



calang.io
empresa júnior

Teste de Performance

Ministrante: MSc. Misael Costa Júnior

I Workshop Calang.io

O que é Teste de Performance?



UNIVERSIDADE
FEDERAL DO CARIRI



Agenda

01 Contextualização

Entendendo o que são requisitos não funcionais, o que os diferenciam de requisitos funcionais e como eles são classificados.

02 Performance

O que é o Requisito de Performance e como testá-lo em aplicações?

03 Ferramentas

Quais as principais ferramentas disponíveis para conduzir Testes de Performance?

04 Prática

Aprendendo como implementar Testes de Performance a partir das aplicações disponíveis

05 Erros mais comuns

Quais os erros mais comuns que Analistas de QA e Desenvolvedores cometem ao conduzir Testes de Performance?

Quem sou eu?

Misael Costa Júnior



- Bacharel em Ciência da Computação pela Universidade Estadual do Piauí (UESPI)
- Mestre e Doutorando em Ciência da Computação e Matemática Computacional no ICMC-USP
- Engenheiro de Garantia de Qualidade (QA) Sênior
- Certificado CTFL ISTQB (Certified Tester Foundation Level)
- Tem experiência nos seguintes temas: Engenharia de Software, Teste de Software, Teste de Software automatizado, Testes de API e Testes Não Funcionais (Segurança de Apps Android e Performance).





Contextualização

Entendendo o que são requisitos não funcionais, o que os diferenciam de requisitos funcionais e como eles são classificados.

Alguns incidentes reais

- [Amazon Prime Day de 2018](#) (Performance)
- [Lançamento do Obamacare em 2013](#) (Performance)
- [Instabilidade no app do Banco do Brasil](#) (Performance)
- [Vazamento de dados do Yahoo em 2013 e 2014](#) (Segurança)
- [Ataque ao site da Sony Pictures em 2014](#) (Segurança)
- [Vazamento de dados do Facebook em 2018](#) (Segurança)

■ ■ ■ ■ O que são Requisitos Não Funcionais (RNFs)?

“São *atributos ou características* do sistema que não estão diretamente relacionados com as *funcionalidades específicas* que o sistema deve realizar, mas sim com as *suas características de performance, segurança, usabilidade, etc.*”

Funcionais x Não Funcionais

Requisitos Funcionais:

Requisitos Não Funcionais:

Funcionais x Não Funcionais

Requisitos Funcionais:

“Especificam o que um sistema de software deve fazer para resolver as necessidades das partes interessadas”

Requisitos Não Funcionais:

Funcionais x Não Funcionais

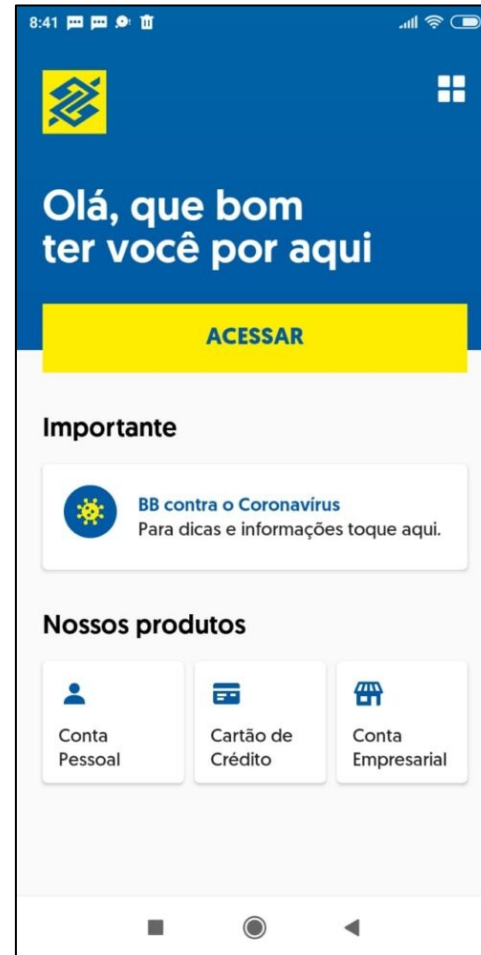
Requisitos Funcionais:

“Especificam o que um sistema de software deve fazer para resolver as necessidades das partes interessadas”

Requisitos Não Funcionais:

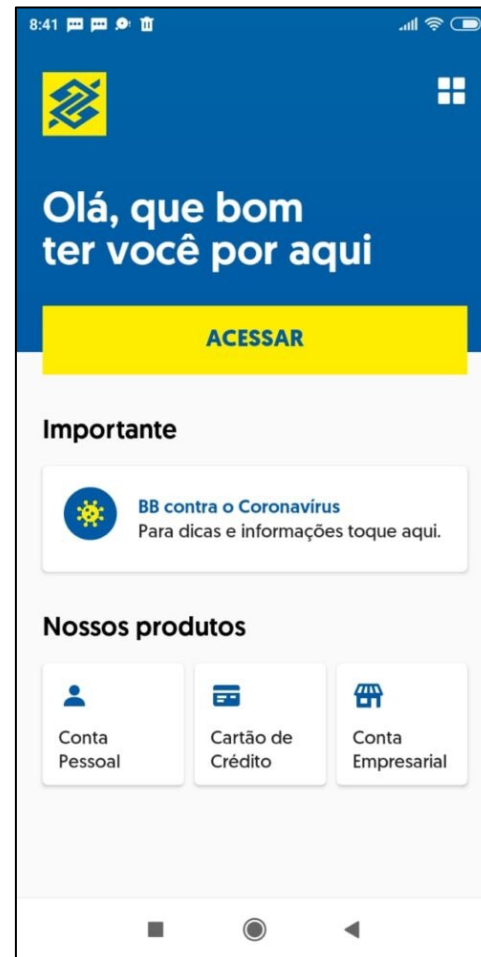
“Definem como o sistema deve fazer algo, em relação as características que o sistema deve ter para garantir o seu desempenho, qualidade, segurança, usabilidade, entre outros aspectos.”

Funcionais x Não Funcionais



Funcionais x Não Funcionais

- O aplicativo deve permitir que o usuário consulte seu saldo bancário;
- O aplicativo deve permitir que o usuário transfira fundos para outras contas bancárias, dentro ou fora do banco;
- O aplicativo deve permitir que o usuário visualize seu histórico de transações e transações pendentes.



Funcionais x Não Funcionais

- O aplicativo deve permitir que o usuário **consulte seu saldo bancário**;
- O aplicativo deve permitir que o usuário **transfira fundos** para outras contas bancárias, dentro ou fora do banco;
- O aplicativo deve permitir que o usuário **visualize seu histórico de transações e transações pendentes**.



