

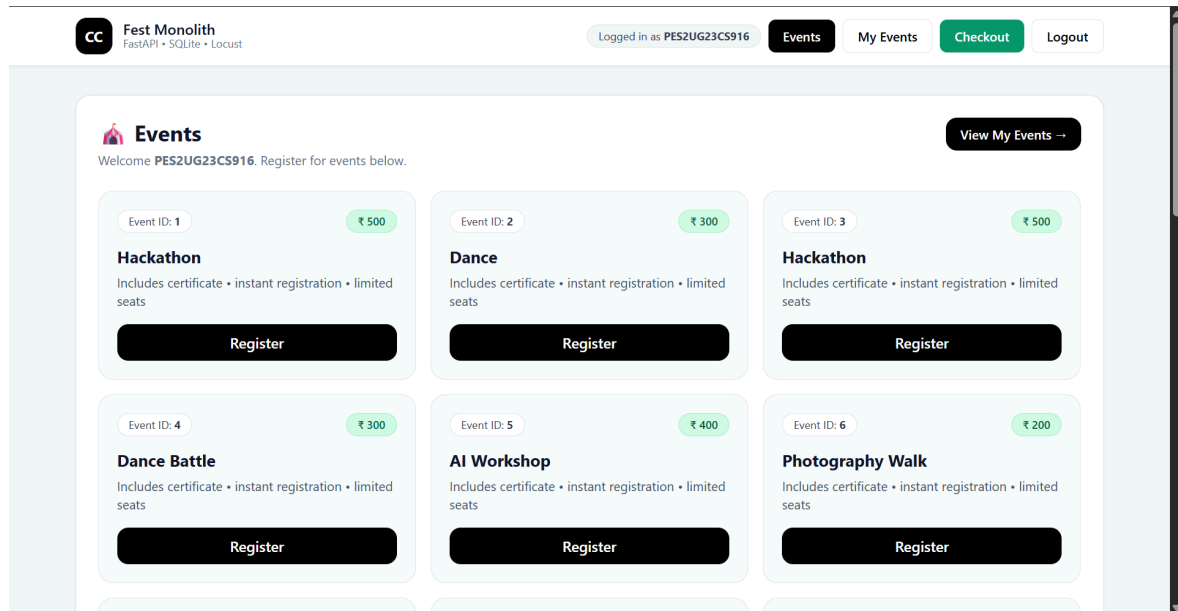
Cloud Computing Lab-2 Monolithic Architecture

Name: Piniseti Sudhiksha

SRN: PES2UG23CS916

Section: 6F

Screenshot 1




Screenshot 2

CC

Fest Monolith
FastAPI • SQLite • Locust

LoginCreate Account

**Monolith Failure**

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

HTTP 500

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 EventsLogin


CC Week X • Monolithic Applications Lab

Screenshot 3

CC

Fest Monolith
FastAPI • SQLite • Locust

LoginCreate Account

**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

Screenshot 4

main.py _init_.py checkout_locustfile.py

main.py PES2UG23CS916 > CC_LAB2 > checkout > _init_.py > checkout_logic

def checkout_logic():
db = get_db()
db.row_factory = None
events = db.execute("SELECT fee FROM events").fetchall()
Uncomment this line initially for the crash screenshot task
lol / 0
total = 0
for e in events:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Aggregated
0(0.00%) | 115 4 2039 9 | 0.66 0.00
Response time percentiles (approximated)
Type Name
50% 60% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100% # reqs
GET /checkout
9 9 10 10 17 2000 2000 2000 2000 2000 2000 19
Aggregated
9 9 10 10 17 2000 2000 2000 2000 2000 2000 19
(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2 >

