

## Cloud Computing LAB 2

Name: Sathwik S K

SRN: PES2UG23CS538

SEC: I

Sem: 6

### LAB 2 - MONOLITHIC ARCHITECTURE

## PART 1: Setup & Run

```
● PS C:\Users\sathw\OneDrive\Documents\cc> mkdir PES2UG23CS538
```

```
Directory: C:\Users\sathw\OneDrive\Documents\cc
```

Mode	LastWriteTime	Length	Name
----	-----	-----	-----
d----	29-01-2026	14:16	PES2UG23CS538

```
● PS C:\Users\sathw\OneDrive\Documents\cc> cd PES2UG23CS538
```

```
○ PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538> █
```

```
Directory: C:\Users\sathw\OneDrive\Documents\cc
```

Mode	LastWriteTime	Length	Name
----	-----	-----	-----
d----	29-01-2026	14:16	PES2UG23CS538

```
● PS C:\Users\sathw\OneDrive\Documents\cc> cd PES2UG23CS538
```

```
PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538> python -m venv .venv
```

```
● >> .\venv\Scripts\activate
```

```
○ (.venv) PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538> █
```

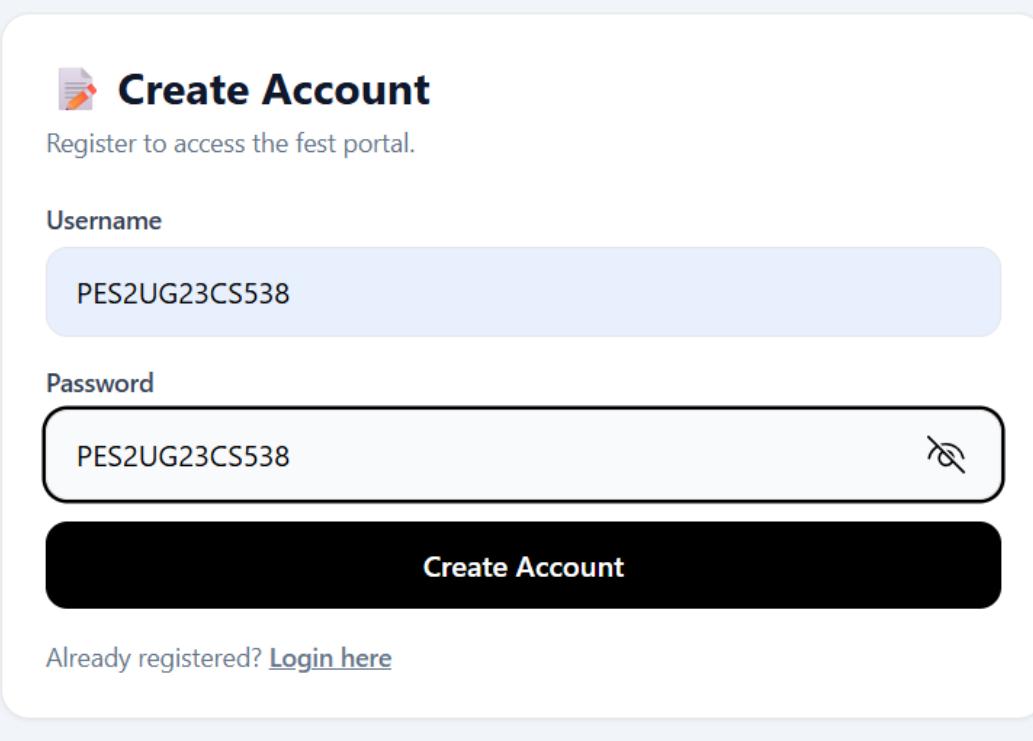
```
● (.venv) PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538> python.exe -m pip install --upgrade pip
Requirement already satisfied: pip in c:\users\sathw\onedrive\documents\cc\pes2ug23cs538\.venv\lib\site-packages (24.0)
Collecting pip
  Using cached pip-25.3-py3-none-any.whl.metadata (4.7 kB)
Using cached pip-25.3-py3-none-any.whl (1.8 MB)
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 24.0
    Uninstalling pip-24.0:
      Successfully uninstalled pip-24.0
```

```
● PS C:\Users\sathw\OneDrive\Documents\cc> cd C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2; pip install -r requirements.txt
Collecting fastapi (from -r requirements.txt (line 2))
  Downloading fastapi-0.128.0-py3-none-any.whl.metadata (30 kB)
Collecting uvicorn (from -r requirements.txt (line 3))
  Downloading uvicorn-0.40.0-py3-none-any.whl.metadata (6.7 kB)
Requirement already satisfied: jinja2 in c:\users\sathw\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from -r requirements.txt (line 4)) (3.1.6)
Collecting python-multipart (from -r requirements.txt (line 5))
  Downloading python_multipart-0.0.22-py3-none-any.whl.metadata (1 kB)
```

```
● PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2> python insert_events.py
 Events inserted successfully!
○ PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2>
```

```
PS C:\Users\sathw\OneDrive\Documents\cc> Set-Location C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2; python.exe -m uvicorn main:app --reload
INFO:     Will watch for changes in these directories: ['C:\\\\Users\\\\sathw\\\\OneDrive\\\\Documents\\\\cc\\\\PES2UG23CS538\\\\CC_Lab2']
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [7624] using StatReload
INFO:     Started server process [22780]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
```

