<智能医学图像分割系统>

测试计划 period2

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <18/11/2018> | <1.0> | <测试计划period2> | <Group 5> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 数据和数据库完整性测试 7

3.1.2 功能测试 7

3.1.3 业务周期测试 9

3.1.4 用户界面测试 10

3.1.5 性能评价 11

3.1.6 负载测试 12

3.1.7 强度测试 13

3.1.8 容量测试 14

3.1.9 安全性和访问控制测试 15

3.1.10 故障转移和恢复测试 16

3.1.11 配置测试 18

3.1.12 安装测试 19

3.2 工具 20

4. 资源 21

4.1 角色 21

4.2 系统 23

5. 项目里程碑 24

6. 可交付工件 25

6.1 测试模型 25

6.2 测试日志 25

6.3 缺陷报告 25

7. 附录 A：项目任务 26

测试计划

# 简介

## 目的

本文档描述了测试智能医学图像分割系统的计划。此测试计划文档支持以下目标：

* 确定现有项目信息和应测试的软件组件。
* 列出推荐的测试要求（高级别）。
* 推荐并描述要采用的测试策略。
* 确定所需资源并提供测试工作的估算。
* 列出测试活动的可交付元素。

## 背景

* 主要的功能和特性：本软件的总体效果是能够实现医学图片的自动分割以及分割数据的共享。智能医学图像分割系统是一个医学图像自动分割软件，通过上传的方式实现医学图像的共享，用计算机图像处理和分析技术，对医学图像进行处理和分析，旨在提高影像科医师分析图像的工作效率，提高影像诊断的准确性。用户只需要进行基本的图像导入等操作就可以实现相应图像的分割目的，包括腹部脂肪分割，颅脑分割，右心室分割三种类型。软件分割结束后，用户可将分割的结果导出，用于进一步的诊断。
* 测试对象的架构：详情请见 智能医学图像分割系统架构文档。
* 项目简史：本项目开发至今已经完成第一个迭代，正在进行第二个迭代。基本的前端和服务器框架已经完成。
* 项目已有的参考文档：
* 智能医学图像分割系统立项建议书
* 智能医学图像分割系统架构文档
* 智能医学图像分割系统需求分析

## 范围

该测试计划适用于将在智能医学图像分割系统上进行的集成和系统测试。计划所针对的测试类型为功能测试和性能测试。

* 各个阶段测试：
* 单元测试：对于选择分割类型、选择分割方式等小单元进行模块化的检验和测试。
* 集成测试：将前端和后端的各种功能集成起来，完成一个整体的软件流程测试，测试需要遍历每部分的小功能。
* 测试对象中将接受测试特性和功能有：
* 图像分割功能
* 容量：系统可容纳的客户数为100，响应时间不超过10s；
* 对事务的响应时间：10mins；吞吐量，例如每秒处理的事务数
* 资源利用情况：对内存、磁盘的占用率不超过50%；
* 列出可能会影响测试设计、开发或实施的所有风险或意外事件:
* 项目开发的迭代没有按照计划准时完成
* 列出可能会影响测试设计、开发或实施的所有约束:
  + 开发时间有限
  + 开发人员都来自软件学院以外的学院，对于软件开发掌握的知识有限

## 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 | 🗹 是 🞎 否 | 🗹 是 🞏 否 | 第五小组 |  |
| 功能性规约 | 🞏 是 🞏 否 | 🞏 是 🞏 否 |  |  |
| 用例报告 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 第五小组 |  |
| 项目计划 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 第五小组 |  |
| 设计规约 | 🞎 是 🞏 否 | 🞏 是 🞏 否 |  |  |
| 用户手册 | 🞏 是 🞏 否 | 🞏 是 🞏 否 |  |  |
| 业务模型或业务流程 | 🗹 是 🞏 否 | 🞏 是 🞏 否 | 第五小组 |  |
| 数据模型或数据流 | 🗹 是 🞏 否 | 🞏 是 🞏 否 | 第五小组 |  |
| 项目或业务风险评估 | 🞏 是 🞏 否 | 🞏 是 🞏 否 |  |  |

# 测试需求

下面列出了那些已被确定为对象的项目（用例、功能性需求和非功能性需求）。此列表说明了测试的对象。

|  |  |
| --- | --- |
| * 测试名称 | * 测试对象及测试目的 |
| 数据以及数据库完整性测试 | * 验证是否正确检索了数据库数据的更新。 * 验证是否能够将手动处理的结果上传到数据库端 |
| 系统测试（功能测试） | * 验证图片自动分割用例 * 验证图像基本处理操作用例 * 验证图片手动分割用例 * 验证处理结果下载用例 |
| 性能测试 | * 验证图像分割的准确性 * 验证自动分割图片的响应时间。 * 验证提交手动处理结果的响应时间 |
| 商业周期测试 | * 下载分割图像后验证操作。 * 分割方式选择跳转时，验证操作是否正确。 * 验证样本屏幕是否符合GUI标准。 |
| 压力测试 | * 在黄金时段使用服务器期间验证系统响应。 * 验证系统响应 |

