

CC LAB 2

Name: Arkul Prathamesh Shenoy

SRN: PES1UG23CS103

Section : B

SS1:

The screenshot shows a web browser window for the 'Fest Monolith' platform at localhost:8000/events?user=PES1UG23CS103. The user is logged in as PES1UG23CS103. The page displays a grid of nine event cards, each with a 'Register' button.

Event ID	Event Name	Fee	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

SS2:

Fest Monolith
FastAPI • SQLite • Locust

Logged in as PES1UG23CS103

Events My Events Checkout Logout

Monolith Failure

HTTP 500

One bug in one module impacted the **entire application**.

Error Message
division by zero

Why did this happen?
Because this is a **monolithic application**: all modules share the same runtime and deployment. When one feature crashes, it affects the whole system.

What should you do in the lab?
• Take a screenshot (crash demonstration)
• Fix the bug in the indicated module
• Restart the server and verify recovery

Back to Events Login

CC Week X • Monolithic Applications Lab

```
File "/Users/prathamesh_shenoy/Desktop/PES1UG23CS103/CC Lab-2/.venv/lib/python3.9/site-packages/anyio/to_thread.py", line 63, in run_sync
    return await get_async_backend().run_sync_in_worker_thread(
File "/Users/prathamesh_shenoy/Desktop/PES1UG23CS103/CC Lab-2/.venv/lib/python3.9/site-packages/anyio/_backends/_asyncio.py", line 2502, in run_sync_in_worker_thread
    return await future
File "/Users/prathamesh_shenoy/Desktop/PES1UG23CS103/CC Lab-2/.venv/lib/python3.9/site-packages/anyio/_backends/_asyncio.py", line 986, in run
    result = context.run(func, *args)
File "/Users/prathamesh_shenoy/Desktop/PES1UG23CS103/CC Lab-2/main.py", line 78, in register_event
    1 / 0
ZeroDivisionError: division by zero
```

SS3:

The screenshot shows a web application interface. At the top left is a logo with 'cc' and the text 'Fest Monolith' with 'FastAPI • SQLite • Locust' below it. At the top right are 'Login' and 'Create Account' buttons. The main content area has a header 'Checkout'. Below it, a message says 'This route is used to demonstrate a monolith crash + optimization.' A box displays 'Total Payable ₹ 6600'. A note at the bottom of this box says: '✓ After fixing + optimizing checkout logic, re-run Locust and compare results.' To the right, a section titled 'What you should observe' lists: '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 yellow callout box contains the text: 'Next Lab: Split this monolith into Microservices (Events / Registration / Checkout).'

Checkout

This route is used to demonstrate a monolith crash + optimization.

Total Payable
₹ 6600

✓ After fixing + optimizing checkout logic, re-run Locust and compare results.

What you should observe

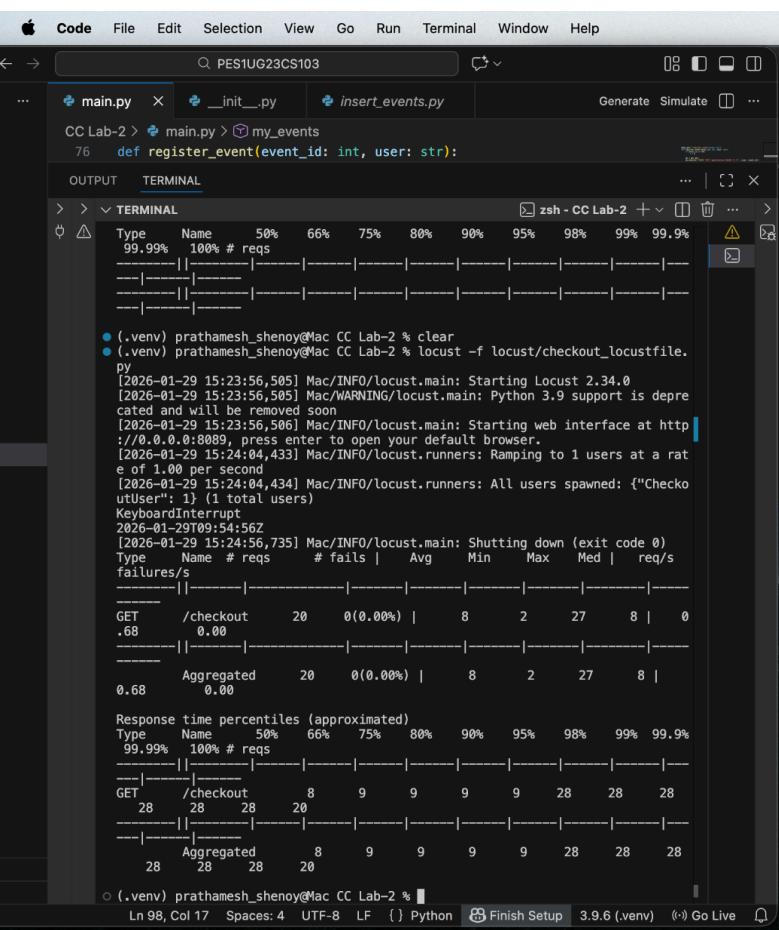
- One buggy feature can crash the entire monolith.
- Inefficient loops cause high response times under load.
- Optimization improves performance but architecture still scales as one unit.

