Test Strategy

1.Revision History

Date	Version	Author	Description
2020-11-26	1.0	Test Manager	Planning Test Strategy documentation.
2020-11-26	1.1	Test Designer	Planning Test environment, Test cases and how to test.
2020-11-27	1.2	Test Manager	Adding information to the Test Environment.
2020-11-30	1.3	Test Manager	Adding Testing tools, Release Control, Risk Analysis and Review and Approvals.
2020-12-1	1.4	Test Manager	Adding information to Roles and Responsibility.
2020-12-06	1.5	Test Manager	Changing name for heading 5, Adding documentation to Testing Tools.

2020-12-07	1.6	Test manager	Removing Roles and responsibility and adding Resources instead with all team members and information about them.

2. Scope

The Test Strategy is made for a simple to deploy web server that is created by the small Software Company (SDC). The goal with the product is to easily deploy a java-web-server that can be deployed onto a lot of different devices. This to attract the attention of a wide range of Internet Of Things (IOT) developers who want an easily deployed java-web-server for their projects. By testing the different requirements given by the stakeholders we can assure that the software is working correctly and therefore build confidence in our software.

SDC has given us the opportunity to make a Test Strategy for SDC and the SDC management will be reviewing this documentation.

Testing Activities

Testing activities that will be carried out in this Test Strategy is **Read Documentation, Test Cases Design, Regression Testing, Bug Reporting, Meetings and Reporting Testing Progress.** This will all be presented in a table below with a description and schedule time for the activity.

Activity	**Description**
Read Documents.	Check the requirements and Use Cases.
Test Cases Design.	Create manual test steps that will be followed step by step for an Expected result
Regression Testing.	Test a new build of the tests by re-run test cases.
Bug Reporting.	Find critical problems.
Meetings.	Daily meetings.
Reporting Testing Progress.	Checking test progress.

3. Testing Areas

JUnit tests, Manual test cases, functional tests and non-functional tests will be used in the project to test and make all requirements fulfilled that have been given by the company SDC.

Testing areas that have been given by the stakeholders are **Start Server, Stop Server and Request Shared Resource.** These testing areas are different from each other and have different testing levels depending on the complexity of the area. The higher testing level the more prioritized and time consuming is the testing area. The different levels are displayed in a table.

Test Area	**Testing Level**
Start Server	High
Stop Server	Medium
Request Shared Resource	Low

4. Resources

Human resources

The test driven development (TDD) team consists of **two** main roles **Test manager** and **Test designer**, however, there are five roles total we have hired a **implementer, Designer** and **system tester** to make all corresponding tasks and responsibilities fulfilled.

Test manager is responsible for providing oversight, making sure that the testing is done and moving in the right direction. Test manager will provide resources to the staff and report to management.

Test designer is responsible for identification, implementation,prioritization of test cases and constructing the test plan with documentation.

Hired roles

Implementer is specifically responsible for implementing test classes with test packages as well as the unit tests and documents the implementation.

Designer is specifically responsible for identification and definition of tests on attributes,operations and associations of the corresponding test classes. Other responsibilities are construct classes based on previous identifications. Construct test packages based on previous identification.

System tester is specifically responsible for execution of all tests, other responsibilities are log results of the test, error-recovery and documentation of possible defects.

Technological Resources

- **Computer specs for computer 1:**
  GPU:GeForce RTX 2060 SUPER.
  CPU: Intel(R)Core(™) i7-4790K CPU @4.00GHz.
  RAM: 16 GB.
  OS: Windows.

- **Computer specs for computer 2:**
  GPU: GeForce RTX 2070 SUPER.
  CPU: Intel(R) Core(™) i7-9700K CPU @3,60GHz.
  RAM : 32 GB.
  OS: Windows.

- **Unit tests**
  Test Driven Development(TDD) is the approach for the project & JUnit test framework is going to be used as an automation tool. Types of Testing that will be included are Load testing, Security testing, Performance testing.

- **Software testing experience**

Learning testing from a course called “Programvaruteknik” (1DV600) at Linnèaus University, learned to create manual test cases as well as creating a small test report, a test plan and Unit tests by using Jest framework in javaScript.

