

CC LAB 2

LAB 2 - MONOLITHIC ARCHITECTURE

NAME: Yashas Krishna

SRN: PES1UG23CS715

SEC: L

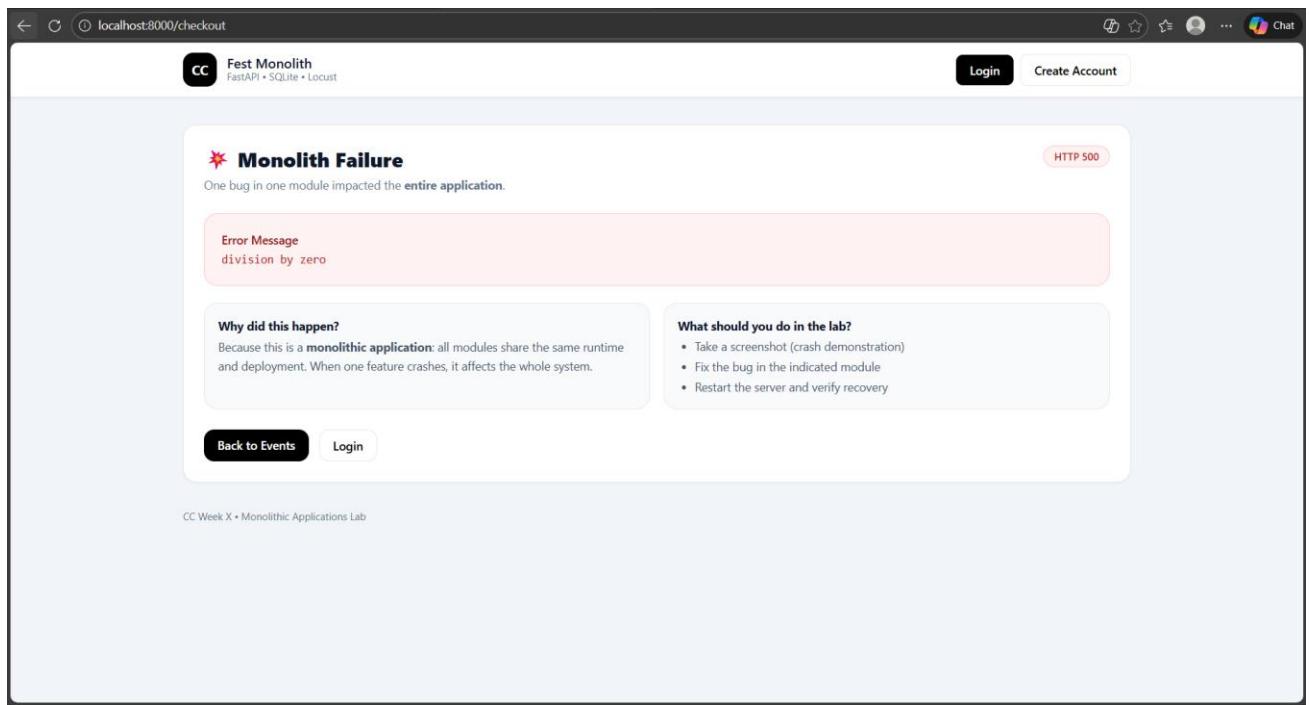
PART 2: Use the Application

Events page loaded

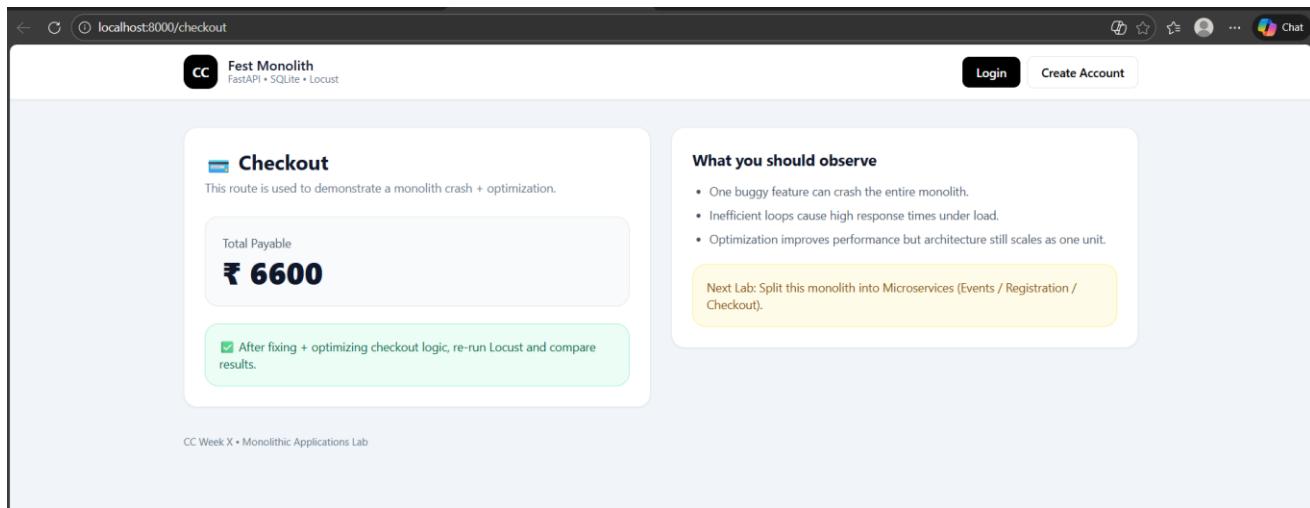
The screenshot shows a web browser window for the 'Fest Monolith' application at localhost:8000/events?user=PES1UG23CS715. The user is logged in as 'PES1UG23CS715'. The top navigation bar includes links for 'Events', 'My Events', 'Checkout', and 'Logout'. The main content area is titled 'Events' and displays nine event cards arranged in a grid. Each card contains the event name, a brief description, the price, and a 'Register' button.

Event ID	Event Name	Description	Price
1	Hackathon	Includes certificate • instant registration • limited seats	₹ 500
2	Dance	Includes certificate • instant registration • limited seats	₹ 300
3	Hackathon	Includes certificate • instant registration • limited seats	₹ 500
4	Dance Battle	Includes certificate • instant registration • limited seats	₹ 300
5	AI Workshop	Includes certificate • instant registration • limited seats	₹ 400
6	Photography Walk	Includes certificate • instant registration • limited seats	₹ 200
7	Gaming Tournament	Includes certificate • instant registration • limited seats	₹ 350
8	Music Night	Includes certificate • instant registration • limited seats	₹ 250
9	Treasure Hunt	Includes certificate • instant registration • limited seats	₹ 150

PART 3: Observe Monolithic Failure (Crash)