- O aplicativo deve ser **fácil de usar e intuitivo**, com uma interface de usuário amigável (**Usabilidade**);
- O aplicativo deve ser **responsivo e rápido**, permitindo que o usuário execute transações com rapidez e eficiência (**Performance**);
- O aplicativo deve ter **recursos de segurança avançados** para proteger as informações do usuário e garantir a integridade dos dados (**Segurança**).



Pontos importantes sobre RNFs





Pontos importantes sobre RNFs



RNFs indicados como mais críticos: **Segurança**, **Performance** e **Usabilidade**.





Pontos importantes sobre RNFs



RNFs são negligenciados. Funcionalidade > Qualidade





Pontos importantes sobre RNFs



Não há **tempo de projeto** para avaliar propriedades de qualidade da aplicação





Pontos importantes sobre RNFs

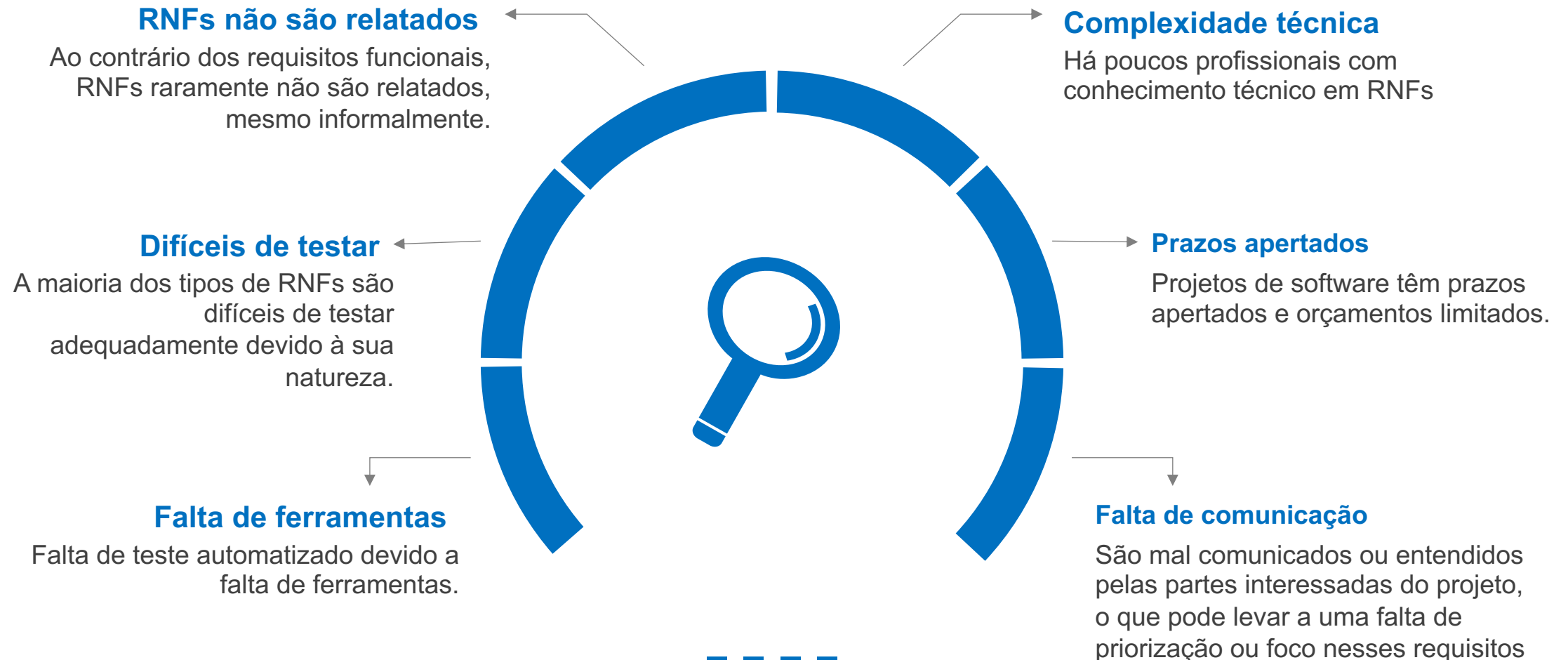


O uso do teste automatizado não é comum em RNFs.





Por que RNFs são negligenciados?

