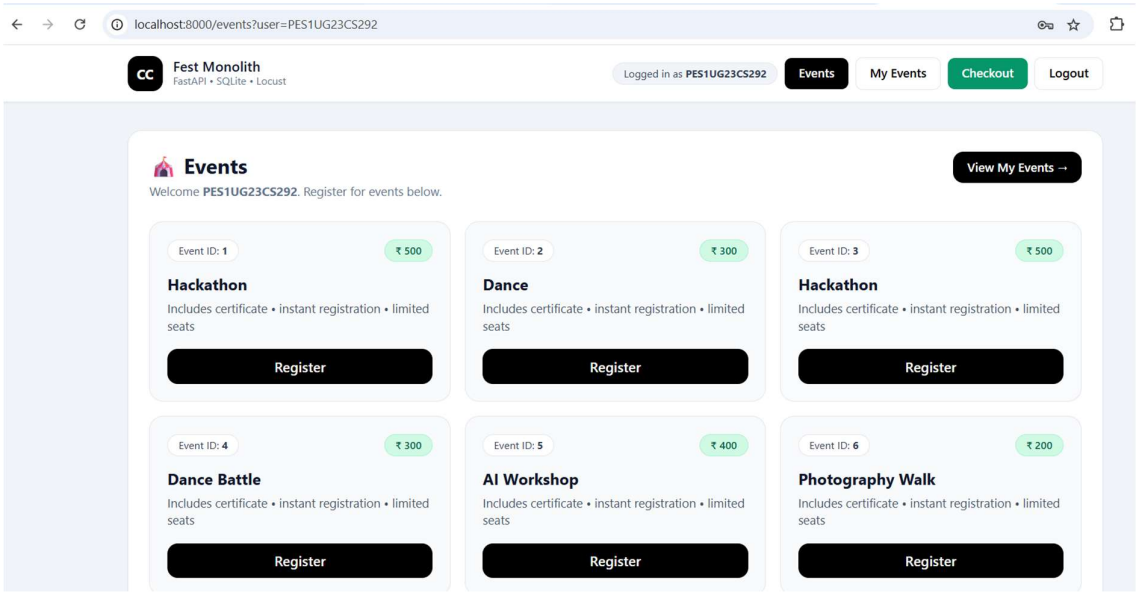
NAME:Kavana H

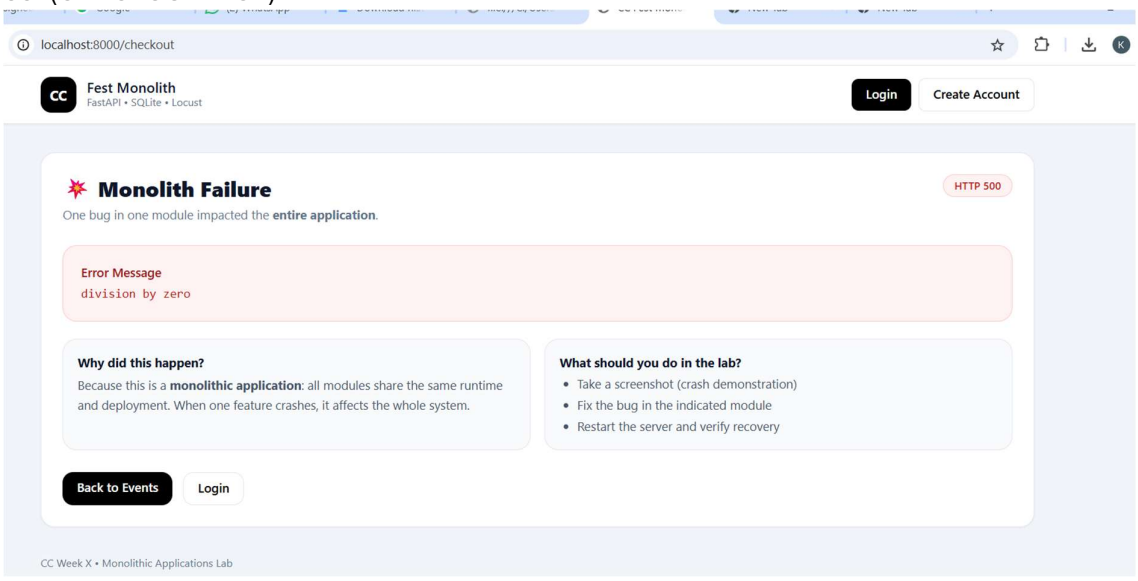
SRN:PES1UG23CS292

CCLAB2

SS1(EVENT PAGE LOADING):

