

Performance Metrics and Load Testing

1.1. Pipeline Execution Statistics

The following table contains the timing results of the CI (Continuous Integration) and CD (Continuous Deployment) processes run in the GitHub Actions and AWS Elastic Beanstalk environments. These statistics were generated using the GitHub CLI, and because the least privilege settings specified in steps 79 and 80 of the first appendix caused many bad data points over 15 iterations, the data had to be cleaned. It was precisely during the analysis of this data set that it became apparent that the CI and CD branches were running in parallel, and the execution of the CD branch was not a prerequisite for the CI branch

CI run number	CI run time(sec)	Deploy run number	Deploy run time (sec)
1	16	1	26
2	29	2	27
3	12	3	22
4	15	4	33
5	18	5	43
6	14	6	22
7	16	7	23
8	19	8	31
9	14	9	93
10	17	10	50
11	14	11	59
12	19	12	60
13	29	13	73
14	22	14	59
15	20	15	140
16	22	16	148
17	20	17	144

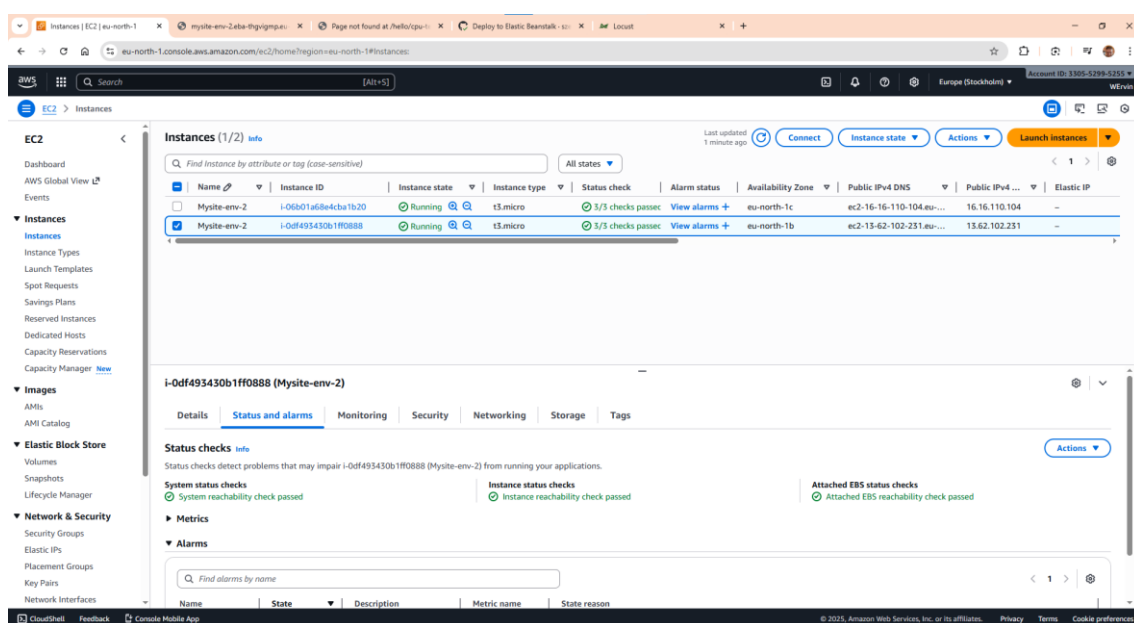
Source: GitHub Actions Workflow Run and AWS Elastic Beanstalk Logs (2025)