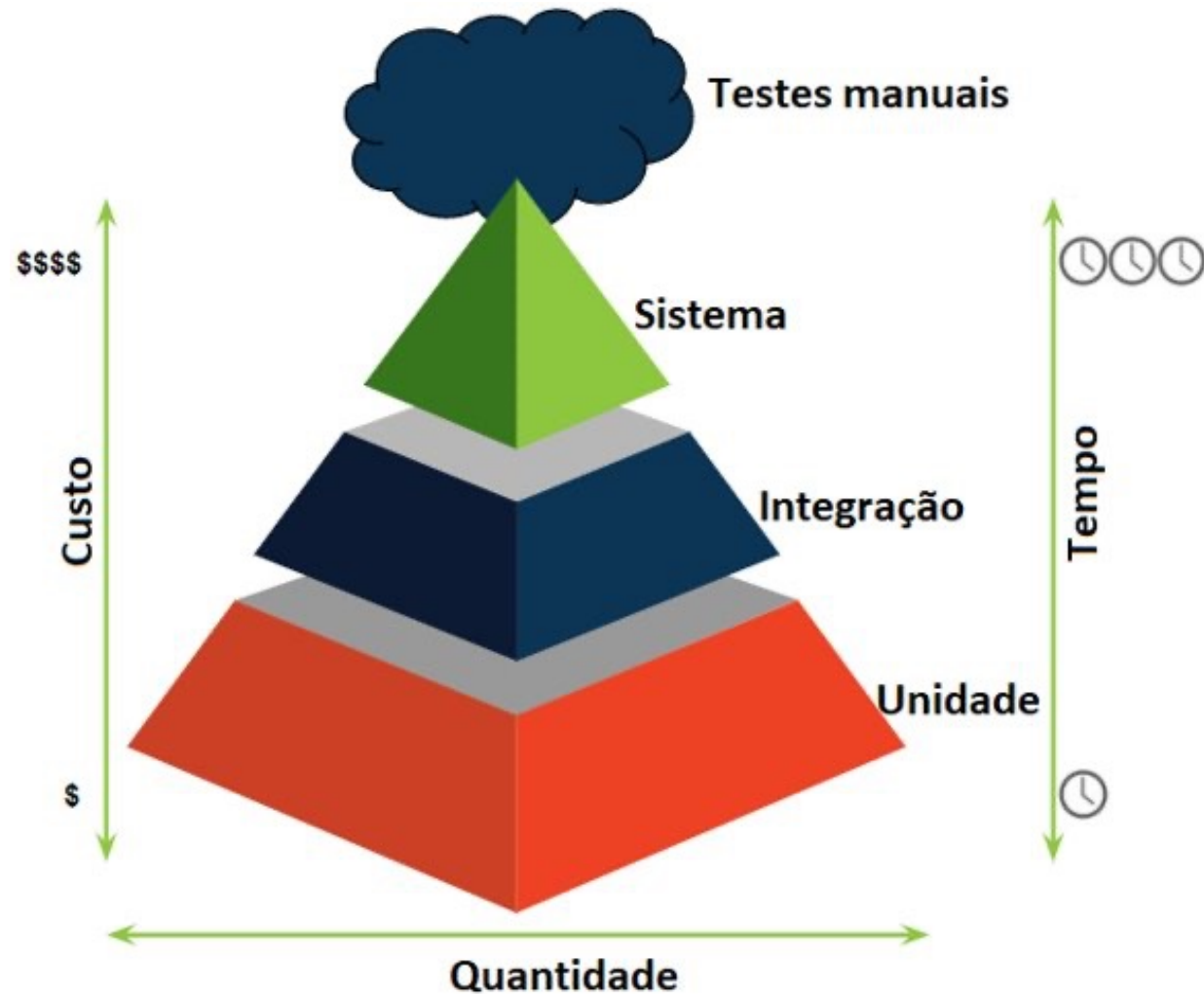


ISO 25010

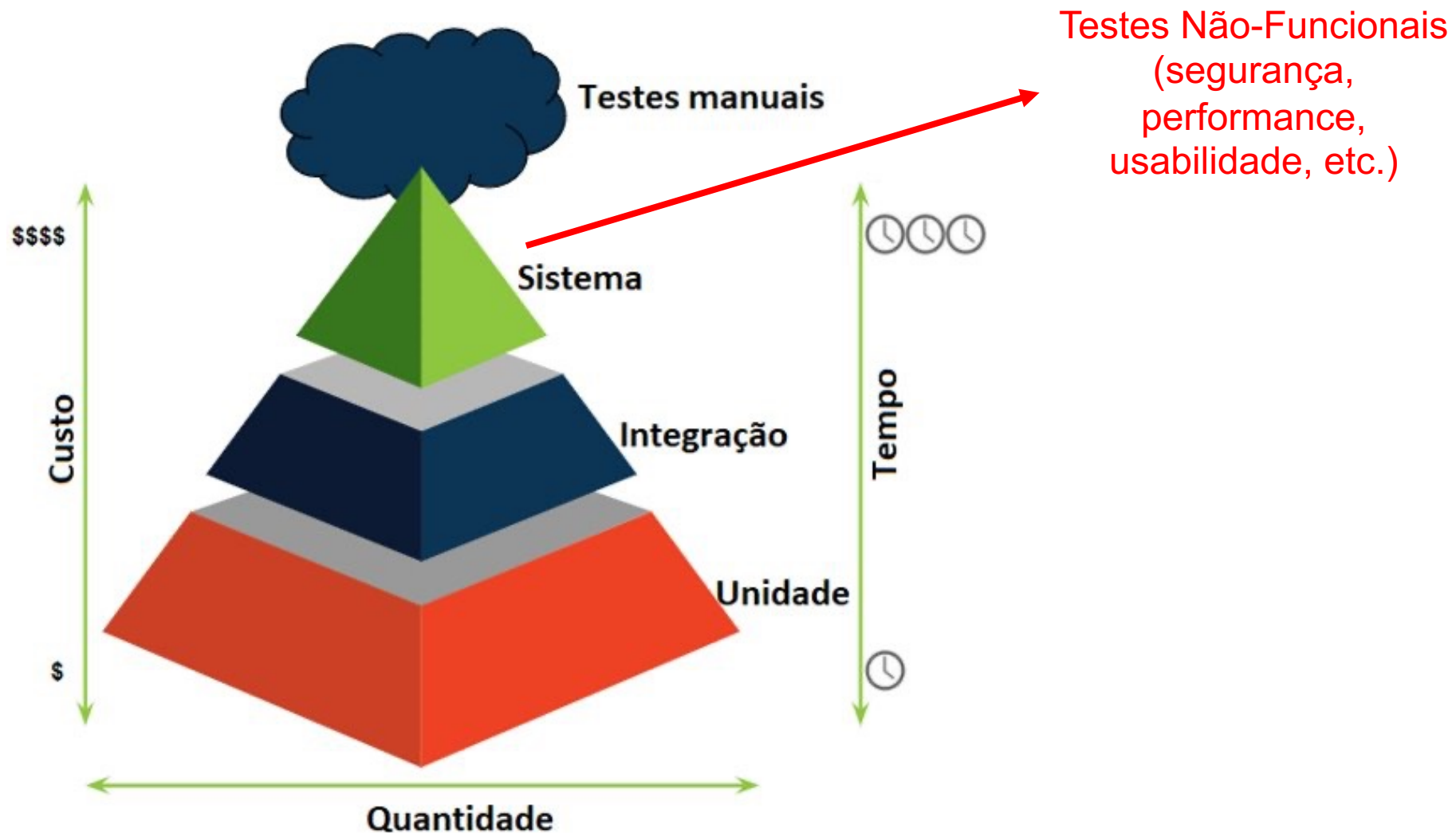


*ISO/IEC 25010

Pirâmide de Testes



Pirâmide de Testes

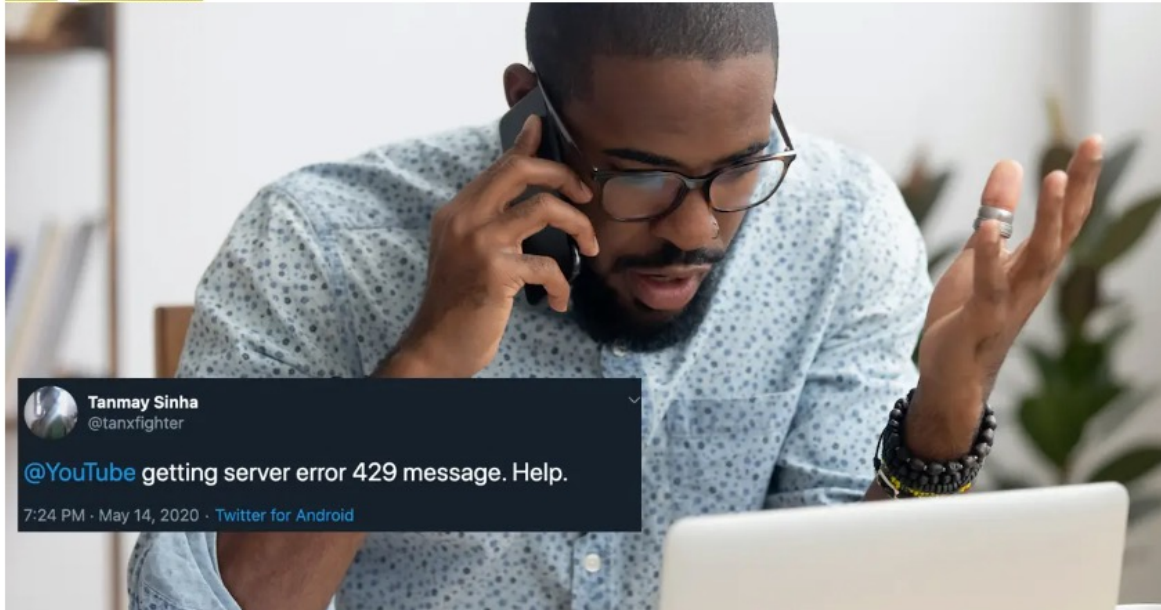




Performance

O que é o Requisito de Performance e como testá-lo em aplicações?

Casos Reais



SOURCE: ISTOCK

Here's Why You're Most Likely Receiving YouTube's "429 Too Many Requests" Error

What does YouTube's '429 Too Many Requests' error mean? Here's how to fix the technical issue while you're trying to access the website or analytics.



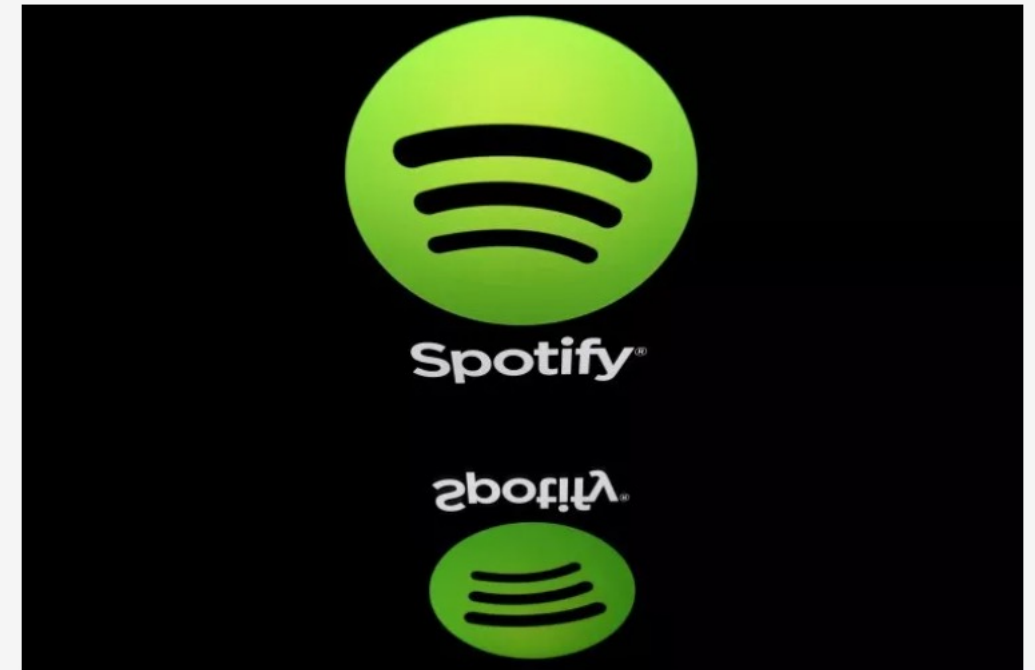
BY **LIZZY ROSENBERG**

MAY 15 2020, UPDATED 2:04 P.M. ET

U.S.

Spotify Down, Not Working: 503 Error, Service Unavailable Users Say

BY **NINA GODLEWSKI** ON 2/6/19 AT 12:19 PM EST



This illustration picture taken on April 19, shows the logo of online streaming music service Spotify displayed on a tablet screen in Paris.

LIONEL BONAVENTURE/AFP/GETTY IMAGES

Requisito de Performance

“Representa o desempenho relativo à
quantidade de recursos utilizados em
condições estabelecidas.”

