Non-Functional Requirements Testing

Table of Contents

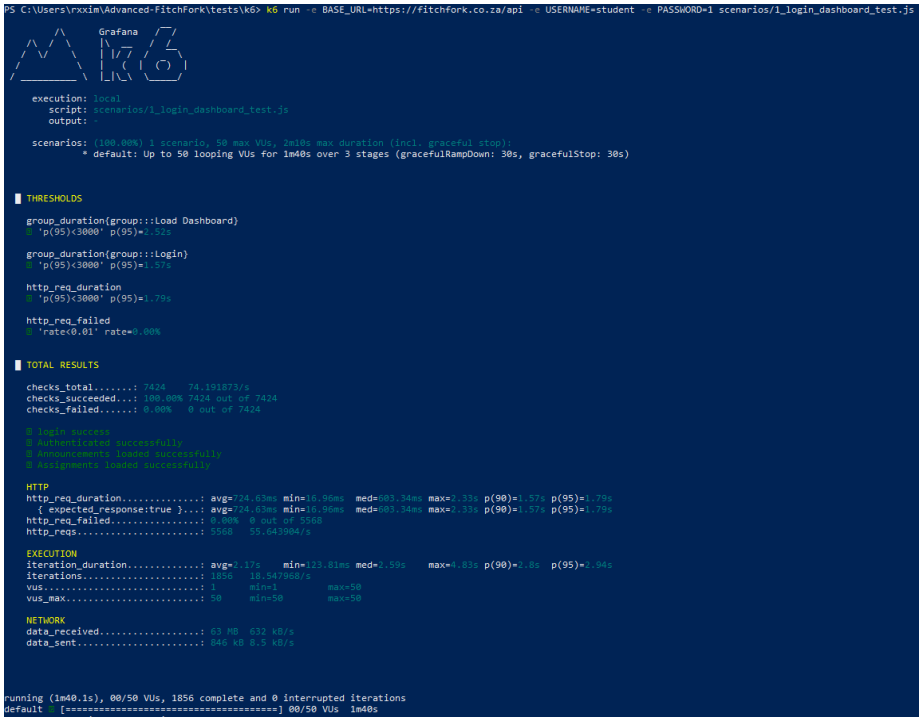
1. Performance Testing

1.1 Login and View Dashboard Test

This test simulates users logging into the system and then loading their dashboard. The test ramps up to 50 concurrent users over 30 seconds, maintains that load for 1 minute, then ramps down, while verifying that users can successfully authenticate and retrieve their announcements and assignments data. The test includes performance thresholds ensuring that HTTP request failure rates stay below 1% and that 95% of requests complete within 3 seconds.

Results:

- Overall result: All thresholds passed. Failure rate = 0.00% (target <1%), and p95 latencies stayed under the 3s threshold.
- Checks: 7,424 checks, 100% succeeded (0 failed).
- HTTP requests: 5,568 requests total (~55.6 req/s).
- Latency (HTTP requests):
    - p95 = 1.79 s
    - p90 = 1.57 s
    - median = 603 ms
    - average = 724 ms
    - max = 2.33 s
- Iteration (user scenario) duration: avg = 2.17 s, p95 = 2.94 s.
- Load: ramped to 50 VUs (max concurrent VUs = 50); run length ≈ 1m40s; 1,856 iterations completed.
- Network: ~63 MB received, 846 kB sent.
- Takeaway: the test passed with solid success rates and p95s well under the 3s target for HTTP requests.

