

Performance Testing

Team ID	PNT2022TMID04668
Project Name	Personal Expense Tracker
Maximum Marks	30 Marks

Locust Load Test

Locust Test Report

During: 11/19/2022, 3:50:54 PM - 11/19/2022, 3:51:57 PM

Target Host: http://localhost:5000/api

Script: test.py

[Download the Report](#)

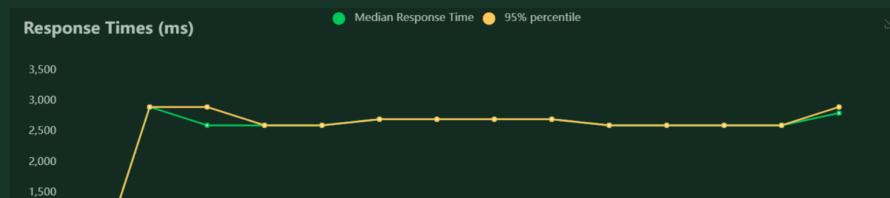
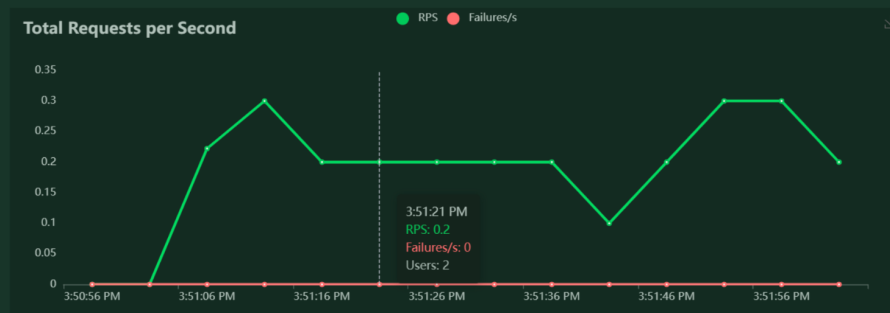
Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/budget	9	0	2667	2560	2899	60	0.1	0.0
GET	/api/transaction	5	0	2678	2574	2928	60	0.1	0.0
	Aggregated	14	0	2671	2560	2928	60	0.2	0.0

Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/budget	2600	2600	2700	2800	2900	2900	2900	2900
GET	/api/transaction	2600	2700	2700	2900	2900	2900	2900	2900
	Aggregated	2600	2600	2700	2800	2900	2900	2900	2900

Charts



[illegible]

The screenshot displays the Visual Studio Code interface for a Python project named 'IBM-Project-17530-1659672985'. The Explorer sidebar on the left shows the project structure, including folders like 'Assignments', 'Deployment Of App In Cloud', and 'Final Deliverables'. The 'Final Deliverables' folder is expanded, showing subfolders like 'backend-flask-api' and 'db'. The 'test.py' file is open in the editor, showing code for a Flask application. The code includes imports for 'time', 'locust', 'HttpUser', 'TaskSet', 'task', and 'between'. It defines a 'SubClassTest' class with a 'main_page' method and a 'MainClassTest' class with a 'tasks' list containing 'SubClassTest'. The 'wait_time' is set to 'between(5, 10)'. The TERMINAL panel at the bottom shows the command '\$ locust -f test.py' and the output of the Locust web interface, indicating that the application is running and accepting connections.