Michael Felderer, Matthias Buchler, Martin Johns, Achim D Brucker, Ruth Breu, and Alexander Pretschner.
Security testing: A survey. In Advances in Computers, volume 101, pages 1–51. Elsevier, 2016



Sub Características de Performance

**Comportamento
de tempo**

Performance

**Utilização
de recursos**

Capacidade





Sub Características de Performance

**Comportamento
de tempo**

capacidade do software de responder e processar as operações do usuário dentro do tempo especificado.

Performance

**Utilização
de recursos**

Capacidade





Sub Características de Performance

**Comportamento
de tempo**

Performance

**Utilização
de recursos**

capacidade do software de utilizar os recursos de hardware e software disponíveis de forma eficiente e eficaz.

Capacidade





Sub Características de Performance

**Comportamento
de tempo**

Performance

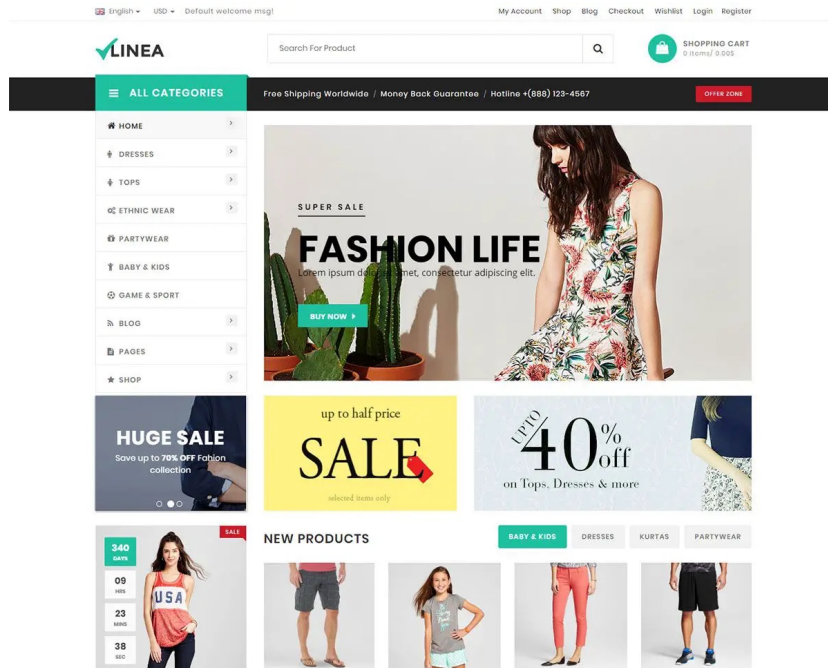
**Utilização
de recursos**

Capacidade

capacidade do software de lidar com volumes crescentes de dados, usuários ou transações sem comprometer seu desempenho ou sua capacidade de resposta

