BookStore 单元测试

测试计划

版本 1.0

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2024/4/3 | 0.1 | 阅读源码，指定初步测试计划 | 王浩丞 |
| 2024/4/4 | 0.2 | 组内讨论，确认分工 | 王浩丞 |
| 2024/4/15 | 0.3 | 小组第一次内容整合 | 王浩丞 |
| 2024/4/22 | 0.4 | 小组第二次内容整合 | 王浩丞 |
| 2024/4/28 | 1.0 | 根据最终反馈完善文档 | 王浩丞 |

目录

1. 简介 4

1.1 目的 4

1.2 背景 4

1.3 范围 4

2. 测试需求 5

3. 测试策略 5

3.1 测试类型 5

单元测试，白盒测试

工具 6

4. 资源 7

4.1 角色 7

4.2 分工 8

4.3 系统

5. 项目里程碑 9

6. 可交付工件 10

6.1 测试模型 10

6.2 测试日志 10

6.3 缺陷报告 10

7. 附录 A：项目任务 11

测试计划

# 简介

## **1.1**目的

BookStore单元测试的“测试计划”文档有助于实现以下目标：

·进一步确定并细化BookStore的功能模块，及对应软件构件的测试需求

·对后端代码进行全方位白盒测试，达到95%以上覆盖率

·进一步了解探索单元测试策略的原则，熟悉dd路径，数据流分析等内容，制定应对各种环境与要求的测试用例

·基于AI生成的测试用例，添加以手工测试用例添加，以完善白盒测试覆盖率

·为最终交付离开Lab环境下可运行的BookStore提供测试支持

## **1.2**背景

BookStore是21级软件学院本科生于大二下学期《互联网应用技术开发》课程中完成的大作业，前端基于React框架，后端基于Spring框架，用于在网页端提供一个网上书店供客户使用。

本次单元测试将针对BookStore的后端展开测试，BookStore后端基于spring框架，基于jdbc repository，dao，service，controller四层架构，负责管理user，book，cart，order等数据结构，并维护与client的session，另外，还有一些Utils用于方便操作。

本次单元测试将针对后端全部代码进行单元测试，保证代码覆盖率达到95%以上。

## **1.3**范围

本次测试为单元测试，白盒测试，对BookStore的后端进行全部测试。测试用例基于AI生成+手动编写获得，并根据测试结果进一步分析。

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** |
| 测试计划 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 王浩丞 |  |
| 测试报告 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 安俊宇 |  |
| 各部分测试与分析报告 | 🗹 是 🞏 否 | 🗹 是 🞏 否 | 黄培正 安俊宇 汪毅恒 王浩丞 |  |

# **2.**2. 测试需求

下面列出了那些已被确定为测试对象的项目（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。

# 本次对BookStore测试的主要需求有：Controller各单元测试（含BookController，CartController，HomeController，LoginController，OrderController，OrderItemController，UserController等模块），Service-Dao-Repository各单元测试（含Order，Book，User，Cart，Home等模块），Utils单元测试等部分。

# 3. 测试策略

## 3.1 测试类型

白盒测试

本次BookStore后端测试的类型为白盒测试

|  |  |
| --- | --- |
| 测试目标： | 对BookStore的后端全部代码进行白盒测试，以保证BookStore在真实业务逻辑下的健壮性 |
| 方法： | 使用白盒测试方法，分析代码的dd路径，并基于分析结果与源代码，使用AI+人工的方式生成测试用例，使得代码覆盖率达到95%以上 |
| 完成标准： | 在基于AI生成测试用例的基础上，手工补充测试用例，使得测试用例完备健壮，且都经过测试获得对应输出，还需要根据输入输出对测试结果进行分析 |
| 需考虑的特殊事项： | • 不同层次的测试单元可能存在依赖，为了确保单元测试的独立性，可能需要将一个单元测试中所依赖的其他单元mock住  • 某些组件可能需要在spring环境下有效，对此需要在Junit5中引入某些Extension  • AI生成的测试用例可能不够完善，需要分析生成结果后，人工补充测试用例 |

### **3.2 工具**

此项目将使用以下工具：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 工具 | 厂商/自行研制 | 版本 |
| 文档撰写 | WPS, WORD，Markdown |  |  |
| AI测试用例生成 | ChatGPT |  |  |
| BookStore环境配置 | Java, Spring, MySQL |  |  |
| 版本控制 | git，github |  |  |
| 测试框架 | Junit5，Mockito |  |  |
| IDE | Idea，Vscode |  |  |

# **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 | 推荐的最少资源  （所分配的专职角色数量） | 具体职责或注释 |
| 测试小组组长 | 1人 | 测试小组组织者  主要职责：  ·测试结果，测试报告的整合与提交  ·与老师助教交流，确定小组任务细节  ·发起组会，定期跟进组内任务进度  ·领取测试任务，参与测试 |
| 测试小组成员 | 3人 | 测试小组组员  主要职责：  ·参与组会，报告测试进度  ·领取测试任务，参与测试 |

## **4.2分工**

|  |  |
| --- | --- |
| 角色 | 具体职责或注释 |
| 黄培正 | ·book和cart部分分析与测试  ·Interception和utils部分分析与测试  ·测试覆盖率报告导出 |
| 安俊宇 | ·单元测试测试报告撰写  ·User部分分析与测试 |
| 汪毅恒 | ·单元测试AI部分报告撰写  ·Order和OrderItem部分分析与测试 |
| 王浩丞 | ·单元测试测试计划撰写  ·Controller部分分析与测试 |

## **4.3 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 |
| 测试与开发 PC | 任意小组成员PC即可 |

# 5. 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% | 2024/4/3 | 2024/4/4 |
| 设计测试 | 30% | 2024/4/4 | 2024/4/4 |
| 实施测试 | 30% | 2024/4/4 | 2024/4/15 |
| 执行测试 | 20% | 2024/4/4 | 2024/4/22 |
| 评估测试 | 20% | 2024/4/15 | 2024/4/29 |

# 可交付工件

## **6.1**测试模型

本次测试的模型为后端Spring + 数据库MySQL架构的Online Bookstore应用。测试针对后端所有代码，采取白盒测试+人工分析+AI生成测试用例的方法，力求测试完备，代码覆盖率高，并尽可能地发现其中的隐藏bug

## **6.2**测试日志

全组于4.3--4.4期间确认各自任务分工

全组于4.4--4.15期间各自负责所属单元测试用例，于4.15进行第一次集中反馈

全组于4.15-4.22进行第二轮测试完善，主要为丰富测试分析与报告，覆盖零碎的utils等模块的代码，整理各自格式结果交付文档撰写人，于4.22进行第二次集中反馈

4.28全部测试工程与文档撰写结束，正式提交测试大作业

## **6.3**缺陷报告

测试过程中未发现bug

# 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