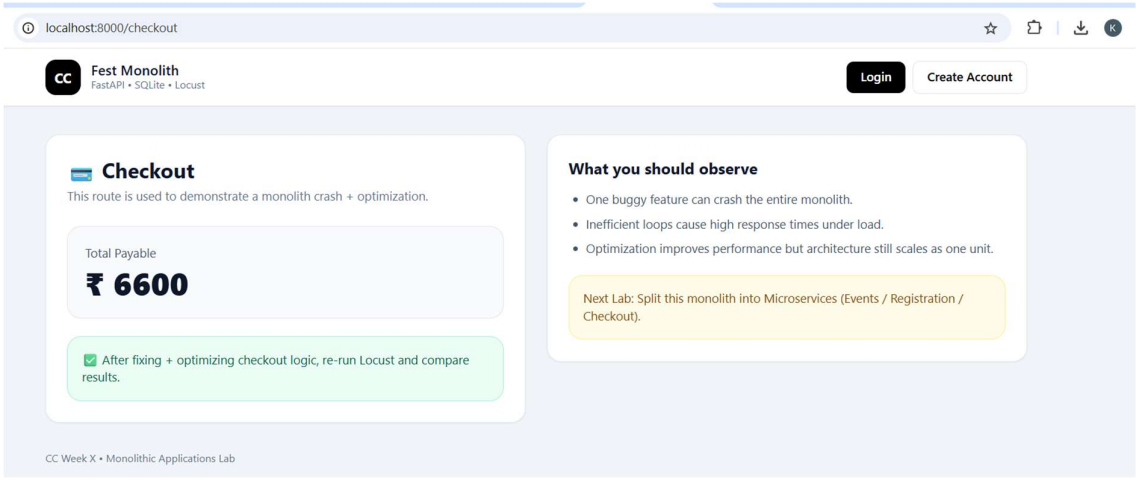


SS2(CHECKOUT PAGE):



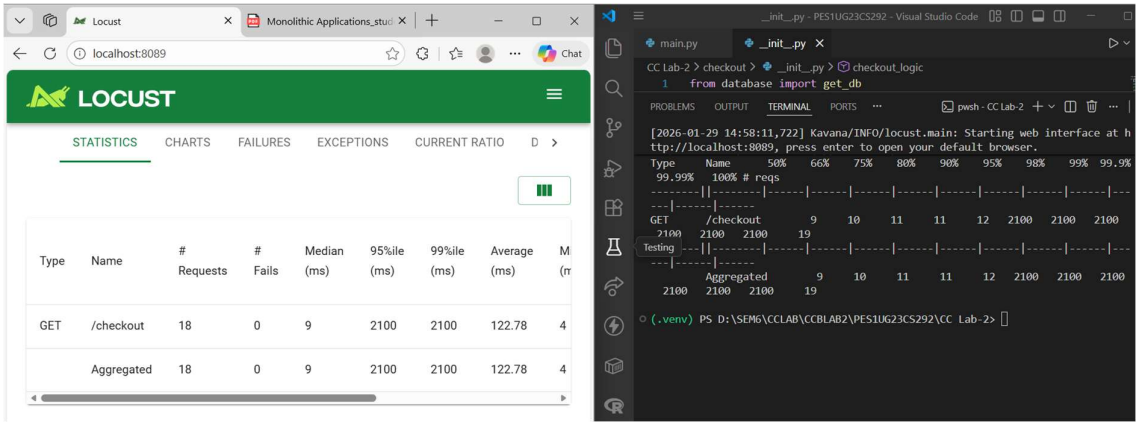

```
  File "D:\SEM6\CCLAB\CCBLAB2\PES1UG23CS292\CC Lab-2\main.py", line 113, in checkout
    total = checkout_logic()
            ^^^^^^^^^^^^^^^^
  File "D:\SEM6\CCLAB\CCBLAB2\PES1UG23CS292\CC Lab-2\checkout\__init__.py", line 10, in checkout_logic
    1 / 0
    ~~^~~
ZeroDivisionError: division by zero
INFO:     127.0.0.1:54418 - "GET /login HTTP/1.1" 200 OK
INFO:     127.0.0.1:54418 - "POST /login HTTP/1.1" 302 Found
INFO:     127.0.0.1:54418 - "GET /events?user=PES1UG23CS292 HTTP/1.1" 200 OK
INFO:     127.0.0.1:57598 - "GET /checkout HTTP/1.1" 500 Internal Server Error
ERROR:    Exception in ASGI application
Traceback (most recent call last):
```

