

# Cloud Computing

## LAB 2 - MONOLITHIC ARCHITECTURE

**Github url :**

[https://github.com/Laxmisah/Monolithic-Application\\_PES1UG23CS327](https://github.com/Laxmisah/Monolithic-Application_PES1UG23CS327)

SS1

The screenshot shows a web application interface for 'Fest Monolith' with the following details:

- Header:** CC Fest Monolith, FastAPI • SQLite • Locust. Logged in as PES1UG23CS327. Navigation buttons: Events, My Events, Checkout, Logout.
- Section:** Events. Welcome message: 'Welcome PES1UG23CS327. Register for events below.'
- Events List:** A grid of event cards.
  - Event ID: 1** (Hackathon): ₹ 500. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 2** (Dance): ₹ 300. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 3** (Hackathon): ₹ 500. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 4** (Dance Battle): ₹ 300. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 5** (AI Workshop): ₹ 400. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 6** (Photography Walk): ₹ 200. Includes certificate • instant registration • limited seats. **Register** button.
  - Event ID: 7**: ₹ 350. **Register** button.
  - Event ID: 8**: ₹ 250. **Register** button.
  - Event ID: 9**: ₹ 150. **Register** button.
- Buttons:** View My Events →.

SS2

The screenshot shows a web application interface for 'Fest Monolith' with the following details:

- Header:** CC Fest Monolith, FastAPI • SQLite • Locust. Navigation buttons: Login, Create Account.
- Section:** Monolith Failure. Message: '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
- Buttons:** Back to Events, Login.

```
INFO:      Waiting for application startup.
INFO:      Application startup complete.
INFO: 127.0.0.1:50060 - "GET /login HTTP/1.1" 200 OK
```

## SS3

The screenshot shows a web application interface titled "Fest Monolith" with the subtext "FastAPI • SQLite • Locust". At the top right are "Login" and "Create Account" buttons. Below the title, there's a section titled "Checkout" with a sub-section "What you should observe". The main content area displays a large button labeled "₹ 6600" with the text "Total Payable" above it. A note below the button says "After fixing + optimizing checkout logic, re-run Locust and compare results." To the right of the main content, under "What you should observe", is a list of bullet points: "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 callout box at the bottom right suggests "Next Lab: Split this monolith into Microservices (Events / Registration / Checkout)". At the bottom left of the main content area, there's a small note: "CC Week X • Monolithic Applications Lab".

INFO: 127.0.0.1:64866 - "GET /checkout HTTP/1.1" 200 OK

## SS4

The screenshot shows the Locust performance testing interface. At the top, it displays "LOCUST" with a logo, the host "http://localhost:8000", and the status "STOPPED" with RPS 0.7 and 0% failures. It also has "NEW" and "RESET" buttons and a settings gear icon. Below this is a navigation bar with tabs: STATISTICS (selected), CHARTS, FAILURES, EXCEPTIONS, CURRENT RATIO, DOWNLOAD DATA, and LOGS. On the far right of the header is a refresh icon. The main content area is a table titled "STATISTICS" showing request details. The table has columns: Type, Name, # Requests, # Fails, Median (ms), 95%ile (ms), 99%ile (ms), Average (ms), Min (ms), Max (ms), Average size (bytes), Current RPS, and Current Failures/s. Two rows are present: one for a GET request to "/checkout" with 19 requests, 0 fails, and various performance metrics; and an aggregated row for all requests with 19 total requests, 0 fails, and similar metrics.

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	19	0	7	2000	2000	114.34	6	2042	2797	0.7	0
	Aggregated	19	0	7	2000	2000	114.34	6	2042	2797	0.7	0

File Edit Selection View Go Run Terminal Help ← → 🔍 PES1UG23CS327

EXPLORER OPEN EDITORS PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PEST1UG23CS327 CC Lab-2 .NET Core 3.1.2

CC Lab-2 > CC Lab-2 > checkout \_init\_.py checkout\_logic

```
3 def checkout_logic():
5     db.row_factory = None
```

powerShell - CC Lab-2

TERMINAL

```
PS C:\Users\LAJMI\OneDrive\ドキュメント\PES\Sem 6\CC\PEST1UG23CS327\CC Lab-2> locust -f locustfile.py
[2026-01-29 15:21:28,664] LAPTOP-NH#7KAUK/INFO:locust.main: Starting Locust 2.43.1
[2026-01-29 15:21:28,666] LAPTOP-NH#7KAUK/INFO:locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 15:24:51,254] LAPTOP-NH#7KAUK/INFO:locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:24:51,257] LAPTOP-NH#7KAUK/INFO:locust.runners: All users spawned: {"checkoutUser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\LAJMI\OneDrive\ドキュメント\PES\Sem 6\CC\PEST1UG23CS327\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
[2026-01-29 15:27:03,053] LAPTOP-NH#7KAUK/INFO:locust.main: Shutting down (exit code 0)
```

