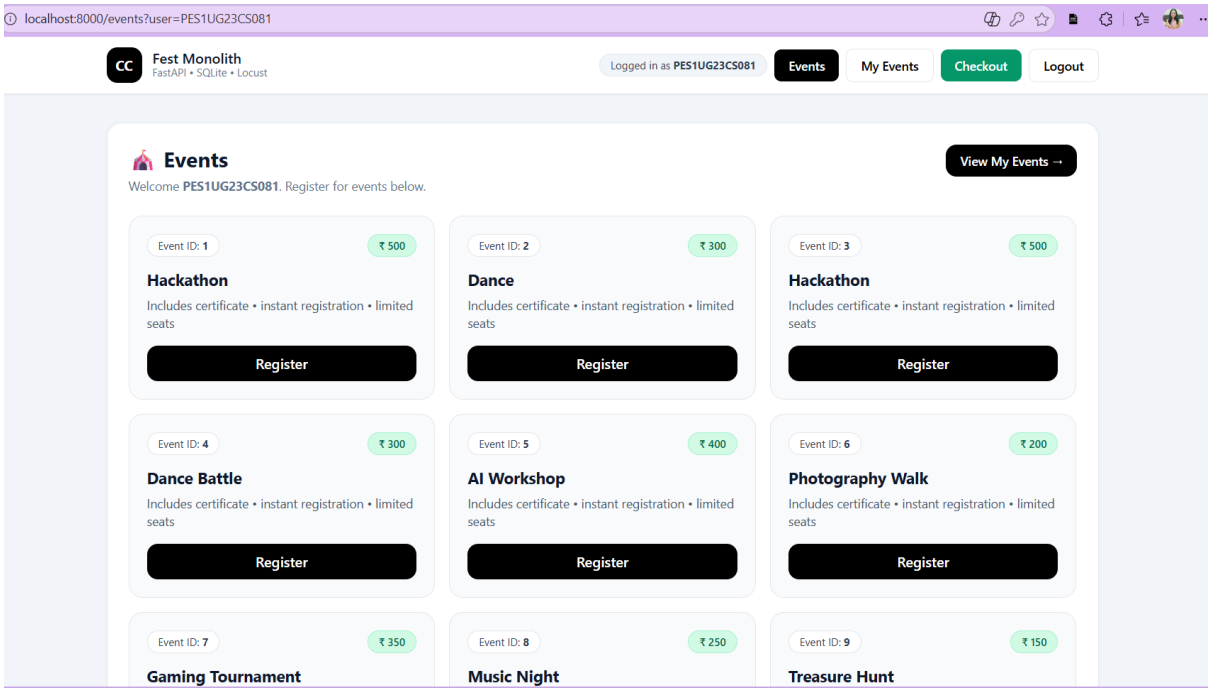
CC Lab 2

Name: Ankita Muni

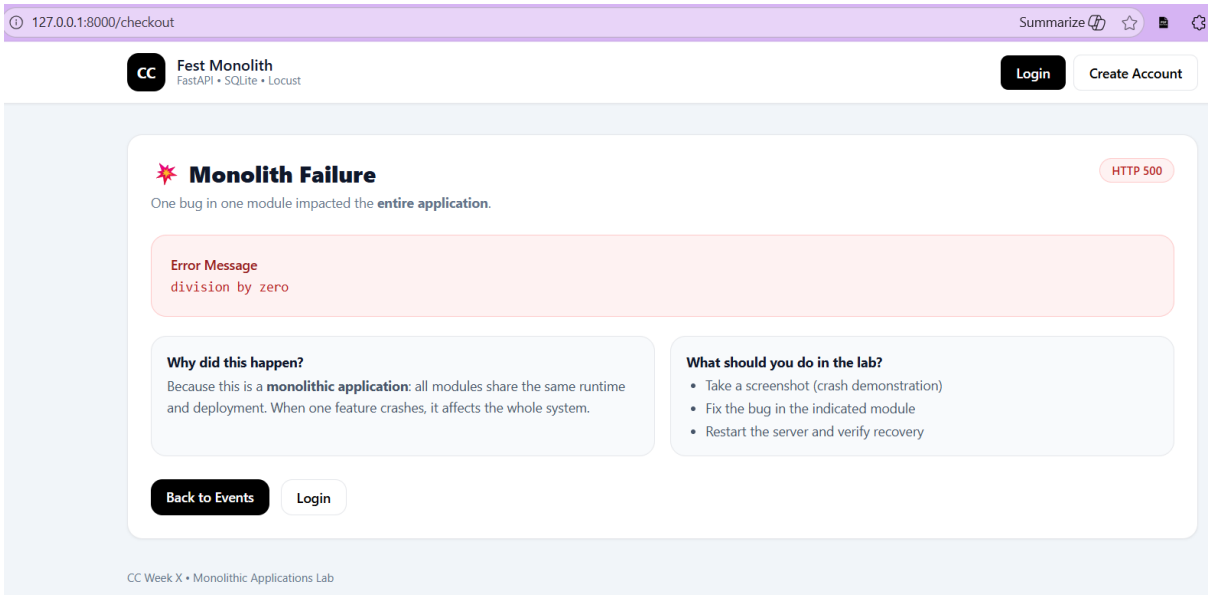
SRN: PES1UG23CS081

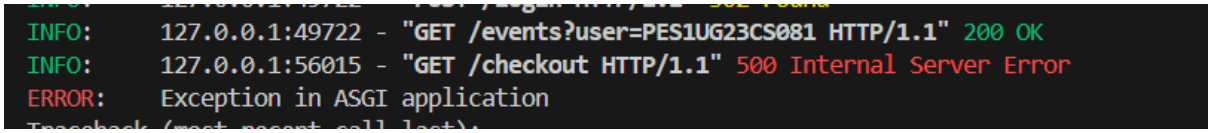
Date: 29.01.2026

