刘鹭语l

**华东师范大学 软件工程学院**

性能测试计划

MeetHere项目

目录

[1 项目概要介绍 4](#_Toc28644926)

[1.1 项目简介 4](#_Toc28644927)

[1.2 项目成员 4](#_Toc28644928)

[1.3 测试范围 4](#_Toc28644929)

[1.3.1 范围内 4](#_Toc28644930)

[1.3.2 范围外 4](#_Toc28644931)

[1.4 前提假设 4](#_Toc28644932)

[1.5 测试目标 4](#_Toc28644933)

[2 性能测试策略 5](#_Toc28644934)

[2.1 性能测试模型 5](#_Toc28644935)

[2.2 性能测试场景 6](#_Toc28644936)

[2.3 重点测试策略 6](#_Toc28644937)

[2.3.1 重点测试原则 6](#_Toc28644938)

[2.3.2 重点测试交易 6](#_Toc28644939)

[3 测试案例设计 6](#_Toc28644940)

[3.1 生产压力分析 6](#_Toc28644941)

[3.2 场景通过标准 6](#_Toc28644942)

[3.3 测试场景设计 7](#_Toc28644943)

[3.3.1 独立场景 7](#_Toc28644944)

[3.3.2 混合场景 9](#_Toc28644945)

[3.3.3 峰值场景 10](#_Toc28644946)

[4 测试实施安排 10](#_Toc28644947)

[4.1 测试进度 10](#_Toc28644948)

[4.2 测试流程 11](#_Toc28644949)

[4.3 测试报告需求 11](#_Toc28644950)

[4.4 性能缺陷管理 11](#_Toc28644951)

[5 性能测试标准 11](#_Toc28644952)

[5.1 启动标准 11](#_Toc28644953)

[5.2 中止标准 12](#_Toc28644954)

[5.3 通过标准 12](#_Toc28644955)

[6 测试环境规划 12](#_Toc28644956)

[6.1 部署环境 12](#_Toc28644957)

[6.2 执行环境 12](#_Toc28644958)

[7 测试风险分析 13](#_Toc28644959)

[8 角色与职责 14](#_Toc28644960)

表目录

[Table 1性能测试类型 5](#_Toc501358463)

[Table 2 场景通过标准 7](#_Toc501358464)

[Table 3 独立场景设计 7](#_Toc501358465)

[Table 4 混合场景设计 8](#_Toc501358466)

[Table 5 峰值场景设计1 8](#_Toc501358467)

[Table 6 峰值测试设计2 8](#_Toc501358468)

[Table 9 测试报告需求 10](#_Toc501358471)

# 项目概要介绍

## 项目简介

MeetHere是一个场馆预约与管理的Web电子商务网站，主要功能包括：

1) 普通用户：注册、登录、个人信息管理、查看场馆介绍和预约信息、场馆预约、场馆预约订单管理、查看新闻、留言管理（发布、浏览、删除、修改）。

2) 管理员：用户管理、场馆信息管理（场馆介绍、场馆位置、场馆租金、场馆空闲时间...）、预约订单审核、预约订单统计（按照场馆、时间等对预约订单进行统计）、新闻动态管理（发布、增、删、改）、留言审核

3) 场馆预约订单：已预定场馆信息的查看、取消、信息修改

## 项目成员

罗亦翔、王晋、刘鹭语、刘珂冰

## 测试范围

### 范围内

MeetHere项目的前端业务，包含登陆注册、发布订单、查看订单、审核订单、查看留言、查看新闻等业务功能。

### 范围外

MeetHere项目的后台业务，包含用户管理、数据处理等业务功能。

## 前提假设

被测系统完成功能测试，测试报告显示达到要求。

## 测试目标

1) 系统在单步操作响应时间方面，能满足用户当前及未来1到3年的发展需求；

2) 系统的批量作业运行稳定，处理能力能满足用户当前及未来1到3年的发展需求；

3) 在业务处理能力方面，能满足当前及未来1到3年的业务增长需求；

4) 发现并解决宕机，内存泄漏等严重问题，使系统具备良好的稳定性、健壮性；

5) 系统在资源使用方面比较合理，各项资源平均利用率在30%左右。

# 性能测试策略

与其它测试类型一样，性能测试周期分为：性能测试需求分析、性能测试设计、性能测试实现和性能测试报告四个存在迭代的阶段。其中，性能测试需求和性能测试设计由测试人员手工完成，而性能测试实现需借助性能测试工具达成，通过对性能测试报告的分析判定系统是否存在性能缺陷。

