

CLOUD COMPUTING LAB - 2

NAME : APHARNA KAMATH R

SRN : PES2UG23CS086

SECTION : B

Screenshot - 1

The screenshot shows a web browser window for the 'Fest Monolith' application at localhost:8000/events?user=PES2UG23CS086. The page title is 'Events'. At the top right, there are links for 'Events', 'My Events', 'Checkout', and 'Logout'. A banner at the top says 'Welcome PES2UG23CS086. Register for events below.' Below the banner, there is a grid of 12 event cards. Each card contains the event ID, name, price, a brief description, and a 'Register' button.

Event ID	Name	Price	Description	Action
1	Hackathon	₹ 500	Includes certificate • instant registration • limited seats	Register
2	Dance	₹ 300	Includes certificate • instant registration • limited seats	Register
3	Hackathon	₹ 500	Includes certificate • instant registration • limited seats	Register
4	Dance Battle	₹ 300	Includes certificate • instant registration • limited seats	Register
5	AI Workshop	₹ 400	Includes certificate • instant registration • limited seats	Register
6	Photography Walk	₹ 200	Includes certificate • instant registration • limited seats	Register
7	Gaming Tournament	₹ 350	Includes certificate • instant registration • limited seats	Register
8	Music Night	₹ 250	Includes certificate • instant registration • limited seats	Register
9	Treasure Hunt	₹ 150	Includes certificate • instant registration • limited seats	Register
10	Stand-up Comedy	₹ 300		
11	Robo Race	₹ 450		
12	Hackathon	₹ 500		

Screenshot - 2 (crash)

The screenshot shows a web browser window for the 'Fest Monolith' application at localhost:8000/checkout. The page title is 'Monolith Failure'. At the top right, there are links for 'Login' and 'Create Account'. Below the title, a message states 'One bug in one module impacted the entire application.' A red box highlights an 'Error Message' which reads 'division by zero'. To the left, a box explains why this happened: 'Because this is a monolithic application, all modules share the same runtime and deployment. When one feature crashes, it affects the whole system.' To the right, a box lists what should be done in the lab: 'Take a screenshot (crash demonstration)', 'Fix the bug in the indicated module', and 'Restart the server and verify recovery'. At the bottom, there are 'Back to Events' and 'Login' buttons.

Screenshot - 3 (crash fixed)

The screenshot shows a web browser window with the URL `localhost:8080/checkout`. The page is titled "Fest Monolith" and includes the subtext "FastAPI + SQLite + Locust". At the top right are "Login" and "Create Account" buttons. The main content area has a yellow header "Checkout" with the subtext "This route is used to demonstrate a monolith crash + optimization.". Below this is a box containing "Total Payable" and "₹ 6600". To the right is a section titled "What you should observe" with a bulleted list: "One buggy feature can crash the entire monolith.", "Inefficient loops cause high response times under load.", and "Optimization improves performance but architecture still scales as one unit." A note below this says "Next Lab: Split this monolith into Microservices (Events / Registration / Checkout)." At the bottom left is the footer "CC Week X - Monolithic Applications Lab".

Screenshot - 4

The screenshot shows a web browser window with the URL `localhost:8089`. The title bar says "LOCUST". The main interface has several tabs: "STATISTICS" (selected), "CHARTS", "FAILURES", "EXCEPTIONS", "CURRENT RATIO", "DOWNLOAD DATA", and "LOGS". The "STATISTICS" tab displays a table with the following data:

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/checkout	23	0	3	4	2.97	2	4	2797	0.6	0
	Aggregated	23	0	3	4	2.97	2	4	2797	0.6	0

Below the table is a "LOGS" section showing command-line logs from Locust, including errors about host specification and shutdown. The "LOGS" section also contains a table of response time percentiles and a command-line prompt.

```
[2026-01-29 14:40:12,124] apharna-ASUS/ERROR/locust.runners: You must specify the base host. Either in the host attribute in the User class, or on the command line using the --host option.
[2026-01-29 14:40:12,125] apharna-ASUS/WARNING/locust.runners: Stopping Locust...
[2026-01-29 14:40:57,455] apharna-ASUS/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 14:40:57,456] apharna-ASUS/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
[2026-01-29 14:43:13,275] apharna-ASUS/INFO/locust.main: Shutting down (exit code 0)
Type      Name           # reqs    # fails | Avg     Min     Max     Med | req/s   failures/s
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET      /checkout       23        0(0.00%) | 2        2        4        3 | 0.79    0.00
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated          23        0(0.00%) | 2        2        4        3 | 0.79    0.00
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Response time percentiles (approximated)
Type      Name           50%    66%    75%    80%    90%    95%    98%    99%    99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET      /checkout       3        3        3        3        4        4        4        4        4        4        4        23
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated          3        3        3        3        4        4        4        4        4        4        4        23
(.venv) apharna@apharna-ASUS:~/PES2UG23CS86/CC_Lab$
```