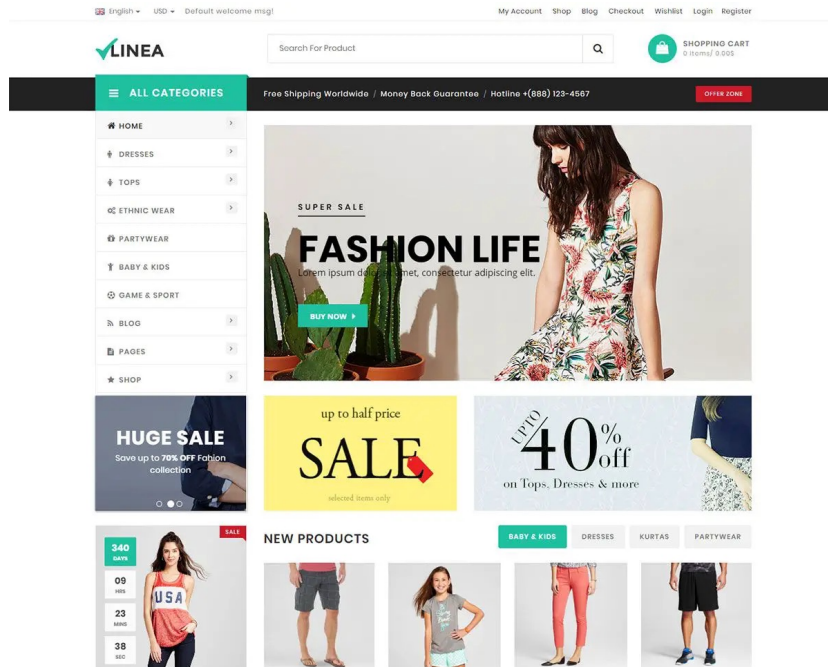


Exemplo de Especificação



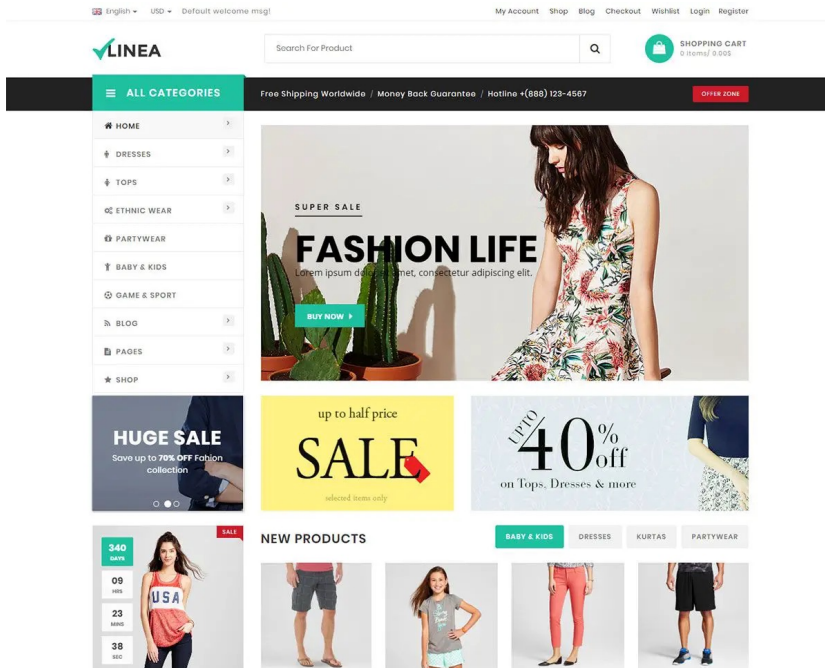
Exemplo de Especificação

1. **Tempo de carregamento da página:** o tempo de carregamento da página inicial deve ser inferior a 3 segundos;



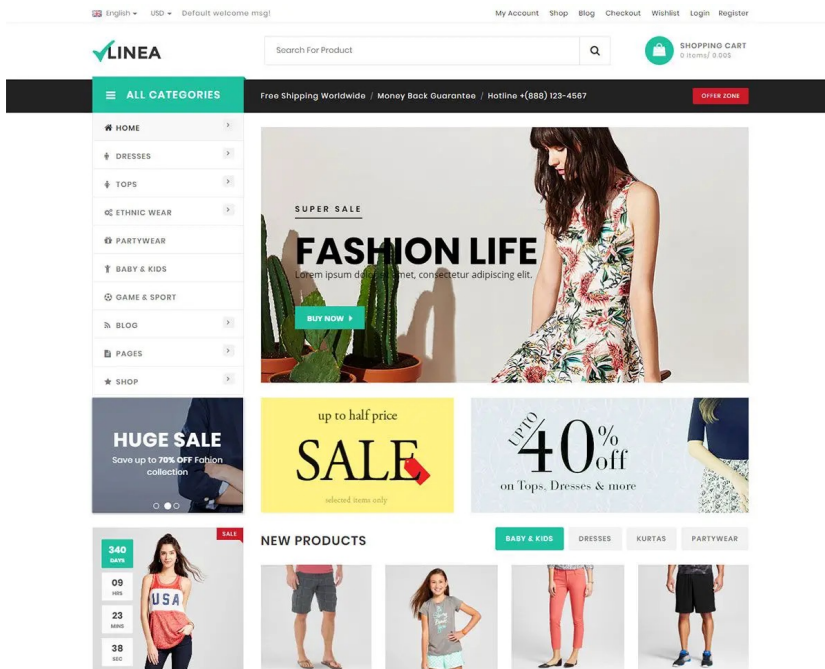
Exemplo de Especificação

1. **Tempo de carregamento da página:** o tempo de carregamento da página inicial deve ser inferior a 3 segundos;
2. **Tempo de resposta:** a resposta para uma solicitação de adicionar um item ao carrinho de compras deve ser inferior a 1 segundo;