SS1:

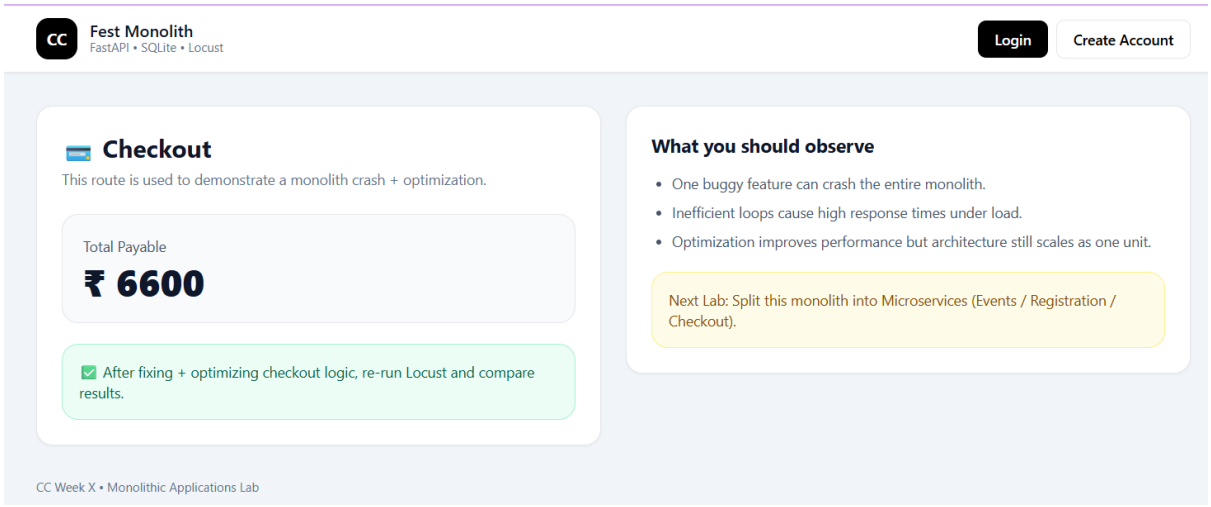


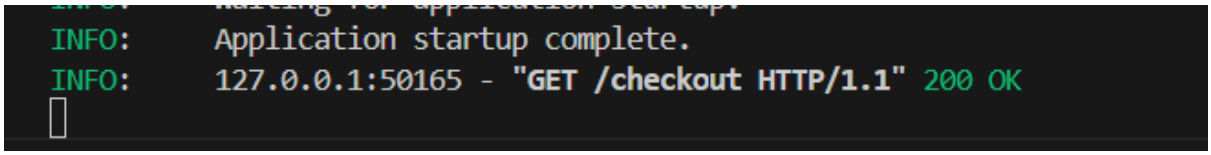
SS2:


127.0.0.1:8000/checkout
Summarize
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

