User Acceptance Testing (UAT) – Performance Monitoring

Date: 20-06-2025
Team ID: LTVIP2025TMID55453
Project Name: ShopMart: Your Digital Grocery Store Experience
Team Lead: Vanjarapu Suryavardhan
Team Members: V. Naga Venkata Pavan, V. Sai Viswanadh, Uppada Sagar

PERFORMANCE MONITORING

Performance monitoring is critical for maintaining the health and responsiveness of the **ShopMart** platform after deployment. It involves the continuous tracking of system metrics such as:

Server CPU usage

Memory consumption

Database performance

API response times

Error rates

The objective is to detect and resolve issues like **slow page loads**, **failed transactions**, or **unresponsive components** *before they affect end-users*.

By utilizing tools such as **Postman**, **Chrome DevTools**, and **MongoDB Atlas Profiler**, our team ensures real-time monitoring of system behavior. This proactive approach enables swift identification and remediation of issues while also allowing us to optimize performance under various usage scenarios—especially crucial during traffic spikes or peak business hours.

Key Metrics and Monitoring Tools

Metric	Description	Tools Used
Response Time	Time taken for APIs and pages to load	Postman, Chrome DevTools
Throughput	Number of requests handled per second	Apache JMeter
Error Rate	Percentage of failed API or system transactions	JMeter, Browser Logs
Server Load	Monitoring CPU and memory usage of application	Node.js Monitoring Tools (e.g., PM2, top, htop)
DB Performance	Query execution times, read/write latency	MongoDB Atlas Profiler

Optimization Plan

Area	Identified Issue	Optimization Strategy

Login API	Latency under high user load	Implement a load balancer to spread incoming traffic across multiple API nodes
Checkout	Slow database operations	Add MongoDB indexing; reduce nested queries to speed up response times
Static Assets	Initial load is slow on poor networks	Enable lazy loading and minify JS/CSS to reduce bundle size
Search	UI delay during real-time input	Add debounce logic; preload collections to reduce lag
Server	Memory/resource usage spikes	Use PM2 clustering and enable autoscaling to handle load more efficiently