

Testes de Desempenho

Avaliando o comportamento de sistemas sob carga



Por que Testar Desempenho?

01

Evitar Falhas Críticas

Prevenir lentidão e indisponibilidade em momentos críticos para o negócio

02

Experiência do Usuário

Garantir resposta rápida e satisfação dos usuários em todas as situações

03

Detectar Gargalos

Identificar limitações em hardware, rede e banco de dados antes da produção

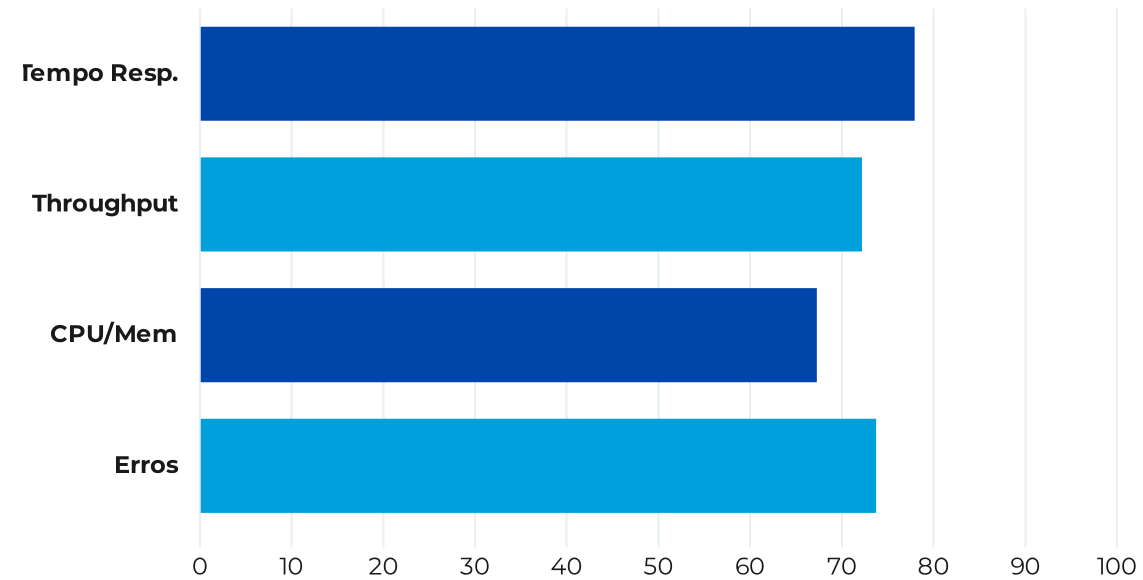
04

Preparar para Crescimento

Dimensionar infraestrutura para suportar aumento de demanda futura

Métricas Principais de Desempenho

Métrica	O que mede	Importância
Tempo de Resposta	Tempo para o sistema responder a uma requisição	Usabilidade
Throughput	Requisições processadas por segundo	Capacidade
CPU/Memória	Recursos consumidos durante execução	Eficiência
Erros/Segundo	Quantidade de falhas sob carga	Confiabilidade



Quatro Tipos de Testes de Desempenho

01

Teste de Carga

02

Teste de Estresse

03

Teste de Volume

04

Teste de Resistência

Teste de Carga: Validando Capacidade Normal

OBJETIVO

Avaliar o comportamento do sistema sob o número esperado de usuários ou requisições simultâneas