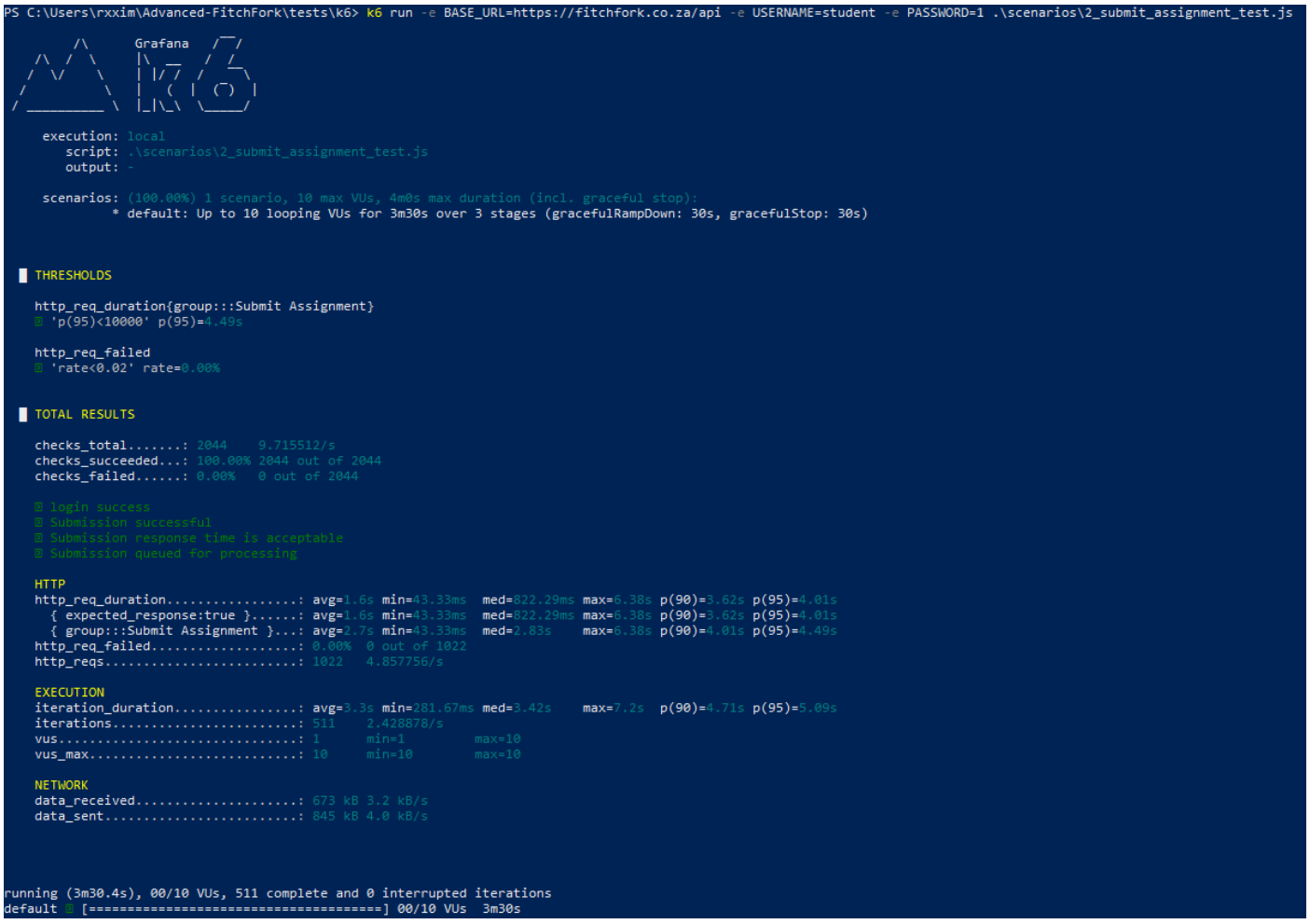


1.2 Submit Assignment Test

This test simulates students submitting assignment files to the system. The test runs with 10 concurrent users for 2 minutes, authenticating users and then submitting assignments in asynchronous mode to avoid waiting for the full grading process to complete. The test validates that submissions are successfully queued and complete within 10 seconds, with a slightly higher failure tolerance (2%) due to the complexity of file upload operations. The test makes about 500 submissions, simulating extremely high load.

Results:

- Overall result: All thresholds passed. Failure rate = 0.00% (target <2%), and p95 latencies stayed under the 10s threshold.
- Checks: 2,044 checks, 100% succeeded (0 failed).
- HTTP requests: 1,022 requests total (~4.86 req/s).
- Latency (HTTP requests):
  - p95 = 4.01 s
  - p90 = 3.62 s
  - median = 822 ms
  - average = 1.6 s
  - max = 6.38 s
- Iteration (user scenario) duration: avg = 3.3 s, p95 = 5.09 s.
- Load: steady 10 VUs (max concurrent VUs = 10); run length ≈ 3m30s; 511 iterations completed.
- Network: ~673 kB received, ~845 kB sent.
- Takeaway: the submission flow passed the configured thresholds (p95 < 10s, failure rate <2%).

