e-Book 网上购书平台

测试计划

版本 1.0

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 17日/3月/2022年 | 1.0 | 初始化测试计划文档 | 康艺潇，林舒怀，林祺龙，徐惠东 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 5

3. 测试策略 5

3.1 测试类型 5

3.1.1 数据和数据库完整性测试 5

3.1.2 功能测试 5

3.1.3 业务周期测试 5

3.1.4 用户界面测试 5

3.1.5 性能评价 5

3.1.6 负载测试 5

3.1.7 强度测试 5

3.1.8 容量测试 5

3.1.9 安全性和访问控制测试 5

3.1.10 故障转移和恢复测试 5

3.1.11 配置测试 5

3.1.12 安装测试 5

3.2 工具 5

4. 资源 5

4.1 角色 5

4.2 系统 5

5. 项目里程碑 5

6. 可交付工件 5

6.1 测试模型 5

6.2 测试日志 5

6.3 缺陷报告 5

7. 附录 A：项目任务 5

测试计划

# 简介

本测试报告是针对 “e-Book网上购书平台”项目的测试文档。用户可以在此平台上查看书籍信息、购买书籍。此平台由e-Book系统管理员维护。本次测试针对6个模块进行测试。本次测试为黑盒测试，采用了等价类，决策表，边界值测试的测试方法，所有测试用例储存在表格文件中。测试中主要使用postman作为测试工具，采用postman测试模块提供的语法编写代码，在给定数据库的情况下，对后端的接口进行测试。

## 目的

“e-Book 网上购书平台”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：Web 前端使用Vue 架构，用WebPack打包管理；后端使用 Spring 来实现，使用Maven打包管理。前后端使用HTTP 通信。
2. 列出推荐的测试需求。
3. 推荐可采用的测试策略，并对这些策略加以说明：本次测试主要使用的方法为等价类测试，基于决策表测试，边界值测试。
4. 本次测试需要在测试环境部署该项目，需要测试服务器1台。
5. 项目最终将交付项目计划，测试计划书，需求规约书。

背景：本课题的任务是开发一个网上购书系统。在此主要实现电子商店部分，完成定购功能并实现一些对书店的管理功能。这样，基于消费者和商家的需求考虑，网上购书系统应该包括以下几部分：

1. 图书展放区：用来展放所售图书，类似于实际书店中的书架功能。
2. 图书查询：使客户能够对书店所售图书进行查询，方便客户查找订购所需图书，节约时间。
3. 会员注册：注册成为会员并且每次在登录后才可以订购图书。
4. 更新资料：会员可以修改密码或更改个人信息（主要是联系信息）。
5. 购物车和购买书籍：购物车用来暂时存放客户所选商品，并可以在购物车中修改购书的数量。 用户可以下单。
6. 查询订单：客户将所选商品提交后，系统生成订单，用户可以通过查询订单对付款行为进行指导。
7. 管理系统：包括对图书、客户及订单等的管理，同时可以录入新书。

还必须有强大的数据库，存储以下信息：

1. 图书信息：包括图书名称、价格、作者、出版社、出版日期以及内容简介等信息。
2. 会员信息：应当尽可能搜集关于客户的详尽信息，包括一般信息：姓名、 账号、密码、身份证号、住址、电话、电子邮件等。
3. 订单信息：存储会员每次交易的内容：会员名、购买的图书名称、数量、交易时间等信息。

## 范围

本次进行的是单元测试，针对用户注册功能、用户登陆功能、购买书籍功能、订单过滤功能和用户管理功能进行黑盒测试。

# 测试策略

## 测试类型

### 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。 |
| 方法： | 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： |  |

|  |  |
| --- | --- |
| 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： |  |