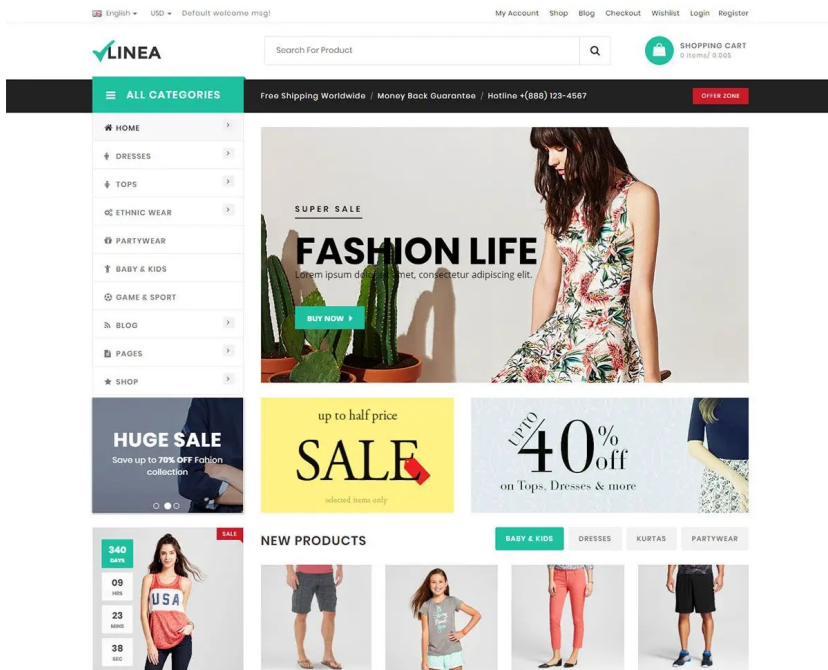
Exemplo de Especificação

1. **Tempo de carregamento da página:** o tempo de carregamento da página inicial deve ser inferior a 3 segundos;
2. **Tempo de resposta:** a resposta para uma solicitação de adicionar um item ao carrinho de compras deve ser inferior a 1 segundo;
3. **Capacidade de carga:** o site deve ser capaz de lidar com um pico de tráfego durante um evento de vendas sem afetar o tempo de resposta;

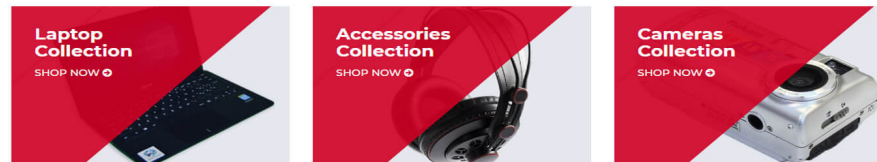
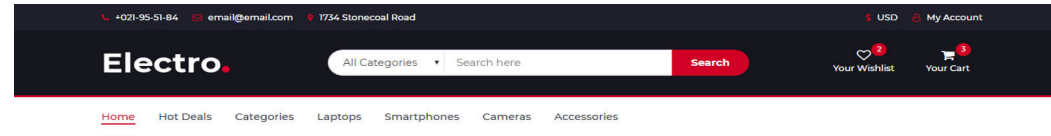


Exemplo de Especificação

1. **Tempo de carregamento da página:** o tempo de carregamento da página inicial deve ser inferior a 3 segundos;
2. **Tempo de resposta:** a resposta para uma solicitação de adicionar um item ao carrinho de compras deve ser inferior a 1 segundo;
3. **Capacidade de carga:** o site deve ser capaz de lidar com um pico de tráfego durante um evento de vendas sem afetar o tempo de resposta;
4. **Escalabilidade:** o site deve ser capaz de lidar com um aumento no número de usuários e transações durante as férias de final de ano.

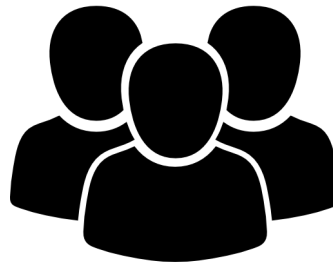
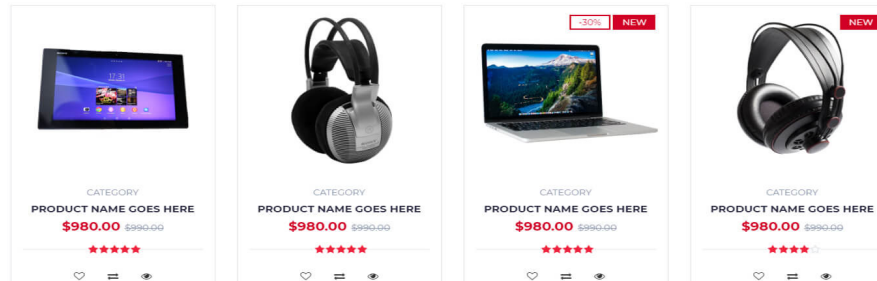


