Performance Test Design for Magento Test Website

Objective:  
To evaluate the scalability, responsiveness, and stability of the Magento eCommerce site under varying load conditions — ensuring smooth customer experience during peak usage.

# Key Areas to Test:

## Homepage Load Test

It’s the first touchpoint for every user; a slow homepage impacts bounce rate and user satisfaction.  
Goal: Ensure homepage loads under 2–3 seconds under normal and peak conditions.

## Product Search and Filter Functionality

A major part of user interaction; critical for conversion.  
Goal: Test response time when 50, 100, 500 virtual users perform search queries simultaneously.

## Add to Cart and Checkout Process

Cart operations directly impact revenue; a delay here leads to abandoned carts.  
Goal: Validate smooth performance when 100–300 concurrent users are adding products and proceeding to checkout.

## User Login and Account Page Access

Registered users will frequently log in to track orders or manage accounts.  
Goal: Ensure no bottleneck when hundreds of users log in at the same time.

## Order Submission (Place Order)

Crucial transaction point, system must handle stress while maintaining data accuracy.  
Goal: Ensure the backend handles 50–200 simultaneous order submissions without delays or errors.

# Test Types to Apply:

• Load Testing: Measure normal user volume handling capacity.  
• Stress Testing: Identify the website’s breaking point under extreme loads.  
• Spike Testing: Evaluate response to sudden traffic increase.  
• Soak Testing: Assess stability over an extended period under average load.

# Performance Parameters:

Response Time: < 3 seconds for static pages; < 5 seconds for checkout.

Throughput: Requests per second must scale with users.

Error Rate: Less than 1% during peak loads.

CPU & Memory Usage: Must remain under 80% to avoid server crash.

Concurrent Users Limit: Identify the max number of users the site can handle.

# Performance Testing Tools:

• Apache JMeter  
• K6.io  
• Gatling  
• BlazeMeter (Cloud JMeter)  
• Locust

# Conclusion:

I would prioritize the checkout, cart, and search modules for performance testing because these directly impact sales and user retention. Optimizing these would ensure both a better user experience and lower risk of server failures during traffic spikes like seasonal sales.