<https://github.com/szelese/ci-cd-gha-aws/actions>

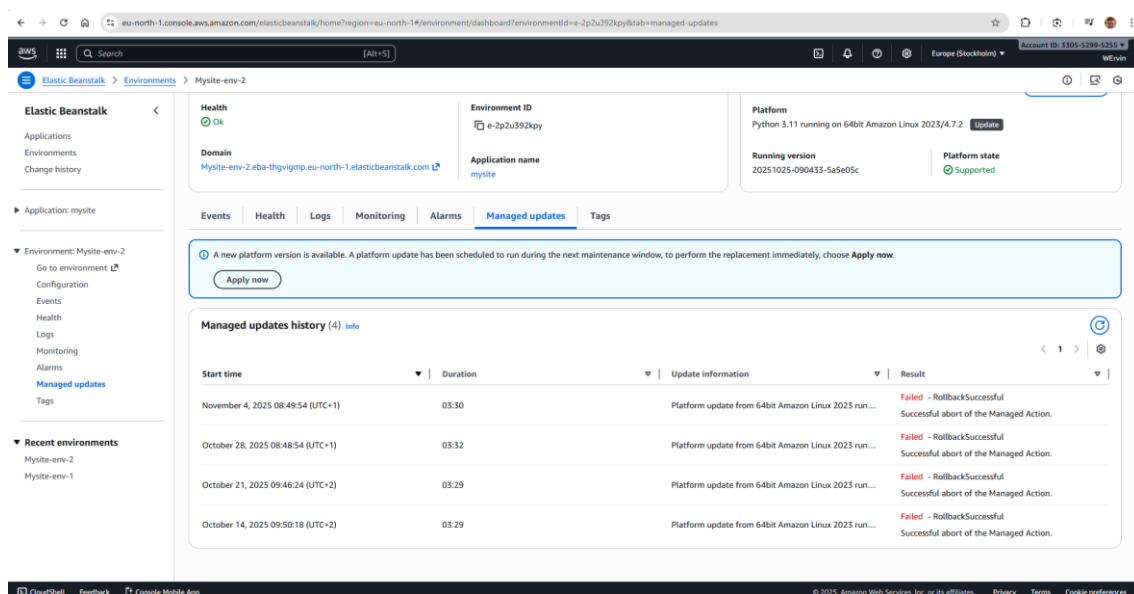
1.2. Demonstration of Automatic Rollback

Error handling is a key element of the CI/CD pipeline's reliability. To measure Mean Time to Recovery (MTTR) among the DORA metrics, a deliberately faulty configuration was uploaded to the Elastic Beanstalk environment to test the system's automatic self-correction capability. The goal was to verify the recovery time described in the thesis.

Error Diagnosis: The immediate effect of the faulty deployment attempt was visible at the EC2 instance level. The application could not start, so the instance became unreachable, as indicated by the "Instance reachability check failed" status.



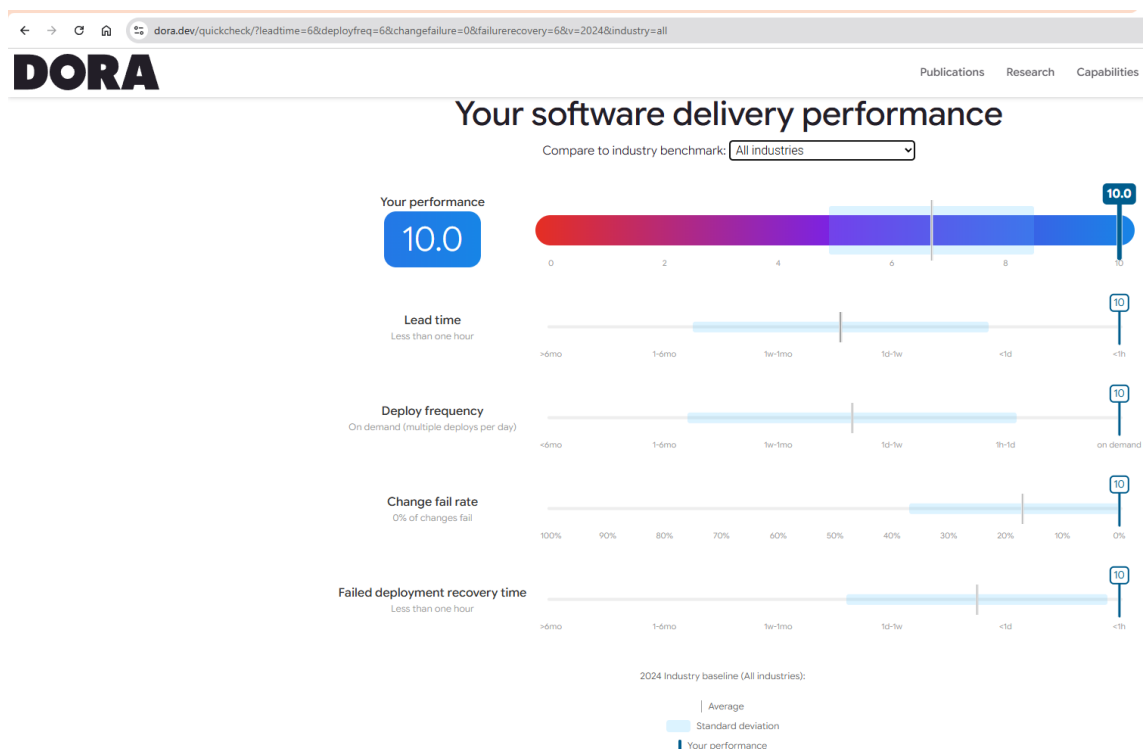
Automatic rollback:



Automatic Rollback: The Elastic Beanstalk management layer immediately detected that the EC2 instance running the new version was "ill" (did not pass the health check). As a result, it set the update status to "Failed" and automatically started a "RollbackSuccessful" process.

Result: As the data indicates, the system successfully rolled back to the previous, stably functioning application version in less than 4 minutes (durations of 03:30, 03:32, 3:29, 3:29). This practical test confirmed that the chosen PaaS solution's automatic error handling meets the "Elite level" MTTR requirement mentioned in the thesis.

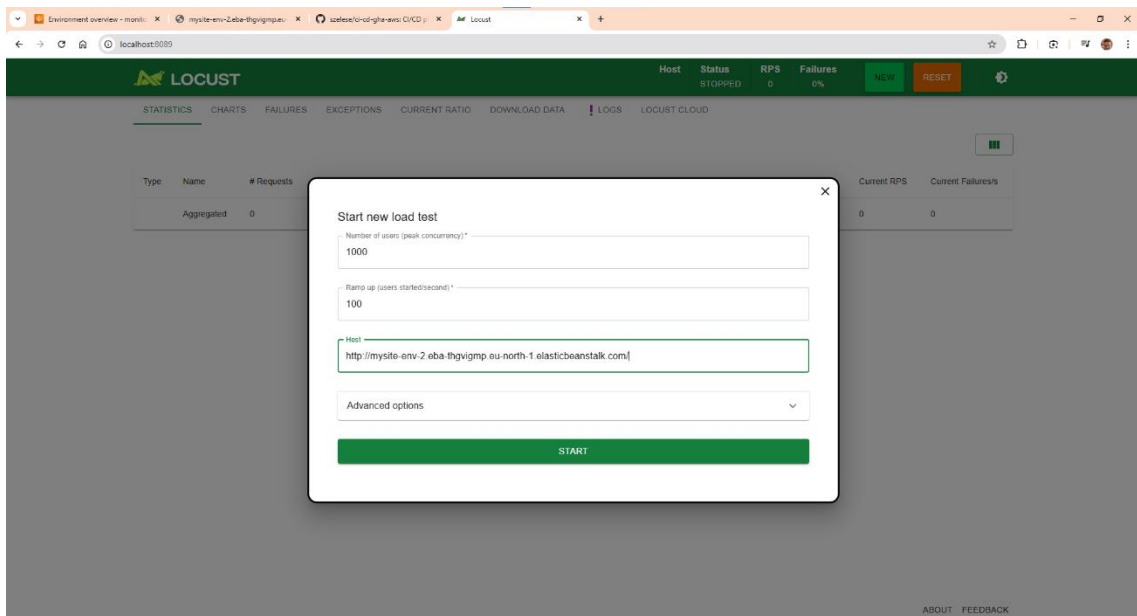
Dora Metrics:



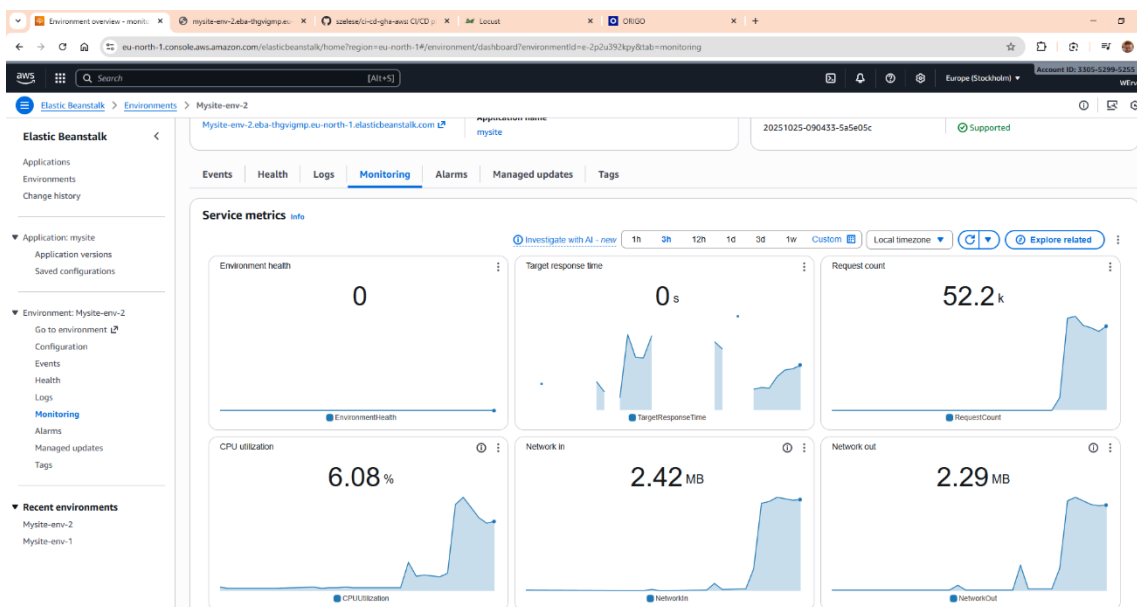
The measured data was entered into the <https://dora.dev/quickcheck/> site (Source: own measurement).

1.3. Load Testing (Locust)

The basic pipeline load was not sufficient to trigger scaling, so the pipeline was expanded with a view specifically targeting the CPU. After installing Locust, the main branch was targeted.



While the initial attempt was not fully successful, it was more effective than PowerShell testing.

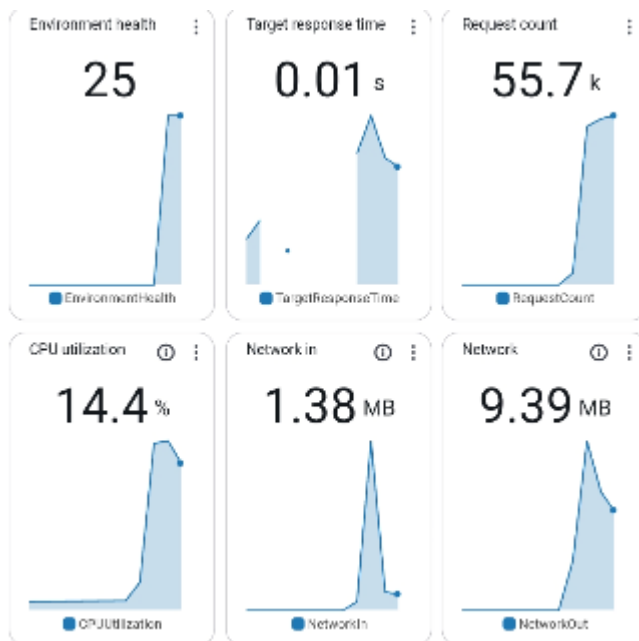
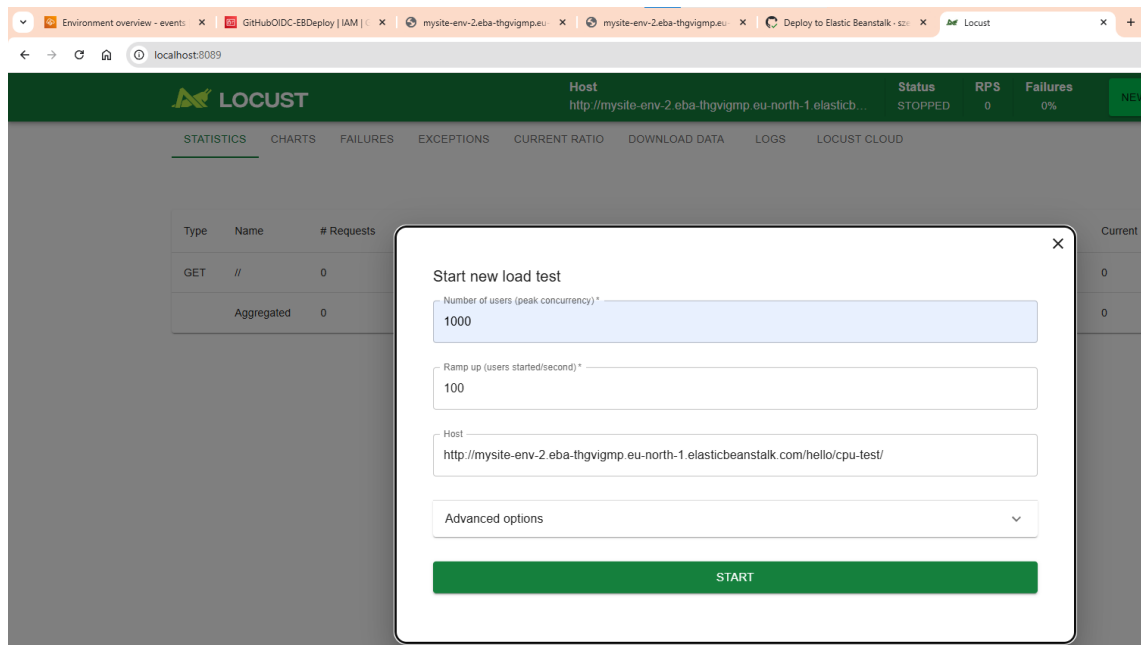


