Assignment 2

Table of Contents

6. Test Report

7. Conclusion

1.1. Introduction

1.1.1. Purpose

The purpose of this document is to find possible problems in given project, show the recommended requirements, suggest testing strategy and describe suitable resources for valuable system test document.

1.1.2. Background

My Web Server is an open source software. The Software Development Company (SDC) plans to use this server on large-scale of Internet of Things (IOT) in order to display the information from sensors. SDC wants to evaluate whether this server fulfills all of their security, functional and performance needs.

1.1.3. Goals

- SDC
- easy deploy
- wide range: many deployable devices
- minimal configuration
- easy integration: unambiguous API

- easy access
- absolute security

1.1.4. Requirements to be tested

Because of lack of time and team members only these requirements will be tested. Other requirements will be left for the next iteration.

Req 1. The Web Server should be able to be stopped

Req 2. If the access log could not be written to during the start of the Web Server an error message should be presented

---

Reg 3. The access log should be viewable from a text editor

Req 4. The web server mush follow minimum requirements for HTTP 1.1

Reg 5. The files in the shared resources should be served to the client.

Req 6. If the resource container can’t be accessed when the webserver is started an error message should be presented.

1.1.5. Scope

This test plan applies to Data and Database Integrity Testing, Unit

Testing, Function Testing, Business Cycle Testing, User Interface Testing, Performance Testing, Load Testing, Stress Testing, Security & Access Testing, API Testing, Configuration Testing and Installation Testing.

1.2. Requirements

Requirement	Description
Req 1.	The web server should be responsive under high load.
Req 2.	The web server must follow minimum requirements for HTTP 1.1
Req 3.	The web server must work on Linux, Mac, Windows*.
Req 4.	The source code should be released under GPL-2.0.
Req 5.	The access log should be viewable from a text editor.

2. Test Strategy

Test Strategy section displays suggested approach of testing the identified testing-targets. Previous section described *what* are the testing-targets, this one will describe *how* the system is going to be tested.

2.1. Resources

This section will list the resources to be used for the MyWebServer testing process.

2.1.1. Team

All normal team member functions (test manager, test designer, tester, implementer) will be performed by one member. This member will take the responsibility of reporting, generating artifacts (documents), executing tests and loging the gathered results.

2.1.2. System

The server module will mainly run on the defined PC’s as localhost. For some function tests it will be emulated on Virtual Machines. The access tests will be done from various client systems to ensure compatibility and meet the requirements.

The server test stations must have the following software installed and properly configured:

- JAVA 8 JDK

The server needs to be setup locally or remotely to run each test suit. The default configuration used:

- port: 1091
- shared resource folder: /MyWebServer/tests/se/lnu/http/resources/inner

2.2. Stakeholders and their Goals

Software Development Company (SDC) would like to have MyWebServer tested to find out what the current state of the product is. Their goal is to see if it is possible to use this open-source project as the foundation for their own easy to deploy web server.

SDC also has certain requirements on MyWebServer that needs to be fullfilled and those are listed under the requirements section.

2.3. Project Milestones

The dates are just estimates and may vary since iteration-efforts may depend on each other. So planning and working will be done in an agile-way as defined in the test-plan.

- Test Plan (15hrs.)
- Test Design (60hrs.)
- Test Implementation (45hrs.)
- Test Execution (30hrs.)
- Test Evaluation (25hrs.)

2.4. Risks

As in every software relating process risks are unavoidable.

The lack of knowledge in programming languages and testing process is one of the biggest risks. These potential risks could cause time issues, problems wiht code and other.

Lack of personnel in the team is another possible risk. Since there will be only one team member to conduct and report tests not all requirements might be finished successfully.

3. Deliverables

3.1. Test Plan

The Test Plan will define *what* and *when* will be done. It divides the test-effort into manageable iterations and sets time-limitations.

3.2. Test Results

For each test executed a test-result form will be created. This should include the name or ID of the test-case or specification it relates to, the execution-date, name of the tester and the result of the test.

3.3. Test Report

A final evaluation of the test-activities and their results will be presented.

4. Test-Plan

The fulfillment of the test-strategy will be distributed in three iterations. To complete means all test-cases for each listed requirement is designed, executed and evaluated. The Test-Project Goal is to meet all completion-criteria from the test-strategy.