- **Condições Normais e Pico**

Mede desempenho em condições normais e de pico planejado

- **Identificar Gargalos**

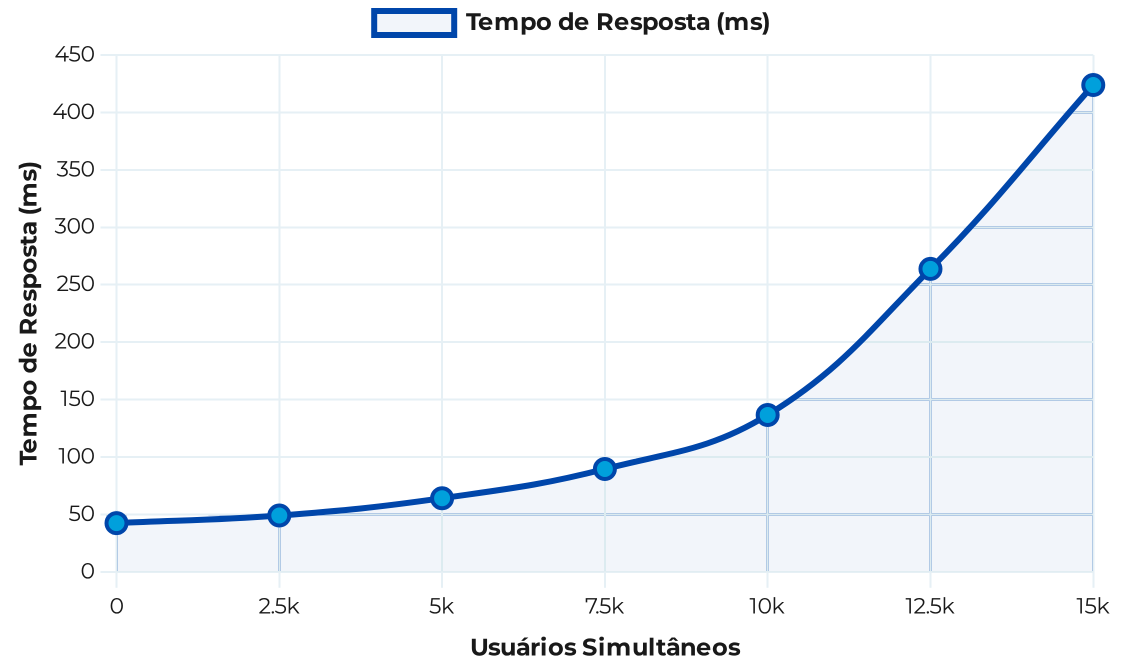
Identifica lentidão e gargalos dentro da capacidade prevista

EXEMPLO PRÁTICO

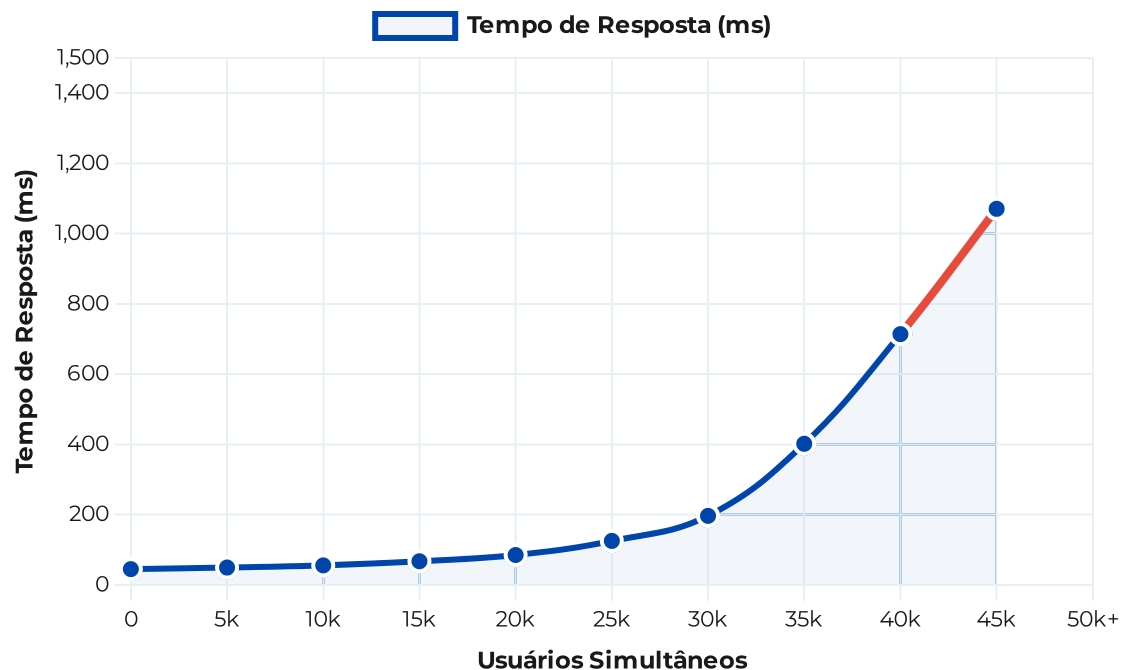
Simular 10.000 usuários simultâneos e verificar se o sistema mantém resposta rápida e estável

FERRAMENTAS TÍPICAS

JMeter • k6 • Locust • Gatling



Teste de Estresse: Encontrando o Limite



OBJETIVO

Verificar até onde o sistema suporta e como reage quando ultrapassa o limite esperado

EXPLICAÇÃO

Eleva progressivamente requisições até ocorrer falha
Mede ponto de ruptura e recuperação do sistema

