<电影订票app>

系统测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <12/6/2023> | <1.0> | 综合测试 | <曾永诚> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写词和缩略语 4

1.4 参考资料 4

1.5 概述 4

2. 测试概要 4

3. 测试环境 4

4. 测试结果及分析 4

4.1 需求覆盖率及缺陷分布 4

4.2 缺陷严重程度 5

5. 缺陷清单 5

5.1 功能性缺陷 5

5.2 非功能性缺陷 6

6. 测试结论与建议 6

系统测试报告

# 简介

## 目的

该系统测试的目的是验证前端app是否满足用户规定的需求。

## 范围 该测试报告仅针对于前端app的系统测试，不包含后端。

## 定义、首字母缩写词和缩略语

详见【glossary】

## 参考资料

（暂无）

## 概述

对app前端进行了各项系统测试。

# 测试概要 2023.6.23，开发人员在本地虚拟机进行app前端的系统测试。

# 测试环境

硬件环境：

处理器 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz 2.30 GHz

机带 RAM 16.0 GB (15.9 GB 可用)

系统类型 64 位操作系统, 基于 x64 的处理器

软件环境：

Android studio 2022 Pixel API 30 RAM：2048mb

# 测试结果及分析

## 需求覆盖率及缺陷分布

**表1 需求覆盖率及缺陷分布**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **测试内容** | | **测试**  **用例数** | **需求**  **覆盖率** | **缺陷数** | **缺陷率** | **备注** |
| **功能项** | **信息浏览** | 4 | 100% | 0 | 0 | - |
| **查看影院** | 10 | 100% | 0 | 0 | - |
| **搜索影院** | 10 | 100% | 0 | 0 |  |
| **选座** | 10 | 100% | 3 | 30% | - |
| **购票** | 10 | 100% | 0 | 0 | - |
|  | **查看我的** | 4 | 100% | 0 | 0 | - |
|  | **查看订单** | 10 | 100% | 0 | 0 | - |
|  | **功能项小计** | 58 | 100% | 3 | 5.17% | - |
| **非功**  **能项** | **响应时间** | 10 | 100% | 2 | 20% | - |
| **抗压能力** | 10 | 100% | 0 | 100% | - |
| **可靠性** | 5 | 100% | 0 | 100% | - |
| **非功能项小计** | 20 | 100% | 2 | 1% | *-* |
|  | **总计** | 78 | 100% | 5 | 6.4% | *-* |

## 缺陷严重程度

**表2 缺陷严重程度**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **严重程度** | **致命** | **严重** | **一般** | **细微** | **总数** |
| **缺陷个数** | 0 | 3 | 0 | 2 | 0 |
| **占缺陷百分比** | 0 | 60% | 0 | 40% | 0 |

# 缺陷清单

## 功能性缺陷

**表3 功能性缺陷列表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序**  **号** | **缺陷**  **编号** | **严重程度** | **功能模块** | **缺陷标题** | **缺陷描述** | **测试用例编号** |
| 1 | 1 | 严重 | 选座模块 | 选座冲突 | 多个用户同时选座，后端返回错误的信息 | 8、9、10 |

表中相关项说明：

1. 缺陷编号：为每个缺陷赋予一个唯一的编号，可以通过此编号对缺陷进行跟踪。例如：Bug001。
2. 缺陷严重程度：缺陷可以根据严重程度分为以下几种情况。
3. 致命（fatal）：致命的错误，测试执行直接导致系统死机、蓝屏、挂起、或是程序非法退出；系统的主要功能或需求没有实现。
4. 严重（critical）：严重错误，系统的次要功能点或需求点没有实现；数据丢失或损坏。执行软件主要功能的测试用例导致系统出错，程序无法正常继续执行；程序执行过于缓慢或是占用过大的系统资源。
5. 一般（major）：不太严重的错误，这样的缺陷虽然不影响系统的基本使用，但没有很好地实现功能，没有达到预期的效果。如次要功能丧失，界面错误，打印内容、格式错误，提示信息不太正确，或用户界面太差，简单的输入限制未放在前台进行控制，删除操作未给出提示，操作时间长等。
6. 细微（minor）：一些小问题，对功能几乎没有影响，产品及属性仍可使用。如软件的实际执行过程与需求有较小的差异；程序的提示信息描述容易使用户产生混淆；有个别错别字、 文字排列不整齐；辅助说明描述不清楚，显示格式不规范，长时间操作未给用户进度提示，提示窗口文字未采用行业术语，可输入区域和只读区域没有明显的区分标志，系统处理未优化等。
7. 功能模块：所测试并出现该缺陷的功能模块名称。
8. 缺陷标题：描述缺陷的标题。
9. 缺陷描述：对缺陷的详细描述。描述缺陷的重现步骤和问题所在，描述需清晰明了，使研发人员可以按照描述将缺陷重现出来。必要时可以放上相应的截图，以便于研发人员理解问题所在。

## 非功能性缺陷

**表4 非功能性缺陷列表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序**  **号** | **缺陷**  **编号** | **严重程度** | **测试类型** | **缺陷标题** | **缺陷描述** | **测试用例编号** |
| 1 | 1 | 细微 | 购票模块 | 响应时间 | 响应时间没有达到预期 | 9、10 |

# 测试结论与建议

对于功能性缺陷，即多人同时购票选座时产生的数据不一致的缺陷。一方面，可以对请求进行异步处理；另一方面，需要修改前后端的代码，如选座产生了冲突，应即使撤销其他用户的操作，可以考虑上锁来解决数据冲突的问题。

对于非功能性缺陷，例如响应时间没有达到预期，可以采用并发编程的思想，多线程地处理用户请求，提高系统的吞吐量。此外，还要对代码路径进行优化，减少不必要的内存访问。