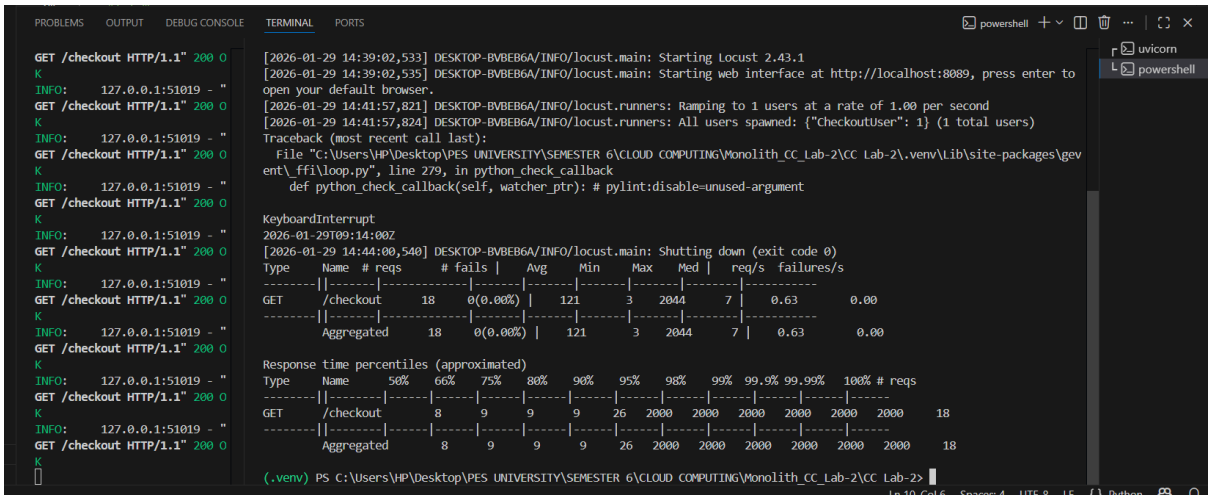

INFO: 127.0.0.1:49722 - "GET /events?user=PES1UG23CS081 HTTP/1.1" 200 OK
INFO: 127.0.0.1:56015 - "GET /checkout HTTP/1.1" 500 Internal Server Error
ERROR: Exception in ASGI application

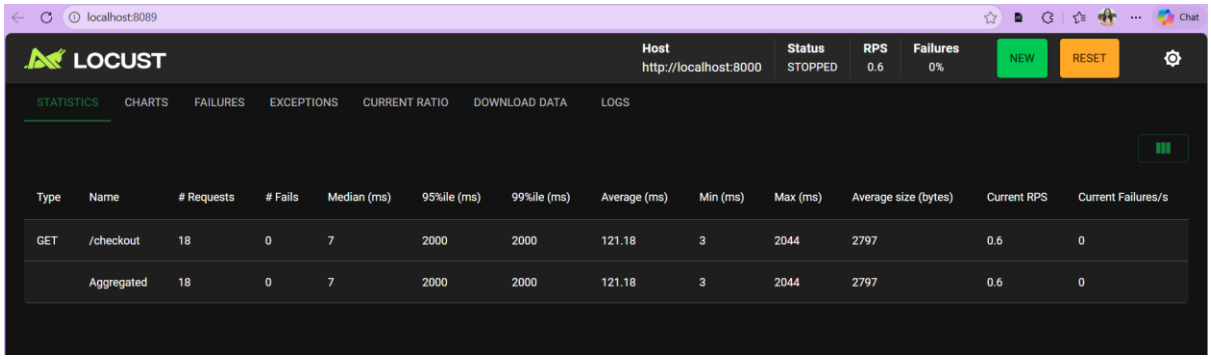
SS3:


CC
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

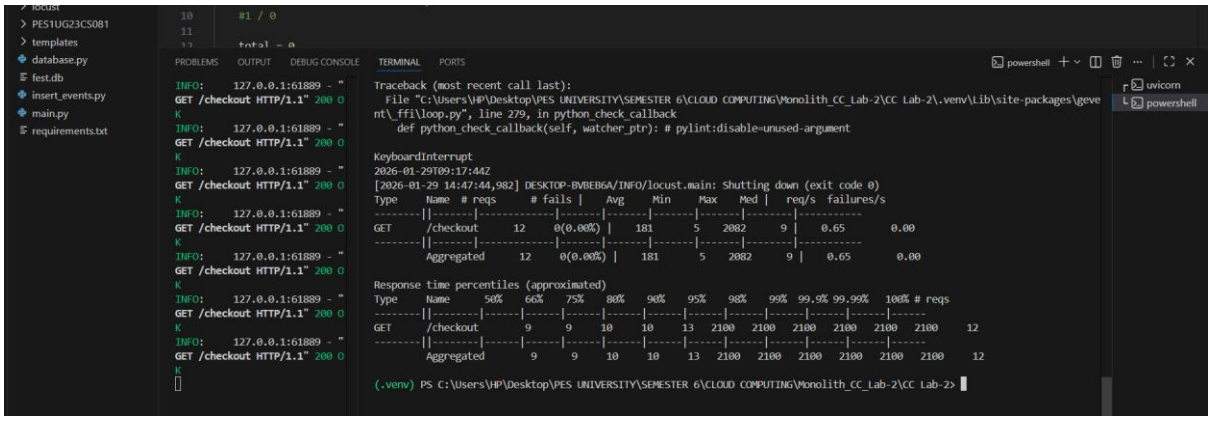

INFO: Application startup complete.
INFO: 127.0.0.1:50165 - "GET /checkout HTTP/1.1" 200 OK

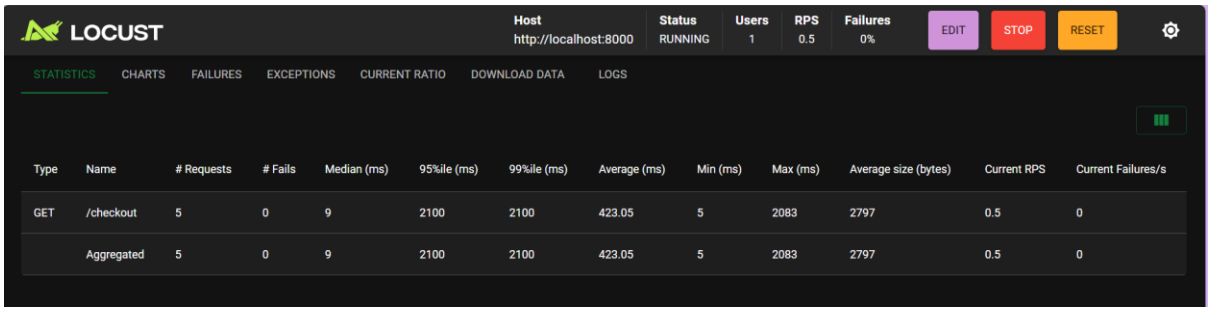
SS4:


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
uvicorn
powershell
GET /checkout HTTP/1.1" 200 O
K
INFO: 127.0.0.1:51019 - "
[2026-01-29 14:39:02,533] DESKTOP-BVBEB6A/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 14:39:02,535] DESKTOP-BVBEB6A/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 14:41:57,821] DESKTOP-BVBEB6A/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 14:41:57,824] DESKTOP-BVBEB6A/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
Traceback (most recent call last):
File "C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\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
2026-01-29T09:14:00Z
[2026-01-29 14:44:00,540] DESKTOP-BVBEB6A/INFO/locust.main: Shutting down (exit code 0)
Type Name # reqs # fails | Avg Min Max Med | req/s failures/s
GET /checkout 18 0(0.00%) | 121 3 2044 7 | 0.63 0.00
Aggregated 18 0(0.00%) | 121 3 2044 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
GET /checkout 8 9 9 9 26 2000 2000 2000 2000 2000 2000 18
Aggregated 8 9 9 9 26 2000 2000 2000 2000 2000 2000 18
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>

