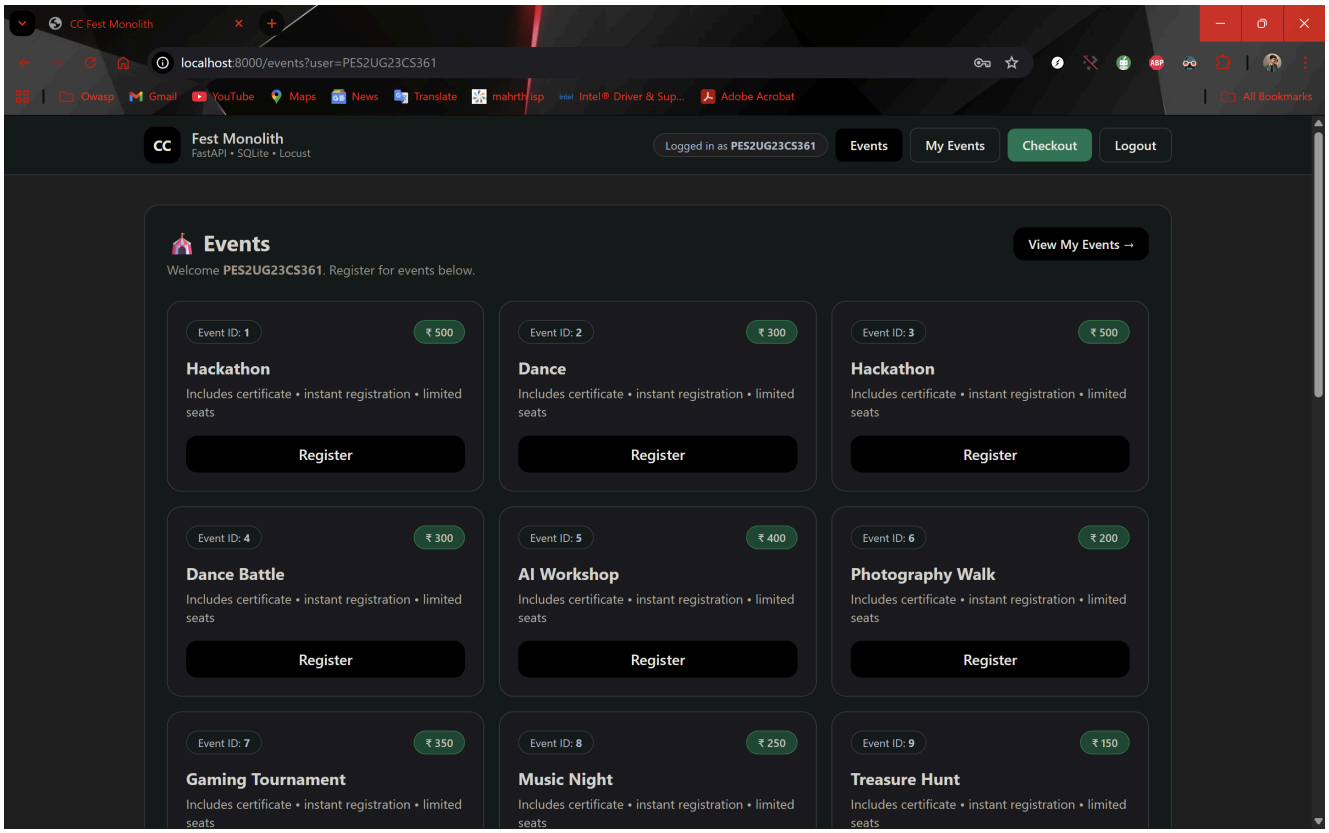
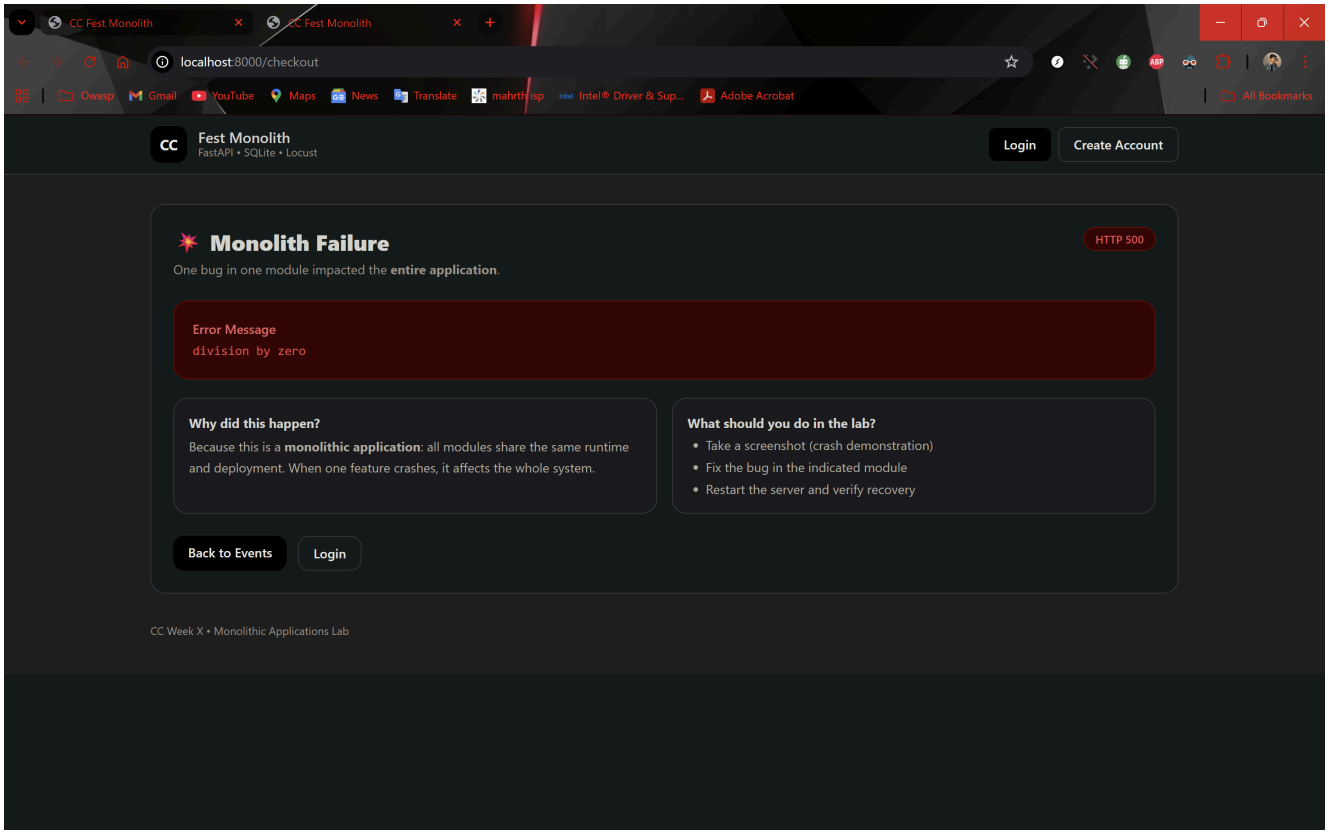