Diagnosis: The Locust test (showing a 4% error rate and increased response time) and the AWS metrics (low CPU utilization despite high network traffic) clearly indicated the problem. The "Hello World" application logic has minimal computational requirements, so it does not generate significant CPU load. The bottleneck in this case

was likely not the CPU, but presumably the limit of Gunicorn connections or the network I/O limit of the t3.micro instance. While the server reached saturation during the requests, the CPU remained underutilized.

Scaling Proof: Since reaching the auto-scaling threshold (60% CPU) was not possible with the simple logic, the basic view was expanded for testing purposes to prove the automatic scalability of the pipeline. A further expansion with a Hash test successfully overloaded the instances

<http://mysite-env-2.eu-north-1.elasticbeanstalk.com/hello/cpu-test/>



Outcome: New instances were created automatically during the stress test. After the load ended, the system automatically turned off the additional instances. This process proves the system's automatic horizontal scalability and self-healing capability.

The screenshot displays two panels from the AWS CloudWatch console. The top panel, titled "Overall health", provides a summary of system performance metrics. The bottom panel, titled "Enhanced instance health (2)", shows the health status of two specific EC2 instances, both of which are in a "Severe" state due to high HTTP 4xx error rates.

Overall health

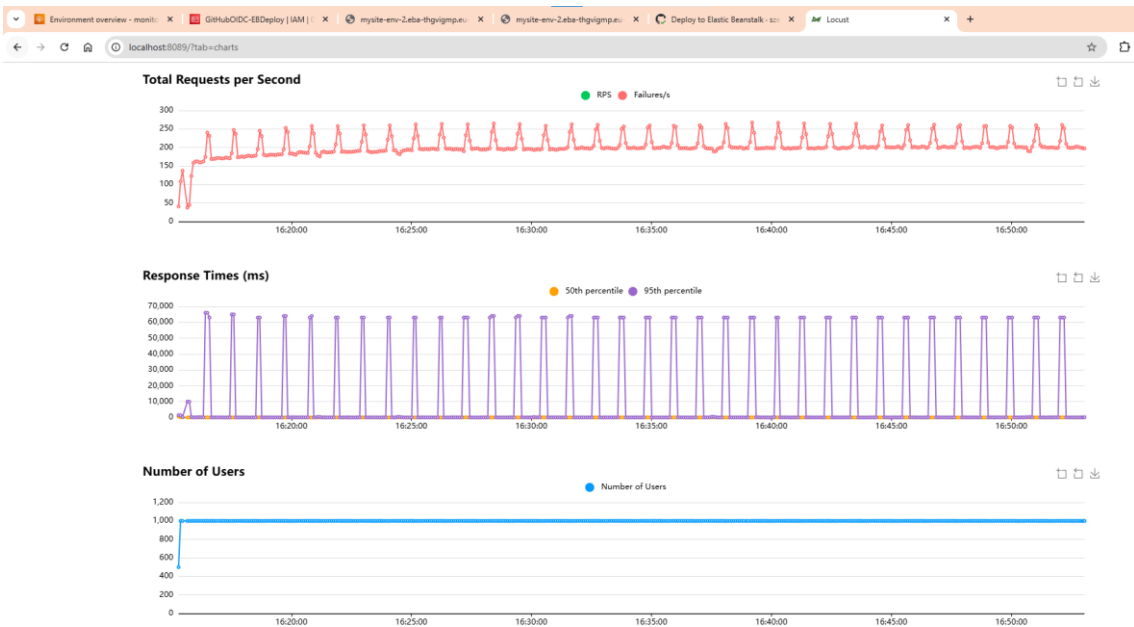
Requests / second	2XX responses	3XX responses	4XX responses
150.9	5	-	1504
P99 latency(ms)	P90 latency(ms)	P75 latency(ms)	P50 latency(ms)
395	120.5	11.5	7
5xx responses	-		
P10 latency(ms)	4.5		