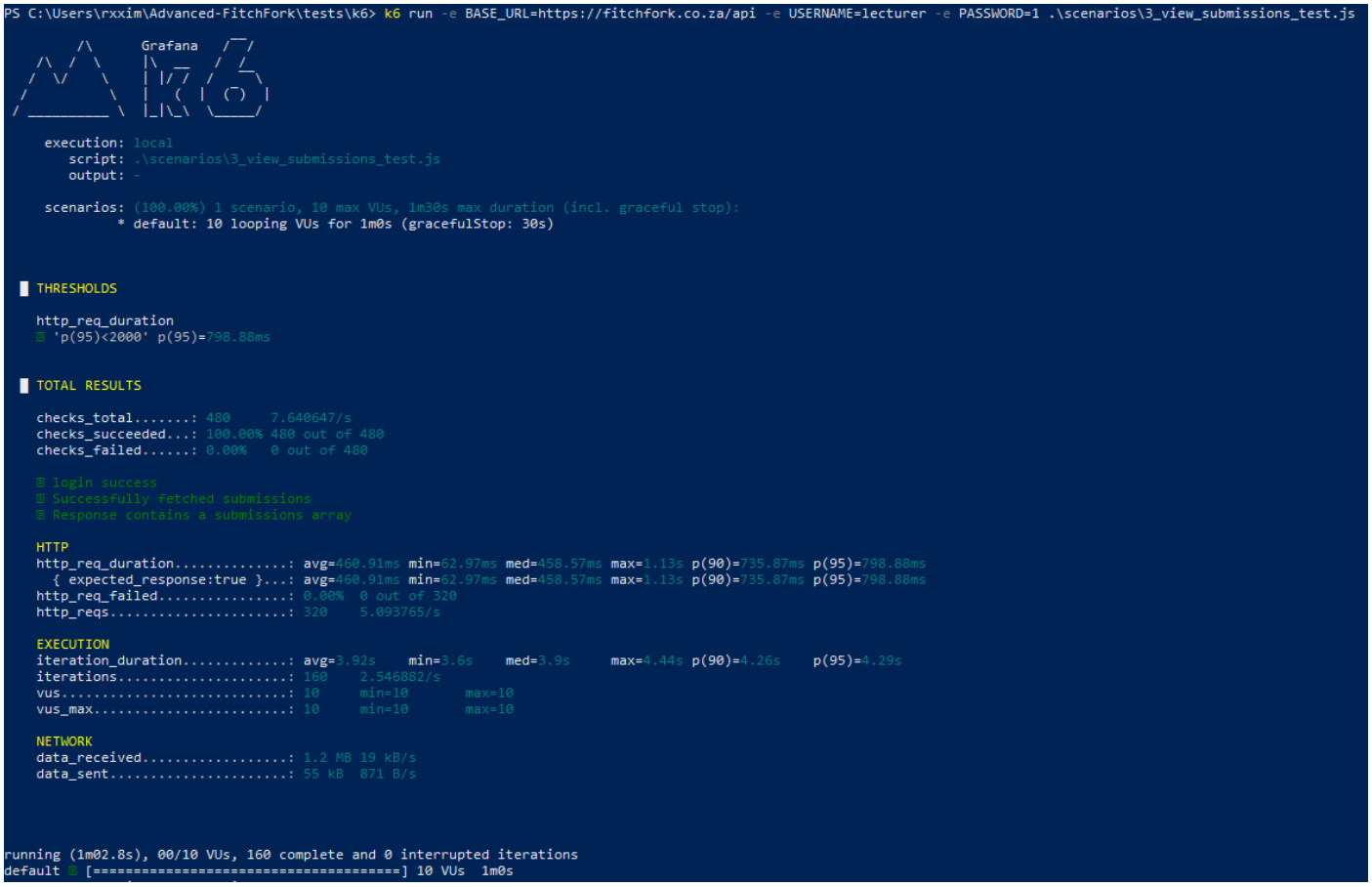


1.3 View Submissions Test

This test simulates lecturers viewing all submissions for a specific assignment in the system. The test runs 10 concurrent users for 1 minute, with each user authenticating as a lecturer and then fetching the submissions list for a given module and assignment. The test validates that the API returns a successful response and that the response contains the expected submissions data structure, with a 3-second delay between requests to simulate realistic user behaviour.

Result:

- Overall result: All thresholds passed. HTTP p95 = 798.88 ms (target <2s).
- Checks: 480 checks, 100% succeeded (0 failed).
- HTTP requests: 320 requests total (~5.09 req/s).
- Latency (HTTP requests):
    - p95 = 798.88 ms
    - p90 = 735.87 ms
    - median = 458.57 ms
    - average = 460.91 ms
    - max = 1.13 s
- Iteration (user scenario) duration: avg = 3.92 s, p95 = 4.29 s.
- Load: steady 10 VUs (max concurrent VUs = 10); run length ≈ 1m02s; 160 iterations completed.
- Network: ~1.2 MB received, ~55 kB sent.
- Takeaway: Viewing submissions is fast, HTTP p95 comfortably under the 2s target.

