Testing Policy V.1

COS 301 Capstone Project: Demo 4

Members:

Maria Petronella Laura-Lee Strydom *u04974359*

LightBot Adaptive Traffic Control System

Contents

1. Introduction

1.1 Purpose

The purpose of this document is to describe all automated testing procedures and tools used for the testing of the LightBot systems non-functional requirements. Also specified are the expected results that each test case should return.

2. Non-functional Requirements

The following non-functional requirements are tested:

2.1 Usability

Usability refers to the quality of a user's experience when interacting with products or systems. Usability is about effectiveness, efficiency and the overall satisfaction of the user.
To test the usability of the LightBot web application as well as the Lightbot Simulator, which are the main component user will interact with, 3 metrics are used:

1. Completion Rate
2. Overall-relative efficiency
3. User satisfaction

2.2 Performance

Performance is the amount of useful work accomplished by a computer system. Performance is estimated in terms of accuracy, efficiency and speed of executing system and user instructions.

1. The Apdex score is used to calculate and track the relative performance of an application. It is done by specifying a goal for the amount of time it takes a specific web request or transaction to be completed.

2.3 Availability

The availability of the system is measured through the probability that it does not fail under certain circumstances or undergoing maintenance when it needs to be up and running.

1. Availability = Uptime / (Uptime + Downtime)

3. Tools

3.1 Usability

The only way to accurately test and quantify the results of user satisfaction, completion rate and overall-relative efficiency is to ask users to manually test the system, complete certain tasks and provide feedback. I used a **System Usability Scale questionnaire** to let users test the usability of the system and provide feedback through **Google Forms.**

The reason I chose to use the SUS questionnaire is that it is internationally recognised as well as a quick and reliable method of assessing the usability of design solutions. It provides easy feedback on the overall ease of use of a site or app from a user's perspective.

3.2 Performance

To test the performance of the LightBot system in terms of the average page load time, system response time and to calculate the Apdex score, I used an online tool called **Freshping**.

Freshping is a simple Uptime & Performance monitoring tool which helps monitoring uptime/downtime, outages and performance of websites, APIs, web services, web applications with instant outage/downtime alerts.

I chose Freshping as it is an automated, online service that efficiently monitors the status of my application without having to install additional packages. It is also a free-to-use tool.

3.3 Availability

To test the availability of the LightBot system in terms of the uptime of the system and the probability that it does not fail under certain circumstances, I used an online tool called **Uptrends**.

Uptrends is an all-in-one website monitoring solution that allows teams to manage the uptime, performance, and functionality of their websites, web applications, servers, APIs, and services so they can deliver the best online experience to customers around the world.

I chose Uptrends as it supports a variety of protocols, provides 1-minute monitoring intervals, and provides me with the resolve, connection and download time that it takes to test the LightBot system from different countries around the world.

4. Procedure

4.1 Usability

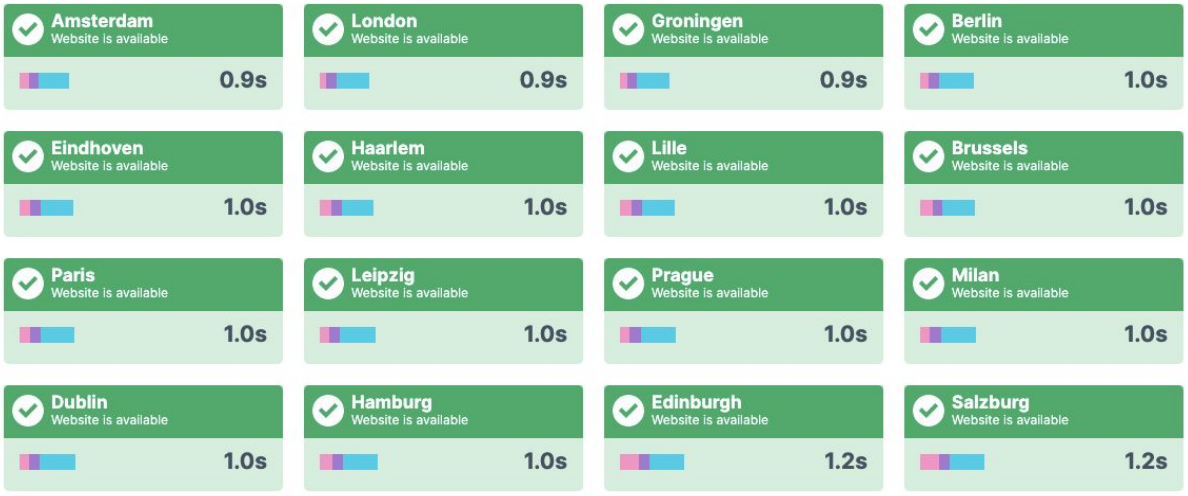
1. Users were asked if they wanted to participate voluntarily and were presented with their rights for completing the test and questionnaire.
2. They were presented with the link to the LightBot system and the Google Forms questionnaire, and I explained the purpose of the system and the test.
3. The users interacted with the system in their own time, filled out the questionnaire and asked questions about the system in which I explained the system a bit more in-depth in return.
4. The results were taken and the SUS score was calculated. This score was then compared to the original desired score specified a few demos back in the SRS document.