## 性能测试模型

性能测试模型定义了性能测试涉及的各个待测方面，是性能测试设计的指南。



Figure 1 性能测试模型

根据Figure 1所示的性能测试模型，可以设计如Table 1所示的性能测试类型

Table 1性能测试类型

|  |  |  |  |
| --- | --- | --- | --- |
| **测试种类** | **测试方法** | **测试内容** | **场景类型** |
| 性能测试 | 以系统最大并发用户数的1-2倍作为上限对关键交易进行梯度加压测试，在压力时间内通过的交易量应接近峰值时段的交易量，甚至超过系统全天的交易量 | 核心业务，需求分析中发现的容易出现性能问题的业务 | 独立场景 |
| 压力测试 | 以系统预期最大并发用户数的1-2倍作为上限对常用/关键交易进行混合场景梯度加压测试，在压力时间内通过的交易量应接近峰值时段的交易量，甚至超过系统全天的交易量 | 存在一定关联关系的常用/核心业务 | 混合场景 |
| 峰值测试 | 按照系统峰值时预期最大并发用户数的2-3倍对峰值时段涉及的核心交易进行的混合场景梯度加压测试，在测试时间内通过的交易数量应接近或超过系统全天的交易量 | 生产上峰值场景中的核心业务 | 峰值场景 |

## 性能测试场景

根据项目实际，确定本次性能测试场景，例如可包含以下场景的性能测试：

独立场景

混合场景

峰值场景

## 重点测试策略

### 重点测试原则

为了全面评估系统性能，本次重点测试策略安排如下：

1. 对关键交易开展全面的独立场景测试；
2. 对常用/核心业务进行混合场景测试；
3. 对关键交易开展全面的峰值测试；

### 重点测试交易

普通用户发布订单

# 测试案例设计

## 生产压力分析

对于本系统的开发

## 场景通过标准

Table 2 场景通过标准

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **场景类型** | **单步操作响应时间** | **加压时间** | **操作账户数据** | **处理业务笔数** | **事务成功率** | **各后台利用率** | **并发/在线上限** | **在线思考时间设置** |
| 独立场景 | <1秒 | 38分钟 | >1万 | >5000 | 100% | <50% | 100 | 无 |
| 混合场景 | <1秒 | 30分钟 | >1万 | >7万 | 98% | <80% | 150 | 无 |
| 峰值场景 | <2秒 | 4小时 | >1万 | >10万 | 90% | <80% | 400 | 无 |

## 测试场景设计

### 独立场景

对于关键场景，将分别对其采用渐进式加压的方式来进行独立场景测试。每个场景测试多组并发、并发数从1逐步增加到200。对于具体的场景，测试几组并发依据案例执行。案例需要按照业务量设计。对于使用频度较低的场景，可以采用独立/混合场景的方式进行测试，即在背景压力下以小并发的方式来进行长时间疲劳测试。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 1 | <1秒 | N/A | 3分钟 | 直接加压 |
| 10 | <1秒 | N/A | 5分钟 | 1个用户/5秒 |
| 20 | <1秒 | N/A | 5分钟 | 1个用户/5秒 |
| 50 | <1秒 | >100笔/秒 | 10分钟 | 1个用户/5秒 |
| 100 | <1秒 | >100笔/秒 | 10分钟 | 1个用户/5秒 |
| 200 | <1秒 | >100笔/秒 | 5分钟 | 1个用户/1秒 |