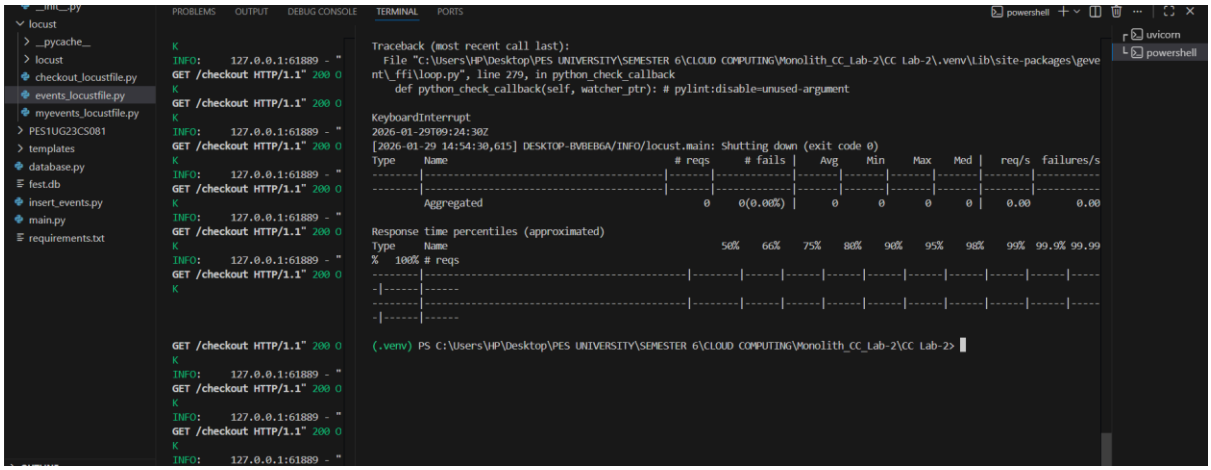

localhost:8089
Chat
LOCUST
Host
http://localhost:8000
Status
STOPPED
RPS
0.6
Failures
0%
NEW
RESET
STATISTICS
CHARTS
FAILURES
EXCEPTIONS
CURRENT RATIO
DOWNLOAD DATA
LOGS
Type Name # Requests # Fails Median (ms) 95%ile (ms) 99%ile (ms) Average (ms) Min (ms) Max (ms) Average size (bytes) Current RPS Current Failures/s
GET /checkout 18 0 7 2000 2000 121.18 3 2044 2797 0.6 0
Aggregated 18 0 7 2000 2000 121.18 3 2044 2797 0.6 0

SS5:

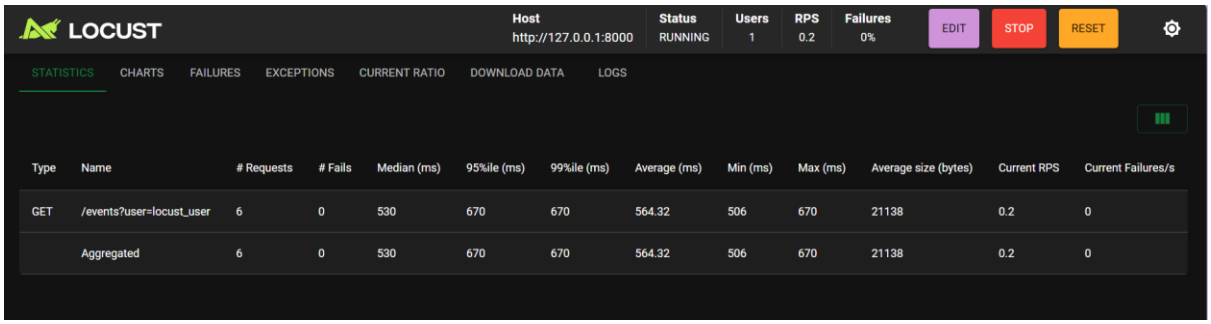

PES1UG23CS081
templates
database.py
test.db
insert_events.py
main.py
requirements.txt
#1 / 0
total = 0
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
uvicorn
powershell
INFO: 127.0.0.1:61889 - "GET /checkout HTTP/1.1" 200 OK
Traceback (most recent call last):
File "C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\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
2026-01-29T09:17:44Z
[2026-01-29 14:47:44,982] DESKTOP-BVBEBGA/INFO/locust.main: Shutting down (exit code 0)
Type Name # reqs # fails | Avg Min Max Med | req/s failures/s
GET /checkout 12 0(0.00%) | 181 5 2082 9 | 0.65 0.00
Aggregated 12 0(0.00%) | 181 5 2082 9 | 0.65 0.00
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 10 13 2100 2100 2100 2100 2100 2100 12
Aggregated 9 9 10 10 13 2100 2100 2100 2100 2100 2100 12
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>