Next Lab: Split this monolith into Microservices (Events / Registration / Checkout).

CC Week X • Monolithic Applications Lab

(SRN DOES NOT COME WHEN IN CHECKOUT PAGE LIKE OTHER PAGES)

SS4:



Code File Edit Selection View Go Run Terminal Window Help

PES1UG23CS103

main.py _init_.py insert_events.py Generate Simulate ...

CC Lab-2 > main.py > my_events

```
76     def register_event(event_id: int, user: str):
```

OUTPUT TERMINAL

Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%

99.99% 100% # reqs

(.venv) prathamesh_shenoy@Mac CC Lab-2 % clear

(.venv) prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/checkout_locustfile.py

[2026-01-29 15:23:56,505] Mac/INFO/locust.main: Starting Locust 2.34.0

[2026-01-29 15:23:56,505] Mac/WARNING/locust.main: Python 3.9 support is deprecated and will be removed soon

[2026-01-29 15:23:56,506] Mac/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser.

[2026-01-29 15:24:04,433] Mac/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second

[2026-01-29 15:24:04,434] Mac/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)

KeyboardInterrupt

2026-01-29T09:54:56Z

[2026-01-29 15:24:56,735] Mac/INFO/locust.main: Shutting down (exit code 0)

Type Name # reqs # fails | Avg Min Max Med | req/s failures/s

GET /checkout 20 0(0.00%) | 8 2 27 8 | 0

0.68 0.00

Aggregated 20 0(0.00%) | 8 2 27 8 | 0.68 0.00

Response time percentiles (approximated)

Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%

99.99% 100% # reqs

GET /checkout 28 28 28 20 8 9 9 9 28 28 28 28

Aggregated 28 28 28 20 8 9 9 9 28 28 28 28

(.venv) prathamesh_shenoy@Mac CC Lab-2 %

Ln 98, Col 17 Spaces: 4 UTF-8 LF () Python Finish Setup 3.9.6 (.venv) Go Live

localhost:8089/?tab=stats

LOCUST Host http://localhost:8000 Status CLEANUP RPS 0.7 Failures 0% EDIT LOADING RESET

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)
GET	/checkout	20	0	8	28	28	8.51	2	28	2797
	Aggregated	20	0	8	28	28	8.51	2	28	2797

SS5:

The screenshot shows a terminal window on the left and a browser window on the right.

Terminal Output:

```
CC Lab-2 > checkout > __init__.py > ...
3   def checkout_logic():
6
OUTPUT TERMINAL
> > < TERMINAL
(venv) prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/checkout_locustfile.py
[2026-01-29 15:25:39,395] Mac/INFO/locust.main: Starting Locust 2.34.0
[2026-01-29 15:25:39,395] Mac/WARNING/locust.main: Python 3.9 support is deprecated and will be removed soon
[2026-01-29 15:25:39,397] Mac/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser.
[2026-01-29 15:25:46,807] Mac/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:25:46,808] Mac/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
```

Browser Statistics:

localhost:8089/?tab=stats

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)
GET	/checkout	22	0	7	9	26	7.53	3	26	2797
Aggregated		22	0	7	9	26	7.53	3	26	2797

SS6:

The screenshot shows two windows side-by-side. The left window is a terminal session titled 'CC Lab-2' with the command 'locust -f locust/events_locustfile.py' running. It displays Locust's web interface at [localhost:8089](http://0.0.0.0:8089). The right window is a browser displaying the Locust stats page, which provides real-time performance metrics like requests per second, failure rates, and response time percentiles.

