My Web-Server

TEST PLAN

Version 2

20/12/2017

VERSION HISTORY

Version #	Author	Describe	Date
1.0	Project Manager	First Version	13/12/2017
2.0	Project Manager	Final Version	20/12/2017

TABLE OF CONTENTS

1 INTRODUCTION

1.1 PURPOSE

This Test Plan document for the Web-Server for IOT Developers supports the following objectives:

1- Identify existing project information and the software components that should be tested
2- List the recommended Requirements for Test.
3- Recommend and describe the testing strategies to be employed.
4- Identify the required resources and provide an estimate of the test efforts.
5- List the deliverable elements of the test.

1.2 BACKGROUND

My Web-Server an open source software that supports multiple platforms. It's an easy to deploy server and an open source. SDC a Software Development Company plans to use this server on a wide range of Internet of Things (IOT) in order to present information from sensors etc. SDC wants to know if this software fulfills all of their functional, performance and security needs.

1.3 OBJECTIVES

To apply the test methods in this test plan, objectives and goals that are demanded should be set to clear any errors and to measure the faults and errors that could possibly affect the software.

1- Test the software for SDC evaluation.
2- Wide range many deployable devices minimal configuration.
3- Easy deploy to support different platforms.
4- Easy Integration: unambiguous API
5- Absolute security
6- Easy access.

1.4 SCOPE

This Test Plan applies to the unit and integration tests that will be conducted on the My Web-Server. In the test strategy will be shown the test requires in both Unit and Integration test.

1.4.1 Requirements

The listing below identifies those items (use cases, functional requirements, and non-functional requirements) that have been identified as targets for testing. This list represents what will be tested.

1.4.1.1 Use Cases

UC #	Description
UC1	Start Server
UC2	Stop Server
UC3	Request Shared Resource

1.4.1.2 Suplimentary Requirements

SR #	Description
R1	The web server should be responsive under high load
R2	Minimum requirements for HTTP 1
R3	The web server must work on Linux, Mac, Windows
R4	The source code should be released under GPL-2.0
R5	The access log should be viewable from a text editor

1.4.1.3 Informal Requirements

UC #	Description
IR1	**Usability**
IR2	Security

1.4.2 Items Not to be Tested

InT #	Description
InT1	PerformanceTesting: Profiling
InT2	Function Testing: Verify behavior in LAN.
InT3	UI Testing: Verify easy access.
InT4	API Testing: Easy integration and adaptation.
InT5	The source code should be released under GPL-2.0
InT6	The web server work on Linux and Mac

1.4.3 Test Case Identification.

Test case ID#	Description	Responsibility	Date
T1.1	Server start	Tester A	2017-12-14
T1.2	Unavailable port	Tester A	2017-12-14
T1.3	Shared resource before socket param.	Tester A	2017-12-15
T1.4	Incorrect socket number	Tester A	2017-12-15
T1.5	Incorrect shared resource	Tester A	2017-12-16
T1.6	Protected shared resource	Tester A	2017-12-16
T1.7	Writing to txt file	Tester B	2017-12-16
T2.1	Stopping server	Tester B	2017-12-16
T2.2	Using capital STOP to stop the server	Tester B	2017-12-16
T3.1	Response 200 OK	Tester B	2017-12-17
T3.2	Response 400 Bad Request	Tester B	2017-12-17
T3.3	Response 403 Forbidden	Tester B	2017-12-18
T3.4	Response 404 Not Found	Tester C	2017-12-18
T3.5	Response 500 Internal Server Error	Tester C	2017-12-19
T4.1	Response under high load	Tester C	2017-12-19
T5.1	Testing the server with https	Tester C	2017-12-20

1.5 RISKS AND CONSTRAINTS

Among the factors that could hinder the steady development of the software are included time constraints, miscommunication or misunderstanding between team members. Furthermore, the following risks have been identified as well as the correct action to mitigate their impact on the project. The severity of the risk is based on how the project would be affected if the risk was triggered. The trigger is what event would cause the risk to become a problem to be handled.

#	Risk	Impact	Trigger	Mitigation Plan
1	Lack of knowledge in programming language and testing process.	High	Delays in implementation date	Choose the best one in the team to check over the testing process and make sure it is done accordingly
2	Aggressive schedule	High (affects delivery date)	Slips in a scheduled phase (delaying an iteration)	No iteration will be postponed or delayed.

2 TEST SCHEDULE

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.

Task Name	Milestones/Goals	Start date	End date	Effort
Test Planning	Planning the iteration schedule along with setting the milestones of the testing methods and objectives.	13-12-2017	20-12-2017	10 h
Iteration 1	**Unit-Testing** : Confirm JUnit-Testsuit Verify unit-coverage.	14-12-2017	16-12-2017	15 h
	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.			
	PerformanceTesting: Performance			
Iteration 2	**Data Integrity Testing**: UC3: System delivers to browser Verify correct retrieval Verify Simultaneous access.	17-12-2017	20-12-2017	30h
	Performance Testing: Start in reasonable time Access-Time in LAN.			
	Load Testing: Responsive under high load			
Iteration 3	**Security and Access Testing**: Webserver security report.	20-12-2017	22-12-2017	30h
	Configuration Testing: Minimal Configuration.			
	Installation Testing: Easy deployment of server			
	Business Cycle Testing: GPL-2 License published			
Delivery date	Submission the results and required documentation	23-12-2017		

3 DELIVERABLES

Deliverable	Responsible	Date / Milestone
Test Plan	Project Manager	13-12-2017
Test Strategy	Tester A	14-12-2017
Develop Test Cases	Tester B	22-12-2017
Produce Traceability Matrix Document (Test Report)	Tester C	21-12-2017

4 RESPONSIBILITIES

Learning and understanding the test tools and methods is each member responsibility.

Moreover, the team should work together in creating test cases executing the tests and gather the test results in the test cases and test report document. The project manager is then responsible for reviewing the test results and take appropriate action.

5 DEPENDENCIES

5.1 PERSONNEL DEPENDENCIES

Due to the lack of experienced testers in the test team, the testers available are required to be more careful and correctly revise their operations before developing, performing and validating tests.

5.2 SOFTWARE DEPENDENCIES

The source code must be unit tested and provided within the scheduled time outlined in the Project Schedule.

6 DOCUMENTATIONS

The following documentation will be available at the end of the test phase:

1- Test Plan
2- Test Cases
3- Test Strategy
4- Test Report

Appendix A: References

The following table summarizes the documents referenced in this document.

Document Name and Version	Description	Location
Web Server Document	List of requirements	Anonymous
My Web-Server Source Code	Application Source Code	https://github.com/dntoll/MyWebServer