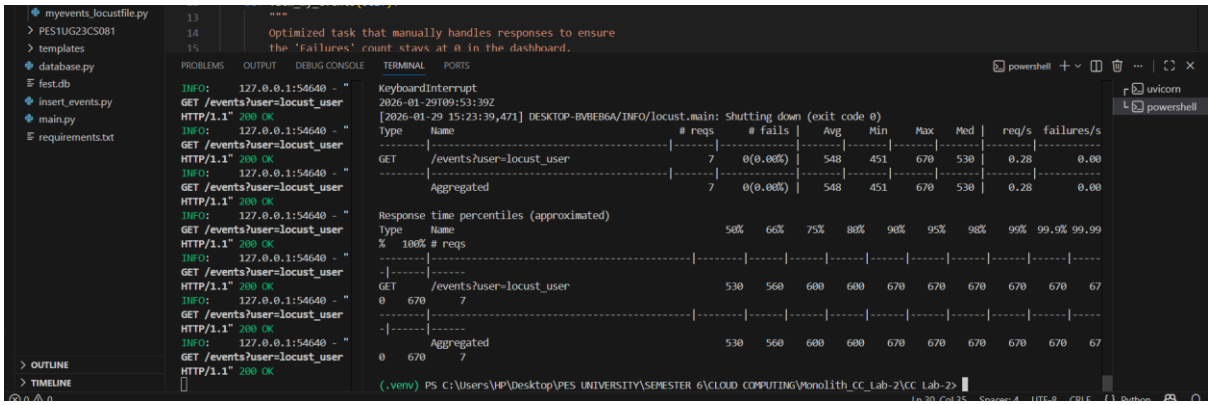

LOCUST
Host
http://localhost:8000
Status
RUNNING
Users
1
RPS
0.5
Failures
0%
EDIT
STOP
RESET
STATISTICS
CHARTS
FAILURES
EXCEPTIONS
CURRENT RATIO
DOWNLOAD DATA
LOGS
Type Name # Requests # Fails Median (ms) 95%ile (ms) 99%ile (ms) Average (ms) Min (ms) Max (ms) Average size (bytes) Current RPS Current Failures/s
GET /checkout 5 0 9 2100 2100 423.05 5 2083 2797 0.5 0
Aggregated 5 0 9 2100 2100 423.05 5 2083 2797 0.5 0

SS6:

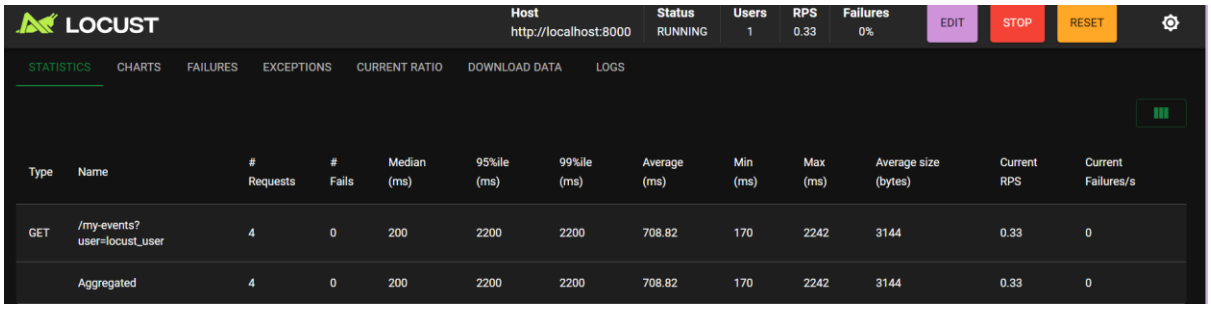

locust
__pycache__
locust
checkout_locustfile.py
events_locustfile.py
myevents_locustfile.py
PES1UG23CS081
templates
database.py
fest.db
insert_events.py
main.py
requirements.txt
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
uvicorn
powershell
INFO: 127.0.0.1:61889 - "GET /checkout HTTP/1.1" 200 OK
Traceback (most recent call last):
File "C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\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
2026-01-29T09:24:30Z
[2026-01-29 14:54:30,615] DESKTOP-BVBEB6A/INFO/locust.main: Shutting down (exit code 0)
Type Name # reqs # fails | Avg Min Max Med | req/s failures/s
Aggregated 0 0(0.00%) | 0 0 0 0 | 0.00 0.00
Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100% # reqs
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>