# 测试策略

## 测试类型

### 数据和数据库完整性测试

|  |  |
| --- | --- |
| 测试目标： | * 数据：存储在数据库中的所有数据值均正确的状态； * 数据库：确保数据库访问方法和进程正常运行，数据不会遭到损坏。 |
| 方法： | * 拟采用的测试方法：边界值、等价值法 * 数据完整性：使用约束实现数据完整性；使用规则和默认值；调用各个数据库访问方法和进程，并在其中填充有效的和无效的数据或对数据的请求； * 数据库完整性： 检查数据库，确保数据已按预期的方式填充，并且所有数据库事件都按正常方式出现；或者检查所返回的数据，确保数据格式及结果正确 |
| 完成标准： | * 数据完整性：实现实体完整，域完整，引用完整等； * 数据库完整性：所有的数据库访问方法和进程都按照设计的方式运行，数据没有遭到损坏。 |
| 需考虑的特殊事项： | * 测试可能需要数据库管理系统，以便定义数据库的模式结构与权限约束，实现对数据的追加、删除等操作。 * 进程应该以手工方式调用。 |

### 功能测试

|  |  |
| --- | --- |
| 测试目标： | 确保医学图像分割系统的功能正常，其中包括基本图像处理、手动分割、自动分割。 |
| 方法： | 基于黑盒方法，即通过图形用户界面 (GUI) 与图像分割系统交互并分析输出结果来验证图像分割系统及其内部进程。手动输入有效和无效的数据来执行各个用例，以核实以下内容：   * 点击基本图像处理工具，确保能执行各个操作； * 点击进行图像分割（自动和手动），确保各功能都得到了正确的应用。 |
| 完成标准： | * 方法中涉及到的测试已全部执行。 * 方法中涉及到的缺陷已全部解决。 |
| 需考虑的特殊事项： | * 手动测试输入的局限性 |

### 业务周期测试

|  |  |
| --- | --- |
| 测试目标 | 确保测试对象及后台进程都按照所要求的业务模型和时间表正确运行。 |
| 方法： | * 通过执行以下活动，测试将模拟若干个业务周期： * 将修改或增强对医学图像分割系统进行的功能测试，以增加每项功能的执行次数，从而在指定的时段内模拟若干个不同的用户。 * 将在适当的时候执行或启动所有周期性出现的功能。 * 在测试中还将使用有效的和无效的数据，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。 |
| 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： | 需要通过业务模型（医学图像分割算法模型）来确定相应的测试需求和测试过程。 |

### 用户界面测试

|  |  |
| --- | --- |
| 测试目标： | 确保用户界面(UI)向用户提供了适当的访问和浏览测试对象功能的操作，确保 UI 功能内部的对象符合预期要求，核实以下内容：   * 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。 |
| 方法： | 手动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： | * 成功核实各个窗口都与基准版本保持一致 |

### 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： * 正常的预期工作量、响应时间、事务处理速率 * 预期的最繁重工作量、响应时间、事务处理速率 |
| 方法： | * 使用为功能或业务周期测试制定的测试过程（手动过程） * 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次数。 * 脚本应该在一台计算机上运行，并在多台客户机上重复 |
| 完成标准： | * 单个事务或单个用户：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。 * 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。 |
| 需考虑的特殊事项： | * 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与实际大小相同或等比例缩放的数据库。 |

### 负载测试

|  |  |
| --- | --- |
| 测试目标： | * 核实医学图像分割系统在不同的工作量条件下的性能行为时间，确保系统在超出最大预期工作量的情况下仍能正常运行 |
| 方法： | * 使用为功能或业务周期测试制定的测试： * 通过修改数据文件来增加事务数量，或通过修改测试来增加每项事务发生的次数 |
| 完成标准： | * 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，没有发生任何故障。 |
| 需考虑的特殊事项： | * 负载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； * 负载测试所用的数据库应该是与实际大小相同或等比例缩放的数据库。 |

