<银行自动柜员机(ATM)模拟软件>

测试计划

版本 <1.0>

修订版历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <24日/10月/2022年> | <1.0> | < T18组ATM机模拟项目缺陷记录> | <牟战> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

测试计划

# 1.     简介

## 1.1     目的

<银行自动柜员机(ATM)模拟软件> 的这一“测试计划”文档有助于实现以下目标：

确定现有项目的信息和应测试的软件构件。

列出推荐的测试需求（高级需求）。

推荐可采用的测试策略，并对这些策略加以说明。

确定所需的资源，并对测试的工作量进行估计。

列出测试项目的可交付元素

## 1.2     背景

## <银行自动柜员机(ATM)模拟软件>只有项目旨在制作一个银行ATM存取款系统。这个系统既是一个多端交互的系统，也是对于安全性、及时性、鲁棒性有着高需求的系统。基于此目的，该软件在正式发布之前对测试的水平和方式提出了较高的要求。在测试时首先要关注系统功能方面的测试，对系统的所有用例、以及用户界面、数据和数据库进行测试，然后对非功能性需求如负载、强度、安全性等进行测试，力求保证无死角测试，谋求最好的测试效果。

## 1.3     范围

## 系统所包含的所有用例、系统的相关功能性需求：用户界面测试、性能测试、数据和数据库测试。以及非功能性测试：负载测试、强度测试、容量测试、故障恢复和转移测试、安全性测试。在测试过程中可能会出现某一功能没有达到测试所预期效果，需要将问题及时反馈给项目经理和设计师。1.4               项目核实

下表列出了制定*测试计划*时所使用的文档，并标明了各文档的可用性：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **文档 （版本/日期）** | **已创建或可用** | **已被接收或已经过复审** | **作者或来源** | **备注** |
| 需求规约 | 🗹 是  否 | 🗹 是  否 | 系统分析员 |  |
| 功能性规约 | 🗹 是  否 | 🗹 是  否 | 系统分析员 |  |
| 用例报告 | 🗹 是  否 | 🗹 是  否 | 系统分析员 |  |
| 项目计划 | 🗹 是  否 | 🗹 是  否 | 项目经理 |  |
| 设计规约 | 🗹 是  否 | 🗹 是  否 | 系统分析员 |  |
| 用户界面设计 | 🗹 是  否 | 🗹 是  否 | 界面设计员 |  |

# 2.     测试需求

下表用于确定被当作测试对象的各项需求（例如用例、功能性需求和非功能性需求）。 表中列出了将要测试的对象。

|  |
| --- |
| 用例 |
| 1.取款 |
| 2.存款 |
| 3.查询余额 |
| 4.修改密码 |
| 5.转账 |
| 6.付款 |
| 7.添加现金 |
| 8.维护ATM设备 |
| 9.收款 |

|  |
| --- |
| 功能性需求 |
| 1.数据和数据库完整测试 |
| 2.用户界面测试 |
| 3.性能评测 |
| 4.功能测试 |
| 5.业务周期测试 |

|  |
| --- |
| 非功能性测试 |
| 1.负载测试 |
| 2.强度测试 |
| 3.容量测试 |
| 4.安全性和访问控制测试 |
| 5.故障转移和恢复测试 |

# 3.     测试策略

## 3.1               测试类型

### 3.1.1     数据和数据库完整性测试

|  |  |
| --- | --- |
| 测试目标： | 确保数据库访问方法和进程正常运行，数据不会遭到损坏。 |
| 技术： |  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）。   检查数据库，确保数据已按预期的方式填充， 并且所有的数据库事件都已正常发生；或者检查 所返回的数据，确保为正当的理由检索到了 正确的数据 |
| 完成标准： | 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 DBMS 开发环境或驱动程序 在数据库中直接输入或修改数据。   进程应该以手工方式调用。   应使用小型或最小的数据库（记录的数量有限） 来使所有无法接受的事件具有更大的可视度。 |

### 3.1.2     功能测试

|  |  |
| --- | --- |
| 测试目标： | 确保测试对象的功能正常，其中包括导航、数据输入、处理和检索等功能。 |
| 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。 |
| 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： | 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） |

### 3.1.3     业务周期测试

### 

|  |  |
| --- | --- |
| 测试目标 | 确保测试对象及背景的进程都按照所要求的业务模型和时间表正确运行。 |
| 技术： | 通过执行以下活动，测试将模拟若干个业务周期：   将修改或改进对测试对象进行的功能 测试，以增加每项功能的执行次数，从而在指定的时间段内模拟 若干个不同的用户。   将使用有效的和无效的数据或时间段来执行 所有与时间或数据相关的功能。   将在适当的时间执行或 启用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。   各业务规则都得到了正确的应用。 |
| 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： |  系统日期和事件可能需要特殊的支持活动   需要通过业务模型来确定相应的测试需求 和测试过程。 |

### 3.1.4     用户界面测试

|  |  |
| --- | --- |
| 测试目标： | 核实以下内容：   通过测试对象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 健、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。 |
| 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。 |
| 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] |
| 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。 |

### 3.1.5     性能评测

|  |  |
| --- | --- |
| 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 |
| 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（虚拟的或 实际的客户机，请参见下面的“需要考虑的特殊事项”）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： | 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“结构化查询语言”(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。 |

### 3.1.6     负载测试

|  |  |
| --- | --- |
| 测试目标： | 核实所指定的事务或商业理由在不同的工作量条件下的性能行为时间。 |
| 技术： |  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实际大小或相同缩放比例 的数据库。 |