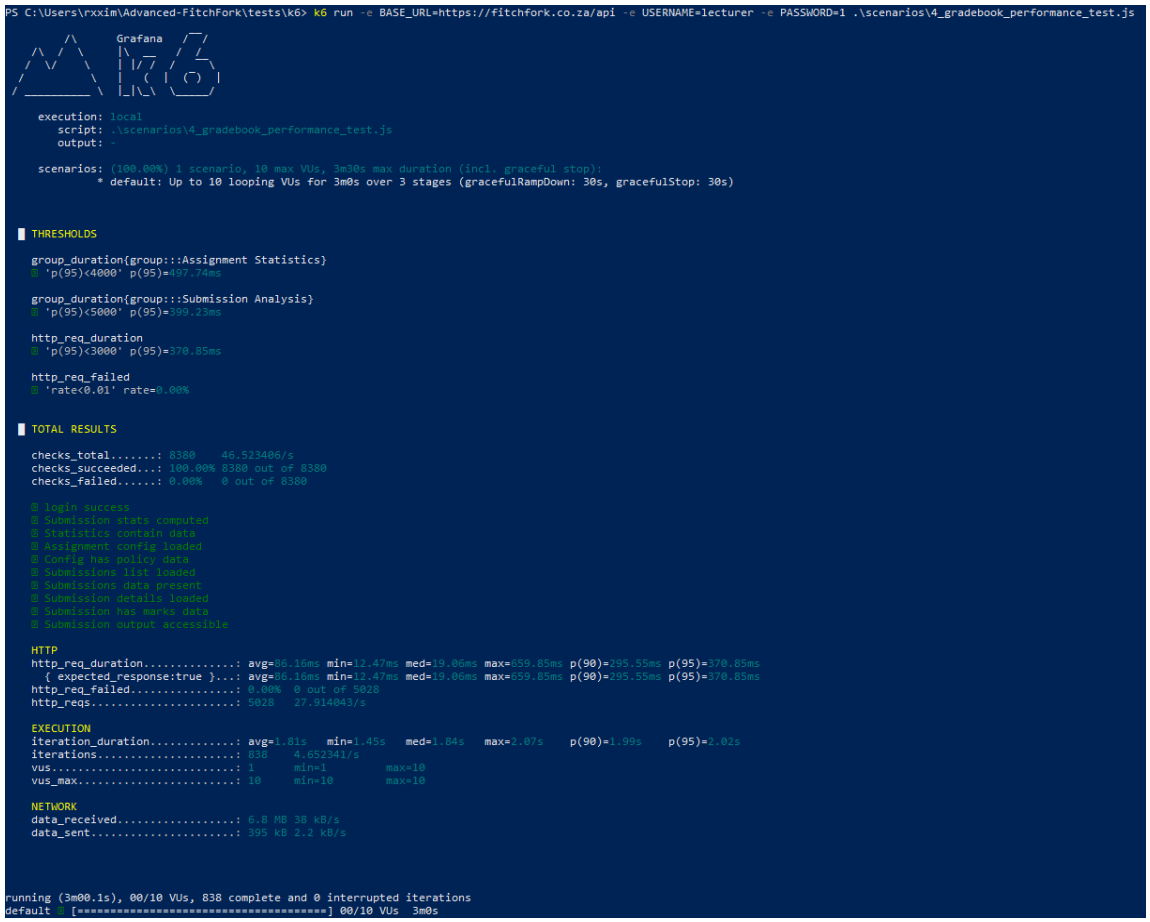


1.4 View Statistics Test

This test simulates lecturers accessing assignment statistics and analysing submissions in the system. The test runs 10 concurrent users for 2 minutes, performing two main groups of operations: fetching assignment statistics and configuration data, then loading and analysing individual submissions, including their details and output files. The test validates that all API endpoints respond successfully within specified time thresholds and that the returned data contains the expected statistical and submission information structures.

Results:

- Overall result: All thresholds passed. HTTP p(95) = 370.85 ms (target <3000ms).
- Checks: 8380 checks, 100% succeeded (0 failed).
- HTTP requests: 5028 requests total (~40.8 req/s).
- Latency (HTTP requests):
    - p(95) = 370.85 ms
    - p(90) = 255.55 ms
    - median = 19.06 ms
    - average = 46.16 ms
    - max = 659.85 ms
- Iteration (user scenario) duration: avg = 1.81 s, p(95) = 2.02 s.
- Load: steady 10 VUs (max concurrent VUs = 10); run length ≈ 2m; 838 iterations completed.
- Network: ~6.8 MB received, ~395 kB sent.
- Takeaway: The system performed very well under a load of 10 concurrent lecturers. The key performance indicator, the p(95) HTTP request duration, was only 370.85ms, which is significantly faster than the 3-second target.