Iteration	Goals/Milestones	Requirement-Number	Start-Date	End-D
1st iteration	**Unit-Testing** : Confirm JUnit-Testsuit Verify unit-coverage. **Function-Testing**: Webserver works on OS's HTTP 1.1 Standard Access log viewable UC1: Start of webserver UC2: Termination of webserver Verify Integration-Tests. **Performance Testing**: Performance Profiling	n/a	15/12/2016	19/12/2
2nd iteration	**Data Integrity Testing**: UC3: System delivers to browser Verify correct retrieval Verify simultaneous access. **Function-**	n/a	19/12/2016	22/12/2

	Testing: Verify behavior in LAN. **Performance Testing**: Start in reasonable time Access-Time in LAN. **Load Testing**: Responsive under high load			
3rd iteration	**UI Testing**: Verify easy access. **Security and Access Testing**: Webserver security report. **Configuration Testing**: Minimal Configuration. **Installation Testing**: Easy deployment of server. **API Testing**: Easy integration and adaptation. **Business Cycle Testing**: GPL-2 License published	n/a	19/12/2016	23/12/2

5. Test-Cases

Test case 1.

Info	Input

Test ID	1
Requirement	1
Actor	Administrator
Precondition	A web server has been started
Postcondition	A note has been written to the access log that the server has been stoped

Step	Action	Expected Result	Pass/Fail
1.	Stop the webserver(1)	System stops the webserver and presents that the webserver has been stopped	Fail

---

Test case 2.

Info	Input
Test ID	2
Requirement	2
Actor	Administrator
Precondition	Restric acess to the access log
Postcondition	Web Server could not be started due to the access log could not be written to

Step	Action	Expected Result	Pass/Fail
1.	Start the server(1)	System asks for socket port number and shared resource container	Pass

2.	Input socket port number and shared resource container	Web server do not start on giving port and presents the error message System presents an error message: “Cannot write to server log file log.txt”	Fail

Test case 3.

Info	Input
Test ID	3
Requirement	3
Actor	Administrator
Precondition	A web server has been started
Postcondition	Log file should be viewable from an editor

Step	Action	Expected Result	Pass/Fail
1.	Open the access file in a Text editor	The file should be viewable	Fail

Test case 4.

Info	Input
Test ID	4
Requirement	4
Actor	Administrator

Precondition	A web server has been started
Postcondition	Successfully handles http 1.1 requests

Step	Action	Expected Result	Pass/Fail
1.	Send HTTP requests using JMeter	System handles the request and specify HTTP 1.1 in the responser	Pass

---

Test case 5.

Info	Input
Test ID	5
Requirement	5
Actor	Administrator
Precondition	A web server has been started
Postcondition	A note in the access log was written, that the access happened with request information and the result of the request

Step	Action	Expected Result	Pass/Fail
1.	Access the shared resource(1)	System delivers the shared resource and a success message is written to the access log	Fail

---

Test case 6.

Info	Input

Test ID	6
Requirement	6
Actor	Administrator
Precondition	Change access on the resource folder so that the WebServer do not have access to it.
Postcondition	Web Server could not be started due to restriction to the shared folder

Step	Action	Expected Result	Pass/Fail
1.	Start the server(1)	System asks for socket port number and shared resource container	Pass
2.	Input socket port number and shared resource container	Web server do not start on giving port and presents the error message System presents an error message: “No access to folder XX”	Fail

6. Test Report

Test Case 1

The web server stops as expected and the console present a message saying the web server has been stopped. But as far as we can see, no access log does exist and there for does the tests for the test case fail since the feature request is not completely fullfilled

Test Case 2

We where not able to perform this test since we where not able to find any access log file. As far as we can see it seems like there is’nt any. Because of our lack of experience in Java we also dont know how to implement one. The tests for the test case has therefor failed but it will also be postponed for the nest iteration.

Test Case 3

We have been searcing both the provided files for the application as well as the source code. And we have not been able to find any indication that a access log file actually exist. The test for the test case has failed.

Test Case 4

The response headers during the test clearly specifies HTTP 1.1. The test for the test case has passed.

Test Case 5

Shared resources are successfully access by viewing localhost on provided port trought the browser and the console utputs the requests when a shared resource is requested. But as we have stated earlier we where not able to find any access log file and therefor there is no way for us to check if required information is logged to the access file or not. The tests for the test case has failed

Test Case 6

The web server did start without any known issues as expected. When trying to access a resource that that has restricted access an HTTP error is given in the browser as expected. How ever no error or other indications where outputed anywhere. The tests for the test case has failed.

7. Conclusion

The application is stable enough to be used for the purpose that SDC wants to use it for. How ever as specified in the tests a few of the features such as the access log file is missing and we suggest we move forward to implement this. And the workload to turn this open source project into a fully functional web server won’t be to heavy and is doable witout investing too much time into the application.