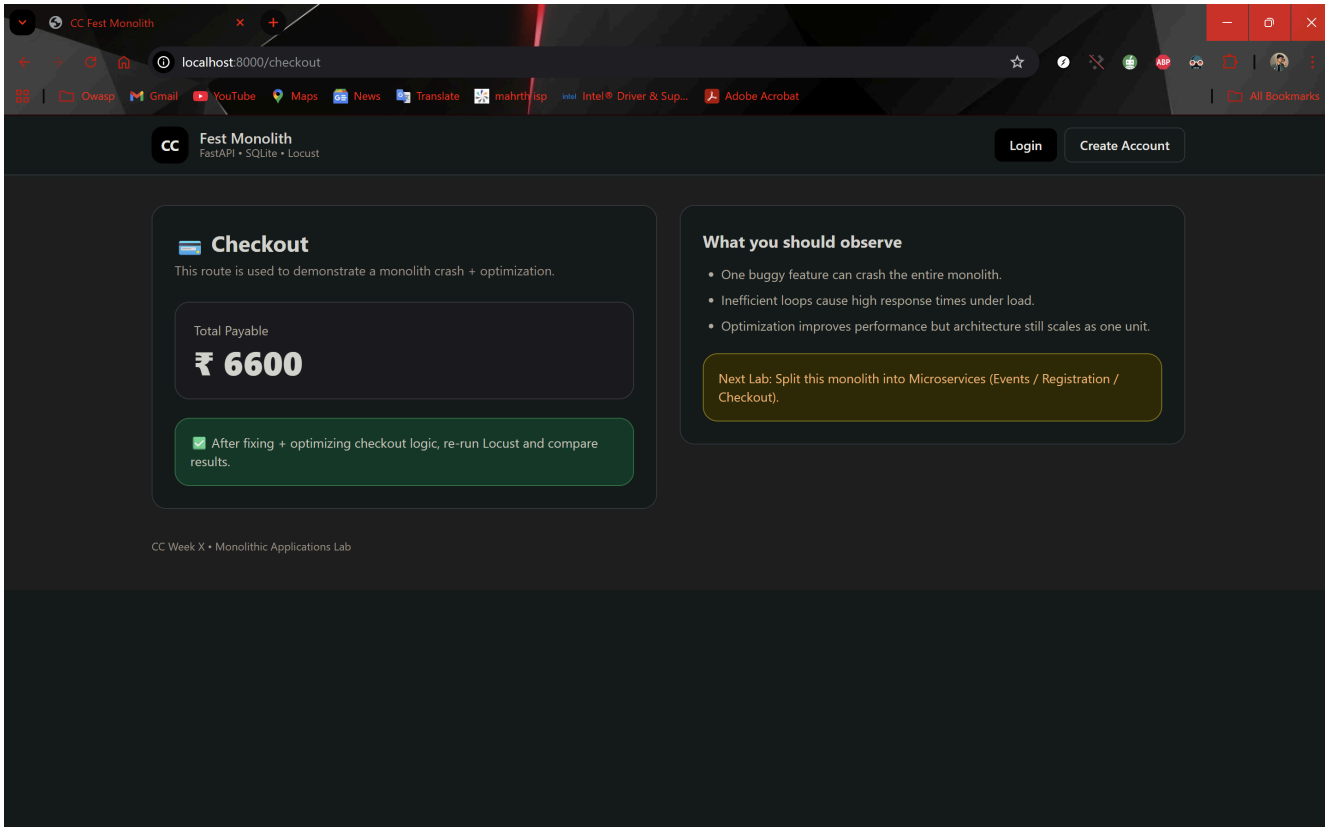