4.2 Performance

1. Freshping is the tool which I used to test the performance of my system.
2. I signed up a profile for the LightBot system, and entered my system URL which is then used to monitor.
3. I set up 2 checks, one the Initial check and one the Second check which pings my system in 5-minute intervals and provides me with feedback on the following:
    1. Availability
    2. Downtime
    3. Apdex
    4. Average response time
    5. Incidents
    6. Longest downtime interval
4. These are all tested automatically and run in the background even if you close the tool.

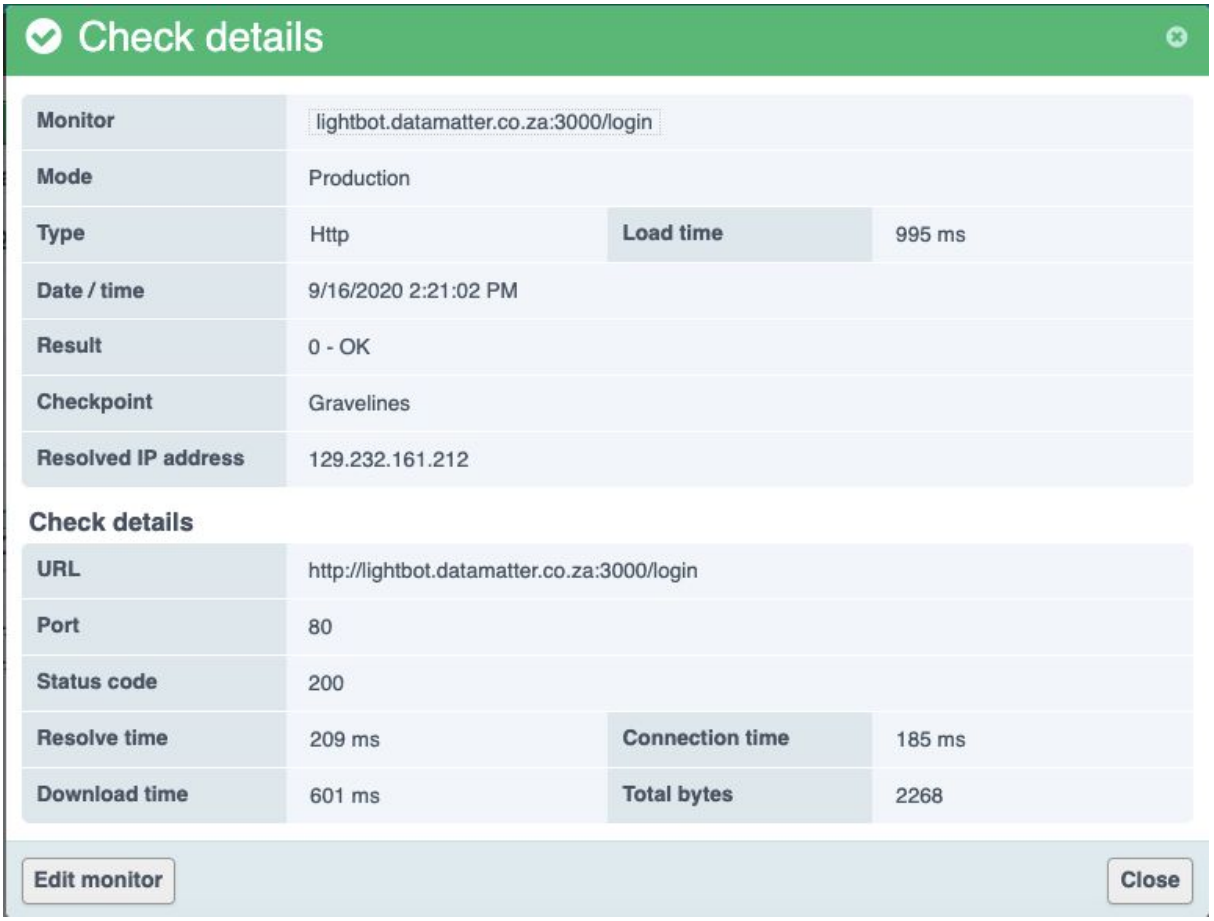
4.3 Availability

1. Uptrends is the tool which I used to test the availability of my system.
2. First, I entered my system URL which was then tested from different servers around teh world with the following output:



As shown in the above screenshots, the fastest system availability time is in AMsterdam, which is 0.9 seconds, and the slowest is in Auckland, which is 2.9 seconds.