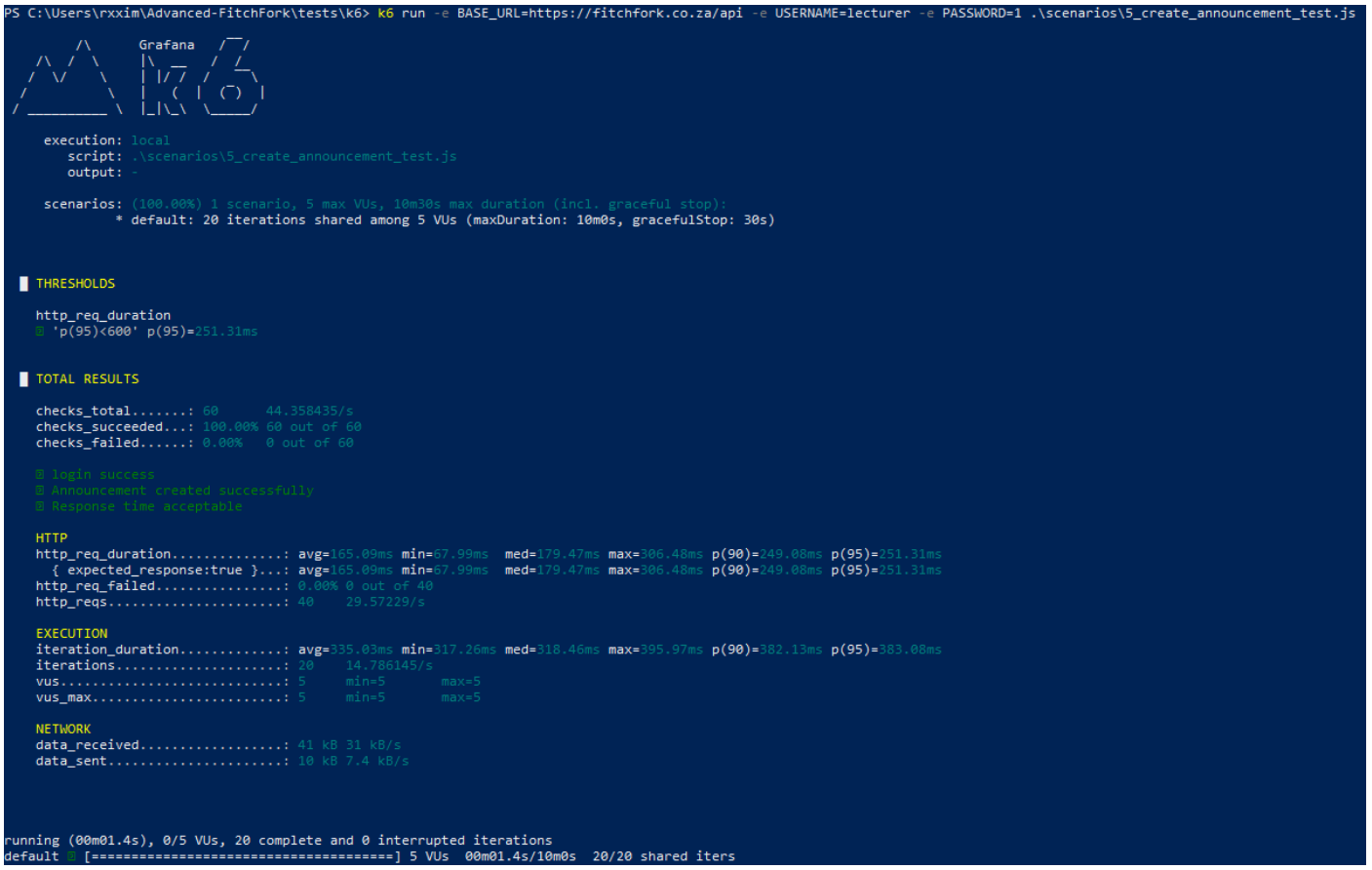


1.5 Create Announcement Test

This test simulates lecturers creating announcements in the system. The test uses 5 concurrent users to perform a total of 20 iterations, with each user authenticating as a lecturer and then creating a new announcement with a randomised title for a specific module. The test validates that announcements are successfully created and that the API responds within 600 milliseconds to ensure acceptable performance for content creation operations.

Results:

- Overall result: All thresholds passed. HTTP p(95) = 151.31 ms (target <600ms).
- Checks: 60 checks, 100% succeeded (0 failed).
- HTTP requests: 40 requests total (~9.75 req/s).
- Latency (HTTP requests):
  - p(95) = 151.31 ms
  - p(90) = 149.00 ms
  - median = 79.47 ms
  - average = 85.00 ms
  - max = 206.48 ms
- Iteration (user scenario) duration: avg = 205.06 ms, p(95) = 183.08 ms.
- Load: 5 VUs (max concurrent VUs = 5); run length ≈ 4s; 20 iterations completed.
- Network: ~41 kB received, ~10 kB sent.
- Takeaway: Creating announcements is very fast. The p(95) HTTP request duration of 151.31ms is well under the 600ms target, showing that the system can handle content creation operations from multiple users with excellent performance.



