

[基于 flask 的 web 计算器 1.0]

# 测 试 报 告

中国科学技术大学软件工程课程设计 I 第九组

李双利 王新 张培宁

# 1. 引言

## 1.1 编写目的

本测试报告为基于 flask 的 web 计算器项目的测试报告，目的在于总结测试阶段的测试以及分析测试结果，描述系统是否符合需求。预期参考人员包括用户、测试人员、开发人员。

## 1.2 项目背景

在日常学习生活中，我们在很多情况下都有计算的需要，可能是某道题目，也可能是计算花费等情形，对于本项目，用户无需安装任何 app，通过简单的浏览器即可随时随地达到计算的目的，用户只需一个简单的网页接口就能即时获取到想要的计算结果，还可以查询历史计算记录，计算复杂的表达式，包括各种常用函数，用户还可以根据自己的喜好更改界面主题。

# 2. 测试概要

## 2.1 说明

测试阶段包括功能集成测试，性能测试，压力测试，异常测试，负载测试并且对测试过程进行评估。本计划所提到的测试类型是需求阶段的测试，即对基于 flask 的 web 计算器进行功能验证的测试过程。

**测试范围：**基本计算功能，用户界面展示

**测试目的：**发现计算器项目中目前存在的问题，并且对影响系统正常运行的部分 BUG 进行修复，以对完善系统。

**测试过程执行：**整体测试过程结束，功能模块测试测试最为顺利，其它模块因技术，条件等原因或多或少存在一些问题。

## 2.2 测试用例设计

本测试用例主要采用黑盒测试方法，功能测试部分采用了边界值，等价类等用例设计方法，其它部分测试如性能测试，兼容性测试，压力测试等涉及到各自不同的测试方式和测试工具。

## 2.3 测试环境

下表列出了测试的系统环境

应用环境（相关软件、操作系统等）
测试操作系统 1: PC, Windows 10
测试操作系统 2: iPhone, ios11
测试操作系统 3: Ipad mini, ios11
服务器操作系统: ubuntu 14.04 lts
Web 服务器: gunicorn 和 nginx
测试浏览器: chrome 和 safari
测试网络环境: ustc 校园网, 带宽 10Mb

### 3. 测试内容和执行情况

#### 3.1 功能测试

##### (1) 科学计算

##### a) 带括号的基本加减乘除运算（包括小数测试）

测试样例	结果	是否正确	测试目标
$1-1+2$	2	正确	基本加减运算
$2*3-6/2$	3	正确	基本乘除运算
$(1-2+(2-1))$	2	正确	带括号的加减运算
$(2*3+(4-2)/2)$	7	正确	带括号的乘除运算
$3.14-3+2.5$	2.64	正确	小数加减运算
$3.14*1.23-6.2/3.1$	1.8622	正确	小数乘除运算
$(3.14-7-(5.31-2.56))$	-1.11	正确	小数带括号加减运算
$(3.14*6-(6.2/4-7/5))$	18.69	正确	小数带括号乘除运算

b) 对数运算

测试样例	结果	是否正确	测试目标
$\ln(9)$	2.19722457734	正确	自然对数
$\ln(e)$	1	正确	常数 e
$\log(9)$	0.95424250943	正确	以十为底的对数

c) 指数运算

测试样例	结果	是否正确	测试目标
$9^{-1}$	0.111111111111	正确	测试计算 $X^{-1}$
$3^2$	9	正确	测试计算 $X^2$
$2^{1/2}$	1.41421356237	正确	测试计算 $X^{1/2}$
$3^3$	27	正确	测试计算 $X^y$ (y 是整数)

d) 三角运算

测试样例	结果	是否正确	测试目标
$\cos(1)$	0.54030230586	正确	测试余弦函数
$\sin(1)$	0.8414709848	正确	测试正弦函数
$\tan(1)$	1.55740772465	正确	测试正切函数

e) 绝对值和阶乘

测试样例	结果	是否正确	测试目标
$\text{abs}(7-9)$	2	正确	测试绝对值函数
$9!$	362880	正确	测试阶乘函数
$30!$	0	溢出	测试阶乘边界