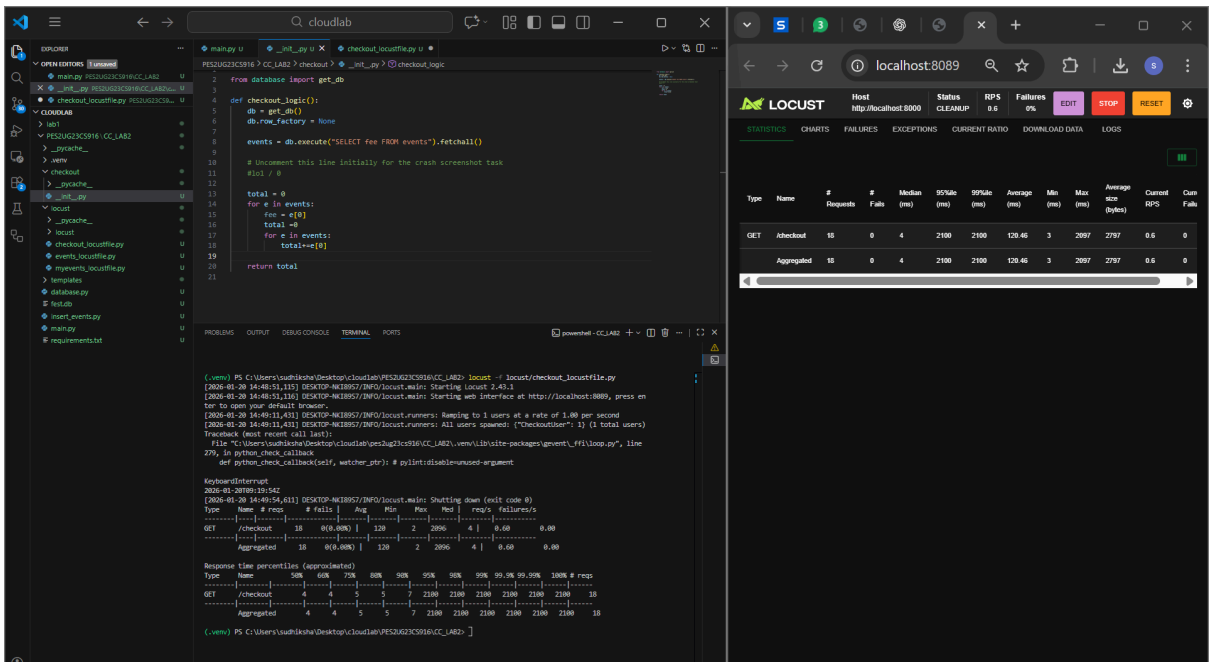
LOCUST

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

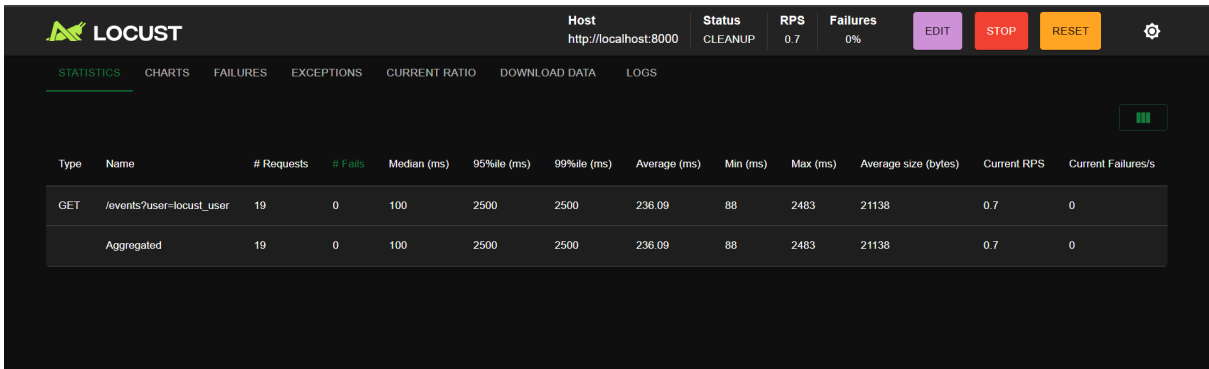
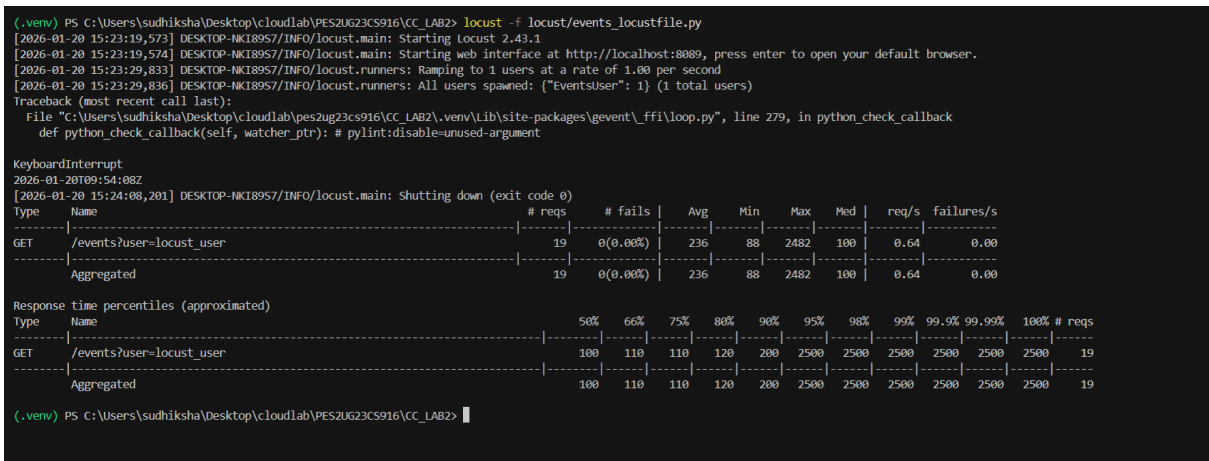
STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average Size (bytes)	Current RPS	Cum Fails
GET	/checkout	19	0	9	2000	2000	115.2	4	2039	2797	0.7	0
Aggregated		19	0	9	2000	2000	115.2	4	2039	2797	0.7	0

Screenshot 5



Screenshot 6




Screenshot 7

```
(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2> locust -f locust/events_locustfile.py
File "C:\Users\sudhiksha\Desktop\cloudlab\pes2ug23cs916\CC_LAB2\.venv\Lib\site-packages\gevent\_ffi\loop.py", line 279, in python_check_callback
def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument

KeyboardInterrupt
2026-01-20T09:56:35Z
[2026-01-20 15:26:35,688] DESKTOP-NKI8957/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                    17          0(0.00%)    149       8      2382      10        0.59        0.00
-----
Aggregated                                     17          0(0.00%)    149       8      2382      10        0.59        0.00
-----

Response time percentiles (approximated)
Type      Name                                     50%      66%      75%      80%      90%      95%      98%      99%      99.9%  99.99%  100% # reqs
-----
GET      /events?user=locust_user                    10       11       12       12       19      2400      2400      2400      2400      2400      2400  17
-----
Aggregated                                     10       11       12       12       19      2400      2400      2400      2400      2400      2400  17
-----

(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2> █
```



Host

http://localhost:8000

Status

CLEANUP

RPS

0.7


Failures

0%

EDIT

STOP

RESET



STATISTICS

CHARTS

FAILURES

EXCEPTIONS

CURRENT RATIO

DOWNLOAD DATA

LOGS

██

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/events?user=locust_user	17	0	10	2400	2400	149.89	8	2383	21138	0.7	0
	Aggregated	17	0	10	2400	2400	149.89	8	2383	21138	0.7	0