### 强度测试

|  |  |
| --- | --- |
| 测试目标： | * 核实测试对象能够在以下强度条件下正常运行，不会出现任何错误： * •连接或模拟了最大实际（或实际可承受）数量的客户机。 * •多个用户对相同的数据/账户执行相同的事务。 * •最繁重的事务量或最差的事务组合（请参见上面的“性能测试”）。 |
| 方法： | * •使用多个浏览器对服务器进行访问操作，检验最大实际可同时连接的客户浏览器端数量是否满足。。 * •在其上基础上，在不同的客户浏览器端运行相同的操作或互补的操作，以产生最繁重的事务量或最差的事务组合。 |
| 完成标准： | 所计划的测试已全部执行，并且在达到或超出指定的系统限制时没有出现任何软件故障，或者导致系统出现故障的条件并不在指定的条件范围之内。 |
| 需考虑的特殊事项： | * •根据我们的测试结果，确定我们需要的服务器数量。 * •根据我们非功能性需求和实际测试医学图像分割速度，确定我们需要的服务器配置。 * •使用多个客户浏览器端对相同的记录或数据账户同时进行的访问时，测试是否达到同布，没有错误。 |

### 容量测试

|  |  |
| --- | --- |
| 测试目标： | * [核实测试对象在以下大容量条件下能否正常运行： * •连接（或模拟了）最大（实际或实际可承受）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）最差的业务功能。 * •已达到最大的数据库大小（实际的或按比例缩放的），而且同时执行了多个查询或报表事务。 |
| 方法： | * •使用为性能评价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（请参见上面的“强度测试”）。 * •创建最大的数据库大小（实际的、按比例缩放的、或输入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。 |
| 完成标准： | * •所计划的测试已全部执行，而且在达到或超出指定的系统限制时没有出现任何软件故障。 |
| 需考虑的特殊事项： | * 服务器何时需要时刻保持该大容量条件？ * 如何能够做到灵活调整服务器容量？ |

### 安全性和访问控制测试

|  |  |
| --- | --- |
| 测试目标： | * 应用程序级别的安全性：核实主角只能访问其所属用户类型已被授权使用的那些功能或数据。 * 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主角才能访问系统和应用程序。 |
| 方法： | 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使用的功能或数据，例如增删改数据库，训练模型权限，更新模型权限等。   * •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。 |
| 完成标准： | * 各种已知的主角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。 |
| 需考虑的特殊事项： | * 只需要给用户增加、删除和修改自己对应图像数据的权限，但是不能处理其他用户数据的权限。而管理员具有增删改所有数据的权限，以及训练更新训练模型的权限。 |

### 故障转移和恢复测试

|  |  |
| --- | --- |
| 测试目标： | * [确保恢复进程（手工或自动）将数据库、应用程序和系统正确地恢复到了预期的已知状态。测试中将包括以下各种情况： * •服务器断电。 * •通过网络服务器产生的通信中断。 * •周期未完成（数据过滤进程被中断，数据同步进程被中断）。 * •数据库指针或关键字无效。 * •数据库中的数据元素无效或遭到破坏。 |
| 方法： | * 应该使用为功能和业务周期测试创建的测试来创建一系列的事务。一旦达到预期的测试起点，就应该分别执行或模拟以下操作： * •服务器断电：模拟或启动服务器的断电过程。 * •通过网络服务器产生的中断：模拟或启动网络的通信中断（实际断开通信线路的连接或关闭网络服务器或路由器的电源）。 * •数据库的内容无效或者遭到破坏：在服务器关闭的情况下手动修改数据库的内容。 * 一旦实现了上述情况（或模拟情况），就应该执行其他事务。而且一旦达到第二个测试点状态，就应调用恢复过程。 * 在测试不完整的周期时，所使用的方法与上述方法相同，只不过应异常终止或提前终止数据库进程本身。 * 对以下情况的测试需要达到一个已知的数据库状态。当破坏若干个数据库字段、指针和关键字时，应该以手工方式在数据库中（通过数据库工具）直接进行。其他事务应该通过使用“应用程序功能测试”和“业务周期测试”中的测试来执行，并且应执行完整的周期。 |
| 完成标准： | * 在所有上述情况中，应用程序、数据库和系统应该在恢复过程完成时立即返回到一个已知的预期状态。此状态包括仅限于已知损坏的字段、指针或关键字范围内的数据损坏，以及表明进程或事务因中断而未被完成的报表。 |
| 需考虑的特殊事项： | * •恢复测试会给其他操作带来许多的麻烦。断开缆线连接的方法（模拟断电或通信中断）可能并不可取或不可行。所以，可能会需要采用其他方法，例如诊断性软件工具。 * •需要系统（或计算机操作）、数据库和网络组中的资源。 * •这些测试应该在工作时间之外或在一台独立的计算机上运行。 |