2. Availability Testing

2.1 Goals

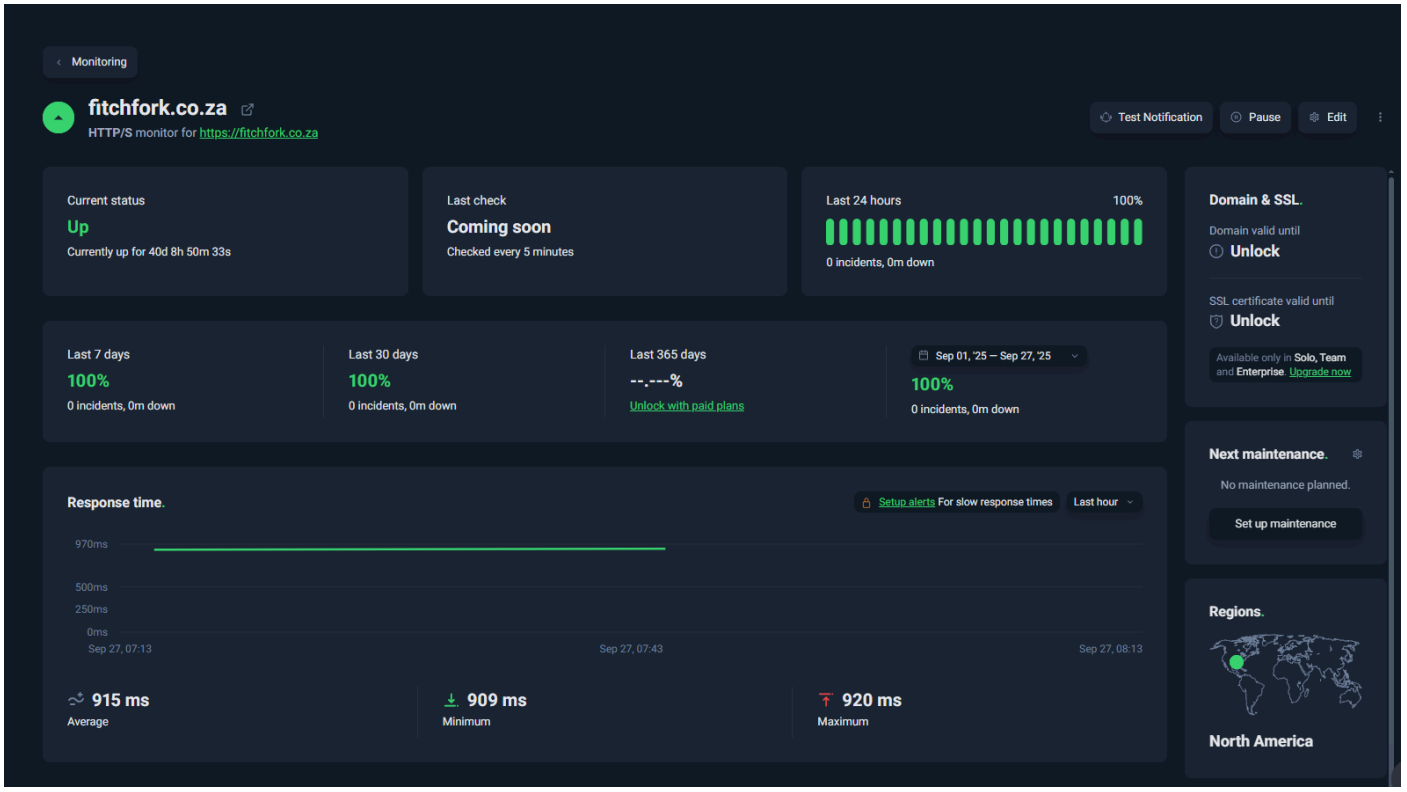
Uptime % >= 99.95%
Avg Response Time <= 50ms (Local)
Incidents per month <= 3