CC Fest Monolith
localhost:8000/events?user=PES2UG23CS361
Owasp
Gmail
YouTube
Maps
News
Translate
mahrthisp
Intel® Driver & Sup...
Adobe Acrobat
All Bookmarks
CC
Fest Monolith
FastAPI • SQLite • Locust
Logged in as PES2UG23CS361
Events
My Events
Checkout
Logout
Events
Welcome PES2UG23CS361. Register for events below.
View My Events →
Event ID: 1
₹ 500
Hackathon
Includes certificate • instant registration • limited seats
Register
Event ID: 2
₹ 300
Dance
Includes certificate • instant registration • limited seats
Register
Event ID: 3
₹ 500
Hackathon
Includes certificate • instant registration • limited seats
Register
Event ID: 4
₹ 300
Dance Battle
Includes certificate • instant registration • limited seats
Register
Event ID: 5
₹ 400
AI Workshop
Includes certificate • instant registration • limited seats
Register
Event ID: 6
₹ 200
Photography Walk
Includes certificate • instant registration • limited seats
Register
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

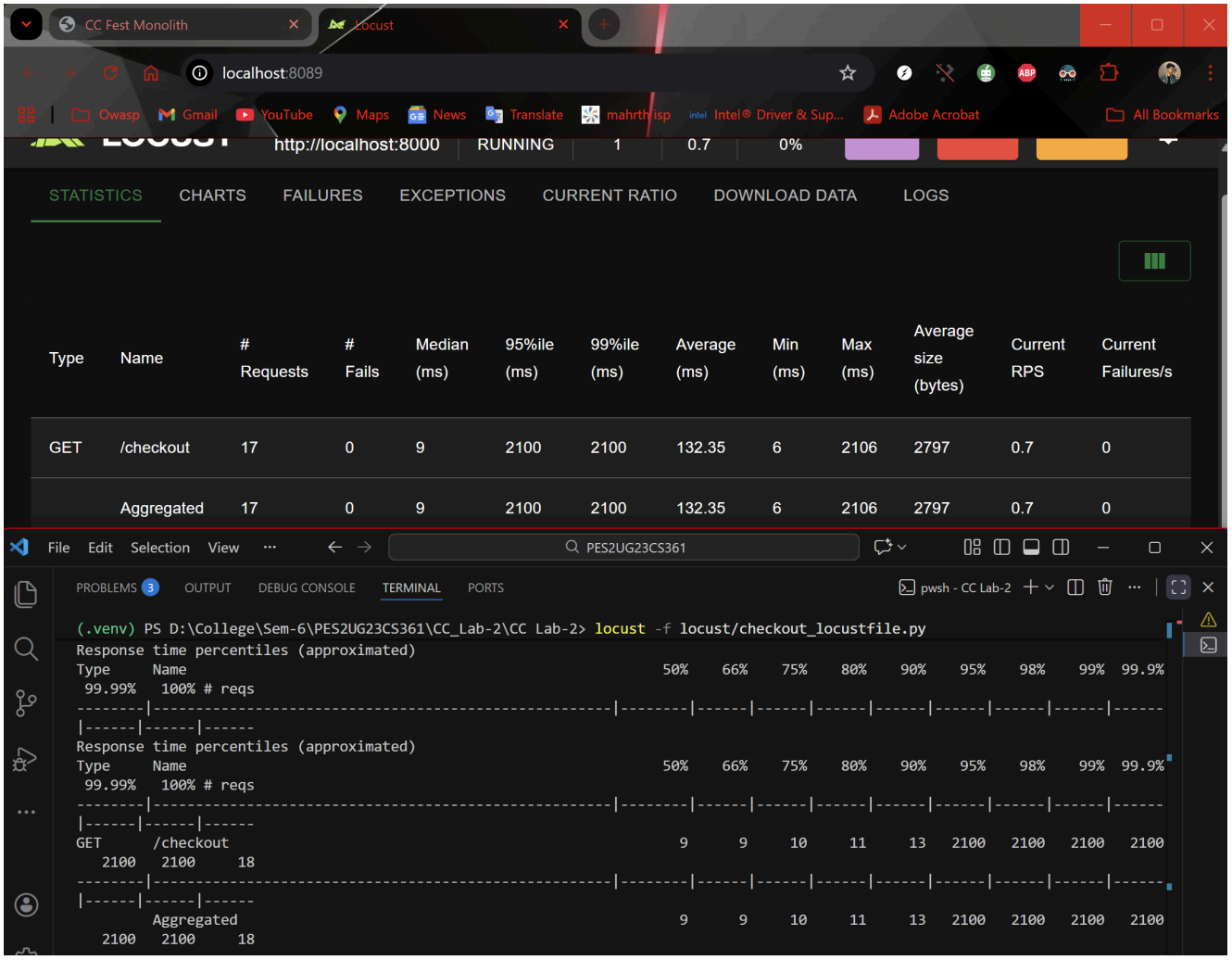
SS2:


CC Fest Monolith
CC Fest Monolith
localhost:8000/checkout
Owasp
Gmail
YouTube
Maps
News
Translate
mahrthisp
Intel® Driver & Sup...
Adobe Acrobat
All Bookmarks
CC
Fest Monolith
FastAPI • SQLite • Locust
Login
Create Account
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

SS3:


CC Fest Monolith
localhost:8000/checkout
Fest Monolith
FastAPI • SQLite • Locust
Login
Create 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

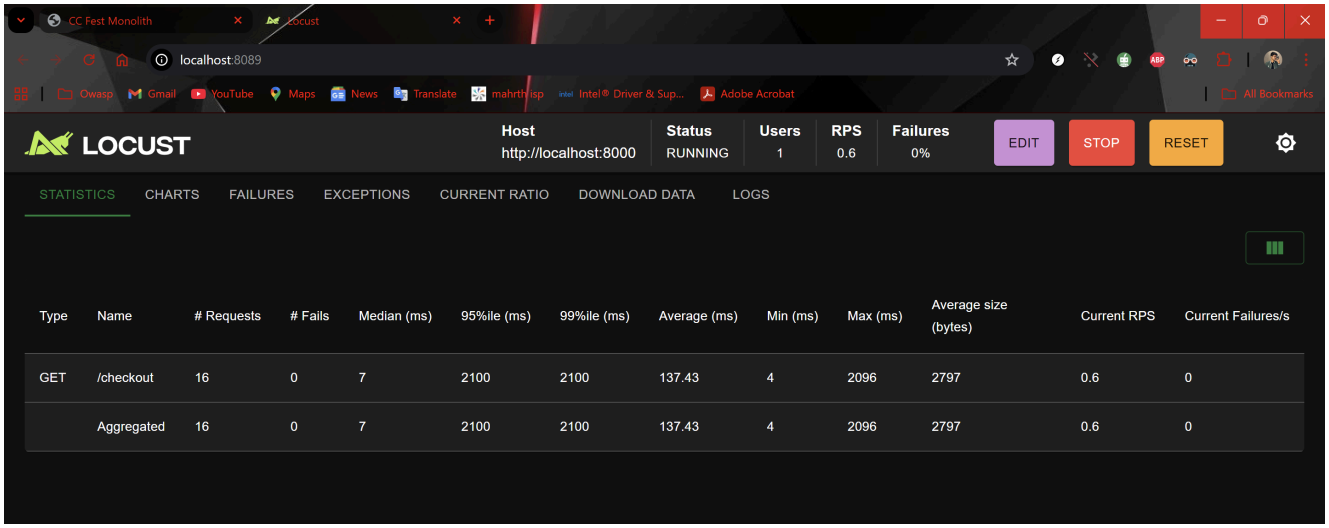
SS4:


CC Fest Monolith
Locust
localhost:8089
http://localhost:8000
RUNNING
1
0.7
0%
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 /checkout 17 0 9 2100 2100 132.35 6 2106 2797 0.7 0
Aggregated 17 0 9 2100 2100 132.35 6 2106 2797 0.7 0
File Edit Selection View
PES2UG23CS361
PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS
pwsh - CC Lab-2
(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2> locust -f locust/checkout_locustfile.py
Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100% # reqs
GET /checkout 9 9 10 11 13 2100 2100 2100 2100 2100 2100 18
Aggregated 9 9 10 11 13 2100 2100 2100 2100 2100 2100 18