### 配置测试

|  |  |
| --- | --- |
| 测试目标： | * 核实测试对象可在要求的硬件和软件配置中正常运行。 |
| 方法： | * •使用功能测试脚本。 * •执行所选的事务，以模拟主角与测试对象软件和非测试对象软件之间的交互。 * •逐渐增加连接用户的数量和用户操作的复杂度，测试服务器的配置是否能够承担相应任务。 |
| 完成标准： | * 对于测试对象软件和非测试对象软件的各种组合，所有事务都成功完成，没有出现任何故障。 |
| 需考虑的特殊事项： | * •我们需要测试在计划最大压力情况下，服务器配置是否能够很好地完成应有的工作。 |

### 安装测试

|  |  |
| --- | --- |
| 测试目标： | * 核实在用户网络连接完好的情况下，能够通过浏览器登陆我们的浏览器网站。 |
| 方法： | * •在网络连接完好的电脑上尝试登陆网站，查看能否登陆并使用软件的相关功能。 |
| 完成标准： | * 事务成功执行，没有出现任何故障。 |
| 需考虑的特殊事项： | * 由于我们的软件为BS架构，无需安装，因此只要在其他电脑上能够登陆网站，实现功能即可。 |

## 工具

此项目将使用以下工具：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 工具 | 厂商/自行研制 | 版本 |
| 测试管理 | Rational RequistePRO Rational Unified Process |  |  |
| 试验设计 | Rational Rose |  |  |
| 缺陷跟踪 | Rational ClearQuest |  |  |
| 功能性测试 | Rational Robot |  |  |
| 性能测试 | Rational Visual Quantify |  |  |
| 测试覆盖监测器或评价器 | Rational Visual PureCoverage |  |  |
| 其他测试工具 | Rational Purify  Rational TestFactory |  |  |
| 项目管理 | Microsoft Project  Microsoft Word  Microsoft Excel | 2016 |  |
| DBMS 工具 | MySQL | 8.0.12 |  |

# 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 |
| 测试项目经理 | 1 | 进行管理监督。  职责：   * 提供技术指导 * 获取适当的资源 * 提供管理报告 |
| 测试设计员 | 1 | 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   确定并定义测试类的操作、属性和关联。  职责：   * 确定并定义测试类 * 确定并定义测试包 |
| 测试员 | 2 | 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 |
| 测试系统管理员兼数据库管理员 | 1 | 确保测试环境和资产得到管理和维护。  职责：   * 管理测试系统 * 授予和管理角色对测试系统的访问权   确保测试数据（数据库）环境和资产得到管理和维护。  职责：   * 管理测试数据（数据库） |
| 实施员 | 1 | 实施测试类和测试包，并对它们进行单元测试。  职责：   * 创建在测试模型中实施的测试类和测试包 |

## 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 |
| 数据库服务器 |  |
| —服务器名服务器名 | ImgProcess |
| —数据库名 | Numpy |
| 客户端测试 PC | 操作系统为64位的普通台式机或者笔记本电脑 |
| 测试存储库 |  |
| —服务器名服务器名 | ImgProcess |
| 测试开发 PC | 操作系统为64位的普通台式机或者笔记本电脑 |

# 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 | 3days | 2018/12/3 | 2018/12/5 |
| 设计测试 | 8days | 2018/12/6 | 2018/12/13 |
| 实施测试 | 10days | 2018/12/14 | 2018/12/23 |
| 执行测试 | 5days | 2018/12/24 | 2018/12/28 |
| 评估测试 | 3days | 2018/12/29 | 2018/12/31 |

# 可交付工件

* [本节列出了将要创建的各种文档、工具和报告，及其创建人员、交付对象和交付时间。]

|  |  |  |  |
| --- | --- | --- | --- |
| 文档/报告类型 | 创建人员 | 交付对象 | 交付时间 |
| 测试计划 | 测试项目经理 | 测试项目经理 | 2018/12/15 |
| 测试设计报告 | 测试设计员 | 测试项目经理 | 2018/12/22 |
| 测试日志 | 测试员，实施员，测试系统兼数据库管理员 | 测试设计员，测试项目经理 | 2018/12/27 |
| 缺陷报告 | 测试员，实施员，测试系统兼数据库管理员 | 测试设计员，测试项目经理 | 2018/12/31 |

## 测试模型

测试计划，测试设计报告，测试日志，缺陷报告。

## 测试日志

Microsoft Word

Microsoft Excel

图文记录测试过程。

## 缺陷报告

Microsoft Word

Microsoft Excel

图文记录测试过程中的缺陷。