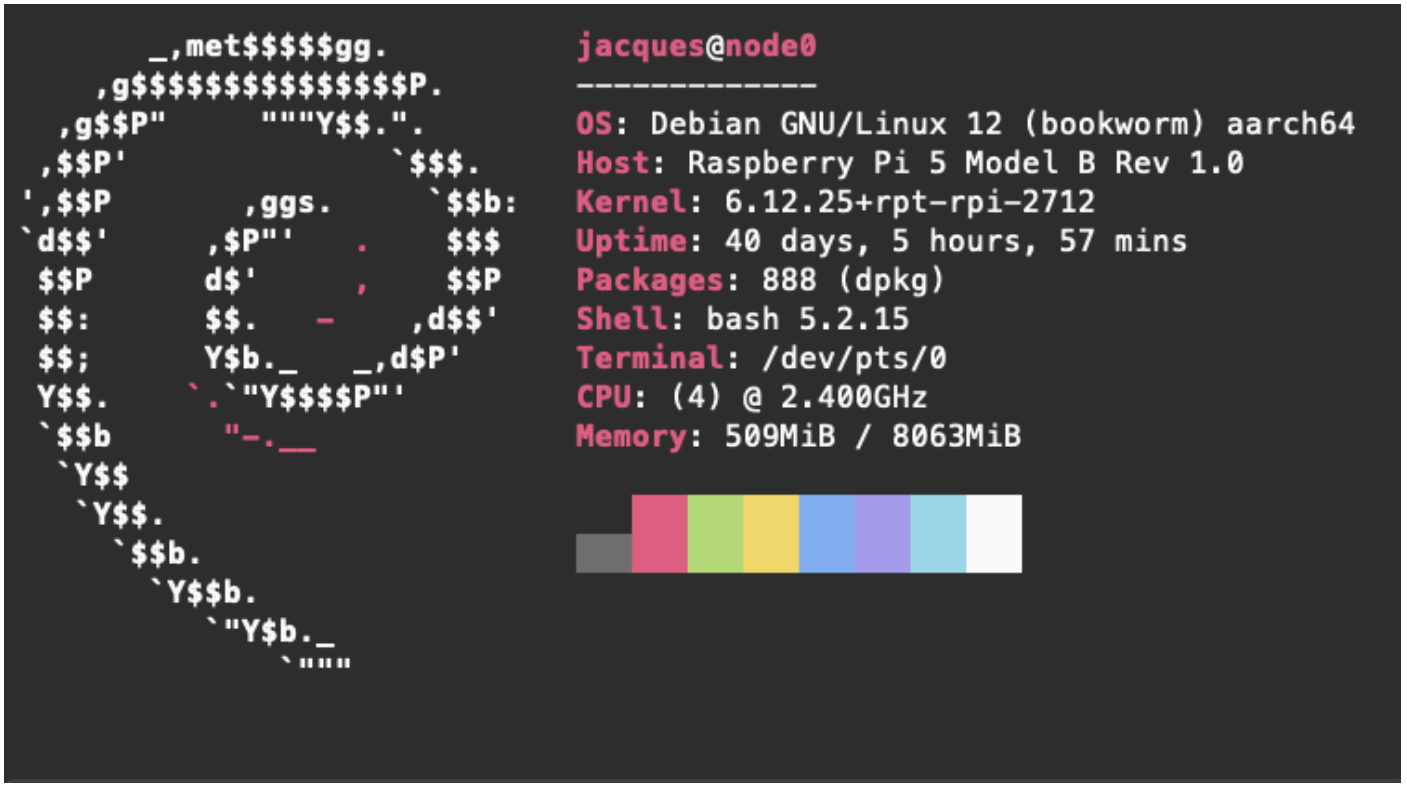
2.2 Achieved

Uptime % = 100%
Avg Response Time = 10ms (Local)
Incidents per month = 0

2.2 UptimeRobot

UptimeRobot was used to track the availability of our website.





```
~  (38.099s)
ping fitchfork.co.za
64 bytes from 172.67.161.103: icmp_seq=18 ttl=54 time=9.954 ms
64 bytes from 172.67.161.103: icmp_seq=19 ttl=54 time=13.985 ms
64 bytes from 172.67.161.103: icmp_seq=20 ttl=54 time=7.139 ms
64 bytes from 172.67.161.103: icmp_seq=21 ttl=54 time=8.301 ms
64 bytes from 172.67.161.103: icmp_seq=22 ttl=54 time=11.102 ms
64 bytes from 172.67.161.103: icmp_seq=23 ttl=54 time=6.603 ms
64 bytes from 172.67.161.103: icmp_seq=24 ttl=54 time=10.594 ms
64 bytes from 172.67.161.103: icmp_seq=25 ttl=54 time=10.520 ms
64 bytes from 172.67.161.103: icmp_seq=26 ttl=54 time=11.711 ms
64 bytes from 172.67.161.103: icmp_seq=27 ttl=54 time=9.901 ms
64 bytes from 172.67.161.103: icmp_seq=28 ttl=54 time=9.296 ms
64 bytes from 172.67.161.103: icmp_seq=29 ttl=54 time=10.124 ms
64 bytes from 172.67.161.103: icmp_seq=30 ttl=54 time=8.135 ms
64 bytes from 172.67.161.103: icmp_seq=31 ttl=54 time=11.548 ms
64 bytes from 172.67.161.103: icmp_seq=32 ttl=54 time=9.689 ms
64 bytes from 172.67.161.103: icmp_seq=33 ttl=54 time=9.330 ms
64 bytes from 172.67.161.103: icmp_seq=34 ttl=54 time=9.955 ms
64 bytes from 172.67.161.103: icmp_seq=35 ttl=54 time=12.130 ms
64 bytes from 172.67.161.103: icmp_seq=36 ttl=54 time=15.491 ms
64 bytes from 172.67.161.103: icmp_seq=37 ttl=54 time=9.374 ms
^C
--- fitchfork.co.za ping statistics ---
38 packets transmitted, 38 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 6.603/9.943/24.247/3.066 ms
```

3. Usability Testing

3.1 Journey: Submit an Assignment

Role: Student
Persona: A first-year computer science student, comfortable with technology but new to this platform.

Quantitative Metrics:

1. Task Success Rate:
    a. Task 1 (Log in): Yes
    b. Task 2 (Find module/assignment): Yes
    c. Task 3 (Upload file): Yes
    d. Task 4 (Confirm submission): Yes
    e. Task 5 (Log out): Yes
2. Time on Task:
    a. Task 1: 15 seconds
    b. Task 2: 45 seconds
    c. Task 3: 30 seconds
    d. Task 4: 10 seconds
    e. Task 5: 12 seconds
    f. Total Time: 1 minute, 52 seconds
3. Error Count: 1 (Initially clicked on "Grades" instead of the specific submission).

Feedback:

1. Overall, how would you rate the ease of use of this process on a scale of 1 to 5?
    - Rating: 4
2. What was the most frustrating part of this experience, if anything?
    - " I clicked 'Grades' first by mistake, which cost me a little time. It wasn't a big deal, though."
3. What did you like most about the process?
    - "The pop-up window for submitting was great. I didn't have to load a whole new page, and the drag-and-drop for the file worked perfectly. The big success message at the end was also very clear."
4. If you could change one thing, what would it be?
    - "I would make it clearer that you can click on a submission to view it."

3.2 Journey: Create and Publish an Announcement

Role: Lecturer
Persona: A lecturer who teaches multiple large classes, with the need to communicate with students efficiently.