1. **用户注册功能**

变量：用户名，密码，重复密码，邮箱

变量要求：

用户名非空

密码要求：1.密码长度超过8位 2. 密码要求数字加英文字母（大小写均可）

重复密码要求和密码一致

邮箱需要符合邮箱格式

**边界值测试**

采用健壮性测试，即需要考虑无效值。因为这些变量非数值型，判定是否合法的依据简单，而且是单缺陷假设，各个变量不影响，即不考虑密码和重复密码是否一致的情况。所以共7个测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用户名 | 密码 | 重复密码 | 邮箱 | 预期注册情况 |
| ebook | ebook12345 | ebook12345 | ebook@126.com | √ |
| 空 | ebook12345 | ebook12345 | ebook@126.com | X |
| ebook | Sdfs343244 | ebook12345 | ebook@126.com | X |
| ebook | ebook12 | ebook12 | ebook@126.com | X |
| ebook | 12345 | 12345 | ebook@126.com | X |
| ebook | ebook | ebook | ebook@126.com | X |
| ebook | ebook12345 | ebook12345 | ebook126.com | X |

**等价类测试**

因为每个变量都有无效值，且密码和重复密码有耦合关系，因此需要考虑变量的相互影响以及无效值，因此采用强健壮等价类测试，即考虑多缺陷的组合和无效值一共 2\*4\*2\*2 = 32

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用户名 | 密码 | 重复密码 | 邮箱 | 预期注册情况 |
| ebook | ebook12345 | ebook12345 | ebook@126.com | √ |
| ebook | ebook12345 | ebook12345 | ebook126 | X |
| ebook | ebook12345 | ebook | ebook@126.com | X |
| ebook | ebook12345 | ebook | ebook126 | X |
| ebook | ebook1 | ebook1 | ebook@126.com | X |
| ebook | ebook1 | ebook1 | ebook126 | X |
| ebook | ebook1 | ebook12345 | ebook@126.com | X |
| ebook | ebook1 | ebook12345 | ebook126 | X |
| ebook | ebook | ebook | ebook@126.com | X |
| ebook | ebook | ebook | ebook126 | X |
| ebook | ebook | ebook12345 | ebook@126.com | X |
| ebook | ebook | ebook12345 | ebook126 | X |
| ebook | 12345 | 12345 | ebook@126.com | X |
| ebook | 12345 | 12345 | ebook126 | X |
| ebook | 12345 | ebook12345 | ebook@126.com | X |
| ebook | 12345 | ebook12345 | ebook126 | X |
| 空 | ebook12345 | ebook12345 | ebook@126.com | X |
| 空 | ebook12345 | ebook12345 | ebook126 | X |
| 空 | ebook12345 | ebook1234567 | ebook@126.com | X |
| 空 | ebook12345 | ebook1234567 | ebook126 | X |
| 空 | ebook1 | ebook1 | ebook@126.com | X |
| 空 | ebook1 | ebook1 | ebook126 | X |
| 空 | ebook1 | ebook12345 | ebook@126.com | X |
| 空 | ebook1 | ebook12345 | ebook126 | X |
| 空 | ebook | ebook | ebook@126.com | X |
| 空 | ebook | ebook | ebook126 | X |
| 空 | ebook | ebook12345 | ebook@126.com | X |
| 空 | ebook | ebook12345 | ebook126 | X |
| 空 | 12345 | 12345 | ebook@126.com | X |
| 空 | 12345 | 12345 | ebook126 | X |
| 空 | 12345 | ebook12345 | ebook@126.com | X |
| 空 | 12345 | ebook12345 | ebook126 | X |

1. **用户登陆功能**

用户输入变量：用户名、密码

用户名要求：存在用户名库中且非空。用户有3种情况：管理员和普通用户和被禁止登陆的用户，都需要测试

密码要求：和数据库中的对应用户名的密码一致且非空

**边界值测试**

因为密码和用户名有依赖关系，所以采用最坏情况测试，即拒绝单缺陷假设，考虑多缺陷。

|  |  |  |
| --- | --- | --- |
| 用户名 | 密码 | 预计登陆状态 |
| Admin | Password1234(正确的密码) | √ |
| Admin | Password123(错误的密码) | X |
| Admin | 空 | X |
| Customer | customer1234(正确的密码) | √ |
| Customer | customer1234(错误的密码) | X |
| Customer | 空 | X |
| Forbidden | customer1234(正确的密码) | X |
| Forbidden | customer1234(错误的密码) | X |
| Forbidden | 空 | X |