Screenshot 8

```
(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2> locust -f locust/myevents_locustfile.py
[2026-01-20 15:28:21,192] DESKTOP-NKI8957/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 15:28:21,192] DESKTOP-NKI8957/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-20 15:28:31,496] DESKTOP-NKI8957/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 15:28:31,527] DESKTOP-NKI8957/INFO/locust.runners: All users spawned: {"MyEventsUser": 1} (1 total users)
Traceback (most recent call last):
File "C:\Users\sudhiksha\Desktop\cloudlab\pes2ug23cs916\CC_LAB2\.venv\Lib\site-packages\gevent\_ffi\loop.py", line 279, in python_check_callback
def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument

KeyboardInterrupt
2026-01-20T09:59:08Z
[2026-01-20 15:29:08,526] DESKTOP-NKI8957/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                  18          0(0.00%)    266      117     2279      140        0.63        0.00
-----
Aggregated                                     18          0(0.00%)    266      117     2279      140        0.63        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                  140      160      160      160      200     2300     2300     2300     2300     2300     2300  18
-----
Aggregated                                     140      160      160      160      200     2300     2300     2300     2300     2300     2300  18
-----

(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2> █
```



Host

http://localhost:8000

Status

CLEANUP

RPS

0.6


Failures

0%

EDIT

STOP

RESET



STATISTICS

CHARTS

FAILURES

EXCEPTIONS

CURRENT RATIO

DOWNLOAD DATA

LOGS

██

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/my-events?user=locust_user	18	0	140	2300	2300	266.63	117	2279	3144	0.6	0
	Aggregated	18	0	140	2300	2300	266.63	117	2279	3144	0.6	0

Screenshot 9


```
(.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2> locust -f locust/myevents_locustfile.py
[2026-01-20 15:30:56,830] DESKTOP-NKI8957/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 15:30:56,831] DESKTOP-NKI8957/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-20 15:31:07,379] DESKTOP-NKI8957/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 15:31:07,385] DESKTOP-NKI8957/INFO/locust.runners: All users spawned: {"MyEventsUser": 1} (1 total users)
* Traceback (most recent call last):
  File "C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2\.venv\Lib\site-packages\gevent\_ffi\loop.py", line 279, in python_check_callback
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument

KeyboardInterrupt
2026-01-20T10:02:03Z
[2026-01-20 15:32:03,019] DESKTOP-NKI8957/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%) 112     3    2070     8     0.68     0.00
-----
Aggregated                               20      0(0.00%) 112     3    2070     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      /my-events?user=locust_user             8      9      9     12     44    2100  2100  2100  2100  2100  2100  20
-----
Aggregated                               8      9      9     12     44    2100  2100  2100  2100  2100  2100  20

% (.venv) PS C:\Users\sudhiksha\Desktop\cloudlab\PES2UG23CS916\CC_LAB2>
```

 LOCUST

Host

http://localhost:8000

Status

CLEANUP

RPS

0.7


Failures

0%

EDIT

STOP

RESET



STATISTICS

CHARTS

FAILURES

EXCEPTIONS

CURRENT RATIO

DOWNLOAD DATA

LOGS

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/my-events?user=locust_user	20	0	8	2100	2100	112.5	3	2071	3144	0.7	0
	Aggregated	20	0	8	2100	2100	112.5	3	2071	3144	0.7	0

Optimisations:

For events:

1. What was the bottleneck?

The route had an unnecessary CPU-intensive loop that ran millions of iterations for every request, even though it did not affect the response.

2. What change did you make?

Removed the wasteful loop:

```
for i in range(3000000):
    waste += i % 3
```

3. Why did the performance improve?

The server no longer wastes CPU cycles on meaningless computation, so requests are processed faster, reducing average response time.

For my-events:

1. What was the bottleneck?

The route contained a dummy loop with 1.5 million iterations, which added artificial delay to every request.

2. What change did you make?

Removed the dummy loop:

```
for _ in range(1500000):  
    dummy += 1
```

3. Why did the performance improve?

By eliminating unnecessary processing, the route now returns results immediately after the database query, lowering response time and improving throughput.