日期计算器

测试报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2020-04-19 | 1.0 | 对日期计算器进行黑盒测试，对其进行测试统计及缺陷分析 | 王笑然、褚宇轩、王志远、吕艺 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

测试报告

# 简介

## 目的

该测试报告将项目的测试过程及结果写成文档，对发现的问题和缺陷进行分析，为纠正软件存在的质量问题提供依据，对日期计算器应用进行质量评估。

## 范围

此报告覆盖的是日期计算器的根据既有日期与日期差获得所求日期的功能和根据两个日期计算日期偏移量的功能

## 定义、首字母缩写词和缩略语

无

## 参考资料

《日期计算器需求规约》

## 概述

本文档主要包含测试的施行情况和出现的问题

# 测试概要

测试时间：2020-4-19

测试方法：黑盒测试(包括边界值测试、等价类测试和决策表测试)

测试人员：王笑然，褚宇轩，王志远,吕艺

测试内容：在网页上输入测试用例并与预期结果对比

# 测试环境

软件：chrome浏览器

硬件：笔记本电脑

# 测试结果及分析

## 需求覆盖率及缺陷分布

下面表格中的需求覆盖率针对的是日期计算器需求规约中的需求

**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** | **计算几天后的日期** | 69 | 100% | 25 | 36.23% |  |
| **计算日期差** | 91 | 100% | 29 | 31.87% |  |
| **总计** | 160 | 100% | 54 | 33.75% |  |
| **非功**  **能项** | **暂无** | | | | | |
| **非功能项小计** | *0* | *0* | *0* | *0* | *0* |
|  | **总计** |  |  |  |  |  |

## 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 | 51 | 0 | 54 |
| **占缺陷百分比** | 0% | 5.56% | 94.44% | 0% | 100% |

# 缺陷清单

## 功能性缺陷

**表3 功能性缺陷列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **缺陷**  **编号** | **严重程度** | **功能模块** | **缺陷标题** | **缺陷描述** |
| 1 | 严重 | 计算几天后的日期 | 日期对应错误 | 当输入的值为1时，输出的结果不是公元1年 |
| 2 | 一般 | 计算几天后的日期 | 未进行错误提示 | 当输入的值不在规定的合法范围时，软件的输出提示用户输入不合法 |
| 3 | 严重 | 计算日期差 | 日期对应错误 | 输入的年份为0时会等同1990年 |
| 4 | 一般 | 计算日期差 | 未进行错误提示 | 当输入的值不在规定的合法范围时，软件的输出提示用户输入不合法 |

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

无

# 测试结论与建议

1. 对于输入的日期判定应进行错误处理，对超出边界的日期进行提示
2. 对于年份为[0, 99]范围内的年份应该进行处理，使之与公元[0, 99]进行对应
3. 日期差的输出结果不应为小数