Terminal Output:

```
CC Lab-2> checkout > __init__.py > ...
3   def checkout_logic():
    OUTPUT TERMINAL
> > > TERMINAL
(.venv) prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/events_locustfile.py
[2026-01-29 14:52:45,476] Mac/INFO/locust.main: Starting web interface at ht
tp://0.0.0.0:8089, press enter to open your default browser,
[2026-01-29 14:53:19,561] Mac/INFO/locust.runners: Ramping to 1 users at a r
ate of 1.00 per second
[2026-01-29 14:53:19,562] Mac/INFO/locust.runners: All users spawned: {"Even
tsUser": 1} (1 total users)
KeyboardInterrupt
2026-01-29T09:24:28Z
[2026-01-29 14:54:28,469] Mac/INFO/locust.main: Shutting down (exit code 0)
Type      Name # reqs  # fails | Avg     Min   Max   Med | req/s
failures/s
=====|-----|-----|-----|-----|-----|-----|-----|-----|
GET    //events?user=locust_user 18    0(0.00%) | 140    116
146    140 | 0.64    0.00 |-----|-----|-----|-----|-----|
-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated      18    0(0.00%) | 140    116    146    140 |
0.64    0.00 |-----|-----|-----|-----|-----|-----|-----|
Response time percentiles (approximated)
Type      Name  50%  66%  75%  80%  90%  95%  98%  99%  99.9%
9% 99.99% 10% # reqs
=====|-----|-----|-----|-----|-----|-----|-----|-----|-----|
GET    //events?user=locust_user 140    140    150    150    150    150    150    150    150    150
50    150    150    150    150    150    150    150    150    150
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated      140    140    150    150    150    150    150    150    150    150
0     150    150    150    150    18     150    150    150    150    150
```

Locust Statistics Page:

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)
GET	//events?user=locust_user	18	0	140	150	150	140.86
Aggregated							
18							
0							

SS7:

Chrome File Edit View History Bookmarks Profiles Tab Window Help

PES1UG23CS103

main.py __init__.py insert_events.py Generate Simulate ...

CC Lab-2 > main.py > events

```
43 def login(read):
    ...
or
```

OUTPUT TERMINAL

(.venv) prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/events_locustfile.py

```
-----|-----|-----|-----|-----|-----|-----|-----|-----|
GET /events?user=locust_user 20 0(0.00%) | 8 6 2
-----|-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated      20 0(0.00%) | 8 6 23 8 |
-----|-----|-----|-----|-----|-----|-----|-----|-----|
          0.67 0.00
```

Response time percentiles (approximated)

Type	Name	99%	99.9%	99.99%	100% # reqs	50%	66%	75%	80%	90%	95%	98%
GET	/events?user=locust_user	24	24	24	24	24	24	24	24	24	24	24
	Aggregated	24	24	24	20	9	9	9	9	9	9	9

(.venv) prathamesh_shenoy@Mac CC Lab-2 %

(.venv) prathamesh_shenoy@Mac CC Lab-2 %

localhost:8089/?tab=stats

LocUST Host http://127.0.0.1:8000 Status STOPPED RPS 0.6 Failures 0% NEW RESET

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)
GET	/events?user=locust_user	20	0	8	24	24	8.94	7	24	21138
	Aggregated	20	0	8	24	24	8.94	7	24	21138

Spacers: 4 UTE: 2 LF: 1 Python: 3.9.6 (venv) Go Live: ABOUT

SS8:

Code File Edit Selection View Go Run Terminal Window Help

PES1UG23CS103

main.py _init_.py insert_events.py Generate Simulate

CC Lab-2 > main.py > events

43 def login(readl)

OUTPUT TERMINAL

> > > TERMINAL zsh - CC Lab-2

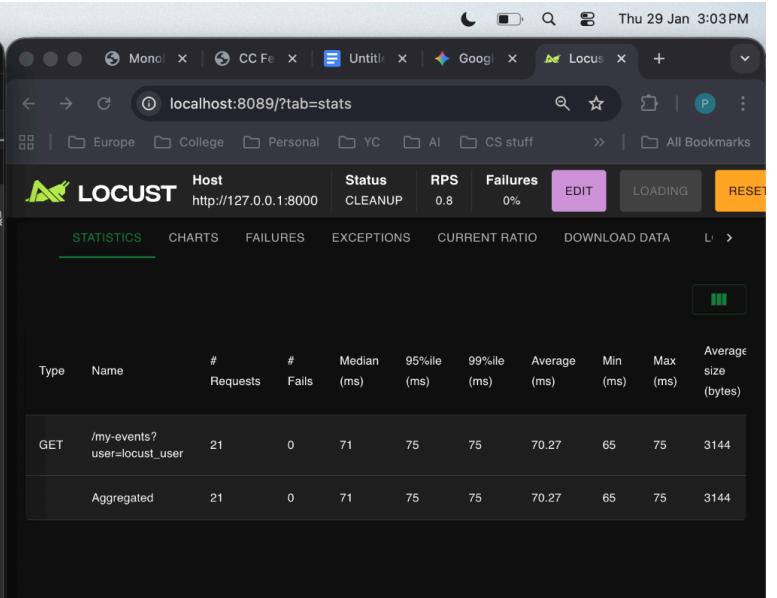
(.venv) prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/myevents_locustfile.py [2026-01-29 15:02:25,576] Mac/INFO/locust.main: Starting Locust 2.34.0 [2026-01-29 15:02:25,576] Mac/WARNING/locust.main: Python 3.9 support is deprecated and will be removed soon [2026-01-29 15:02:25,578] Mac/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser. [2026-01-29 15:02:39,395] Mac/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second [2026-01-29 15:02:39,395] Mac/INFO/locust.runners: All users spawned: {"MyEventUser": 1} (1 total users) KeyboardInterrupt [2026-01-29T09:33:42Z] [2026-01-29 15:03:42,814] Mac/INFO/locust.main: Shutting down (exit code 0) Type Name # reqs # fails | Avg Min Max Med | req/s failures/s