Type	Name	# reqs	# fails	Avg	Min	Max	Med	req/s	failures/s
GET	/checkout	19	0(0.00%)	114	6	2842	7	0.66	0.00
	Aggregated	19	0(0.00%)	114	6	2842	7	0.66	0.00

Response time percentiles (approximated)

Type	Name	50%	66%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%
GET	/checkout	7	7	7	9	12	2000	2000	2000	2000	2000	2000
	Aggregated	7	7	7	9	12	2000	2000	2000	2000	2000	2000

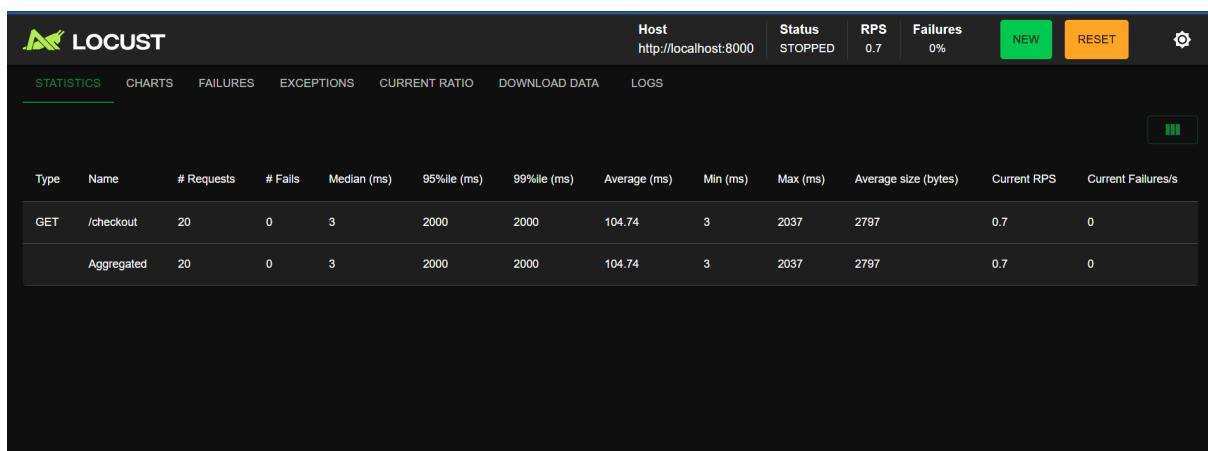
(.venv) PS C:\Users\LAJMI\OneDrive\ドキュメント\PES\Sem 6\CC\PEST1UG23CS327\CC Lab-2>

OUTLINE TIMELINE

Ln 10, Col 6 Spaces: 4 UTF-8 LF { } Python 🏛 3.12.4 Go Live 🎯 Prettier 🎭 ENG IN 03:27 PM 29-01-2026

27°C Sunny

SS5



```

PESTIUG23CS327
PS C:\Users\LAWINI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTIUG23CS327\CC Lab-2> cd "CC Lab-2\CC Lab-2"
PS C:\Users\LAWINI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTIUG23CS327\CC Lab-2> .\venv\Scripts\activate
(.venv) PS C:\Users\LAWINI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTIUG23CS327\CC Lab-2> locust -f locust/checkout_locustfile.py
>>>
[2026-01-29 15:39:46,881] LAPTOP-NHNTKAU/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 15:39:46,882] LAPTOP-NHNTKAU/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 15:40:53,964] LAPTOP-NHNTKAU/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:40:53,965] LAPTOP-NHNTKAU/INFO/locust.runners: All users spawned: {"CheckoutUser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\LAWINI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTIUG23CS327\CC Lab-2\CC Lab-2\venv\Lib\site-packages\gevent\ffl\loop.py", line 279, in python_check_callback
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
KeyboardInterrupt
2026-01-29T10:13:15Z
[2026-01-29 15:43:15,644] LAPTOP-NHNTKAU/INFO/locust.main: Shutting down (exit code 0)