**等价类测试**

采用强一般等价类测试

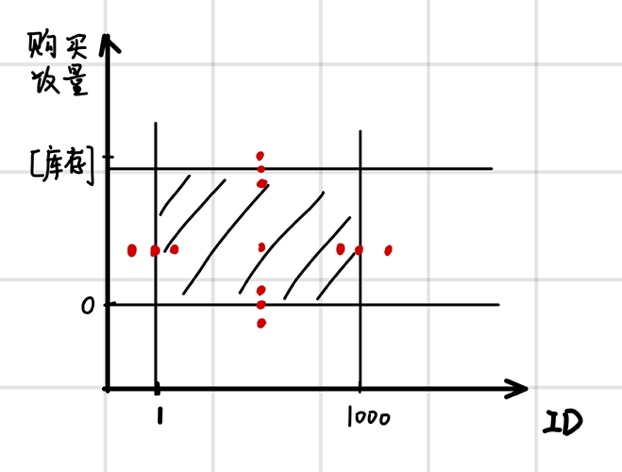
|  |  |  |
| --- | --- | --- |
| 用户名 | 密码 | 预计登陆状态 |
| Admin | Password1234(正确的密码) | √ |
| Customer | customer1234(正确的密码) | √ |

1. **购买书籍功能**

变量：书本ID, 数量，书本ID的范围【1，10000】，数量要求不能超过数据库内的库存，至少一本。这里用【库存】表示该本书的库存数值。

**边界值测试**

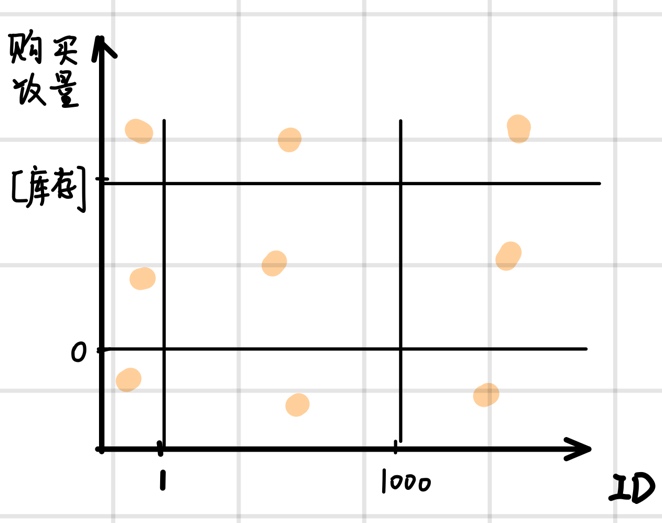
因为书籍的ID和数量都是一个范围，因此需要边界值测试。



假设该本书的库存为2000本

|  |  |  |
| --- | --- | --- |
| ID | 购买数量 | 预计购买状态 |
| 500 | 1000 | √ |
| 0 | 1000 | X |
| 1 | 1000 | √ |
| 2 | 1000 | √ |
| 999 | 1000 | √ |
| 1000 | 1000 | √ |
| 1001 | 1000 | X |
| 500 | 0 | X |
| 500 | 1 | √ |
| 500 | 2 | √ |
| 500 | 1999 | √ |
| 500 | 2000 | √ |
| 500 | 2001 | X |

**强健壮等价类测试**



|  |  |  |
| --- | --- | --- |
| ID | 购买数量 | 预计购买状态 |
| -2 | 3000 | X |
| 500 | 3000 | X |
| 1005 | 3000 | X |
| -2 | 20 | X |
| 500 | 20 | √ |
| 1005 | 20 | X |
| -2 | -2 | X |
| 500 | -2 | X |
| 1005 | -2 | X |