## PART 2:Use the application



The image shows a "Create Account" form. At the top, there is a logo of a pencil writing on a piece of paper next to the text "Create Account". Below the logo, a sub-instruction says "Register to access the fest portal." The form has two input fields: "Username" and "Password". The "Username" field contains the value "PES2UG23CS538". The "Password" field also contains "PES2UG23CS538" and includes a small eye icon with a slash to its right, indicating it's a password field. Below the password field is a large black button with the white text "Create Account". At the bottom of the form, there is a link "Already registered? [Login here](#)".

## Login

Login to browse events, register, and checkout. This app is a **monolith**.

Username: PES2UG23C5538

Password:  [Reset](#)

**Login**

New user? [Create an account](#)

### Why FastAPI in this Monolith?

FastAPI is modern, cloud-friendly and supports `async` endpoints, type-hint based validation and auto docs. But this application is still a **monolith** since all modules run together in one deployment unit.

Optional: [Auto API docs: /docs](#)

The screenshot shows a web browser window with the URL `localhost:8000/events/home - PES2UG23C5538%20`. The page title is "Fast Monolith". The top navigation bar includes links for "Events", "My Events", "Checkout", and "Logout".

The main content area is titled "Events" and displays a grid of nine event cards:

- Event ID: 1** **Hackathon**: Includes certificate + instant registration + limited seats. **₹300**. **Register**
- Event ID: 2** **Dance**: Includes certificate + instant registration + limited seats. **₹300**. **Register**
- Event ID: 3** **Hackathon**: Includes certificate + instant registration + limited seats. **₹300**. **Register**
- Event ID: 4** **Dance Battle**: Includes certificate + instant registration + limited seats. **₹300**. **Register**
- Event ID: 5** **AI Workshop**: Includes certificate + instant registration + limited seats. **₹400**. **Register**
- Event ID: 6** **Photography Walk**: Includes certificate + instant registration + limited seats. **₹300**. **Register**
- Event ID: 7** **Gaming Tournament**: **₹300**. **Register**
- Event ID: 8** **Music Night**: **₹250**. **Register**
- Event ID: 9** **Treasure Hunt**: **₹300**. **Register**

A "View My Events" button is located in the top right corner of the card grid.

# PART 3: Observe Monolithic failure(crash)

The screenshot shows a browser window for the 'Fest Monolith' application at 'localhost:8000/checkout'. The page title is 'Monolith Failure'. It displays an error message: 'division by zero'. Below the message, it says 'Why did this happen?' and 'Because this is a **monolithic application**, all modules share the same codebase and deployment. When one feature crashes, it affects the whole system.' To the right, under 'What should you do in the lab?', there is a list: 'Take a screenshot (crash demonstration)', 'Fix the bug in the indicated module', and 'Restart the server and verify recovery'. At the bottom, there are 'Back to Events' and 'Login' buttons.