```

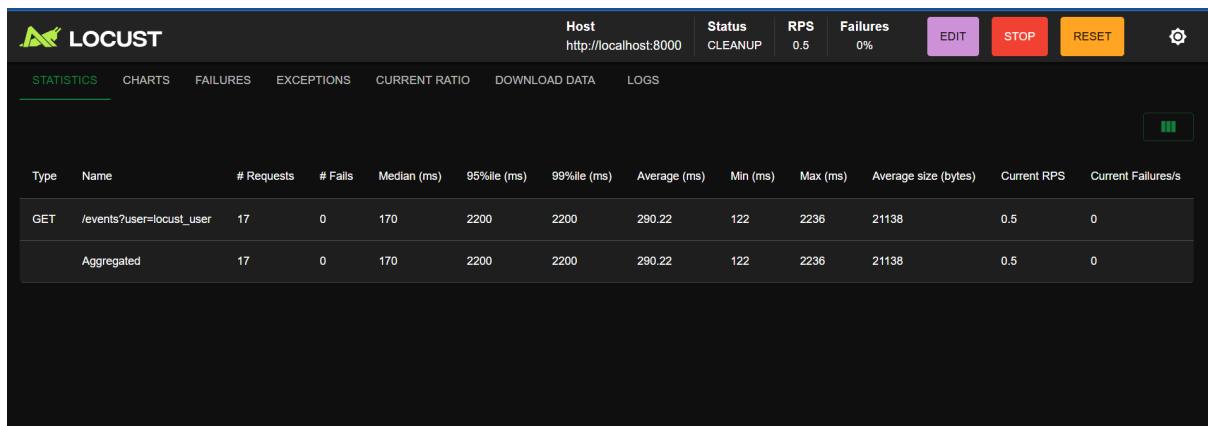
LOCUST

Type	Name	# reqs	# fails	Avg	Min	Max	Med	req/s	failures/s
GET	/checkout	20	0(0.00%)	104	2	2036	3	0.67	0.00
	Aggregated	20	0(0.00%)	104	2	2036	3	0.67	0.00

Response time percentiles (approximated)

Type	Name	50%	66%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%
GET	/checkout	3	3	3	4	4	2000	2000	2000	2000	2000	2000
	Aggregated	3	3	3	4	4	2000	2000	2000	2000	2000	2000
	20											

## SS6



File Edit Selection View Go Run Terminal Help ← → 🔍 PESTUG23CS327

OPEN EDITORS main.py \_init\_.py events\_locustfile.py

PESTUG23CS327 CC Lab-2 .pycache\_ database.cpython-312.pyc main.cpython-312.pyc venv checkout .pycache\_ \_init\_.py locust \_pycache\_ locust checkout\_locustfile.py events\_locustfile.py myevents\_locustfile.py templates database.py fest.db insert\_events.py main.py requirements.txt

problems 3 output DEBUG CONSOLE TERMINAL PORTS

powershell - CC Lab-2

```
PS C:\Users\LAWM\OneDrive\ドキュメント\PESTUG23CS327\CC Lab-2> locust -f locust/events_locustfile.py
>>>
[2026-01-29 15:50:31,624] LAPTOP-NH7KAUK/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 15:50:31,625] LAPTOP-NH7KAUK/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 15:50:59,480] LAPTOP-NH7KAUK/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 15:50:59,481] LAPTOP-NH7KAUK/INFO/locust.runners: All users spawned: {"eventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\LAWM\OneDrive\ドキュメント\PESTUG23CS327\CC Lab-2\venv\lib\site-packages\gevent\ffl\loop.py", line 279, in python_check_callback
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
KeyboardInterrupt
[2026-01-29 15:52:53,964] LAPTOP-NH7KAUK/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%)	290	121	2236	170	0.57	0.00
	Aggregated	17	0(0.00%)	290	121	2236	170	0.57	0.00

Response time percentiles (approximated)

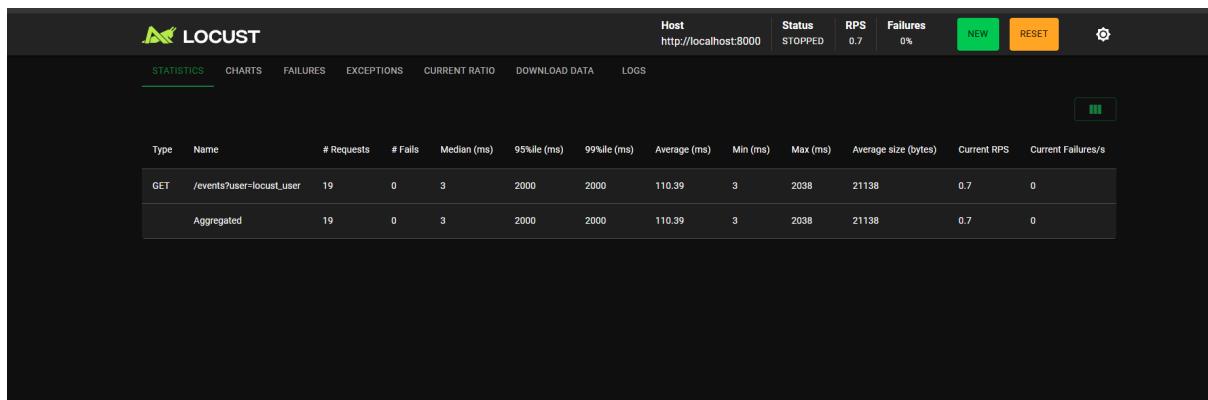
Type	Name	50%	66%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%
GET	/events?user=locust_user	170	190	190	190	250	2200	2200	2200	2200	2200	2200
	Aggregated	170	190	190	190	250	2200	2200	2200	2200	2200	2200

Ln 1, Col 1 Spaces: 4 UTF-8 3.12.4 Go Live Prettier

27°C Sunny

Search 🌐 📱 🎨 🎭 🎵 🎥 🎪 🎧 🎧 🎧 ENG IN 03:53 PM 29-01-2026

SS7



File Edit Selection View Go Run Terminal Help ← → Q PES1UG23CS327

OPEN EDITORS

- main.py CC Lab-2... 3
- \_init\_.py
- events\_locustfile.py

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

(.venv) PS C:\Users\LAWXI\OneDrive\ドキュメント\PES\Sem 6\CC\PES1UG23CS327\CC Lab-2\CC Lab-2> locust -f locust/events\_locustfile.py