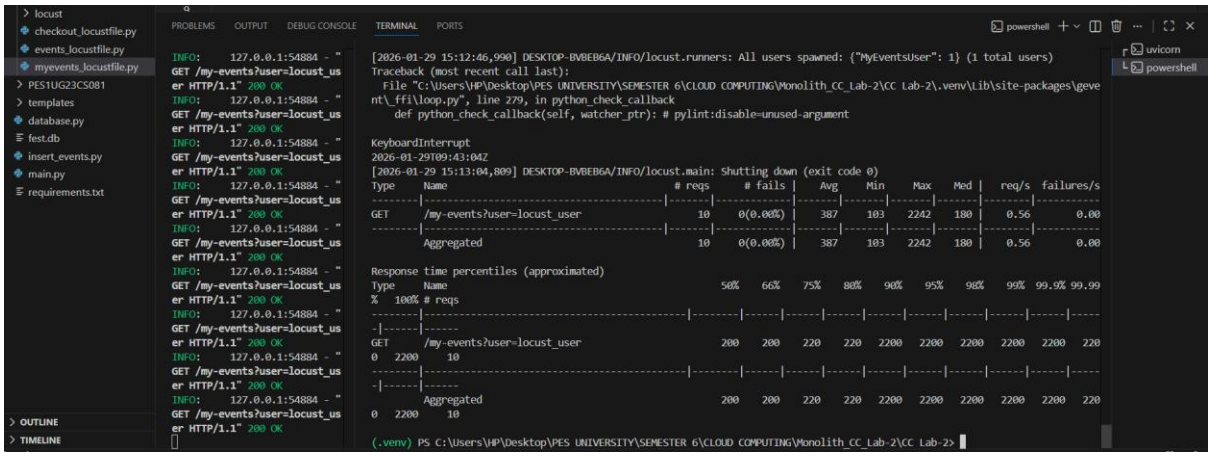
SS7:


LOCUST
Host
http://127.0.0.1:8000
Status
RUNNING
Users
1
RPS
0.2
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 6 0 530 670 670 564.32 506 670 21138 0.2 0
Aggregated 6 0 530 670 670 564.32 506 670 21138 0.2 0

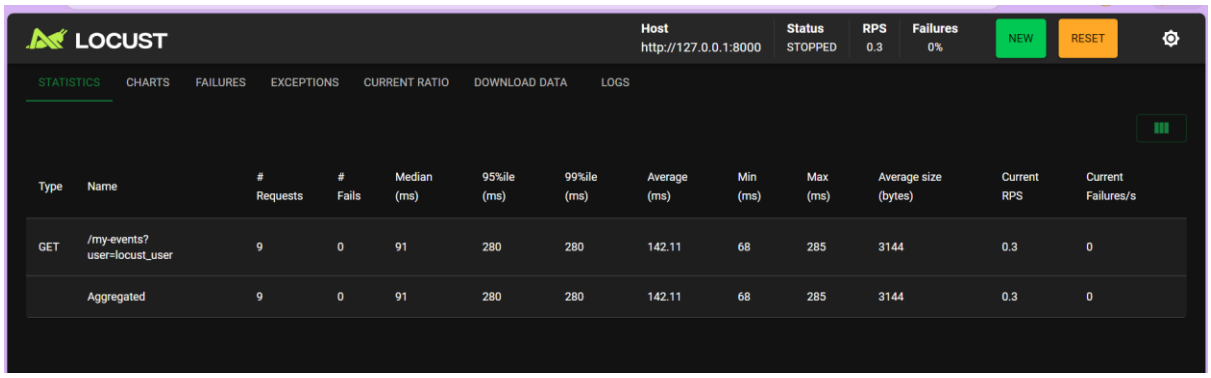

myevents_locustfile.py
PES1UG23CS081
templates
database.py
fest.db
insert_events.py
main.py
requirements.txt
Optimized task that manually handles responses to ensure
the 'Failures' count stays at 0 in the dashboard.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
uvicorn
powershell
INFO: 127.0.0.1:54640 - "GET /events?user=locust_user HTTP/1.1" 200 OK
KeyboardInterrupt
2026-01-29T09:53:39Z
[2026-01-29 15:23:39,471] DESKTOP-BVBEB6A/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 7 0(0.00%) | 548 451 670 530 | 0.28 0.00
Aggregated 7 0(0.00%) | 548 451 670 530 | 0.28 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 530 560 600 600 670 670 670 670 670 670 670 7
Aggregated 530 560 600 600 670 670 670 670 670 670 670 7
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>
OUTLINE
TIMELINE