Table 3 独立场景设计

具体的独立场景测试用例如下所示：

1、登陆部分

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 场景步骤 | 步骤描述 | 输入 | 预期结果 |
| 1、 | 场景1:用户成功登录 | 1、进入登录页面  2、输入账号  3、输入密码  4、点击“登录” | 账号：aaaa  密码：111111 | 跳转到场馆预约页面 |
| 2、 | 场景2:用户账号或密码错误 | 1、进入登录页面  2、输入账号  3、输入密码  4、点击“登录” | 账号：aaaa  密码：111 | 提示“用户账号密码错误” |

2、注册部分

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 场景步骤 | 步骤描述 | 输入 | 预期结果 |
| 1、 | 场景1:成功注册 | 1、进入注册页面  2、输入账号  3、输入昵称  3、输入密码  4、输入确认密码  6、点击“提交” | 账号：aaaa  昵称：aaaa  密码：111111 | 提示“用户注册成功” |
| 2、 | 场景2:账号已被注册 | 1、进入注册页面  2、输入账号  3、输入昵称  3、输入密码  4、输入确认密码  6、点击“提交” | 账号：aaaa  昵称：aaaa  密码：111111 | 提示“用户注册失败” |

3、发布订单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 场景步骤 | 步骤描述 | 输入 | 预期结果 |
| 1、 | 场景1：成功发布订单 | 1、进入场馆预约页面  2、选择场馆  3、选择日期  4、选择时间  5、选择时长  5、确认预约 |  | 提示“场馆预约成功，请管理员审核“，并跳转到订单页面 |
| 2、 | 场景2:订单时间存在冲突 | 1、进入场馆预约页面  2、选择场馆  3、选择日期  4、选择时间  5、选择时长  5、确认预约 |  | 提示“场馆预约失败，预约时间冲突“ |

4、发布留言

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 场景步骤 | 步骤描述 | 输入 | 预期结果 |
| 1、 | 场景1:发布留言 | 1、进入留言管理页面  2、选择发布留言  3、输入内容  4、点击发布 | 内容：aaa | 提示“发布留言成功 请管理员审核” |

### 混合场景

对于相关联的关键交易，对其采用渐进式加压的方式来进行混合场景测试，以测试这些关联交易是否存在性能问题。每个场景测试多组并发，并发数从20逐步增加到150。对于具体的场景，测试几组并发依据案例来执行（案例主要依据业务量来设计）。对于使用频度较低的场景，可以采用独立/混合场景的方式进行测试，即在背景压力下以小并发的方式来进行长时间疲劳测试。

具体场景内的操作流程包括：

1、用户注册—>用户登录—>用户退出

2、用户登录—>用户查看场馆信息—>用户退出

3、用户登录—>用户查看订单—>用户退出

4、用户登录—>用户查看新闻—>用户退出

5、用户登录—>用户查看留言—>用户退出

6、用户登录—>用户查看留言—>用户修改自己发布的留言—>用户退出

7、用户登录—>用户查看个人信息—>用户退出

8、用户登录—>用户修改密码—>用户退出—>用户登陆

9、管理员登录—>管理员对场馆信息做出修改—>管理员退出

10、管理员登录—>管理员对用户密码进行重置—>管理员退出

11、管理员登录—>管理员审核订单—>管理员退出

12、管理员登录—>管理员发布新闻—>管理员退出

13、管理员登录—>管理员修改新闻—>管理员退出

14、管理员登录—>管理员审核留言—>管理员退出

15、管理员登录—>管理员删除留言—>管理员退出

对于操作流程1，3，11，该流程在系统中出现率非常高，采用从20到150逐渐加压的方式，以得到该场景通过测试的成功率。而对于2，4，5，14等操作流程，使用频率较高，因此采用20到100的加压方式。最后对于操作6，7，8，9，10，12，13，15等，此类操作在系统中出现次数较少，所以采用20到50的加压方式。这样可以保证测试的顺利进行并且有效利用资源。