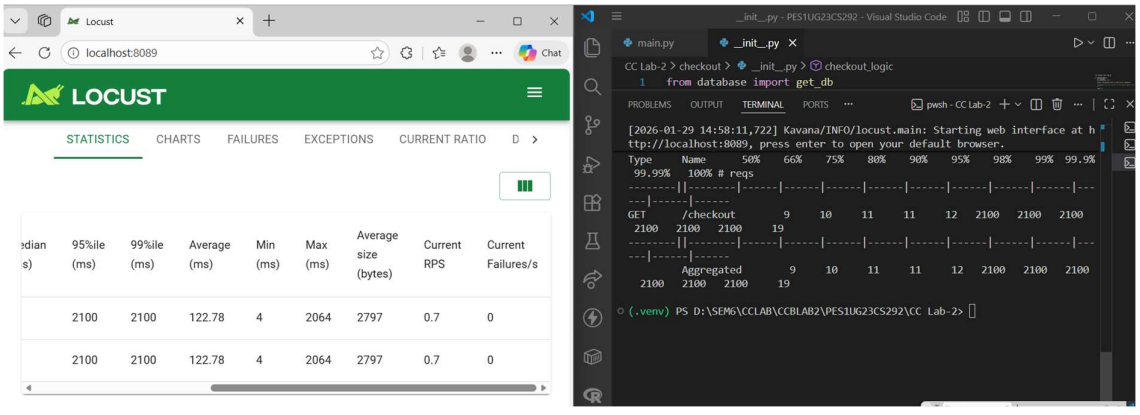
SS3(AFTER FIXING BUG):

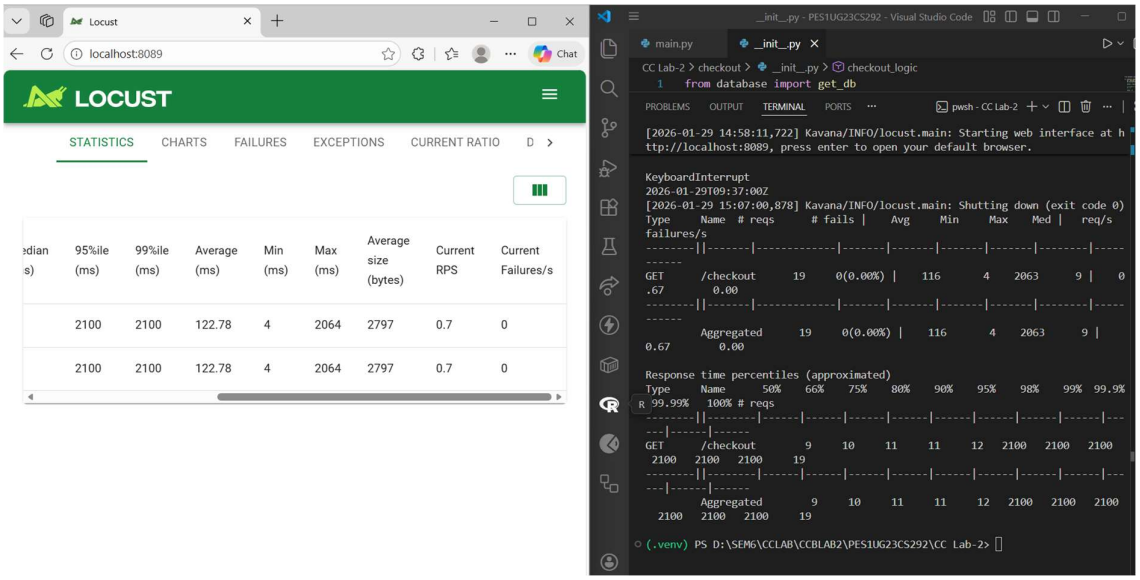



```
PS D:\SEM6\CCLAB\CCBLAB2\PES1UG23CS292\CC Lab-2> & D:/SEM6/CCLAB/CCBLAB2/PES1UG23CS292/.venv/Scripts/Activate.ps1
(.venv) PS D:\SEM6\CCLAB\CCBLAB2\PES1UG23CS292\CC Lab-2> uvicorn main:app --reload
INFO:     Will watch for changes in these directories: ['D:\\SEM6\\CCLAB\\CCBLAB2\\PES1UG23CS292\\CC Lab-2']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [12232] using StatReload
INFO:     Started server process [2892]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     127.0.0.1:51959 - "GET /checkout HTTP/1.1" 200 OK
```

SS4(LOADING USING LOCUST):