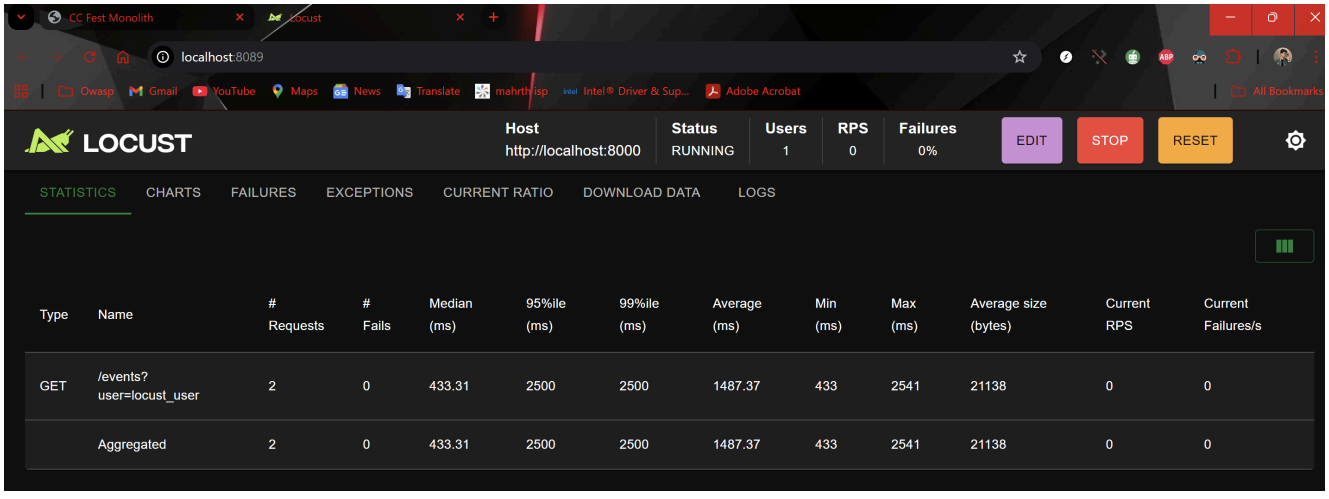
SS5:

```
(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2> locust -f locust/checkout_locustfile.py
[2026-01-20 14:58:42,619] Charlieee/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 14:58:42,620] Charlieee/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browse
r.
[2026-01-20 14:59:14,766] Charlieee/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 14:59:14,768] Charlieee/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
Traceback (most recent call last):
  File "D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\.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
2026-01-20T09:29:41Z
[2026-01-20 14:59:41,855] Charlieee/INFO/locust.main: Shutting down (exit code 0)
Type     Name                                                          # reqs      # fails |    Avg     Min     Max    Med |   req/s  failures/s
--------|-------------------------------------------------------------|-------|-------------|-------|-------|-------|-------|--------|-----------
GET      /checkout                                                        17     0(0.00%) |    129       4    2096      7 |    0.63        0.00
--------|-------------------------------------------------------------|-------|-------------|-------|-------|-------|-------|--------|-----------
         Aggregated                                                       17     0(0.00%) |    129       4    2096      7 |    0.63        0.00

Response time percentiles (approximated)
Type     Name                                                                 50%    66%    75%    80%    90%    95%    98%    99%  99.9% 99.99
%   100% # reqs
--------|-------------------------------------------------------------------|--------|------|------|-  80%    90%    95%    98%    99%  99.9% 99.99
%   100% # reqs
  80%    90%    95%    98%    99%  99.9% 99.99%   100% # reqs
--------|-------------------------------------------------------------------|--------|------|------|------|------|------|------|------|-----
-|------|------
GET      /checkout                                                            7      7      8      8      9   2100   2100   2100   2100   210
0   2100     17
--------|-------------------------------------------------------------------|--------|------|------|------|------|------|------|------|-----
-|------|------
         Aggregated                                                           7      7      8      8      9   2100   2100   2100   2100   210
0   2100     17
```



SS6:

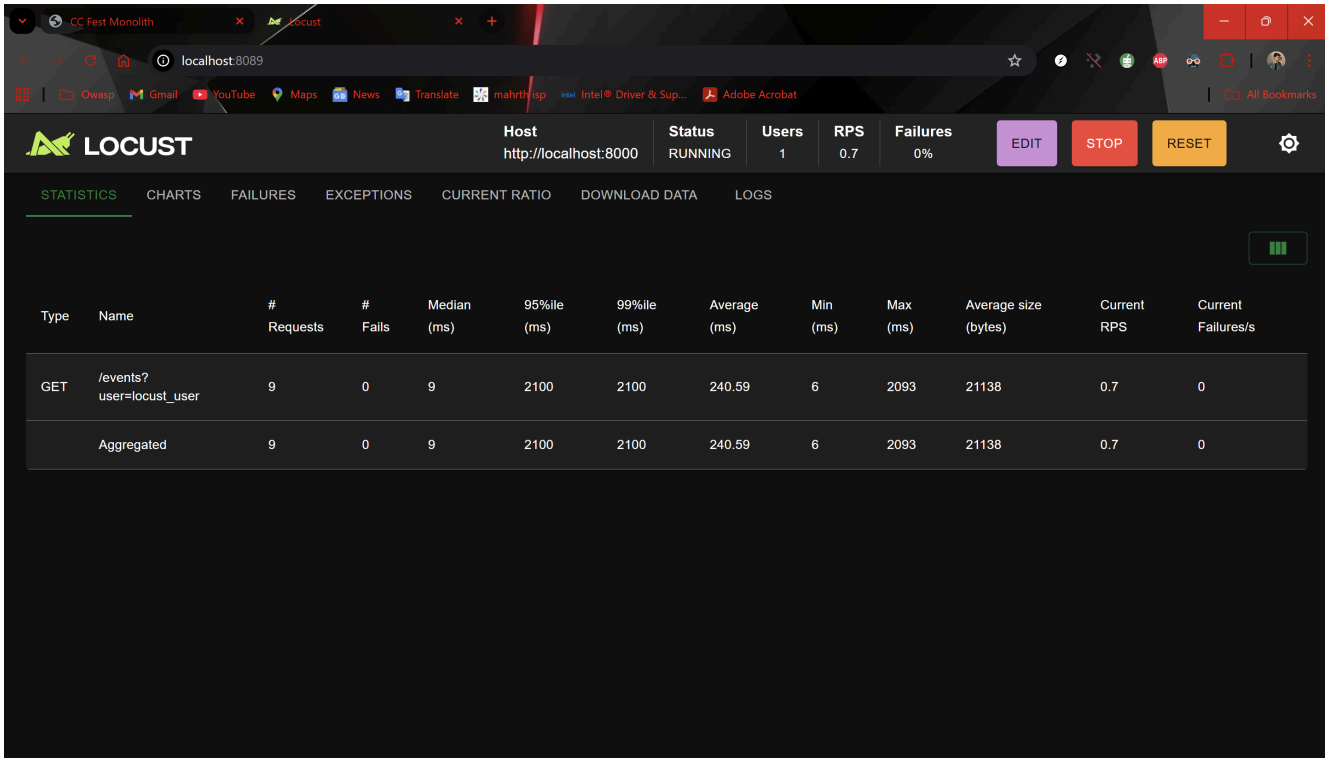


```
(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2> locust -f locust/events_locustfile.py
[2026-01-20 15:04:23,589] Charlieee/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 15:04:23,590] Charlieee/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browse
r.
[2026-01-20 15:05:29,913] Charlieee/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 15:05:29,916] Charlieee/INFO/locust.runners: All users spawned: {"EventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\.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
2026-01-20T09:35:41Z
[2026-01-20 15:05:41,672] Charlieee/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                                          6     0(0.00%) |    689     182    2541    350 |    0.52        0.00
--------|-------------------------------------------------------------|-------|-------------|-------|-------|-------|-------|--------|-----------
         Aggregated                                                        6     0(0.00%) |    689     182    2541    350 |    0.52        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                                          430    430    440    440   2500   2500   2500   2500   2500   250
0   2500      6
--------|-------------------------------------------------------------|--------|------|------|------|------|------|------|------|------|-----
-|------|------
         Aggregated                                                        430    430    440    440   2500   2500   2500   2500   2500   250
0   2500      6

(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2>
```