```
>>>
[2026-01-29 16:01:52,913] LAPTOP-NH#7KAUK/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 16:01:52,913] LAPTOP-NH#7KAUK/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 16:02:09,686] LAPTOP-NH#7KAUK/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 per second
[2026-01-29 16:02:09,686] LAPTOP-NH#7KAUK/INFO/locust.runners: All users spawned: {"eventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\LAWXI\OneDrive\ドキュメント\PES\Sem 6\CC\PES1UG23CS327\CC Lab-2\CC Lab-2\venv\site-packages\gevent\ffl\loop.py", line 279, in python_check_callback
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
KeyboardInterrupt
[2026-01-29 16:03:20,888] LAPTOP-NH#7KAUK/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	19	0(0.00%)	110	2	2038	3	0.67	0.00
	Aggregated	19	0(0.00%)	110	2	2038	3	0.67	0.00

Response time percentiles (approximated)

# reqs	50%	66%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%
19	3	3	4	4	4	2000	2000	2000	2000	2000	2000
Aggregated	3	3	4	4	4	2000	2000	2000	2000	2000	2000
19	3	3	4	4	4	2000	2000	2000	2000	2000	2000

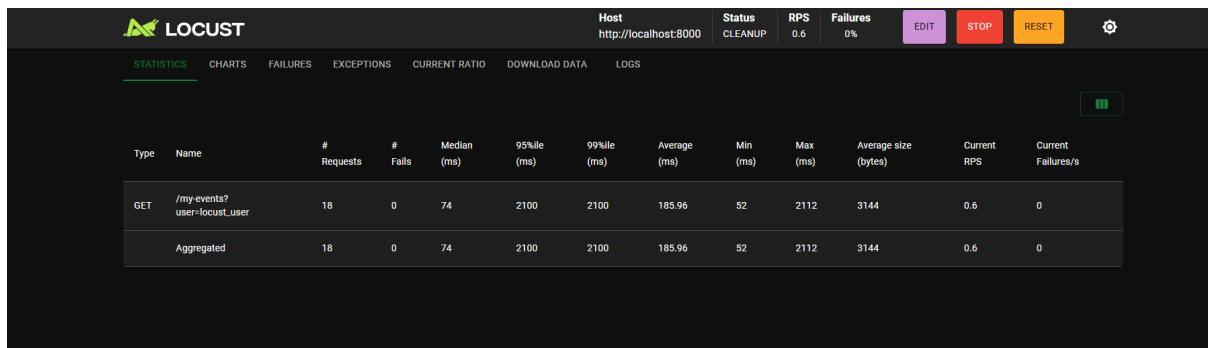
Host http://localhost:8000 Status CLEANUP 0.6 RPS 0 Failures 0% EDIT STOP RESET