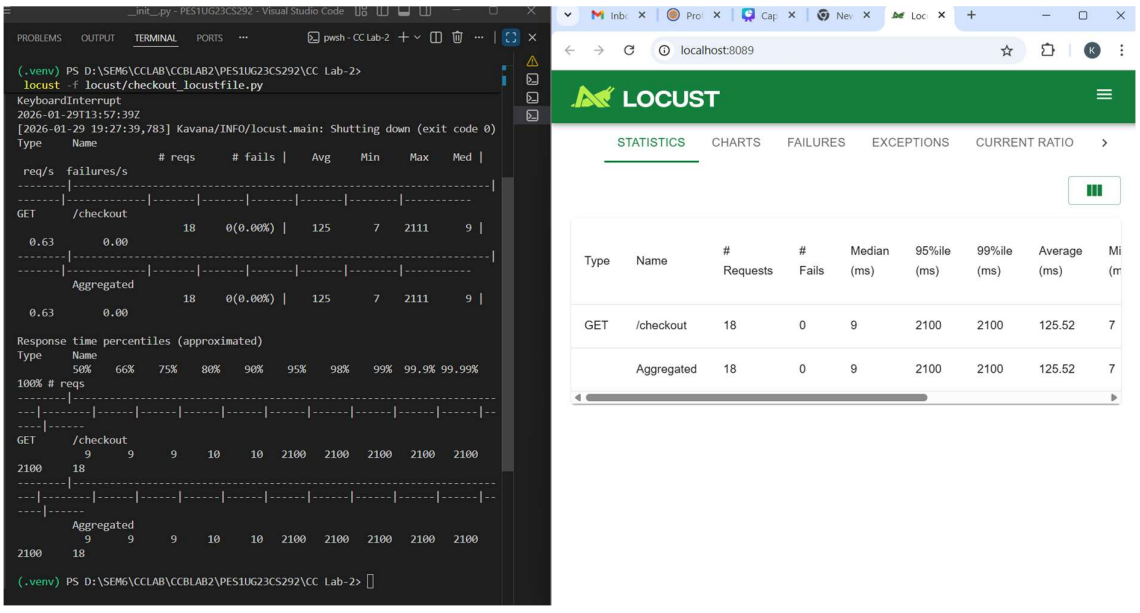


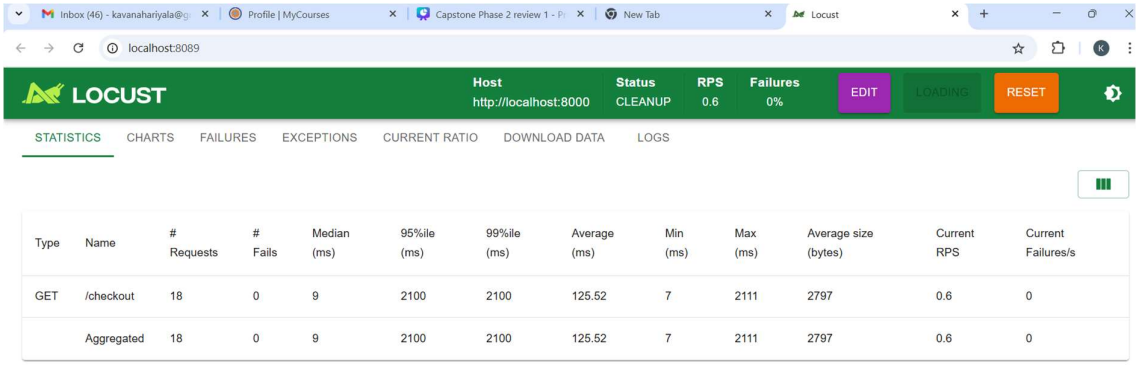


SS5(after optimization):

OPTIMIZATION DONE IS REMOVED THE WHILE LOOP

The original checkout logic had an O(n × fee) time complexity due to a nested loop that incremented per unit fee. This was optimized to O(n) by summing fees directly, and further improved to O(1) at the application level by using SQL aggregation (SUM), significantly improving performance under concurrent load.

