

CC LAB 2

Name: Arkul Prathamesh Shenoy

SRN: PES1UG23CS103

Section : B

SS1:

The screenshot displays a web application interface for event registration. The browser address bar shows the URL `localhost:8000/events?user=PES1UG23CS103`. The application header includes a logo for "Fest Monolith" (FastAPI • SQLite • Locust), a user login status "Logged in as PES1UG23CS103", and navigation buttons for "Events", "My Events", "Checkout", and "Logout".

The main content area, titled "Events", welcomes the user "PES1UG23CS103" and lists nine events in a 3x3 grid. Each event card includes an event ID, a price tag, the event name, a description, and a "Register" button. A "View My Events" button is located in the top right corner of the event list.

| Event ID | Price (₹) | Event Name | Description |
|-------------|-----------|-------------------|---|
| Event ID: 1 | ₹ 500 | Hackathon | Includes certificate • instant registration • limited seats |
| Event ID: 2 | ₹ 300 | Dance | Includes certificate • instant registration • limited seats |
| Event ID: 3 | ₹ 500 | Hackathon | Includes certificate • instant registration • limited seats |
| Event ID: 4 | ₹ 300 | Dance Battle | Includes certificate • instant registration • limited seats |
| Event ID: 5 | ₹ 400 | AI Workshop | Includes certificate • instant registration • limited seats |
| Event ID: 6 | ₹ 200 | Photography Walk | Includes certificate • instant registration • limited seats |
| Event ID: 7 | ₹ 350 | Gaming Tournament | Includes certificate • instant registration • limited seats |
| Event ID: 8 | ₹ 250 | Music Night | Includes certificate • instant registration • limited seats |
| Event ID: 9 | ₹ 150 | Treasure Hunt | Includes certificate • instant registration • limited seats |

SS2:

🔥 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

```
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:

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).

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

SS4:

Code File Edit Selection View Go Run Terminal Window Help

PE\$1UG23CS103

main.py x _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

zsh - CC Lab-2

Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%
99.99% 100% # reqs
GET /checkout 20 0 8 28 28 8.51 2 28 2797
Aggregated 20 0 8 28 28 8.51 2 28 2797
Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%
99.99% 100% # reqs
GET /checkout 8 28 20 9 9 9 9 28 28 28
Aggregated 8 28 20 9 9 9 9 28 28 28

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

localhost:8089/?tab=stats

Monolith CC Lab-2 Local Google Prathal Cloud

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

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA

STATISTICS

| 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 |

ABOUT

SS5:

main.py__init__.pyinsert_events.py

CC Lab-2 > checkout > __init__.py > ...
3 def checkout_logic():
6

OUTPUTTERMINAL

o (.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)

ChromeFileEditViewHistoryBookmarksProfilesTabWindowHelp

PESTUG23CS103

EuropeCollegePersonalYCAI CS stuffAll Bookmarks

LOCUSTHosthttp://localhost:8000

StatusCLEANUPRPS0.7Failures0%EDITLOADINGRESET

STATISTICSCHARTSFAILURES EXCEPTIONSCURRENT RATIODOWNLOAD DATA

GET /checkout22079267.533262797

Aggregated22079267.533262797

Ln 16, Col 1Spaces: 4UTF-8LFPythonFinish Setup3.9.6 (.venv)Go LiveABOUT

SS6:

Code File Edit Selection View Go Run Terminal Window Help

main.py _init_.py

CC Lab-2 > checkout > _init_.py > ...

def checkout logic():

OUTPUT TERMINAL

zsh - CC Lab-2

```
(.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 http://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 rate 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
-----|-----|-----|-----|-----|-----|-----|
GET //events?user=locust_user 18 0(0.00%) | 140 116 146 140 |
146 140 | 0.64 0.00
-----|-----|-----|-----|-----|-----|
0.64 Aggregated 18 0(0.00%) | 140 116 146 140 |
0.00

Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%
9% 99.99% 100% # reqs
-----|-----|-----|-----|-----|-----|-----|
GET //events?user=locust_user 140 140 150 150 150 1
50 150 150 150 150 18
-----|-----|-----|-----|-----|-----|
0 Aggregated 140 140 150 150 150 150 15
0 150 150 18
```

localhost:8089/?tab=stats

LOCUST

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO

STATISTICS

| 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 | 140 | 150 | 150 | 140.86 |

SS7:

The image shows a development environment with a terminal and a web browser. The terminal on the left shows the execution of the Locust command-line interface, displaying performance metrics for a GET request to `/events?user=locust_user`. The web browser on the right shows the Locust web interface, which provides a visual overview of the test results, including a table of statistics.

Terminal Output:

```
(.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
3 0.67 0.00
-----
0.67 Aggregated 20 0(0.00%) | 8 6 23 8 |
0.00

Response time percentiles (approximated)
Type Name 99.9% 99.99% 100% # reqs 50% 66% 75% 80% 90% 95% 98%
GET /events?user=locust_user 24 24 24 24 24 20 9 9 9 9 24
24 24 24 24 20 9 9 9 9 24
-----
24 24 24 24 20 9 9 9 9 24
Aggregated 24 24 24 24 20 9 9 9 9 24

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

Locust Web Interface:

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 |

SS8:

CodeFileEditSelectionViewGoRunTerminalWindowHelp

main.pyinit.pyinsert_events.pyGenerateSimulate

CC Lab-2 > main.py > events

43 def login(requ... or

OUTPUTTERMINAL

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
-----|-----|-----|-----|-----|-----|-----|
GET /my-events?user=locust_user 21 0(0.00%) | 70 64 71 | 0.71
-----|-----|-----|-----|-----|-----|
Aggregated 21 0(0.00%) | 70 64 75 71 | 0.71

Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9%
-----|-----|-----|-----|-----|-----|-----|-----|
GET /my-events?user=locust_user 71 75 75 75 71 72 73 73 74
75 75 75 75 21
-----|-----|-----|-----|-----|-----|-----|
Aggregated 71 72 73 73 74 75 75 75
75 75 75 21
```

localhost:8089/?tab=stats

LOCUST Host http://127.0.0.1:8000

Status CLEANUP RPS 0.8 Failures 0%

EDIT LOADING RESET

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA L >

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 21 0 71 75 75 70.27 65 75 3144

Aggregated 21 0 71 75 75 70.27 65 75 3144

SS9:

The screenshot displays a development environment with a code editor on the left and a web browser on the right. The code editor shows a Python file named `main.py` with a `register_event` function that interacts with a database and returns a `RedirectResponse`. The terminal window shows the output of running `locust -f locust/myevents_locustfile.py`, including startup logs, a warning about Python 3.9 support, and a performance summary table.

The web browser shows the Locust web interface at `localhost:8089/?tab=stats`. The interface displays the status of the load test (CLEANUP), RPS (0.6), and Failures (0%). The **STATISTICS** tab is active, showing a table of request metrics.

| 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.