Table 4 混合场景设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 20 | <1秒 | N/A | 5分钟 | 1个用户/5秒 |
| 50 | <1秒 | >100笔/秒 | 10分钟 | 1个用户/5秒 |
| 100 | <1秒 | >100笔/秒 | 10分钟 | 1个用户/5秒 |
| 150 | <2秒 | >100笔/秒 | 5分钟 | 1个用户/1秒 |

### 峰值场景

对于该系统，大批量的操作多数集中于14:00-17:00，期间会有大批量的登录，发布和查看订单操作。因此，峰值最长持续时间2-3小时。考虑到系统的扩展性及实际情况，峰值测试共测试三组：第一组测试正常峰值压力的处理能力；第二组测试2倍、4倍峰值压力下的系统处理能力，以满足未来的业务发展需求，测试时各组中的功能操作按照业务比例同时加载。

根据相关资料，该系统的生产压力为100。

第一组：先用生成峰值时压力的70%加压1小时作为基准压力，之后采用当前生产上的实际峰值压力并发进行加压。

Table 5 峰值场景设计1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 生产压力的70% | <1秒 | >100笔/秒 | 1小时 | 1个用户/5秒 |
| 生产压力 | <1秒 | >100笔/秒 | 3小时 | 1个用户/5秒 |

第二、三组：分别采用生产压力的2到4倍来进行加压，第二组达标后再进行后面的测试。

Table 6 峰值测试设计2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 生产压力的2倍 | <2秒 | >100笔/秒 | 4小时 | 1个用户/5秒 |
| 生产压力的4倍 | <3秒 | >100笔/秒 | 4小时 | 1个用户/5秒 |

对于性能较好的系统，可以直接加压到最大并发数，持续加压5小时来测试系统的峰值处理能力。

# 测试实施安排

## 测试进度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **任务** | **开始日期** | **结束日期** | **参与人员** |
| 1 | 熟悉被测试系统，确定典型事务 | 2019.12.24 | 2019.12.25 | 测试人员 |
| 2 | 搭建测试环境，  录制典型事务的脚本，增强脚本 | 2019.12.25 | 2019.12.26 | 测试人员 |
| 3 | 执行测试并收集相关数据 | 2019.12.26 | 2019.12.29 | 测试人员 |
| 4 | 数据分析，编写测试报告 | 2019.12.29 | 2019.12.30 | 测试人员 |

## 测试流程

流程分为需求分析、设计、实现和测试报告四个阶段。

性能测试需求分析与设计：

熟悉被测试系统，定义性能测试要求，例如并发用户的数量、典型业务流程和所需响应时间。

性能测试实现：

创建虚拟用户脚本：将最终用户活动捕获到自动脚本中。

定义场景：使用JMeter设置负载测试环境。

运行场景：通过 JMeter驱动、管理和监控负载测试。

生成性能测试报告：

使用 JMeter 创建图和报告并评估性能。

## 测试报告需求

Table 9 测试报告需求

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **报告类型** | **响应时间要求** | **报告者** | **接受者** | **报告内容** |
| 邮件 | 每日 | 各渠道测试组负责人 | 测试中心，项目经理 | 每日结果总结报告 |
| 邮件 | 每日 | 各渠道测试组负责人 | 测试中心，项目经理 | 每日测试结果记录 |
| 邮件 | 每日 | 性能测试负责人 | 测试中心，项目经理 | 周报 |
| 文档 | 各测试阶段结束 | 各渠道测试组负责人 | 测试中心，项目经理 | 性能测试报告 |

## 性能缺陷管理

测试过程采用Quality Center进行缺陷管理

# 性能测试标准

## 启动标准

1. 测试环境满足计划需求
2. 基准参数配置完成校验
3. 关键交易通过冒烟测试

## 中止标准

1. 测试环境或关键系统不可用
2. 测试环境距生产标准差距太大
3. 缺陷周转周期不符合规定的时间
4. 出现宕机、不响应等严重的性能问题
5. 系统的交易成功率低于90%

