# 1 Testing Description

To estimate the performance of xxx, we need to do stress test to verify the performance on xxx DEV2.

# 2 Standard

|  |  |  |
| --- | --- | --- |
| Average Concurrent Users(C) | Total User(n)\* Cost Time(L)/Total Working Time(T) | C=nL/T |
| Peak Concurrent Users(C’) | C’ ≈C+3\*SQRT(C) |  |

Example:

Normally, one operation will cost around 5s.

We assume user operate around 10 times one day.

And we take work time as total 8 hours, and max user number is 8000.

So the average concurrent numbers of uses shall be around 8000\*5\*10/8\*60\*60 = 14.

And peak value shall be 14+3\*SQRT(14) ≈25.

We just set the group as 20, 40, 60, 100 to test more about the performance.

# 3 Detailed Plan

## 3.1 Test Plan Name

Stress Test For xxx

## 3.2 Test Background

This part is mainly about description of hardware and software requirements of the user testing environment.

Tested server: xxx DEV2

CPU：xx

Memory：xxG

Software：jMeter zabbix

## 3.3 Testing Items

The items must be recorded in the test, such as number of thread, minimum responding time, maximum responding time and so on.

Number of users 20

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | Average | MIN | MAX | Err |
| xxxx |  |  |  |  |
| xxxx |  |  |  |  |

Number of users 40

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | Average | MIN | MAX | Err |
| xxxx |  |  |  |  |
| xxxx |  |  |  |  |

Number of users 60

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | Average | MIN | MAX | Err |
| xxxx |  |  |  |  |
| xxxx |  |  |  |  |

Number of users 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | Average | MIN | MAX | Err |
| xxxx |  |  |  |  |
| xxxx |  |  |  |  |

## 3.4 Test Scenario

The test scenario will be accessed in this test.

**xxx**

https://xxdev2.tkeview.com/xxx/xxx/xxx

**xxxx**

https://xxdev2.tkeview.com/xxx/xxx/xxx

CPU User

## 3.5 Testing Strength Estimation

This part is about the testing strength which is need by the system based on the usage of users. It is required to calculate the estimate result of testing stress and the input condition in the testing process.

|  |  |  |  |
| --- | --- | --- | --- |
| Amount of Access | CPU | Httpd Connections | MySQL Connections |
| 20 |  |  |  |
| 40 |  |  |  |
| 60 |  |  |  |
| 100 |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL(CPU average) | 20 | 40 | 60 | 100 |
| xxx |  |  |  |  |
| xxxx |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL(CPU MAX) | 20 | 40 | 60 | 100 |
| xxx |  |  |  |  |
| xxxx |  |  |  |  |

## 3.6 Preparation of Testing Environment

This part is about what must be possessed in both server side and client side, and the functions must be realized in testing devices. Additionally, the number of testing devices and how to allocate the devices need to be concerned.

Client ： jMeter

Server：zabbix

## 3.7 Testing Method and Tools

## We use jMeter to create the script of simulating the requests from users to server, then use zabbix to monitor the server status.

## 3.8 Testing Time Plan

## Testing start time 201x-xx-xx xx:00 PM(Beijing Time).

|  |  |
| --- | --- |
| Preparation for environment | xx DEV2 |
| Preparation for Test Script | 201x-xx-xx xx:00 AM |
| Strength stability test | 201x-xx-xx xx:00 PM |

## 3.9 Problem and solution in the test

### 3.9.1 Pause standard and restart requirements

Pause standard: While the tested application prompts errors frequently in strength stability test (more than once per hour), users or company require to pause the test.

Restart requirements: After debugging, if reliability of tested application is improved, the test can be restarted.

### 3.9.2 Unforeseeable problem

Unforeseeable problem includes:

Data error

Testing environments is broken so that the test cannot proceed.

When the problem above appears, the test stops and is finished. The testing report should be created and indicates the reason for stopping the test.

## 3.10 Testing Report

Date for submitting the testing report：201x-xx-xx

### 3.10.1 Testing files should be created

Testing record

Testing report

### 3.10.2 Template for testing report

|  |  |  |  |
| --- | --- | --- | --- |
| Tested Module | xxx | Reporter | xxx |
|  | Server Memory and CPU Usage | Refer to data above | |
| Responding Time | Refer to data above | |
| Maximum Concurrency Number | Refer to data above | |
| Failed Times | Refer to data above | |
| The longest running time of normal operation | Refer to data above | |
| The shortest running time of normal operation | Refer to data above | |
| Relationship between failed times and concurrency number | Refer to data above | |
| Testing Environment | xx DEV2 | | |

# 4 Personnel and Responsibilities

Division of duty, personnel distribution

Developer name: Be responsible to generate stress data by jMeter and execute concurrency test.

Architecture member name：Be responsible for monitoring and recording the server data, and handling issue on server

# ５　Conclusion

# For normal concurrent visit number: xx, test functions: xxx work well.

# And can handle around xx concurrent accesses.