- **TDD Resources**
  Resources about test driven development From course 1DV609 “Mjukvarutestning” Lectures, Guest Lectures, Course Literature, External Resources and TDD Guide.

- **Programming Language**
  The programming language used is Java and Intellij IDEA is used as a text editor and development environment.

5. Testing Requirements

Requirements

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.

Testing Requirements

Req 1. This requirement is tested with a load test. A load test is when a system is given a great amount of requests from clients in a short period of time, which can be done with a JUnit stress test or by using JMeter.

Req 2. This requirement is tested by checking the request object in headers for HTTP 1.1. In the browser when the web server is connected to a port and started.

Req 3. This requirement is tested by using LambdaTest to make local tests on the web server in different operating systems.

Req 4. The source code will be released with General Public License (GPL), this is done by making copyright of the software and the license gives legal permission to copy, distribute and /or modify the software by other developers.

Req 5. Access log will be saved in a text editor, to be viewed by stakeholder and tester.

Storing Test Data

Test data from JUnit tests received after testing will be stored in documentation, in TestPad which is a test management tool where it is indicated if the test failed or was successful this data can later be updated if the test changes and passes or fails. Manual test cases will be saved in the manual test cases artifact.

6. Testing Tools

TESTPAD will be used as a test management tool to keep track of how the tests are going and to share the tests result with all developers on this project. This is a simple to use tool since you can write in the test yourself and then press if the test passed or failed. This is then stored and can be shared with all other team members and can be updated if the test result is changed.

The source code will be released with General Public License (GPL) which makes this software an open source software. Open source software has been really beneficial to the development community. Open source is a **software-licensing model** where software is used for free to all individual programmers, to add more functionalities.

The development process becomes a collaboration with multiple independent contributors. The project is open for anyone to download, study, change or update and then use software for any purpose.

There are usually some restrictions with open source software, to redistribution, modification and addition.

Pros:

- No initial discussions about buying software
- No need to worry about licenses. It’s free to use
- The code is open and so we can extend based on our needs
- More reachable because it's free
- Managed by open source community
- Attracting more customers by giving competitive pricing

Cons:

- Security
- Less resource availability in the market
- Time is taken to develop
- An employee with that tool knowledge leaving the company
- Not all the features available
- More time required to get code issues fixed

There is also **Commercial software**, the biggest difference between open source software and Commercial software is it is not free for all users. Commercial software is developed by a dedicated team and supports the same requirements of what the industry is looking at. Access to commercial software is given to the users who have purchased the license or procured an agreement to use it.

Pros**:**

- Business focused
- Full Featured and target based that will have all features as a package
- Mostly all in one solution (Includes Test Development, Management, Tracking and so on.)
- On Time / Quick Support
- Secured and reliable
- Avail training services or step-by-step documentation
- Quick fixes

Cons**:**

- Expensive
- Some companies even charge separately even for support
- Limited platforms
- Cannot customize
- Increased dependency

- Less reachable

7. Release Control

For every new release all functionality is tested to make sure that there are no bugs created with new code nor any bugs hiding in the newest version. This will be done by using regression testing, smoke testing and system integration. All this to make the most confident release for the simple to deploy web server.

8. Review and Approvals

Role	**Assignment**	**Description**	**Estimated Time(H)**	**Actual Time(H)**	**Date**
Test manager	Planning Test Strategy.	Adding structure for the Test Strategy.	5	3	2020-11-26
Test designer	Planning Test Strategy.	Planning Test environment, Test cases and how to test.	3	2	2020-11-26

Test manager	Adding documentation.	Adding information to the Test Environment.	2	2	2020-11-27
Test manager	Adding documentation.	Adding Testing tools, Release Control, Risk Analysis and Review and Approvals.	6	6	2020-11-30
Test manager	Adding documentation..	Adding information to Roles and Responsibility.	4	2	2020-12-01
Test manager	Adding documentation.	Changing name for heading 5, Adding documentation to Testing Tools.	2	1	2020-12-06
Test manager	Adding documentation.	Removing Roles and responsibility and adding Resources	3	2	2020-12-07

		instead with all team members and information about them.			