(2) 输入模式

a) 键盘输入

经过测试把鼠标放在计算器显示主屏幕处，右键点击后可以进行键盘的表达式输入，并且按下回车即返回结果。

b) 触屏输入（针对移动设备）

所有按键可以正常进行触摸操作，测试通过。

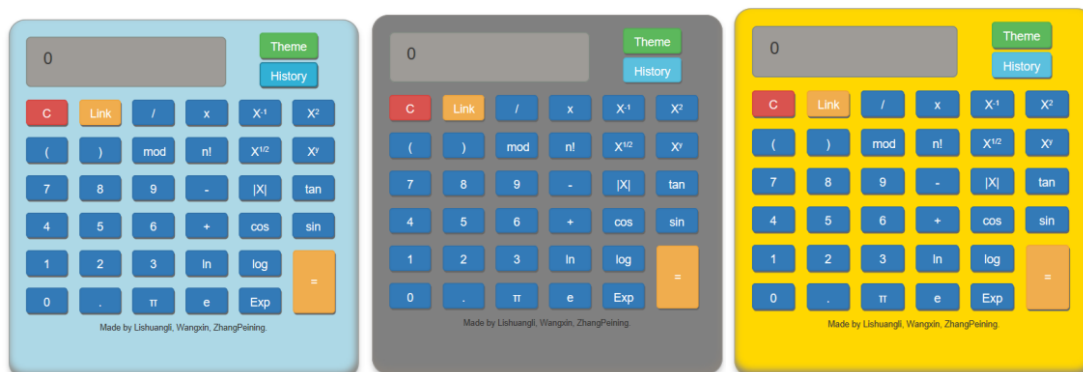
(3) 查看历史记录

- 点击屏幕上的 history 按钮后，即可显示历史记录。
- 最多显示五条历史，如果超过五条按照队列的 FIFO 原则显示。



#### (4) 主题切换

按下 theme 按钮后，可以进行三种主题的切换。

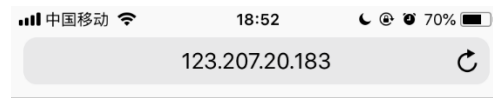


### 3. 2 性能测试

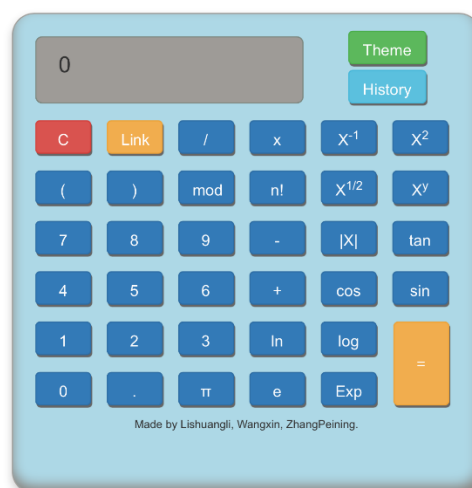
用例名称	基本要求	测试情况	测试通过	
			是	否
运行情况-打开 WEB (1)	测试 1 人打开 WEB 时，系统的运行情况	响应时间达到预期结果	<input checked="" type="checkbox"/>	<input type="checkbox"/>
运行情况-打开 WEB (4)	测试 4 人打开 WEB 时，系统的运行情况	响应时间达到预期结果	<input checked="" type="checkbox"/>	<input type="checkbox"/>
运行情况-计算返回	测试 4 人同时向服务器端请求计算结果	结算结果响应时间达到预期结果	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPU 和内存占用 (PC 的 Edge 为例)	在打开计算机的 web 界面运行的占用情况	CPU 占用 0.1%，内存占用 81M (这与浏览器有关)	<input checked="" type="checkbox"/>	<input type="checkbox"/>