Quantitative Metrics:

1. Task Success Rate:
    a. Task 1 (Log in): Yes
    b. Task 2 (Find module & announcements section): Yes
    c. Task 3 (Create the announcement): Yes
    d. Task 4 (Publish and verify): Yes
    e. Task 5 (Log out): Yes
2. Time on Task:
    a. Task 1: 12 seconds
    b. Task 2: 25 seconds
    c. Task 3: 1 minute, 10 seconds (mostly typing the message)
    d. Task 4: 15 seconds
    e. Task 5: 10 seconds
    f. Total Time: 2 minutes, 12 seconds
3. Error Count: 0

Feedback:

1. Overall, how would you rate the ease of use of this process on a scale of 1 to 5?
    - Rating: 5
2. What was the most frustrating part of this experience, if anything?
    - "Nothing."
3. What did you like most about the process?
    - "The interface was clean, fast, and logical. It did exactly what I needed it to do without getting in my way."
4. If you could change one thing, what would it be?
    - "Add two checkboxes on the creation page: 'Pin to top' and 'Send email notification to all students'. That would make it perfect for my needs."

3.3 Journey: Remark a Batch of Submissions

Role: Assistant Lecturer
Persona: An assistant lecturer who needs to remark 50 submissions for a single assignment.

Quantitative Metrics:

1. Task Success Rate:
    a. Task 1 (Log in & navigate to submissions): Yes
    b. Task 2 (Select all 50 submissions): Yes, with assistance
    c. Task 3 (Find and trigger the remark action): Yes
    d. Task 4 (Confirm and verify): Yes
2. Time on Task:
    a. Task 1: 40 seconds
    b. Task 2: 2 minutes, 30 seconds
    c. Task 3: 15 seconds
    d. Task 4: 20 seconds
    e. Total Time: 3 minutes, 45 seconds
3. Error Count: 1 (Initially tried to find a "remark" button on an individual submission before looking for a bulk action).

Feedback:

1. Overall, how would you rate the ease of use of this process on a scale of 1 to 5?
    - Rating: 4
2. What was the most frustrating part of this experience, if anything?
    - “The remark-all button was a bit hard to find, because it was hidden under the resubmit-all button."
3. What did you like most about the process?
    - "The remark-all feature made this process very easy."
4. If you could change one thing, what would it be?
    - "I would make the remark-all button easier to find."

3.4 Journey: Set up a New Assignment

Role: Lecturer
Persona: A new lecturer is setting up their first course on the platform. They are detail-oriented and want to ensure all assignment parameters (due dates, files, and grading criteria) are configured correctly.

Quantitative Metrics:

1. Task Success Rate:
    a. Task 1 (Navigate and start creation): Yes
    b. Task 2 (Complete initial configuration): Yes
    c. Task 3 (Upload files): Yes
    d. Task 4 (Set up tasks and marking): Yes, with assistance
    e. Task 5 (Review and publish): Yes
2. Time on Task:
    a. Task 1: 35 seconds
    b. Task 2: 3 minutes, 15 seconds
    c. Task 3: 1 minute, 20 seconds
    d. Task 4: 4 minutes, 50 seconds
    e. Task 5: 1 minute, 5 seconds
    f. Total Time: 10 minutes, 5 seconds
3. Error Count: 1 (Misunderstood one of the marking settings).

Feedback:

1. Overall, how would you rate the ease of use of this process on a scale of 1 to 5?
    - Rating: 4
2. What was the most frustrating part of this experience, if anything?
    - "I felt confused by the GATLAM settings. It seemed too complex and overwhelming for a new user to understand."
3. What did you like most about the process?
    - "The checklist on the side. It was my anchor throughout the entire setup. I always knew where I was, what I had completed, and what was next. The final summary page was also a huge confidence booster."
4. If you could change one thing, what would it be?
    - "I would make GATLAM more user-friendly or easier to understand."

3.5 Journey: Report a Technical Issue via a Ticket

Role: Student
Persona: A final-year student is working on her capstone project. She has encountered a bug that is blocking her from submitting her work. She is stressed and needs to report the issue clearly and track its status.

Quantitative Metrics:

1. Task Success Rate:
    a. Task 1 (Find the support/ticket system): Yes
    b. Task 2 (Create a new ticket): Yes
    c. Task 3 (Attach a file and submit): Yes
    d. Task 4 (Confirm submission and check status): Yes
2. Time on Task:
    a. Task 1: 28 seconds
    b. Task 2: 1 minute, 5 seconds (mostly typing)
    c. Task 3: 18 seconds
    d. Task 4: 25 seconds
    e. Total Time: 2 minutes, 16 seconds
3. Error Count: 0

Feedback:

1. Overall, how would you rate the ease of use of this process on a scale of 1 to 5?
    - Rating: 5
2. What was the most frustrating part of this experience, if anything?
    - "Honestly, nothing about the ticketing process itself was frustrating."
3. What did you like most about the process?
    - "The instant feedback. Seeing my ticket appear in a list with a status of 'Open' was a huge relief."
4. If you could change one thing, what would it be?
    - "When you create a ticket from an assignment page, it would be amazing if the form could automatically pre-fill the module and assignment name. I had to type that in myself, and it would save a little bit of time and effort."