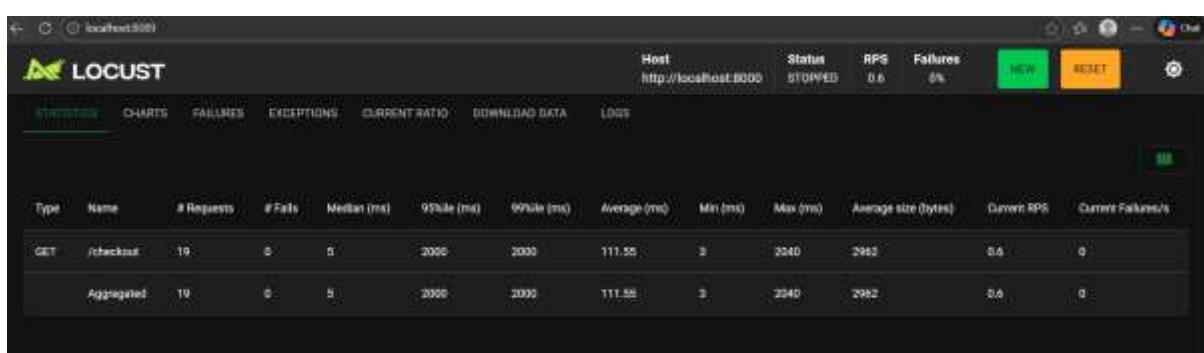
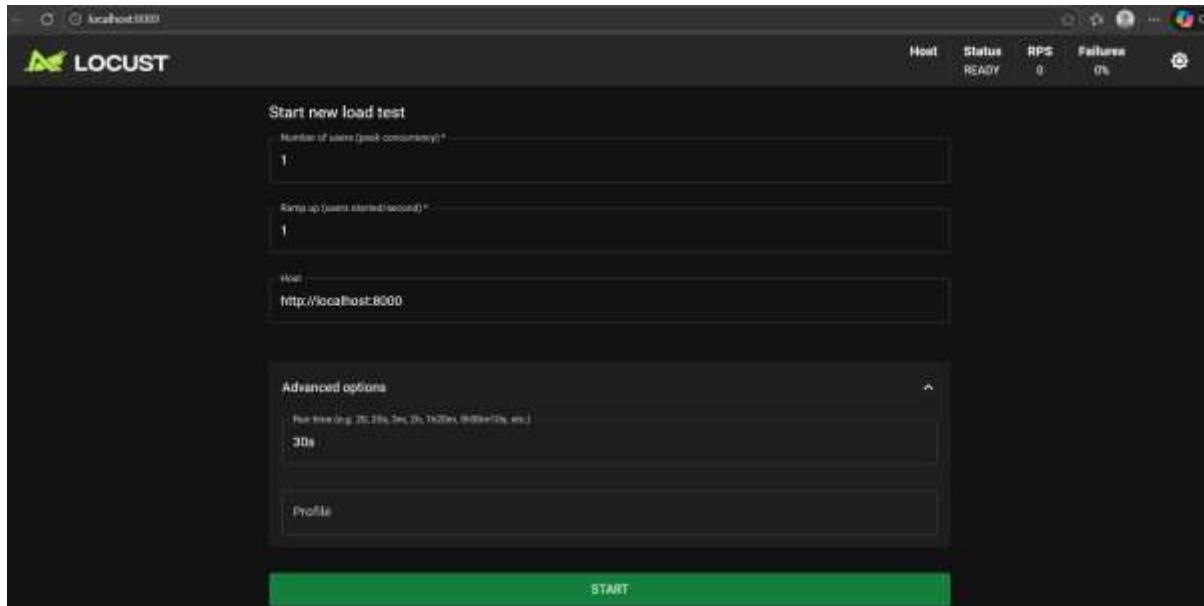
INFO: 127.0.0.1:9440 - "GET /checkout HTTP/1.1" 500 Internal Server Error  
ERROR: Exception in ASGI application

# PART 4: Fix the bug

The screenshot shows a browser window for the 'Fest Monolith' application at 'localhost:8000/checkout'. The page title is 'Checkout'. It displays a total payable amount of '₹ 6600'. A note below says: 'After fixing + optimizing checkout logic, re-run Checkout and compare results.' To the right, under 'What you should observe', there is a list: '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 says: 'Next Lab: Split this monolith into Microservices (Events / Registration / Checkout)'.

INFO: 127.0.0.1:4678 - "GET /checkout HTTP/1.1" 200 OK

# PART 5:Load Testing using Locust



```
Response time percentiles (approximated)
Type      Name      50%    66%    75%    80%    90%    95%    98%    99%    99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
GET      /checkout   8       8       9       9       10      13      22      48      41      41      41      189
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated                      8       8       9       9       10      13      22      48      41      41      41      189
PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2>
```

```
Response time percentiles (approximated)
Type      Name      50%    66%    75%    80%    90%    95%    98%
99% 99.9% 99.99% 100% # reqs
-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
GET      /checkout   8       8       9       9       10      13      22
40     41     41     41     189
-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
Aggregated                      8       8       9       9       10      13      22
40     41     41     41     189
PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2>
```

**LOCUST**

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

Type	Name	# Requests	# Fails	Median (ms)	50%ile (ms)	90%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures
GET	/checkout	38	0	9	14	16	9.15	7	16	2962	0.7	0
	Aggregated	38	0	8	14	15	9.15	7	16	2962	0.7	0

The chart displays response time percentiles (50%, 66%, 75%, 88%, 90%, 95%, 98%, 99%, 99.9%, 99.99%, 100%) across various request counts (42, 42, 188). The Y-axis represents time in milliseconds, and the X-axis represents the number of requests.