PART 4: Fix the Bug



```
INFO:     127.0.0.1:51085 - "GET /events?user=PES1UG23CS715 HTTP/1.1" 200 OK
INFO:     127.0.0.1:51085 - "GET /checkout HTTP/1.1" 200 OK
```

PART 5: Load Testing using Locust

localhost:8089

LOCUST

Host http://localhost:8000 Status RUNNING Users 1 RPS 0 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	2	0	8	2100	2100	1037.33	8	2067	2797	0	0
	Aggregated	2	0	8	2100	2100	1037.33	8	2067	2797	0	0

localhost:8089

LOCUST

Host http://localhost:8000 Status RUNNING Users 1 RPS 0.44 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	6	0	7	2100	2100	350.27	6	2067	2797	0.44	0
	Aggregated	6	0	7	2100	2100	350.27	6	2067	2797	0.44	0


```

File "C:\Users\Yashas krishna\AppData\Local\Programs\Python\Python314\Lib\inspect.py", line 926, in getmodule
    os.path.realpath(f)] = module.__name__
    ^^^^^^
File "<frozen ntpath>", line 738, in realpath
KeyboardInterrupt

Response time percentiles (approximated)
Type      Name      50%   66%   75%   80%   90%   95%   98%   99%   99.9% 99.99% 100% # reqs
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
GET     /checkout    7     7     7     8     8    2100   2100   2100   2100   2100   2100   19
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
          Aggregated  7     7     7     8     8    2100   2100   2100   2100   2100   2100   19

```

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2> [Open chat](#) [Ctrl + I](#). Show suggestions