EXEMPLO

Simular 20.000, 30.000, 50.000 usuários até o servidor falhar e medir recuperação

RESULTADOS ESPERADOS

Falhas controladas
Recuperação automática ou rápida

Teste de Volume: Desempenho com Grandes Dados

Avaliar o desempenho do sistema com grandes volumes de dados armazenados ou processados

EXPLICAÇÃO

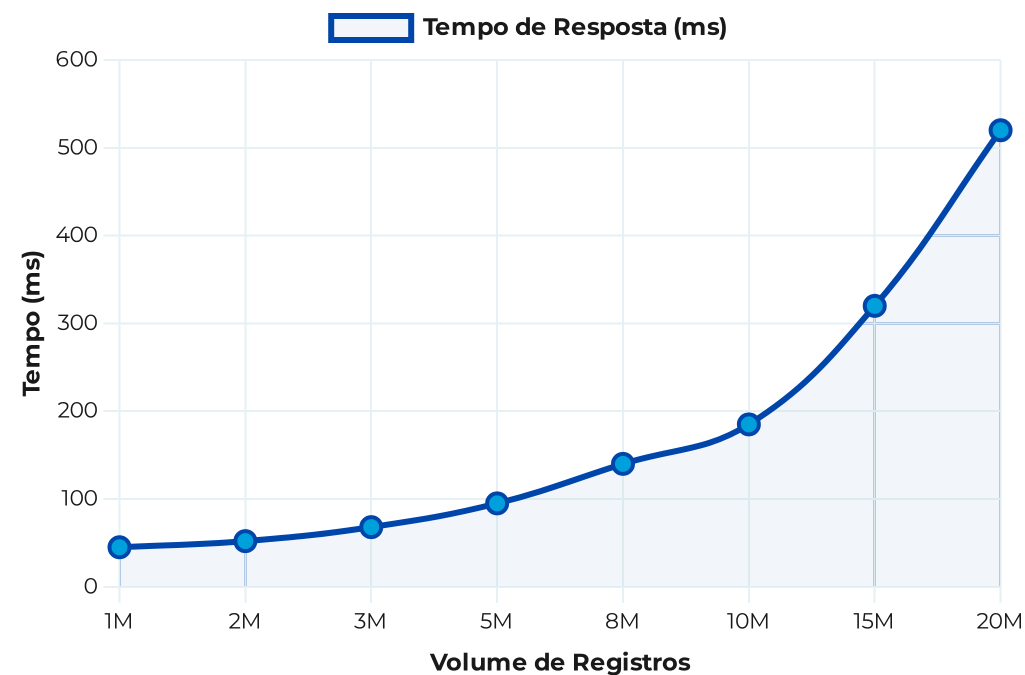
- Mede o impacto do tamanho do banco de dados no tempo de resposta
- Detecta lentidão em consultas e travamentos
- Identifica falta de otimização e índices inadequados

EXEMPLO PRÁTICO

Popular o banco com **10 milhões de registros** e medir tempo de geração de relatórios

FERRAMENTAS AUXILIARES

- Scripts SQL para geração de dados
- Mockaroo para dados realistas
- DBMonster para volume em massa



Teste de Resistência: Estabilidade Prolongada

OBJETIVO

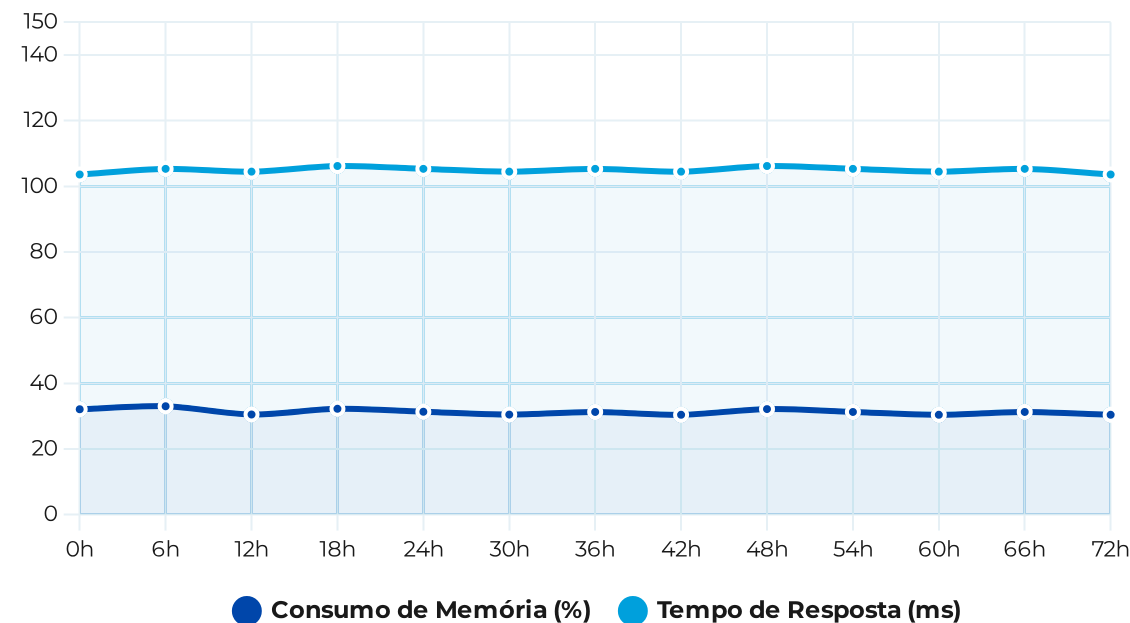
Verificar estabilidade do sistema ao longo do tempo, sob carga constante e contínua.