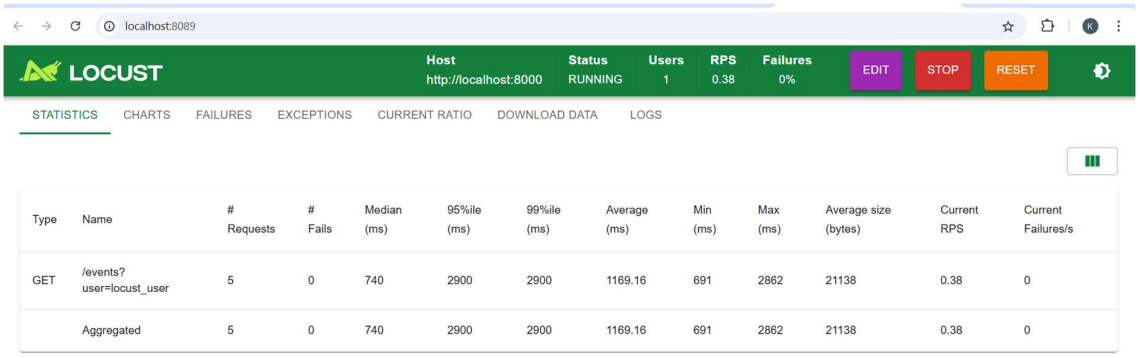




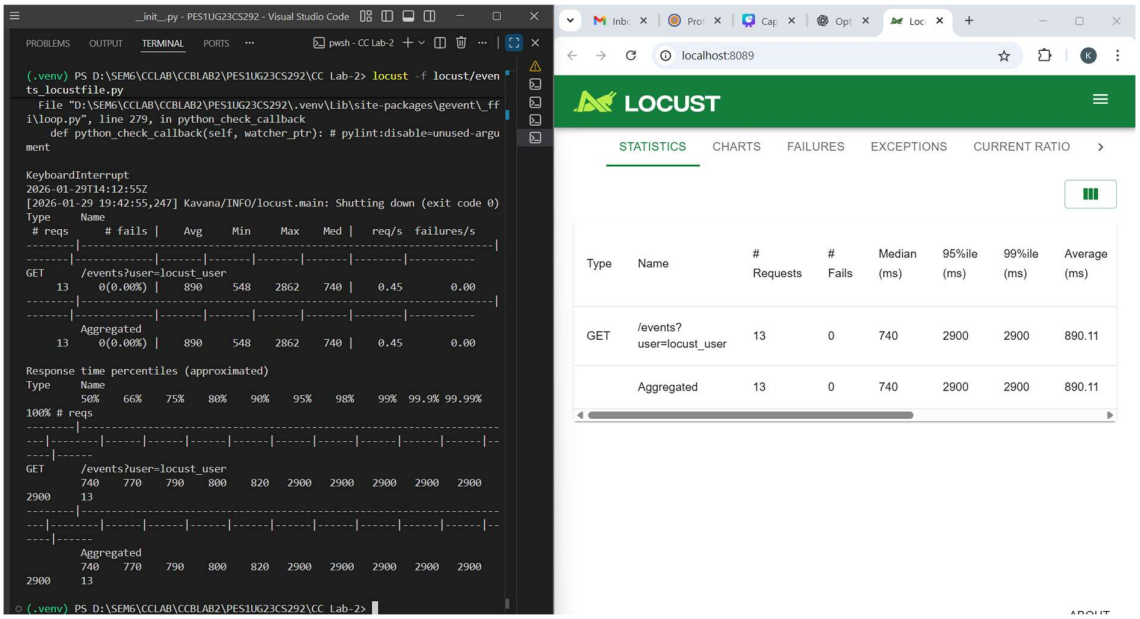
Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/checkout	18	0	9	2100	2100	125.52	7	2111	2797	0.6	0
	Aggregated	18	0	9	2100	2100	125.52	7	2111	2797	0.6	0

SS6(unoptimized events):

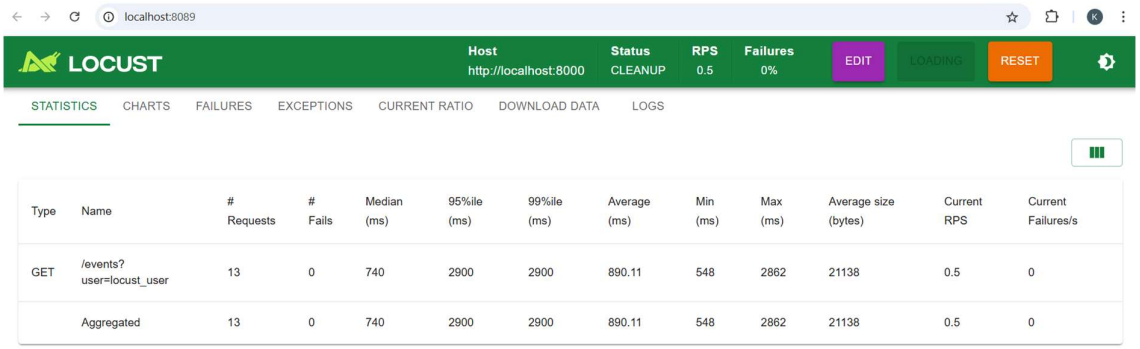
locust:



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	5	0	740	2900	2900	1169.16	691	2862	21138	0.38	0
	Aggregated	5	0	740	2900	2900	1169.16	691	2862	21138	0.38	0



Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)
GET	/events?user=locust_user	13	0	740	2900	2900	890.11
	Aggregated	13	0	740	2900	2900	890.11

