制定测试计划

* + 确定测试需求
  + 评估风险
  + 制定测试策略
  + 确定测试资源
  + 创建时间表
  + 生成测试计划

• 设计测试

- 准备工作量分析文档

- 确定并说明测试用例

- 确定并结构化测试过程

- 复审和评估测试覆盖

• 实施测试

* + 记录或通过编程创建测试脚本
  + 确定设计与实施模型中的测试专用功能
  + 建立外部数据集

• 执行测试

- 执行测试过程

- 评估测试的执行情况

- 恢复暂停的测试

- 核实结果

- 调查意外结果

- 记录缺陷

• 评估测试

- 评估测试用例覆盖

- 评估代码覆盖

- 分析缺陷

- 确定是否达到了测试完成标准与成功标准