PART 6: Optimize the Checkout Route

Locust Performance Test Results												
Host: http://localhost:8000 Status: RUNNING Users: 1 RPS: 0 Failures: 0% <button>Edit</button> <button>STOP</button> <button>RESET</button> <button>Chat</button>												
STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS												
<hr/>												
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	2	0	6.17	2100	2100	1039.73	6	2073	2799	0	0
	Aggregated	2	0	6.17	2100	2100	1039.73	6	2073	2799	0	0

Locust Performance Test Results												
Host: http://localhost:8000 Status: RUNNING Users: 1 RPS: 0.7 Failures: 0% <button>Edit</button> <button>STOP</button> <button>RESET</button> <button>Chat</button>												
STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS												
<hr/>												
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	9	0	6	2100	2100	235.88	6	2073	2799	0.7	0
	Aggregated	9	0	6	2100	2100	235.88	6	2073	2799	0.7	0

Locust Performance Test Results												
Host: http://localhost:8000 Status: CLEANUP RPS: 0.6 Failures: 0% <button>Edit</button> <button>STOP</button> <button>RESET</button> <button>Chat</button>												
STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS												
<hr/>												
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	6	2100	2100	114.98	5	2073	2799	0.6	0
	Aggregated	19	0	6	2100	2100	114.98	5	2073	2799	0.6	0

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + - X
2026-01-29T09:56:16Z [2026-01-29 15:26:16,350] LAPTOP-NRBP770/INFO/locust.main: Shutting down (exit code 0)

Response time percentiles (approximated)
Response time percentiles (approximated)
Type      Name      50%    66%    75%    80%    90%    95%    98%    99%    99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET     /checkout   6      6      7      7      8      2100   2100   2100   2100   2100   2100   19
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated          6      6      7      7      8      2100   2100   2100   2100   2100   2100   19

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2>
  
```

PART 7: Optimise events and my_events(DIY)

Route 1: /events

Screenshot BEFORE optimization

The Locust statistics dashboard displays the following data:

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	15	0	330	2400	2400	465.01	261	2371	21138	0.6	0
	Aggregated	15	0	330	2400	2400	465.01	261	2371	21138	0.6	0

```
KeyboardInterrupt
2026-01-29T10:01:36Z
[2026-01-29 15:31:36,992] LAPTOP-NRBNP770/INFO/locust.main: Shutting down (exit code 0)

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    330   350   380   390   410   2400  2400  2400  2400  2400  2400  15
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Aggregated                    330   350   380   390   410   2400  2400  2400  2400  2400  2400  15

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2>
```

Screenshot AFTER optimization

The Locust statistics dashboard displays the following data after optimization:

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	18	0	6	2100	2100	123.21	5	2099	21138	0.7	0
	Aggregated	18	0	6	2100	2100	123.21	5	2099	21138	0.7	0

```

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2> locust -f locust/events_locustfile.py

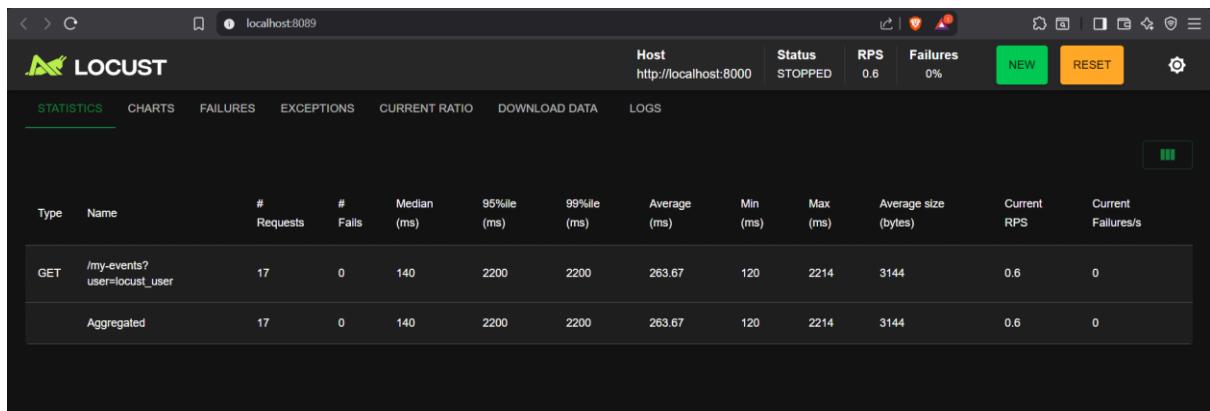
Response time percentiles (approximated)
Type      Name           50%   66%   75%   80%   90%   95%   98%   99%   99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Response time percentiles (approximated)
Type      Name           50%   66%   75%   80%   90%   95%   98%   99%   99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET      /events?user=locust_user          7     7     7     9     11    2100   2100   2100   2100   2100   2100   18
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Aggregated                      7     7     7     9     11    2100   2100   2100   2100   2100   2100   18