Enhanced instance health (2)

Instance ID	Status	Running time
i-0938739428...	Severe <ul style="list-style-type: none">99.7 % of the requests are erroring with HTTP 4xx.	1 hour, 29 min...
i-06b01a68e4c...	Severe <ul style="list-style-type: none">99.6 % of the requests are erroring with HTTP 4xx.	37 days, 3 hou...

November 10, 2025 16:25:00 (UTC+1)	INFO	Added instance [i-0ecb21d1da95b542a] to your environment.
November 10, 2025 16:18:01 (UTC+1)	INFO	Added instance [i-00f024bbe73579e3e] to your environment.
November 10, 2025 16:16:01 (UTC+1)	WARN	Environment health has transitioned from Ok to Severe. 99.2% of the requests are erroring with HTTP 4xx.

Stress test execution in Locust:



After end of the test, the system regenerated:

The screenshot shows the AWS Elastic Beanstalk console interface. The main content area displays the 'Events (100)' page for the environment 'Mysite-env-2'. The events are listed in a table with columns for Time, Type, and Details. Two INFO events are highlighted with red lines, indicating the removal of instances from the environment.

Time	Type	Details
November 10, 2025 17:11:56 (UTC+1)	INFO	Environment health has transitioned from Degraded to Ok.
November 10, 2025 17:10:56 (UTC+1)	INFO	Removed instance [i-0938739428ee67ff2] from your environment.
November 10, 2025 17:09:56 (UTC+1)	WARN	Environment health has transitioned from Ok to Degraded. ELB processes are not healthy on 1 out of 3 instances. ELB health is failing or not available for 1 out of 3 instances.
November 10, 2025 17:05:57 (UTC+1)	INFO	Environment health has transitioned from Warning to Ok.
November 10, 2025 17:04:57 (UTC+1)	INFO	Removed instance [i-0ecb21d1da95b542a] from your environment.
November 10, 2025 17:03:57 (UTC+1)	WARN	Environment health has transitioned from Severe to Warning. ELB processes are not healthy on 1 out of 4 instances. No data received from 1 out of 4 instances. ELB health is failing or not available for 1 out of 4 instances.
November 10, 2025 16:25:00 (UTC+1)	INFO	Added instance [i-0ecb21d1da95b542a] to your environment.
November 10, 2025 16:18:01 (UTC+1)	INFO	Added instance [i-00f024bbe73579e3e] to your environment.
November 10, 2025 16:16:01 (UTC+1)	WARN	Environment health has transitioned from Ok to Severe. 99.2 % of the requests are erroring with HTTP 4xx.
November 10, 2025 15:34:05 (UTC+1)	INFO	Environment health has transitioned from Info to Ok.
November 10, 2025 15:32:11 (UTC+1)	INFO	Environment update completed successfully.
November 10, 2025 15:32:11 (UTC+1)	INFO	New application version was deployed to running EC2 instances.