1. **订单过滤功能**

变量：时间范围、书籍名称、用户登陆状态。要求用户已登录，书籍信息已存在。时间范围需要在2000.01.01到当天时间（假设为2022.03.12），后一个时间要比之前的时间延后。

**强健壮等价类测试**

|  |  |  |  |
| --- | --- | --- | --- |
| 时间 | 书籍 | 用户登陆状态 | 查询状态 |
| 2021.01.02-2021.01.01 | 《已知书籍》 | 登陆 | √ |
| 2021.01.02-2021.01.01 | 《已知书籍》 | 登陆 | X |
| 1977.01.02-2021.01.01 | 《已知书籍》 | 登陆 | X |
| 2021.01.02-2099.01.01 | 《已知书籍》 | 登陆 | X |
| 2021.01.02-2021.01.01 | 《未知书籍》 | 登陆 | X |
| 2021.01.02-2021.01.01 | 《已知书籍》 | 未登陆 | X |

**决策表**

因为登陆是前提条件，所以就不用考虑多缺陷组合造成的冗余。基于决策表的决策方法是所有功能测试方法中最严谨的，因为它考虑到了测试用例整体的逻辑严密性。因此采用决策表的测试方法。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| C1:是否登陆？ | F | T | T | T | T | T | T |
| C2：时间是否符合要求？ | - | F | T | T | F | T | T |
| C3: 书籍名称是否在库？ | - | - | - | - | - | T | F |
| A1: 查询成功 | X | X | X | X | X | √ | X |
| A2: 查询失败 | √ | √ | √ | √ | √ | X | √ |

1. **用户管理功能**

变量: 用户角色（普通用户用0表示，被禁止的用户用1表示）、解禁状态（变成0或者变成1）。要求应该是0，1互换。

**决策表**

因为这里的变量比较少，只有两个，并且两个变量的无效值，有效值都各只有一个，所以组合比较少，用哪种测试方法都是可以的，这里使用比较明确的决策表。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
| C1:用户角色为普通角色 | T | T | F | F |
| C2:用户角色为被禁止角色 | F | F | T | T |
| C3:用户角色变成普通角色 | T | F | F | T |
| C1:用户角色变成普通角色 | F | T | T | F |
| A1:提示不用改变 | T | F | F | T |
| A2:提示改变成功 | F | T | T | F |

**强一般等价类测试**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 角色=0 | 角色=1 | 变为0 | 变为1 | 改变情况 |
| √ |  | √ |  | X |
| √ |  |  | √ | √ |
|  | √ | √ |  | √ |
|  | √ |  | √ | X |

## 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 | Postman | Postman |  |
| 用于功能性测试的 ASQ 工具 | Postman | Postman |  |
| 用于性能测试的 ASQ 工具 | Postman | Postman |  |
| DBMS 工具 | Mysql | Oracle |  |

# 

# 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 * 分配工作 |
| 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 | 2 | 执行测试。  职责：   * 编写代码，执行测试 * 记录结果 * 从错误中恢复 * 记录变更请求 * 生成测试报告 |

## 

## 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 | Mysql |
| —数据库名 | e\_book |
| 客户端测试 PC | Windows 10 |
| —包括特殊的配置需求 | 无 |

# 

# 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 | 10 | 2022年3月10日 | 2022年3月15日 |
| 设计测试 | 20 | 2022年3月15日 | 2022年3月17日 |
| 实施测试 | 20 | 2022年3月17日 | 2022年3月19日 |
| 执行测试 | 10 | 2022年3月19日 | 2022年3月19日 |
| 评估测试 | 10 | 2022年3月19日 | 2022年3月20日 |

# 

# 可交付工件

* 交付的文件如下：

1. 《测试计划》

2. 《需求规格说明书》

3. 《测试报告》

交付时间为2022年3月20日

* 交付的工具如下：

1. e-book 网上购书平台测试代码

交付时间为2022年3月20日

## 测试模型

详见测试报告

## 测试日志

详见测试报告

## 缺陷报告

详见测试报告

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