### 3.1.7     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：   服务器上几乎没有或根本没有可用的内存（RAM 和 DASD）   连接或模拟了最大实际（实际允许）数量的客户机   多个用户对相同的数据或账户执行相同的事务   最繁重的事务量或最差的事务组合 |
| 技术： |  使用为性能评测或负载测试制定的测试。   要对有限的资源进行测试，就应该在一台计算机上运行测试，而且 应该减少或限制服务器上的 RAM 和 DASD。   对于其他强度测试，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： |  如果要增加网络工作强度，可能会需要使用网络工具来给网络加载 消息或信息包。   应该暂时减少用于系统的 DASD，以 限制数据库可用空间的增长。   使多个客户机对相同的记录或数据账户 同时进行的访问达到同步。 |

### 3.1.8     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），而且同时执行了 多个 查询或报表事务。 |
| 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（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。 |
| 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： | 对于上述的高容量条件，哪个时间段是可以接受的时间？ |

### 3.1.9     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。 |
| 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。           系统级别的访问：请参见以下的“需考虑的特殊事项” |
| 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。 |
| 需考虑的特殊事项： | 必须与相应的网络或系统管理员一起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或系统管理的职能，可能会不需要执行此测试。 |

### 3.1.10     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：   客户机断电   服务器断电   通过网络服务器产生的通信中断   DASD 和/或 DASD 控制器被中断、断电或与 DASD 和/或DASD 控制器的通信中断   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 |
| 技术： | 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：   客户机断电： 关闭 PC 机的电源。   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）。   DASD 和 DASD 控制器被中断、断电或与 DASD 和 DASD 控制器的通信中断：模拟与一个或多个 DASD 控制器或设备 的通信，或实际取消 这种通信。  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，所使用的技术与上述技术相同，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。 |

### 3.1.11     安装测试

|  |  |
| --- | --- |
| 测试目标： | 核实在以下情况下，测试对象可正确地安装到各种所需的硬件配置中：   首次安装。以前从未 安装过 <项目名称> 的新计算机   更新。 以前安装过相同版本的 <项目名称> 的计算机   更新。 以前安装过 <Project Name> 的较早 版本的计算机 |
| 技术： |  手工开发脚本或开发自动脚本，以验证 目标计算机的状况   启动或执行安装。   使用预先确定的功能测试脚本子集来运行 事务。 |
| 完成标准： | <银行自动柜员机(ATM)模拟软件>事务成功执行，没有出现任何故障。 |
| 需考虑的特殊事项： | 应该选择 <银行自动柜员机(ATM)模拟软件> 的哪些事务才能准确地测试出 <银行自动柜员机(ATM)模拟软件> 应用程序已经成功安装，而且没有遗漏主要的软件构件？ |

## 3.1.12 用例测试

|  |  |
| --- | --- |
| 测试目标： | 对系统所涉及的所有用例的基本事件流和备选事件流进行测试，以及尽力考虑出多种可能发生的情况，一并进行测试。 |
| 技术： | 在用户界面对每个测试用例输入各式各样的内容，观察各测试用例执行完毕后的结果 |
| 完成标准： | 所有测试用例执行结果无异常 |
| 需考虑的特殊事项： | 哪些情况可能会出现在用户的实际使用过程中，怎样能更全面的对用例进行测试。 |

## 3.2     工具

此项目将使用以下工具：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 工具 | 产商/自产 | 版本 |
| 测试管理 | RequisitePro | Rational |  |
| 缺陷跟踪 | ClearQuest | Rational |  |
| 用于功能性测试的 ASQ 工具 | Rational Robot | Rational |  |
| 用于性能测试的 ASQ 工具 | Visual Quantify | Rational |  |
| 测试覆盖监测器或评测器 | Visual Pure Coverage | Rational |  |
| 项目管理 | Project Word Excel | microsoft |  |
| DBMS 工具 |  |  |  |

# 4.     资源

## 4.1     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 | 所推荐的最少资源  （所分配的专职角色数量） | 具体职责或注释 |
| 测试经理，  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 |
| 测试员 | 1 | 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 | 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           分配和管理角色对测试系统的访问权 |
| 数据库管理员 | 1 | 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） |
| 设计员 | 1 | 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 |
| 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 |

## 4.1     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 |  |
| -网络或子网 | TBD |
| -服务器名称 | TBD |
| -数据库名称 | TBD |
| 客户端测试 PC |  |
| -包括特殊的配置需求 | TBD |
| 测试存储库 |  |
| -网络或子网 | TBD |
| -服务器名称 | TBD |
| 测试开发 PC | TBD |

# 5.     项目里程碑

|  |  |  |  |
| --- | --- | --- | --- |
| **里程碑任务** | **工作** | **开始日期** | **结束日期** |
| 制定测试计划 | 完成测试计划的编写 | 2022.10.7 | 2022.10.12 |
| 设计测试 | 对不同测试类型编写测试设计 | 2022.10.13 | 2022.10.18 |
| 实施测试 | 按照测试计划和设计的测试进行测试 | 2022.11.3 | 2022.11.7 |
| 执行测试 | 执行相关测试策略 | 2022.11.8 | 2022.11.15 |
| 对测试进行评估 | 对测试结果进行总结和反馈 | 2022.11.16 | 2022.11.18 |

# 6.     可交付工件

# 1.完整的测试模型 2.丰富的测试记录 3.记录测试过程中相关问题的缺陷报告7.     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

- 确定测试过程，并建立测试过程的结构

- 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

 对测试进行评估

- 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