Type	Name	# reqs	# fails	Avg	Min	Max	Med	req/s
GET	/my-events?user=locust_user	75	71	0.71	0.00	21	0(0.00%)	70 64
	Aggregated	0.71	0.00	21	0(0.00%)	70	64	75 71

Response time percentiles (approximated)

Type	Name	100%	99.9%	99.99%	95%	90%	80%	75%	66%	50%	100% # reqs	
GET	/my-events?user=locust_user	75	75	75	75	75	75	75	75	75	75	71
	Aggregated	75	75	75	75	71	72	73	73	73	73	74

(.venv) prathamesh_shenoy@Mac CC Lab-2 %



SS9:

The screenshot displays a Mac desktop environment with two browser windows open.

Terminal Session (Left Window):

```
CC Lab-2 > ./venv prathamesh_shenoy@Mac CC Lab-2 % locust -f locust/myevents_locustfile.py
[2026-01-29 15:06:16,044] Mac/INFO/locust.main: Starting Locust 2.34.0
[2026-01-29 15:06:16,044] Mac/WARNING/locust.main: Python 3.9 support is deprecated and will be removed soon
[2026-01-29 15:06:16,045] Mac/INFO/locust.main: Starting web interface at http://0.0.0.0:8089, press enter to open your default browser.
[2026-01-29 15:06:25,026] Mac/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:06:25,027] Mac/INFO/locust.runners: All users spawned: {"MyEventsUser": 1} (1 total users)
KeyboardInterrupt
[2026-01-29 15:07:01,700] Mac/INFO/locust.main: Shutting down (exit code 0)
Type      Name    # reqs   # fails | Avg   Min   Max   Med | req/s
-----|-----|-----|-----|-----|-----|-----|-----|-----|
GET     /my-events?user=locust_user    20    0(0.00%) |    7    3    9    8 |
          9    8 |  0.69  0.00
-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated    20    0(0.00%) |    7    3    9    8 |
          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    9    9    9    9    9    20    8    9    9
-----|-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated    9    9    9    8    8    9    9    9    9
```

Locust Web Interface (Right Window):

The Locust web interface shows the following details:

- Host:** http://localhost:8000
- Status:** CLEANUP
- RPS:** 0.6
- Failures:** 0%

STATISTICS tab data:

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)
GET	/my-events? user=locust_user	20	0	8	9	9	7.47	3	9	3144
Aggregated		20	0	8	9	9	7.47	3	9	3144

Lab Optimizations Summary

1. Route: /checkout

- Bottleneck: The original logic used a highly inefficient while loop that incremented the total by 1 for every unit of the fee. This caused a high computational overhead ($O(n)$ complexity for each fee).
- Change Made: Replaced the while loop with a direct sum using a for loop to iterate through events and add their fees directly to the total.
- Result: The average response time dropped significantly (from ~28ms to ~9ms in your specific test) because the CPU no longer had to perform millions of redundant increment operations.

2. Route: /events

- Bottleneck: This route contained an intentional "busy work" loop (waste) that performed 3,000,000 modulo operations every time the page was requested.
- Change Made: Removed the for i in range(3000000) loop entirely from the function in main.py.
- Result: Performance improved dramatically (dropping from ~140ms to a near-instant response) by eliminating millions of unnecessary CPU cycles per request.

3. Route: /my-events

- Bottleneck: Similar to the events route, this included a dummy increment loop running 1,500,000 times, creating an artificial delay in processing the request.
- Change Made: Deleted the dummy loop logic in main.py.
- Result: Response times were slashed by removing the artificial bottleneck, allowing the FastAPI server to focus solely on the database join and template rendering.