<项目名称>

系统测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <日/月/年> | <x.x> | <详细信息> | <姓名> |
| 2023/6/17 | 1.0 | 进行系统测试并完成此表 | 彭奕涛，朱亦腾，雷宇昊，陈一鸣 |
|  |  |  |  |
|  |  |  |  |

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

本报告的目的是对系统进行全面评估和记录，以便获取该项目的有关系统质量、可行性和符合性的信息，具体可分为以下几点

1. 评估系统质量：系统测试报告提供了系统在不同测试阶段的测试结果和性能数据，用于评估系统是否满足预期的质量标准和要求。这包括功能是否按照规格说明书的要求正常工作，系统性能是否符合性能目标等。
2. 发现问题和风险：系统测试报告记录了在测试过程中发现的问题、缺陷和异常行为，以及对系统功能和性能的潜在风险和威胁。这些问题和风险的详细描述和分类可以帮助项目团队进行问题解决和改进。
3. 沟通和交流：系统测试报告作为一个正式文档，可以用于与项目团队、开发人员、测试人员、管理层和其他利益相关者之间的沟通和交流。它提供了对系统测试活动的综合概述，帮助各方了解系统的状态和质量。

## 范围

交大点评项目

## 定义、首字母缩写词和缩略语

## 参考资料

## 

## 概述

本报告主要分为简介，测试概要，测试环境，测试结果及分析，缺陷清单以及结论与建议六块，同时附有系统测试用例一份

# 测试概要

测试主要由小组成员在线下集体对项目功能进行测试，主要依靠人力对各项功能以及可能的错误进行测试

# 测试环境

使用小组成员的个人电脑，在Windows系统上进行测试。

# 测试结果及分析

共进行32项测试，通过数为29，通过率为90.63%，缺陷主要分布在搜索功能中

## 需求覆盖率及缺陷分布

需求覆盖率达到100%。

**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** | **用户注册/登录** | 11 | 100% | 0 | 0% |  |
| **菜品上传及获取** | 5 | 100% | 0 | 0% |  |
| **搜索功能** | 4 | 100% | 2 | 50% | 搜索功能返回错误 |
| **餐厅浏览** | 3 | 100% | 1 | 33% | 界面返回美观程度不足 |
| **个人信息查看及编辑** | 5 | 100% | 0 | 0% |  |
| **社区功能** | 4 | 100% | 0 | 0% |  |
| **功能项小计** | 32 | 100% | 3 | 9.37% |  |
|  | **总计** | *32* | 100% | 3 | 9.37% |  |

## 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 | 2 | 0 | 1 | 3 |
| **占缺陷百分比** | 0 | 67% | 0 | 33% |  |

# 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 | bug1 | 严重 | 搜索 | 对不存在的菜品搜索 | 对数据库中没有的菜品搜索，系统会出现报错，页面错误 | 20 |
| 2 | bug2 | 严重 | 搜索 | 对菜品部分名称的搜索 | 对已有餐品进行搜索，若不完整打出菜品名字，系统会出现报错且页面错误 | 17 |
| 3 | bug3 | 细微 | 餐厅浏览 | 返回界面有误 | 返回菜品信息时前端界面格式有误 | 22 |

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。

# 测试结论与建议

测试中主要依靠人力对系统各项功能逐一进行测试，这一点仍有较大的改进空间，在今后应尝试引入更高效的测试方法提高系统测试效率。在本次测试中发现漏洞主要集中于搜索功能中，说明搜索功能仍有较多的问题需要修复，在下一步的开发中应对搜索功能进行进一步的，更加细致的完善。同时，在生成前端页面时，仍有部分页面的排版不够美观，需要对前端页面的细节进行进一步的改进。