Request Type	Name	50%	66%	75%	88%	90%	95%	98%	99%	99.9%	99.99%	100%	# reqs
GET	/checkout	8	9	10	11	14	28	37	67	71	71	71	175
	Aggregated	8	9	10	11	14	28	37	67	71	71	71	175

PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC\_Lab2>

## PART 6:Optimise the checkout

**LOCUST**

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

Type	Name	# Requests	# Fails	Median (ms)	50%ile (ms)	90%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures
GET	/checkout	33	0	8	9	13	7.94	7	13	2962	0.7	0
	Aggregated	33	0	8	9	13	7.94	7	13	2962	0.7	0

```
PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC_Lab2> python.exe -m locust -f .\locust\checkout_locust.py --host=http://127.0.0.1:8000 --user-count=1 --spawn-rate=1 --run-time=10s --headless
```

Response time percentiles (approximated)

Type	Name	50%	66%	75%	88%	90%	95%	98%	99%	99.9%	99.99%	100%	# reqs
GET	/checkout	8	9	10	11	14	28	37	67	71	71	71	175
	Aggregated	8	9	10	11	14	28	37	67	71	71	71	175

PS C:\Users\sathw\OneDrive\Documents\cc\PES2UG23CS538\CC\_Lab2>

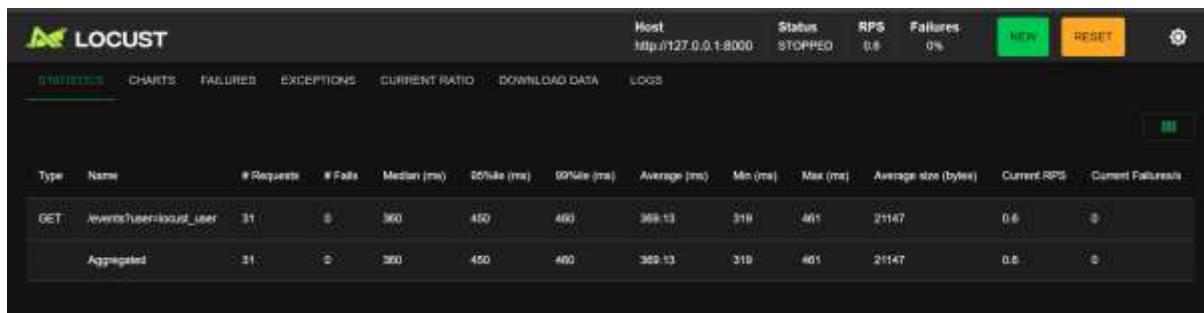
**LOCUST**

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

Type	Name	# Requests	# Fails	Median (ms)	50%ile (ms)	90%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures
GET	/checkout	65	0	8	12	18	8.21	6	13	2962	0.6	0
	Aggregated	65	0	8	12	18	8.21	6	13	2963	0.6	0

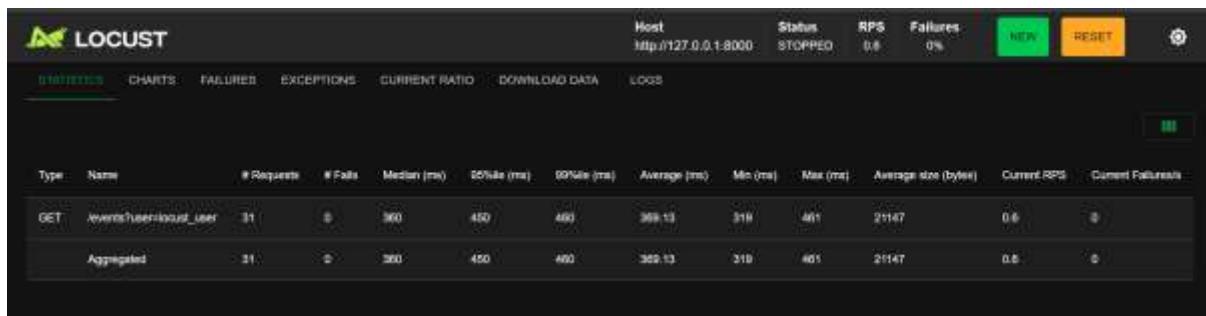
# PART 7: Optimise events and my\_events(DIY)

## Route 1: /events



## Route 2: /my-events





## Explanation

### What was the bottleneck?

**The bottleneck was the slowest resource limiting overall system performance, which was [e.g., disk I/O / CPU processing / network latency / single-threaded execution].**

**Because of this, other resources remained underutilized while waiting for this component to complete its task.**

### What change did you make?

- Optimized the algorithm / query
- Introduced parallel processing or multithreading
- Reduced disk access by caching data
- Replaced blocking operations with non-blocking/asynchronous calls
- Increased resource allocation (CPU, memory, bandwidth)

### Why did the performance improve?

**Performance improved because the change reduced waiting time and resource contention.**

**By eliminating or reducing the bottleneck, the system was able to:**

- Process tasks faster
- Utilize resources more efficiently

- Improve throughput and reduce response time

**As a result, overall system efficiency and execution speed increased.**

**Github link [https://github.com/PES2UG23CS538/cc\\_lab2.git](https://github.com/PES2UG23CS538/cc_lab2.git)**