SS7:

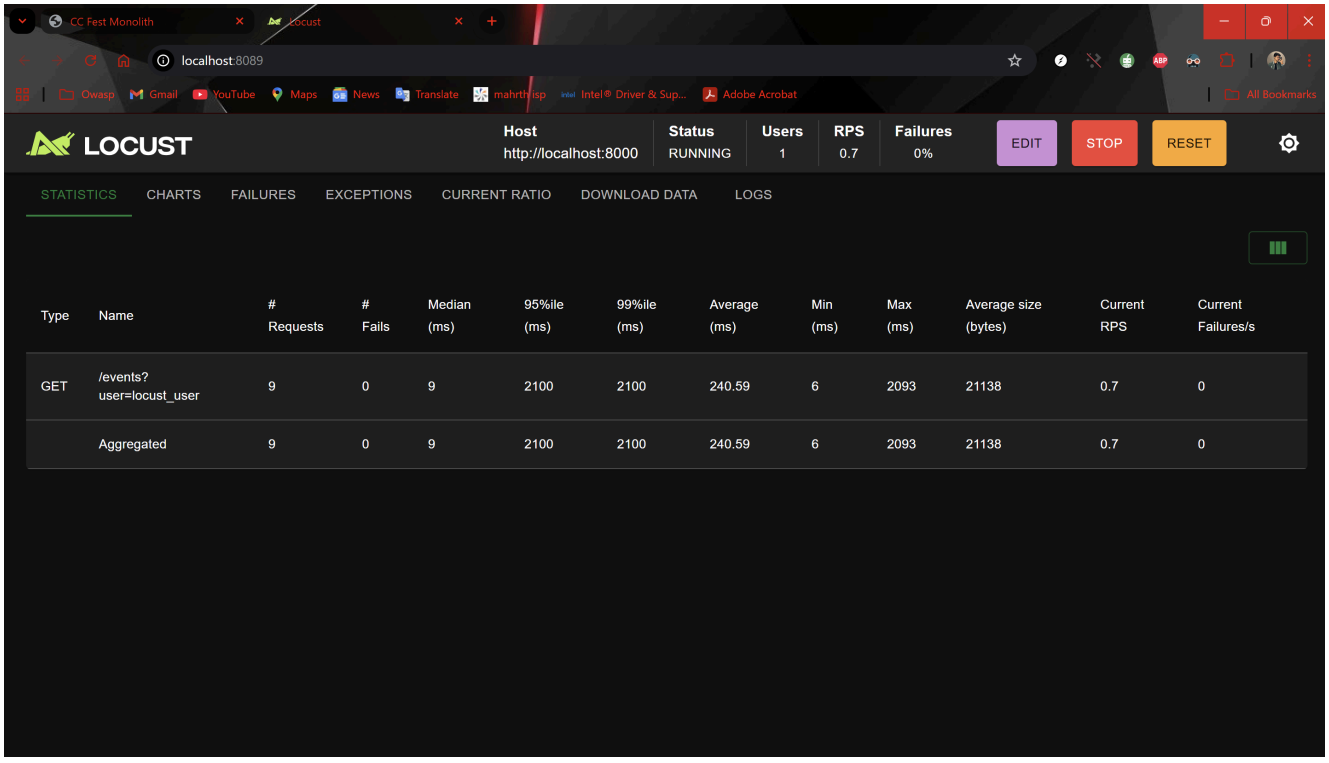


```
(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2> locust -f locust/events_locustfile.py
[2026-01-20 15:10:14,963] Charlieee/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 15:10:14,964] Charlieee/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browse
r.
[2026-01-20 15:10:34,085] Charlieee/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 15:10:34,087] Charlieee/INFO/locust.runners: All users spawned: {"EventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\.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
2026-01-20T09:40:54Z
[2026-01-20 15:10:54,838] Charlieee/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                                         13     0(0.00%) |    169       4    2092      9 |    0.66        0.00
--------|-------------------------------------------------------------|-------|-------------|-------|-------|-------|-------|--------|-----------
         Aggregated                                                       13     0(0.00%) |    169       4    2092      9 |    0.66        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                                           9     10     10     11     11   2100   2100   2100   2100   210
0   2100     13
--------|-------------------------------------------------------------|--------|------|------|------|------|------|------|------|------|-----
-|------|------
         Aggregated                                                         9     10     10     11     11   2100   2100   2100   2100   210
0   2100     13

(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2>
```