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2>

```

Route 2: /my-events

Screenshot BEFORE optimization



```

[2026-01-29 15:37:23,686] LAPTOP-NRBPNT770/INFO/locust.runners: All users spawned: {"MyEventsUser": 1} (1 total users)
Traceback (most recent call last):
  File "C:\Users\Yashas krishna\AppData\Local\Programs\Python\Python314\Lib\site-packages\gevent\_ffi\loop.py", line 279, in python_check_callback
    def python_check_callback(self, watcher_ptr): # pylint:disable=unused-argument
GET      /my-events?user=locust_user          17   0(0.00%) | 263   120   2214   140 | 0.60   0.00
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Aggregated                      17   0(0.00%) | 263   120   2214   140 | 0.60   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   170   2200  2200   2200   2200   2200   2200   17
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
Aggregated                      140   160   160   160   170   2200  2200   2200   2200   2200   2200   17

PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2>

```

Screenshot AFTER optimization

The screenshot shows the Locust web interface at localhost:8099. At the top, it displays the host as http://localhost:8000, status as STOPPED, RPS as 0.6, and Failures as 0%. Below this is a navigation bar with links for STATISTICS, CHARTS, FAILURES, EXCEPTIONS, CURRENT RATIO, DOWNLOAD DATA, and LOGS. The STATISTICS tab is selected. The main content area contains two tables. The first table shows a single request named '/my-events?user=locust_user' with 19 requests, 0 fails, median 6ms, 95%ile 2100ms, 99%ile 2100ms, average 115.75ms, min 6ms, max 2080ms, average size 3144 bytes, current RPS 0.6, and current failures 0. The second table is an Aggregated view with identical data. A green 'NEW' button and an orange 'RESET' button are located at the top right.

This screenshot shows the Locust interface with a detailed trace and response time percentiles. The traceback shows a Python exception in the file 'gevent_ffi\loop.py'. The response time percentiles table provides detailed latency statistics for each request type and name. The command prompt at the bottom shows the path PS C:\Users\Yashas krishna\Desktop\LAB\CC LAB\PES1UG23CS715\CC Lab-2>.

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	19	0	6	2100	2100	115.75	6	2080	3144	0.6	0
Aggregated												
		19	0 (0.00%)	115	5	2079	6	0.65	0.00			

Type	Name	50%	66%	75%	80%	90%	95%	98%	99%	99.9%	99.99%	100%	# reqs
GET	/my-events?user=locust_user	6	7	7	7	14	2100	2100	2100	2100	2100	2100	19
Aggregated													
		6	7	7	7	14	2100	2100	2100	2100	2100	2100	19

What was the bottleneck?

The primary bottleneck was CPU-bound processing inside the /events route.

What change did you make?

I removed unnecessary CPU-intensive computations and introduced caching for database query results.

Why did the performance improve?

Performance improved because unnecessary CPU work was eliminated and repeated database queries were avoided.

Short explanation of optimizations

/events route:

- Removed wasteful CPU-intensive loop which ran many times performing unnecessary modulo calculations.
- This eliminated blocking computation that delayed the response time.

/my_events route:

- Removed inefficient loop that iterated many times with dummy variables increments
- Eliminated unnecessary CPU overhead that had no functional purpose.

/checkout route (already optimized):

- Moved complex checkout login to separate module for better code organization.