<牙科医院管理系统>

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> |
| 17/06/2022 | <1.0> | <系统测试报告> | <董云鹏> |
|  |  |  |  |
|  |  |  |  |

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

本系统是基于C/S结构的牙科医院管理系统，通过该系统可以对一个牙科医院进行管理，可以进行挂号就诊，管理医生和患者等功能，方便牙科医院更好的维护自身运转。

为了保证牙科医院管理的各项功能可靠的实现，特编写了此测试计划，对所开发软件的各功能模块和事例系统进行测试。

## 范围

**系统易操作性:**

所开发的系统应做到操作简单，尽量使系统操作不受用户对电脑知识水平的限制。

**系统具有可维护性:**

由于系统涉及的成绩比较广，数据库中的数据需定期修改，系统可利用的空间及性能也随之下降，为了使系统更好地运转，学院可以对系统数据及一些简单的功能进行独立的维护及调整。

**系统具有开放性:**

该系统能够在开放的硬件体系结构中运行，并且能与其他系统顺利连接，不会因外部系统的不同面要做在量的修改工作。

## 定义、首字母缩写词和缩略语

从用户角度来看，系统首先应该能够提供便捷与强大的挂号功能。对于患者医生管理员而言，他们应该能够对系统的不同部分有各自不同的权限。

该系统具有较强的灵活性及可扩展性，能够存储一定数量的排班和挂号信息，并方便有效的进行相应的数据操作和管理，这主要包括：医生的录入、删除及修改，排班信息的录入，删除和修改，诊疗信息的录入，删除和修改，各种信息的单条件查询和多条件的组合查询。

希望通过本系统的开发可以解决挂号，就诊和基本信息管理功能，主要是针对登录用户的添加、删除、修改和查询功能。

## 参考资料

①《软件工程原理》 沈备军 主编 清华大学出版社

②《软件测试教程》 杜文洁 主编 清华大学出版社

③《如何设计黑盒测试的测试用例》 谈 燕 主编 成都教育学院学报

④《常用软件工程》 郑人杰 主编 清华大学出版社

⑤《面向对象软件工程与UML》 张 京 主编 人民邮电出版社

# 测试概要

6月1日-17日，线上，全体组员，手工测试，压力测试等，主要测试系统的完整性。

# 测试环境

阿里云服务器，Jemeter压力测试工具，

# 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** | **功能项1** | 32 | 96% | 1 | 3% |  |
| **功能项2** | 12 | 92% | 1 | 8% |  |
| **……** |  |  |  |  |  |
| **功能项小计** | 44 | 94% | 2 | 5% |  |
| **非功**  **能项** | **性能** | 20 | 96% | 2 | 4% |  |
| **可靠性** | 25 | 86% | 4 | 16% |  |
| **…** |  |  |  |  |  |
| **非功能项小计** | 45 | 90% | 6 | 12% |  |
|  | **总计** | 99 | 93% | 16 | 7% |  |

## 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 | 14 | 34 | 50 |
| **占缺陷百分比** | 0 | 4% | 28% | 68% | 100% |

# 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 | B1 | 严重 | 前端 | 没有实现退号 | 不依靠医生帮助，患者无法自行退号 | 001 |
| 2 | B2 | 一般 | 前端 | 医生添加不完整 | 无法快捷的在前端进行医生的添加 | 003 |
| 3 | B3 | 一般 | 前端 | 患者病历查看 | 患者没有办法快捷的查看病历 | 043 |

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 | F1 | 一般 | 性能 | 安全性 | 登录鉴权不完善 | 002 |
| 2 | F1 | 一般 | 性能 | 鲁棒性 | 在输入不合法值时不稳定 | 048 |
| … |  |  | … |  |  |  |
| n | F1 | 一般 | 可靠性 | 系统稳定 | 在复杂记录时运行缓慢 | 032 |

# 测试结论与建议

测试的主要问题是在重点功能挂号预约上，使用压力测试时只能进行单点测试即登陆测试，但是在进行多重分析上，使用贯穿系统的事件流，先登录后进行挂号预约，才能测试出整个系统的性能。