localhost:8089

Locust

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

Host http://localhost:8080 Status STOPPED RPS 0.6 Failures 0% NEW RESET

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/checkout	14	0	3	4	2.76	2	4	2797	0.6	0
Aggregated		14	0	3	4	2.76	2	4	2797	0.6	0

ABOUT

```
(.venv) apharna@pharna-ASUS:~/PES2UG23CS086/CC_Lab-$ locust -f locust/checkout_locustfile.py
[2026-01-29 14:45:17,221] apharna-ASUS/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 14:45:17,221] apharna-ASUS/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser.
[2026-01-29 14:45:36,651] apharna-ASUS/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 14:45:36,651] apharna-ASUS/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
[2026-01-29 14:45:48,080] apharna-ASUS/INFO/locust.runners: Ramping to 4 users at a rate of 1.00 per second
[2026-01-29 14:45:48,080] apharna-ASUS/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
KeyboardInterrupt
2026-01-29T09:17:51Z
[2026-01-29 14:47:51,903] apharna-ASUS/INFO/locust.main: Shutting down (exit code 0)
```

Type	Name	# reqs	# fails	Avg	Min	Max	Med	req/s	failures/s
GET	/checkout	14	0 (0.0%)	2	2	3	3	0.77	0.00
Aggregated		14	0 (0.0%)	2	2	3	3	0.77	0.00

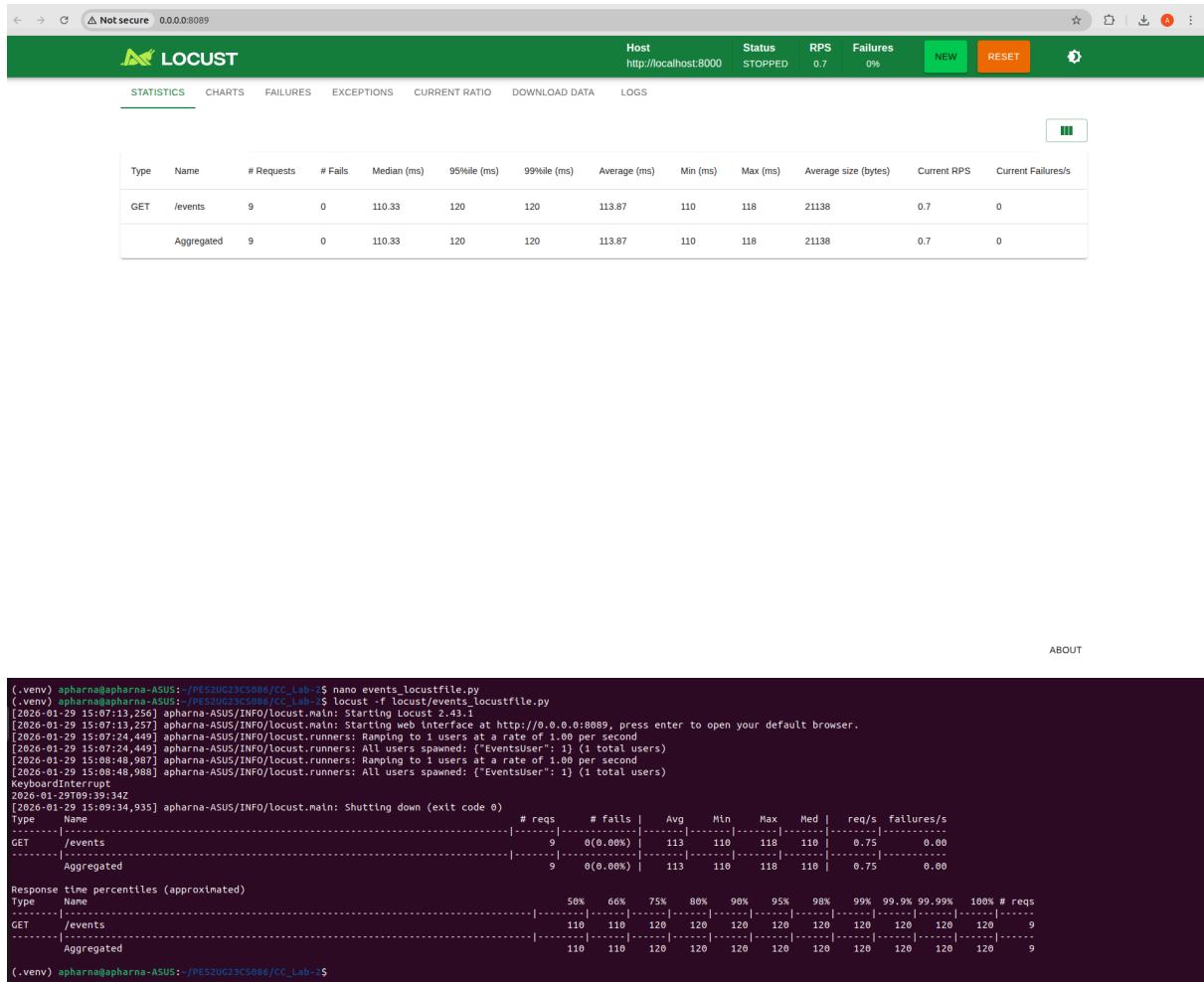
Response time percentiles (approximated)