### 3.3 界面测试

#### (1) 手机端 web 界面



#### (2) 平板 web 界面（和 PC 的 web 界面形同）



### 3.4 异常测试

用例名称	基本要求	测试情况	测试通过	
			是	否
用户输入异常测试（1）	用户在按键输入时，第一个输入的不能为操作符	达到预期效果，用户不能第一个输入加减乘数等	<input checked="" type="checkbox"/>	<input type="checkbox"/>
用户输入异常测试（2）	用户错误的输入表达式后会反馈错误输入的提示	辅助屏幕显示预期的提示 “Input Error”	<input checked="" type="checkbox"/>	<input type="checkbox"/>
用户输入异常测试（3）	用户错误的输入连续多个操作符（比如连续加减）会反馈错误输入的提示	未达到预期效果，返回了错误的显示结果。	<input type="checkbox"/>	<input checked="" type="checkbox"/>
计算异常测试	用户输入格式合法的表达式但是无法完成计算时（比如除法操作的除数为 0 时），会提示用户输入错误。	辅助屏幕显示预期的提示 “Input Error”	<input checked="" type="checkbox"/>	<input type="checkbox"/>
超出长度限制测试	如果计算结果或者用户输入在显示区域无法完全显示时（即长度过长），会反馈提示用户。	辅助屏幕显示预期的提示 “Reach Digit Limit”	<input checked="" type="checkbox"/>	<input type="checkbox"/>



## 4. 缺陷统计

经过本次测试分析统计得到重要缺陷以及修复情况。

模块名称	缺陷编号	简要描述	缺陷修复	测试通过 (Pass/Fail)
显示效果	ERROR-1	在移动端的 web 界面计算器显示排版异常	是	是
计算异常	ERROR-2	部分错误输入未作出正确的反馈提示	否	-
性能测试	ERROR-3	偶尔出现响应延迟	是	是

未修复的 bug: ERROR-2 部分错误输入未作出正确的反馈提示。

## 5. 测试结论与建议

### 5.1 结论

“基于 flask 的 web 计算器”在用户现场环境进行功能、性能、界面、异常、兼容性五个方面进行了全面、严格、规范的测试。测试结果表明:“基于 flask 的 web 计算器”项目达到预期要求,并具有以下特点:

(1) **项目架构便于维护。**该项目采用 html-jquery-flask 架构,前端和后台开发完全独立并且由 jQuery 建立中间的桥梁,系统结构清晰明确,可满足三个开发者相互独立又彼此协作。

(2) **功能全面。**该计算器支持科学计算,能够进行三角函数,指数,对数等函数计算,还可以进行绝对值、整数阶乘的计算,同时还支持历史计算记录的保存和查看,具有较好的用户体验。

(3) **交互界面美观。**项目是基于 Bootstrap 和 jQuery 框架开发前端,具有较为美观的前端显示界面,同时还设有辅助显示区域用以提示用户相关的信息,此外,还支持三种颜色的主题切换,给予用户更多的界面选择。

(4) **可靠性高。**项目的 web 后端框架为 Flask,同时使用 gunicorn 作为 web 服务器,其稳定性和可靠性优于 flask 自带的 web 服务器,同时还有 nginx 进行反向代理和负载均衡,系统的稳定性和可靠性都得到保证。

(5) **兼容性好。**该计算器项是 web 端的,所以对平台没有限制,用户可以在任何平台打开网页使用,能满足用户在各种操作系统,各种 web 浏览器使用。

(6) **测试结论:**通过。

### 5.2 建议和进一步完善

该项目经过严格的测试,还需要在以下几个方面进一步完善。

- (1) 由于开发时间的限制,对于进制运算的模块未做实现,在接下来的工作中可以把进制运算的模块加入,进一步完善功能。
- (2) 项目还存在部分异常不能捕获的问题,可以对用户的输入做进一步的检测,做到完整的检测异常输入。
- (3) 目前服务器端采用的是四线程并发,可以支持四个用户同时计算,可以考虑进一步增加并行线程数。