OUTLINE > OUTLINE TIMELINE

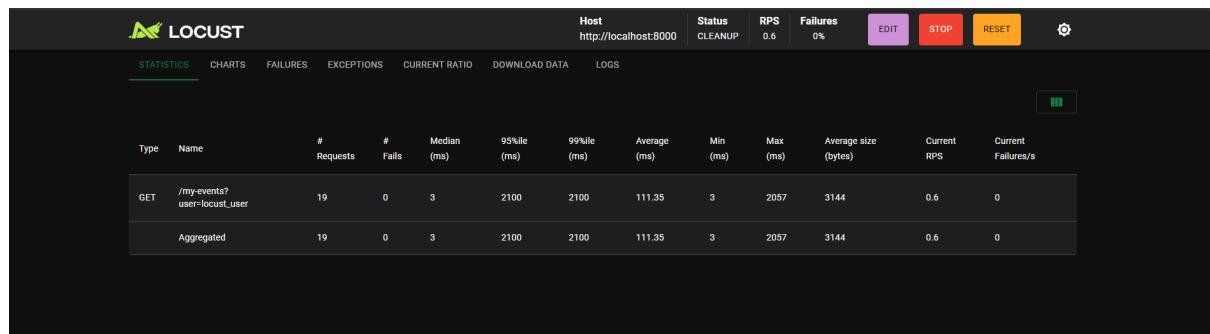
27°C Sunny

In 64, Col 1 Spaces: 4 UTF-8 LF {} Python ENG IN 04:03 PM 29-01-2026

## SS8



ss9



```
File Edit Selection View Go Run Terminal Help < > PES1UG23CS327
EXPLORER ... main.py 3 _init_.py events_locustfile.py
OPEN EDITORS CC Lab-2>CC Lab-2> main.py > my_events
main.py CC Lab-2>CC Lab-2> _init_.py CC Lab-2>CC Lab-2> events_locustfile.py CC ...
PESTUG23CS327 powershell - CC Lab-2 + - X
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Traceback (most recent call last):
  File "C:\Users\LAUMI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTUG23CS327\CC Lab-2\CC Lab-2\.venv\lib\site-packages\gevent\ffi\loop.py", line 279, in python_check_callback
    ck
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
(.venv) PS C:\Users\LAUMI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTUG23CS327\CC Lab-2> locust -f locust\myevents_locustfile.py
[2026-01-29 16:08:33,590] LAPTOP-NIN7KALK/INFO/locust.main: Starting Locust 2.43.1
[2026-01-29 16:08:33,590] LAPTOP-NIN7KALK/INFO/locust.main: Starting web interface at http://localhost:8089, press enter to open your default browser.
[2026-01-29 16:08:46,251] LAPTOP-NIN7KALK/INFO/locust.runners: Ramping to 1 users at a rate of 1.00 user per second
[2026-01-29 16:08:46,251] LAPTOP-NIN7KALK/INFO/locust.runners: All users spawned: {"myeventsuser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\LAUMI\OneDrive\ドキュメント\PES\Sem 6\CC\PESTUG23CS327\CC Lab-2\CC Lab-2\.venv\lib\site-packages\gevent\ffi\loop.py", line 279, in python_check_callback
    ck
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
KeyboardInterrupt
[2026-01-29 16:10:04,093] LAPTOP-NIN7KALK/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 19 0(0.00%) 111 2 2057 3 0.66 0.00
Aggregated 19 0(0.00%) 111 2 2057 3 0.66 0.00
Response time percentiles (approximated)
Type Name # reqs 50% 66% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100%
GET /my-events?user=locust_user 19 3 3 3 4 4 2100 2100 2100 2100 2100 2100
Aggregated 19 3 3 3 4 4 2100 2100 2100 2100 2100 2100
```

## **Short Expanation :**

### **Route 1: events**

#### **What was the bottleneck?**

The `events` route was fetching all event records and then filtering them in Python using loops. This caused unnecessary processing on every request and increased response time.

#### **What change did you make?**

The filtering logic was optimized by reducing Python-level loops and moving the filtering to a more efficient approach (single query / simplified logic). Redundant computations were removed.

#### **Why did the performance improve?**

By reducing looping and unnecessary data processing, the server performed less work per request, resulting in faster response times during load testing.

### **Route 2: my-events**

#### **What was the bottleneck?**

The `my-events` route used inefficient logic where multiple checks and iterations were performed to find events registered by a user, leading to repeated processing.

#### **What change did you make?**

The code was optimized to directly fetch only the required user-specific events, minimizing loops and avoiding repeated data access.

#### **Why did the performance improve?**

The optimized logic reduced the number of operations executed per request, lowering CPU usage and improving average response time under load.