Exemplo prático



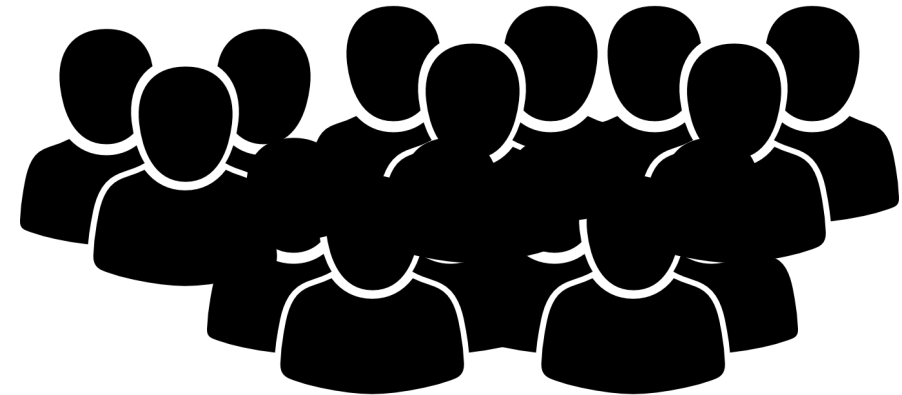
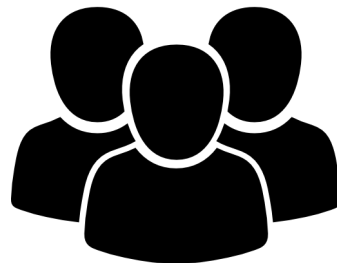
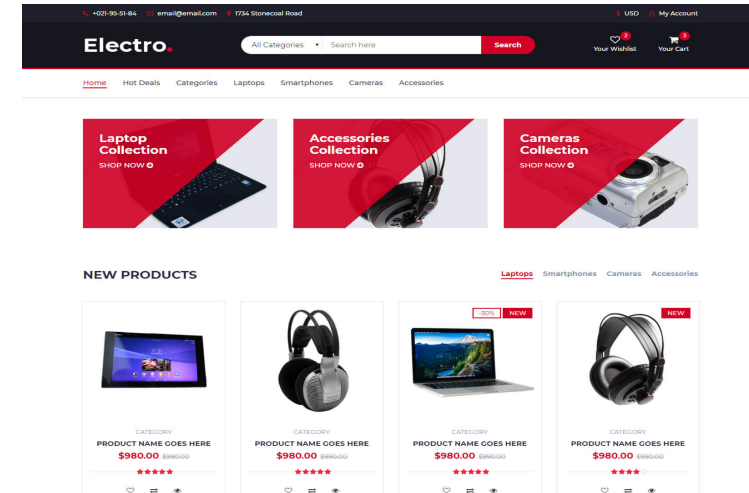
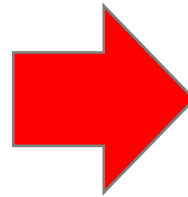
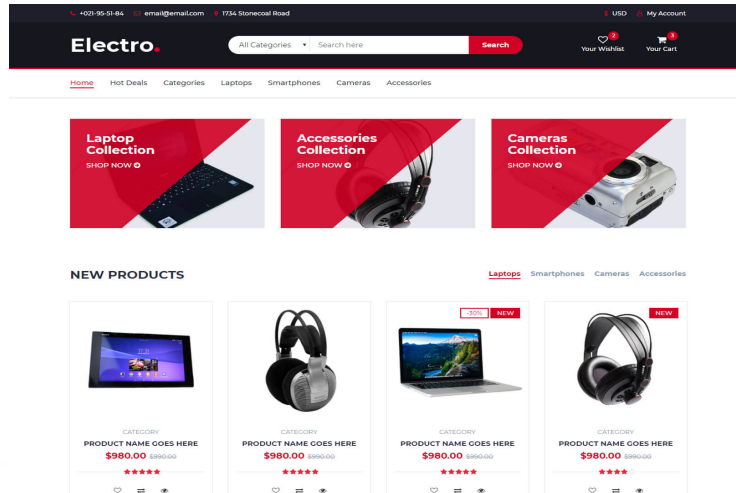
NEW PRODUCTS

Laptops Smartphones Cameras Accessories



Exemplo prático

**BLACK
FRIDAY**



O que é Teste de Performance?

*“Investigação técnica que pode ser realizada para determinar ou validar as características da aplicação relacionadas a **velocidade**, **escalabilidade**, **estabilidade** e **responsividade**.”*



Benefícios do Teste de Performance

- 1 - Identificação antecipada de problemas de Performance
- 2 - Melhoria da experiência do usuário
- 3 - Redução de custos
- 4 - Melhoria da reputação da marca
- 5 - Identificação de problemas de infraestrutura
- 6 - Aumento da eficiência





Quais aspectos devem ser
avaliados?





Quais aspectos devem ser avaliados?



**Tempo de
resposta**



Quais aspectos devem ser avaliados?



**Tempo de
resposta**



Throughput

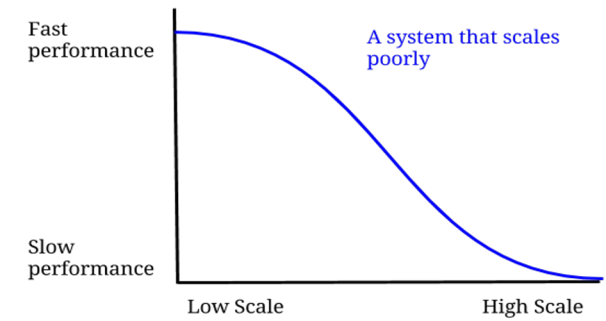
Quais aspectos devem ser avaliados?



Tempo de resposta



Throughput



Escalabilidade

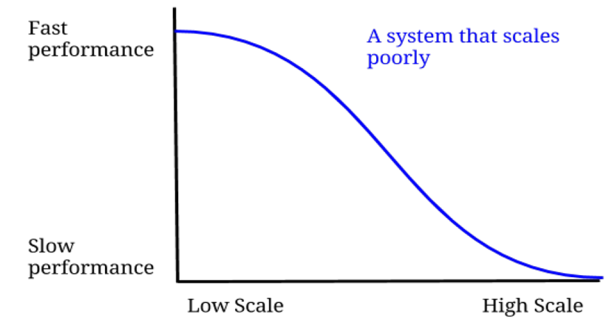
Quais aspectos devem ser avaliados?



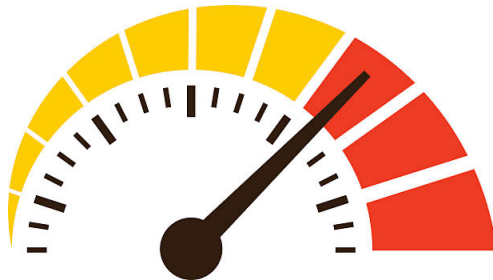
Tempo de resposta



Throughput



Escalabilidade



Uso de recursos

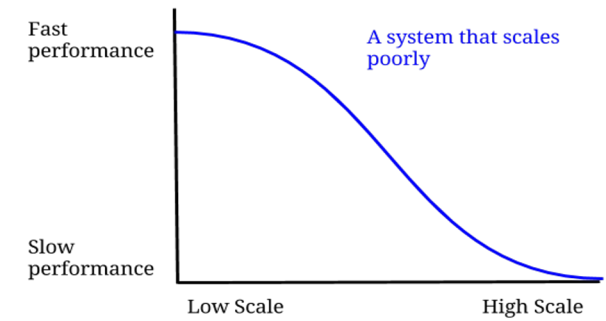
Quais aspectos devem ser avaliados?