Type	Name	50%	60%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%	# reqs
GET	/checkout	3	3	3	3	3	4	4	4	4	4	4	14
Aggregated		3	3	3	3	3	4	4	4	4	4	4	14

```
(.venv) apharna@pharna-ASUS:~/PES2UG23CS086/CC_Lab-$
```

Screenshot - 6 (before optimizing events)

Screenshot - 7 (after optimizing events)



Bottleneck:

The `/events` endpoint included inefficient processing that increased request handling time and reduced performance under concurrent load.

Change Made:

The route logic was optimized by removing unnecessary computation and simplifying request handling.

Why Performance Improved:

With reduced computational overhead, the server handled requests more efficiently, resulting in improved response times and more stable performance during load testing.

Screenshot - 8 (before optimizing my-events)

Screenshot - 9 (after optimizing my-events)

The screenshot shows the Locust web interface at <http://localhost:8089>. The top navigation bar includes links for STATISTICS, CHARTS, FAILURES, EXCEPTIONS, CURRENT RATIO, DOWNLOAD DATA, and LOGS. The STATISTICS tab is selected. The main content area displays a table of performance metrics:

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/my-events? user=locust_user	20	0	43	47	43.56	43	47	3144	0.6	0
	Aggregated	20	0	43	47	43.56	43	47	3144	0.6	0

Below the table, there is a terminal window showing the command-line output of the Locust run:

```
(.venv) apharna@apharna-ASUS:~/PES2UG23CS086/CC_Lab-2$ locust -f locust/myevents_locustfile.py
[2026-01-29 15:10:07,449] apharna-ASUS/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 15:10:02,449] apharna-ASUS/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser.
[2026-01-29 15:10:16,154] apharna-ASUS/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:10:16,154] apharna-ASUS/INFO/locust.runners: All users spawned: {"myEventsUser": 1} (1 total users)
[2026-01-29 15:12:21,388] apharna-ASUS/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:12:21,388] apharna-ASUS/INFO/locust.runners: All users spawned: {"myEventsUser": 1} (1 total users)
[2026-01-29 15:13:17,198] apharna-ASUS/INFO/locust.main: Shutting down (exit code 0)
Type      Name           # reqs | # fails | Avg | Min | Max | Med | req/s | failures/s
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET   /my-events?user=locust_user | 20 | 0(0.00%) | 43 | 42 | 47 | 43 | 0.69 | 0.00
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated | 20 | 0(0.00%) | 43 | 42 | 47 | 43 | 0.69 | 0.00
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Response time percentiles (approximated)
Type      Name           50% | 66% | 75% | 80% | 90% | 95% | 98% | 99% | 99.9% | 99.99% | 100% | # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET   /my-events?user=locust_user | 43 | 44 | 44 | 44 | 45 | 47 | 47 | 47 | 47 | 47 | 47 | 20
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated | 43 | 44 | 44 | 44 | 45 | 47 | 47 | 47 | 47 | 47 | 47 | 20
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
```

Bottleneck:

The /my-events route performed unnecessary processing for each request, resulting in additional overhead when handling concurrent users.

Change Made:

Redundant operations were removed and request handling was simplified to ensure that only required logic executes during each request.

Why Performance Improved:

By reducing unnecessary processing per request, CPU usage was lowered and the endpoint responded faster under load, improving overall throughput and response time.