SS8:

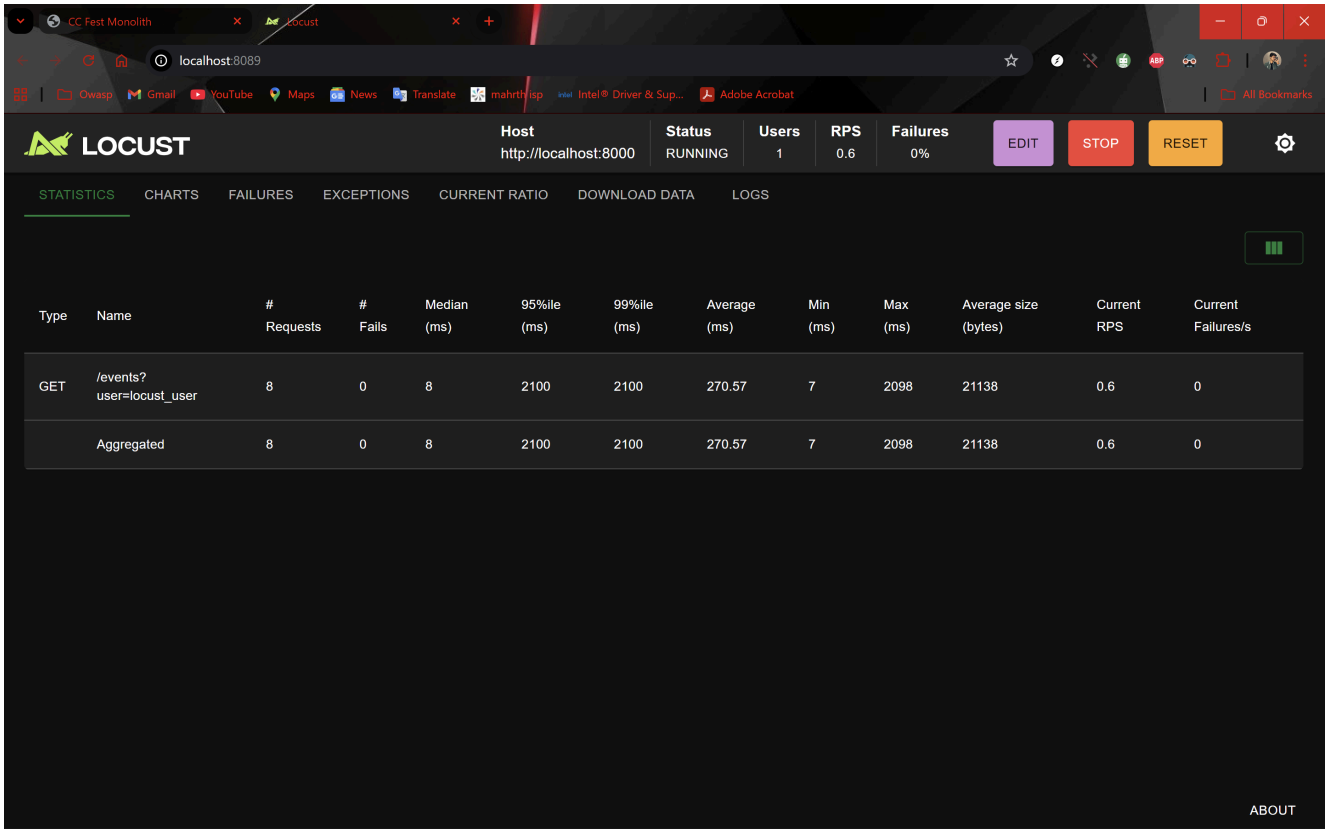


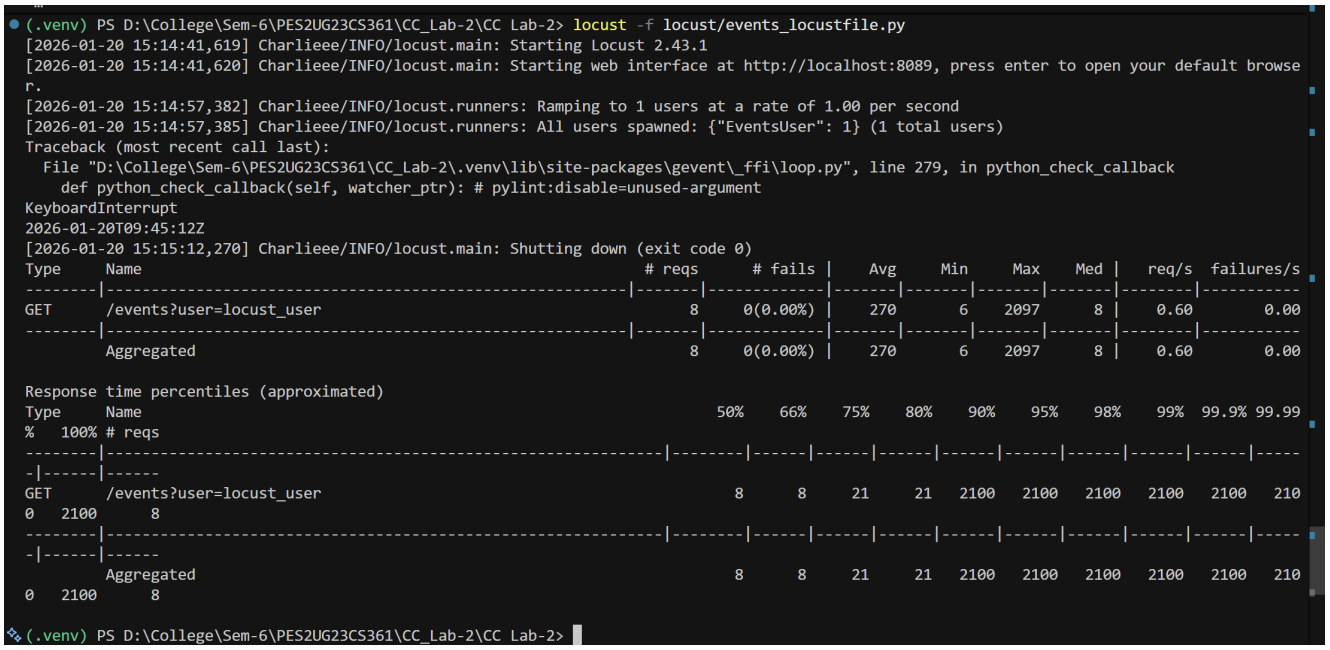
```
(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2> locust -f locust/events_locustfile.py
[2026-01-20 15:10:14,963] Charlieee/INFO/locust.main: Starting Locust 2.43.1
[2026-01-20 15:10:14,964] Charlieee/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browse
r.
[2026-01-20 15:10:34,085] Charlieee/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-20 15:10:34,087] Charlieee/INFO/locust.runners: All users spawned: {"EventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\.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
2026-01-20T09:40:54Z
[2026-01-20 15:10:54,838] Charlieee/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                                         13     0(0.00%) |    169       4    2092      9 |    0.66        0.00
--------|-------------------------------------------------------------|-------|-------------|-------|-------|-------|-------|--------|-----------
         Aggregated                                                       13     0(0.00%) |    169       4    2092      9 |    0.66        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                                           9     10     10     11     11   2100   2100   2100   2100   210
0   2100     13
--------|-------------------------------------------------------------|--------|------|------|------|------|------|------|------|------|-----
-|------|------
         Aggregated                                                         9     10     10     11     11   2100   2100   2100   2100   210
0   2100     13

(.venv) PS D:\College\Sem-6\PES2UG23CS361\CC_Lab-2\CC Lab-2>
```

SS9:





What was the bottleneck?

The bottleneck was the CPU-bound busy loops in events() and my_events() — the large for-loops (3,000,000 and 1,500,000 iterations) that block request handling and exhaust the worker's CPU.

What change did you make?

Commented out the CPU-bound busy loops in events() and my_events() (the 3,000,000 and 1,500,000 iteration for-loops), eliminating the CPU bottleneck.

Why did the performance improve?

- Removing the large CPU-bound for-loops eliminated synchronous work that was saturating the worker CPU and blocking request handling.
- With no busy loops, the server can process I/O (DB queries, template rendering, network) without being blocked, so latency and throughput improved. Resulted in much much lower CPU usage, faster response times, and better concurrency.