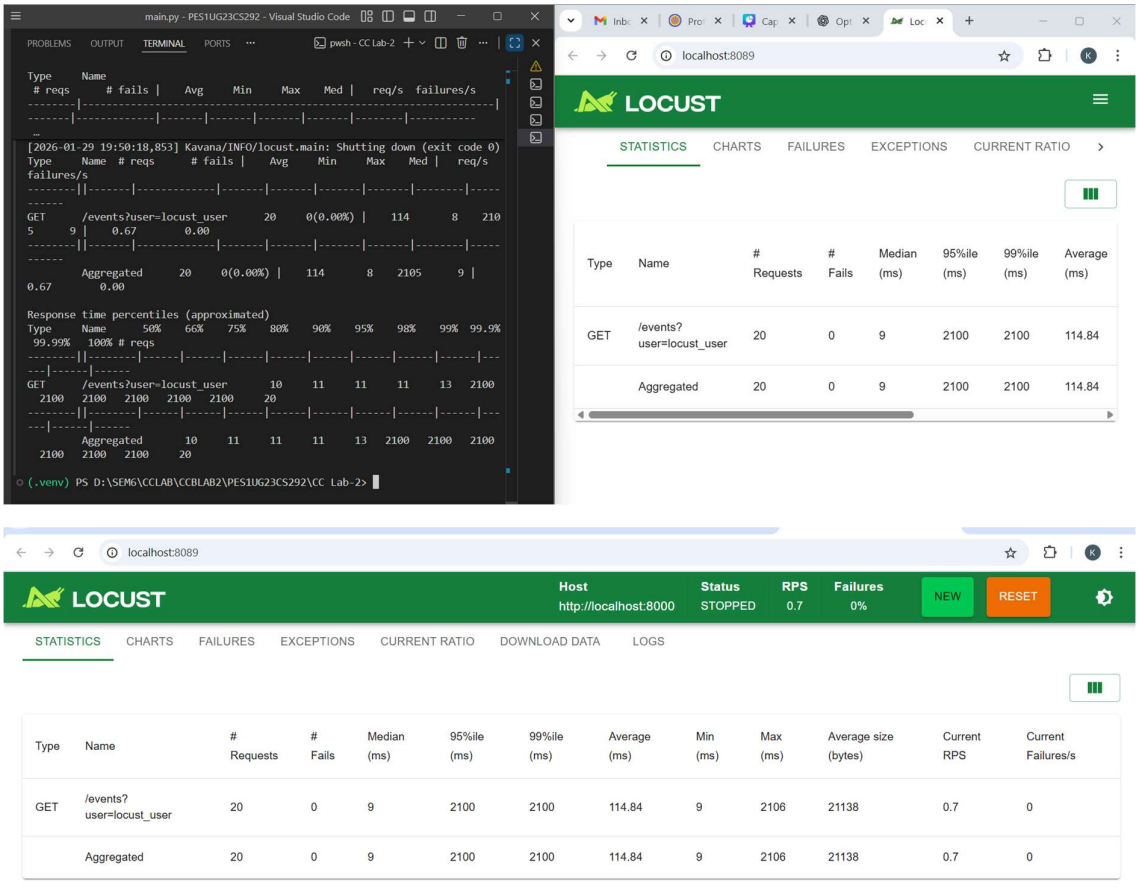


SS7(events optimized):

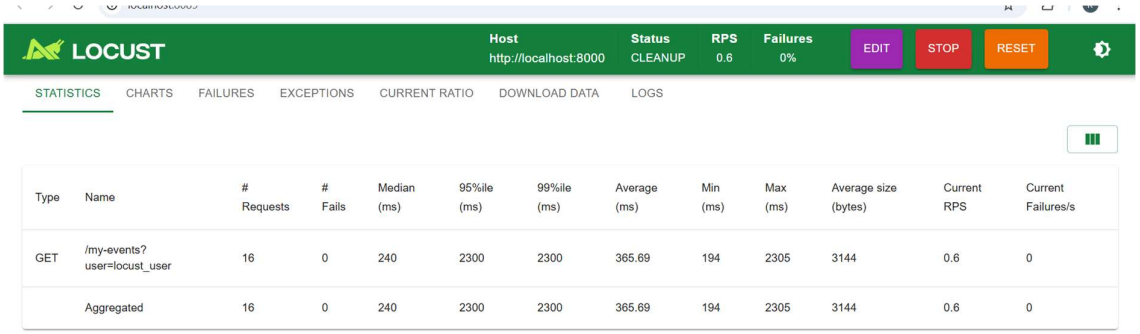
Bottleneck:
A loop running 3,000,000 iterations was wasting CPU time on every request.