EXPLICAÇÃO

- Executa o sistema por horas ou dias simulando uso contínuo
- Detecta vazamentos de memória e lentidão progressiva
- Identifica falhas intermitentes que só aparecem com tempo

EXEMPLO PRÁTICO

Simular 5.000 usuários por 72 horas contínuas e acompanhar consumo de memória, CPU e estabilidade do sistema.



FERRAMENTAS TÍPICAS

JMeter (modo contínuo) • k6 • Grafana • Prometheus

Comparativo dos Quatro Tipos de Testes

TIPO DE TESTE	FOCO PRINCIPAL	DURAÇÃO	QUANDO APLICAR	EXEMPLO
Carga	Usuários esperados	Curto/Médio	Antes da entrega em produção	10.000 usuários simultâneos
Estresse	Limite máximo	Curto	Antes de períodos de picos esperados	50.000 usuários até falhar
Volume	Grande massa de dados	Médio	Após popular banco com dados reais	10 milhões de registros
Resistência	Estabilidade prolongada	Longo	Antes de operação contínua 24/7	Teste contínuo por 72 horas

Boas Práticas em Testes de Desempenho

01

Defina Metas Mensuráveis

Estabeleça métricas claras e objetivos quantificáveis para cada teste de desempenho.

02

Cenários Realistas

Crie cenários com dados reais e fluxos que representem uso efetivo do sistema.

03

Ambiente Similar

Teste em ambiente o mais semelhante possível à infraestrutura de produção.

04

Monitore Recursos

Acompanhe CPU, memória, rede e banco de dados durante toda execução dos testes.

05

Automatize e Integre

Integre testes com pipelines CI/CD para execução automática e contínua.

06

Documente Resultados

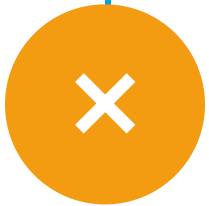
Registre e compare resultados com testes anteriores para rastrear evolução.

Caso Real: Falha em Sistema de Matrículas (2014)



O Problema

O sistema de matrículas de uma universidade **nunca foi testado** com mais de **1.000 acessos simultâneos**. No dia da matrícula, **15.000 alunos** tentaram acessar ao mesmo tempo.



As Consequências

Travamento do banco de dados sob a carga inesperada

Perda de sessões e dados de matrícula

Sistema fora do ar por 2 dias consecutivos

Insatisfação massiva de alunos e reputação abalada



A Solução

Após implementar **testes de desempenho rigorosos**, o sistema foi ampliado para suportar **20.000 usuários simultâneos** com estabilidade garantida.

Resumo: Desempenho é Requisito de Qualidade

Tipo de Teste	Objetivo Principal	Exemplo Prático
	Validar desempenho com usuários esperados em condições normais e pico planejado	10.000 usuários simultâneos
Teste de Estresse	Identificar limite máximo do sistema e comportamento sob sobrecarga	50.000 usuários até falhar
Teste de Volume	Testar desempenho com grande quantidade de dados armazenados	10 milhões de registros
Teste de Resistência	Avaliar estabilidade do sistema sob carga constante e prolongada	Teste contínuo por 72 horas

"Desempenho não é opcional — é um requisito de qualidade"

Testes de desempenho são essenciais para garantir sistemas confiáveis, escaláveis e satisfatórios para os usuários