3. I then signed up a profile for the LightBot system, and automated testing was set up for me.
4. The automated test dashboard provides me with feedback on the following details:



5. These are all tested automatically and run in the background even if you close the tool.

5. Expected Results

5.1 Usability

As specified in my SRS document, these were the desired results/outcomes of the following:

1. Completion Rate -> Effectiveness = (4 / 5) * 100% = 80%
2. Overall-relative efficiency -> Approx. between 1 - 6 seconds
3. User satisfaction - > SUS score of 80.3 or higher

The tested results were as follows:

4. Completion Rate ->
    a. User 1 = (4 / 5) * 100% = 80%
    b. User 2 = (4 / 5) * 100% = 80%
    c. User 3 = (5 / 5) * 100% = 100%
    d. Average Effectiveness = (80 + 80 + 100) / 3 = 86.67%
5. Overall-relative efficiency -> Approx. between 2.6 - 5.7 seconds
6. User satisfaction - > Average SUS score= (87.5 + 92.5 + 92.5) / 3 => 90.83
    a. To access the complete SUS questionnaire & score results that was used in my usability testing, it can be found in my Google Drive.

b. The results can be found here.

All usability tests passed with higher scoring results as were desired and anticipated.

5.2 Performance

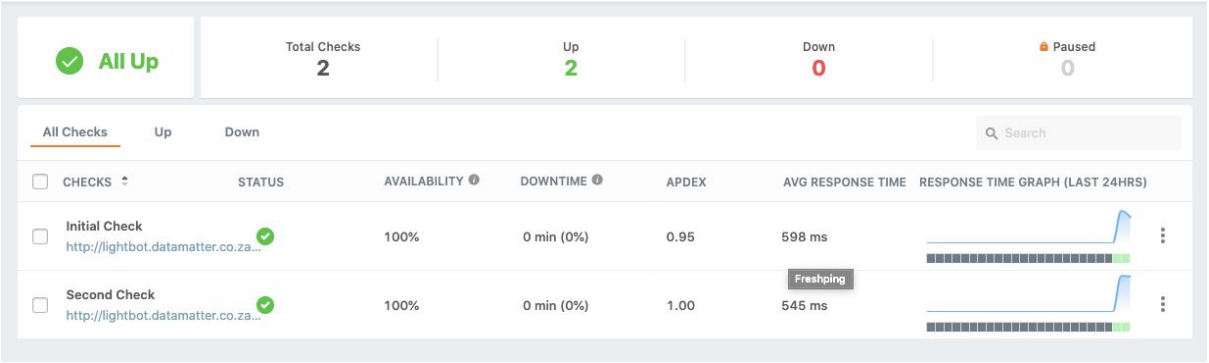
As specified in my SRS document, these were the desired results/outcomes of the following:

1. Availability -> 95.2%
2. Downtime -> N/S
3. Apdex -> Apdex score is to be at least 0.8
4. Average response time -> 3000 milliseconds / 3 seconds
5. Incidents -> N/S
6. Longest downtime interval -> N/S

The tested results were as follows:

7. Availability -> 100%
8. Downtime -> 0 min / 0%
9. Apdex -> Between 0.95 - 1.00
10. Average response time -> Approx. between 545 - 598 milliseconds
11. Incidents -> 0
12. Longest downtime interval -> 0 mins

All performance tests passed with higher scoring results as were desired and anticipated.
The results can be found here.



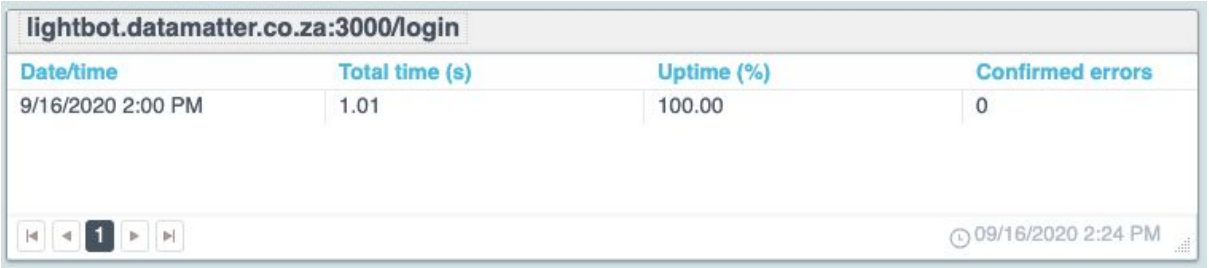
5.3 Availability

As specified in my SRS document, these were the desired results/outcomes of the following:

- For the Availability = Uptime / (Uptime + Downtime), I suspected it to be 95,2%

The tested results were as follows:

- Availability/Uptime = 100%



lightbot.datamatter.co.za:3000/login

Date/time	Total time (s)	Uptime (%)	Confirmed errors
9/16/2020 2:00 PM	1.01	100.00	0

1

09/16/2020 2:24 PM

All availability tests passed with higher scoring results as were desired and anticipated.