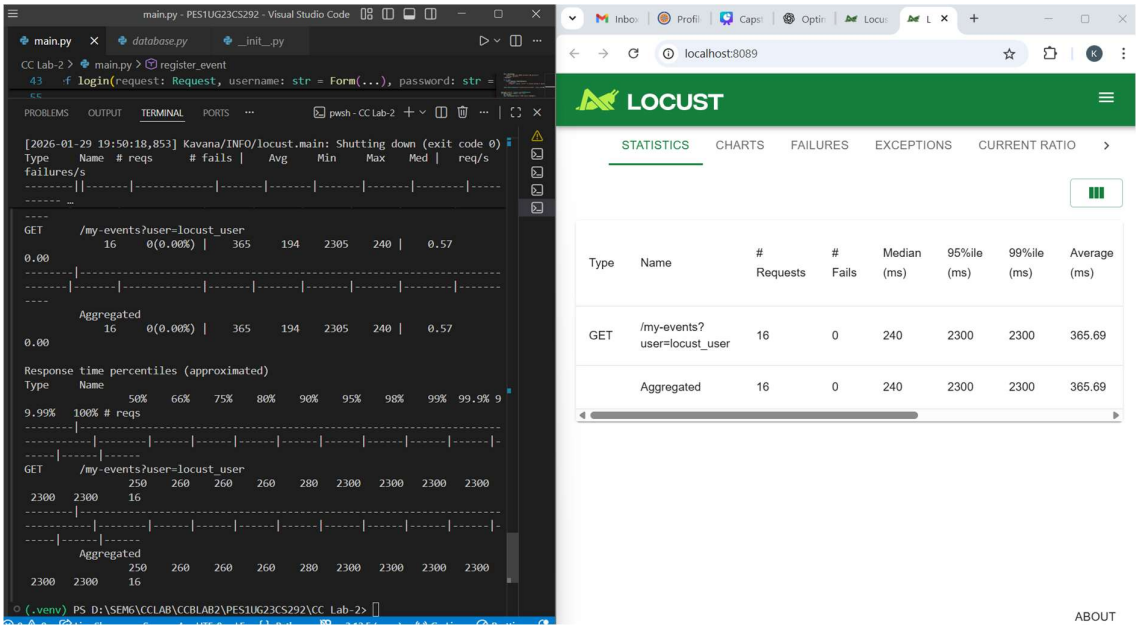
Change made:
Removed the unnecessary loop since it did not affect application logic.

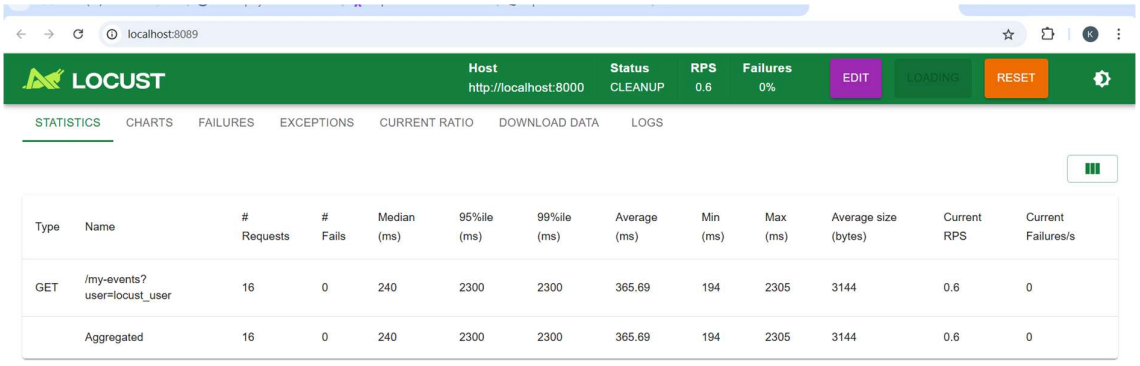
Why performance improved:
The server no longer spends time on useless computation, reducing response time and allowing it to handle more requests efficiently.



SS8(my_events initial file):