## 通过标准

系统上线至少满足下面标准：

1. 系统无宕机、不响应类的严重性能问题
2. 系统响应时间80%达到系统的期望值
3. 系统的业务吞吐量达到预期目标，即当前生产需求的3倍以上
4. 柜台类系统要求通过7\*12小时以上的疲劳强度测试
5. 电子渠道类系统要求通过7\*24小时以上的疲劳强度测试

# 测试环境规划

## 部署环境

测试环境要求尽量和真实环境相同，至少要求服务器配置和网络带宽和拓扑结构应该相似。主要内容：服务器数量和配置，操作系统和数据库版本，软硬件部署等。

|  |  |  |  |
| --- | --- | --- | --- |
| **软件环境（相关软件、操作系统等）** | | | |
| **名称** | **版本** | **数量** | **获得途径** |
| Centos | 7.2 |  |  |
| mysql | 8.0.15 |  |  |
| **硬件环境（网络、设备等）** | | | |
| **名称** | **版本** | **数量** | **获得途径** |
| Web服务器 |  |  |  |
| 数据库服务器 |  |  |  |
| 前台客户机 |  |  |  |

## 执行环境

|  |  |  |  |
| --- | --- | --- | --- |
| **软件环境（相关软件、操作系统等）** | | | |
| **名称** | **版本** | **数量** | **获得途径** |
| MAC OSX | 10.15.2 |  |  |
| mysql | 8.0.15 |  |  |
| Google Chrome | 79.0.3945.88 |  |  |
| Firefox | 71.0 |  |  |
| LoadRunner | 最新版本 |  |  |
| **硬件环境（网络、设备等）** | | | |
| **名称** | **版本** | **数量** | **获得途径** |
| Web服务器 |  |  |  |
| 数据库服务器 |  |  |  |
| 前台客户机 |  |  |  |

# 测试风险分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **风险因素** | **可能结果** | **可能发生时间** | **风险**  **级别** | **应对措施** |
| 工具缺陷 | 测试工具和监控工具无法全部支持信贷业务系统的测试和监控 | 随时 | 中 | 评估被测系统，分析所有需求。  通过其它工具实现对需求的支持程度。 |
| 测试数据的准备备份及恢复无法正常完成 | 测试过程中数据用尽或不满足测试需求，将导致测试无法实施。 | 测试执行时 | 高 | 运维方配合完成数据的准备、备份和恢复 |
| 测试环境有其他用户连接进行操作，服务器产生性能缺陷 | * 1. 测试方获得最大负载压力与实际最大负载有差距   2. 服务器出现性能缺陷的现象，运维方定位性能缺陷模块并非真正性能缺陷的模块 | 测试执行时 | 高 | 测试方进行负载测试时，保证测试环境无其他连接和用户操作 |
| 测试服务器访问状态不稳定 | 测试准备和测试执行中断，测试计划时间延后 | 随时 | 高 | 保证测试期间测试环境访问畅通 |

# 角色与职责

|  |  |  |  |
| --- | --- | --- | --- |
| **角色** | **姓名** | **具体职责** | **技能要求** |
| 性能测试经理 | 刘鹭语 | 性能测试项目管理和协调  性能测试需求分析  测试计划和测试报告编写 | 1. 组织协调能力 2. 熟悉性能测试架构 3. 熟悉应用架构 |
| 性能测试设计人员 | 编制测试案例，开发测试脚本，设计测试场景，执行性能测试 | 1. 熟悉性能测试体系 2. 熟悉性能测试方法和工具 3. 脚本编写和调试能力 |
| 测试环境和测试数据准备人员 | 准备性能测试环境  准备基础数据和测试数据 | 1. JMeter部署能力 2. 熟悉核心系统、数据库 |
| 性能调优人员 | 核心主机系统优化 | 1. 熟悉数据库优化 2. 熟悉应用架构 |