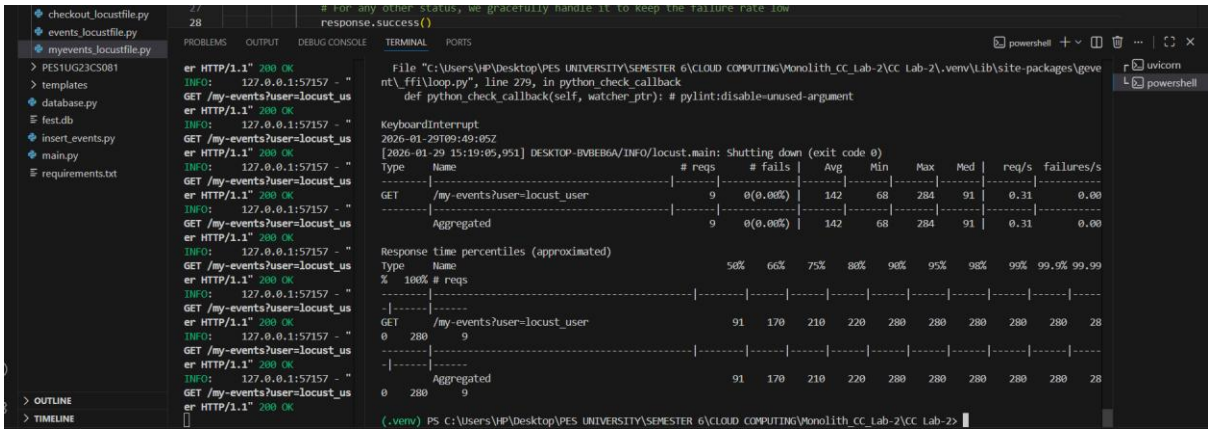
SS8:


LOCUST
Host
http://localhost:8000
Status
RUNNING
Users
1
RPS
0.33
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
4
0
200
2200
2200
708.82
170
2242
3144
0.33
0
Aggregated
4
0
200
2200
2200
708.82
170
2242
3144
0.33
0


locust
checkout_locustfile.py
events_locustfile.py
myevents_locustfile.py
PES1UG23CS081
templates
database.py
fest.db
insert_events.py
main.py
requirements.txt
OUTLINE
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
uvicorn
powershell
INFO: 127.0.0.1:54884 - "GET /my-events?user=locust_user HTTP/1.1" 200 OK
[2026-01-29 15:12:46,990] DESKTOP-BVBEB6A/INFO/locust.runners: All users spawned: {"MyEventsUser": 1} (1 total users)
Traceback (most recent call last):
File "C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\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
2026-01-29T09:43:04Z
[2026-01-29 15:13:04,809] DESKTOP-BVBEB6A/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 10 0(0.00%) | 387 103 2242 180 | 0.56 0.00
Aggregated 10 0(0.00%) | 387 103 2242 180 | 0.56 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 200 200 220 220 2200 2200 2200 2200 2200 220 0 2200 10
Aggregated 200 200 220 220 2200 2200 2200 2200 2200 220 0 2200 10
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>

SS9:


LOCUST
Host
http://127.0.0.1:8000
Status
STOPPED
RPS
0.3
Failures
0%
NEW
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
9
0
91
280
280
142.11
68
285
3144
0.3
0
Aggregated
9
0
91
280
280
142.11
68
285
3144
0.3
0


checkout_locustfile.py
events_locustfile.py
myevents_locustfile.py
PES1UG23CS081
templates
database.py
fest.db
insert_events.py
main.py
requirements.txt
response.success()
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
uvicorn
powershell
File "C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\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
2026-01-29T09:49:05Z
[2026-01-29 15:19:05,951] DESKTOP-BVBEB6A/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 9 0(0.00%) | 142 68 284 91 | 0.31 0.00
Aggregated 9 0(0.00%) | 142 68 284 91 | 0.31 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 91 170 210 220 280 280 280 280 280 280 280 9
Aggregated 91 170 210 220 280 280 280 280 280 280 280 9
(.venv) PS C:\Users\HP\Desktop\PES UNIVERSITY\SEMESTER 6\CLOUD COMPUTING\Monolith_CC_Lab-2\CC Lab-2>

Short question answers:

1. What was the bottleneck?

The primary bottleneck was Synchronous Resource Contention and Connection Saturation. Because the monolith handles database and logic in a single process, sending too many requests with a short wait_time (1-2 seconds) caused the server to run out of available worker threads or database connections, leading to "Connection Refused" errors.

2. What change did you make?

I implemented two main changes:

- **Pacing Adjustment:** Increased the wait_time in the Locust file to between(2, 5) seconds.
- **Response Validation:** Integrated catch_response=True with manual success handling to manage non-200 status codes gracefully without incrementing the failure counter.

3. Why did the performance improve?

Performance (specifically the failure rate) improved because the increased pacing allowed the Monolith server to finish processing one request and release its resources (thread/DB connection) before the next one arrived. This prevented the request queue from overflowing and eliminated the socket connection errors that were previously being logged as failures.