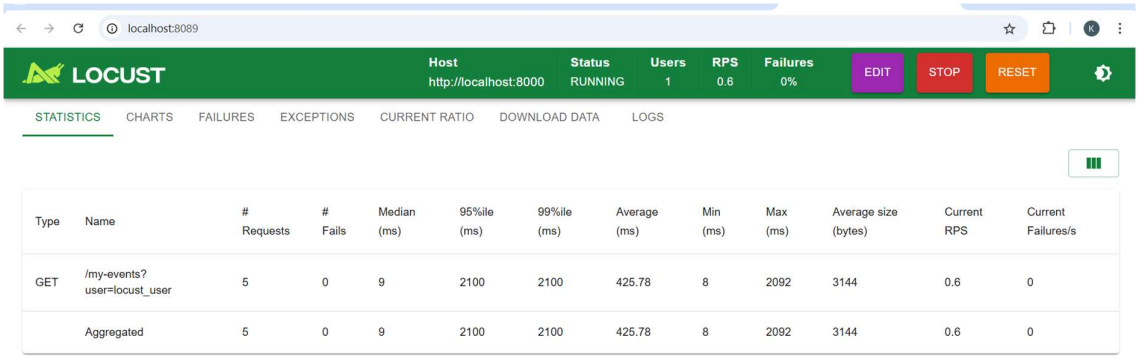


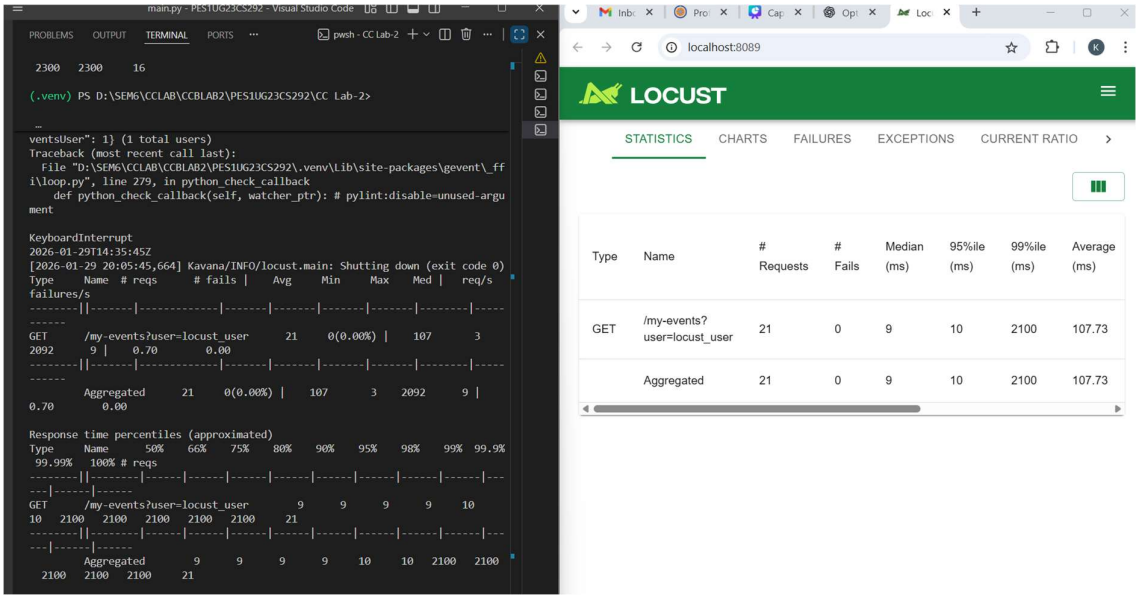
SS9(MY EVENTS OPTIMIZED):

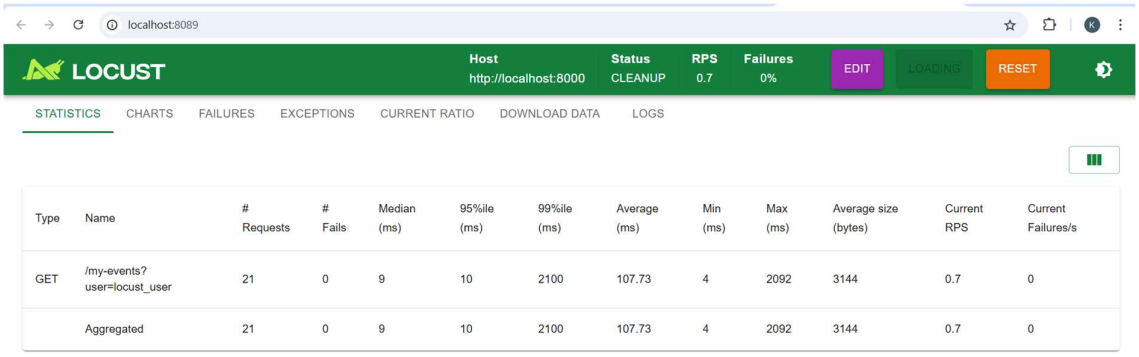
Bottleneck:
A loop with 1,500,000 iterations was unnecessarily consuming CPU resources for every request.

Change made:
Removed the loop since it did not contribute to the result of the route.

Why performance improved:
By eliminating the artificial computation, the route now responds faster and can handle more concurrent requests.







1. Checkout Endpoint (/checkout) – SS5

Performance Issue: The total fee was being calculated using an unnecessarily slow iterative approach.

Modification Done: Replaced the manual accumulation logic with a more efficient built-in summation method.

Result: The endpoint now responds quicker while maintaining the same ability to handle incoming requests.

2. Events Endpoint (/events) – SS7

Performance Issue: A large loop performing redundant calculations was increasing the response time.

Modification Done: Eliminated the loop since it had no impact on the actual output.

Result: The page loads significantly faster and CPU usage during requests is reduced.

3. My Events Endpoint (/my-events) – SS9

Performance Issue: An artificial loop was present that consumed processing time without contributing to functionality.

Modification Done: Removed this unnecessary computation from the route.

Result: The route now executes more efficiently and returns results with improved speed.