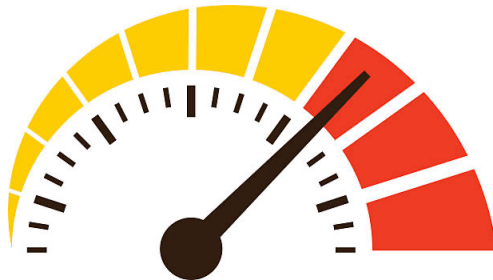
Tempo de resposta



Throughput



Escalabilidade



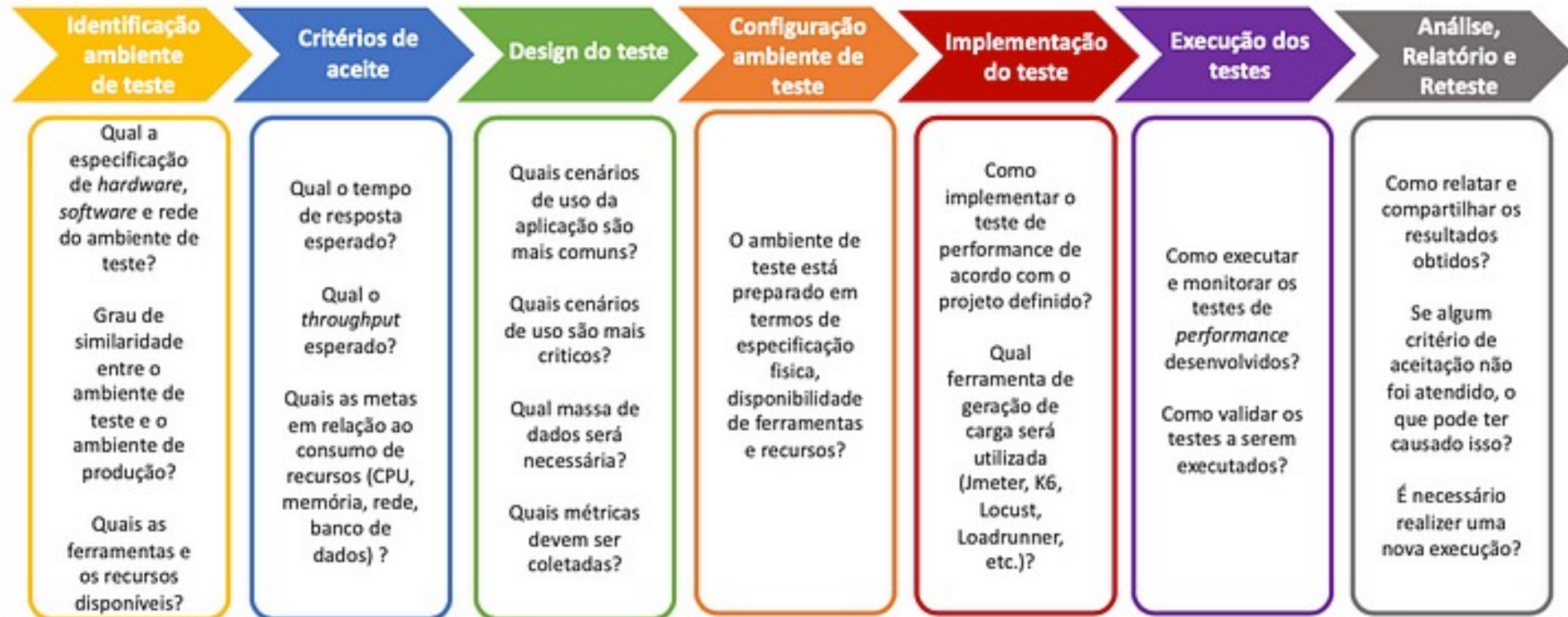
Uso de recursos



Acertos e Erros



Ciclo de Vida Teste de Performance



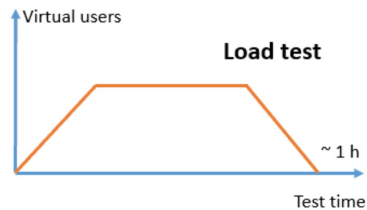


Ciclo de Vida Teste de Performance

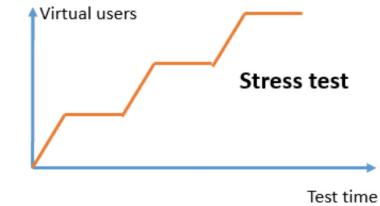


Tipos de Teste de Performance

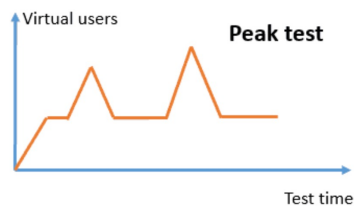
Teste de Carga



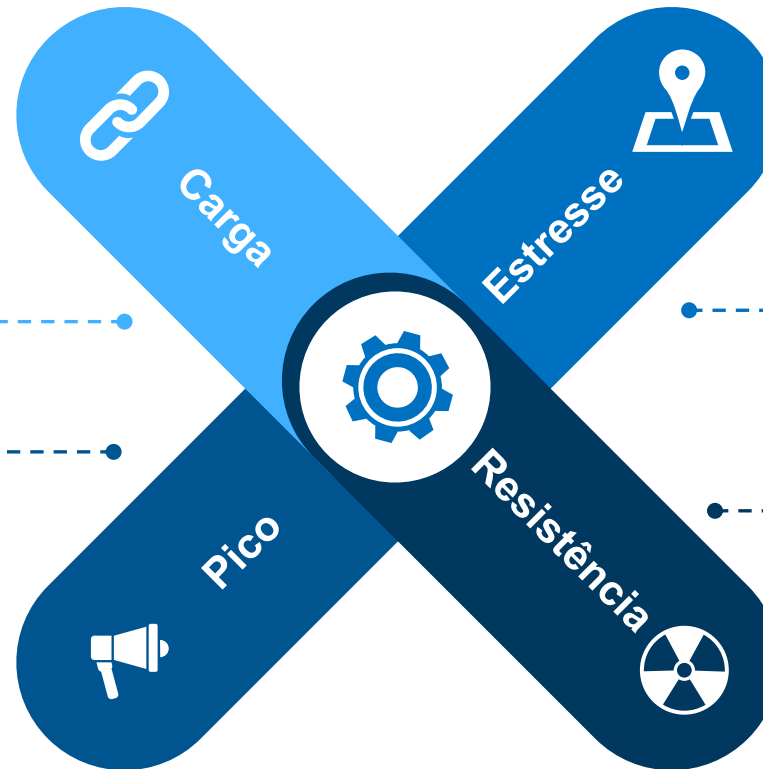
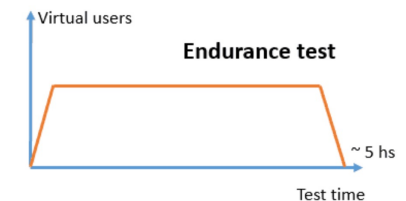
Teste de Estresse



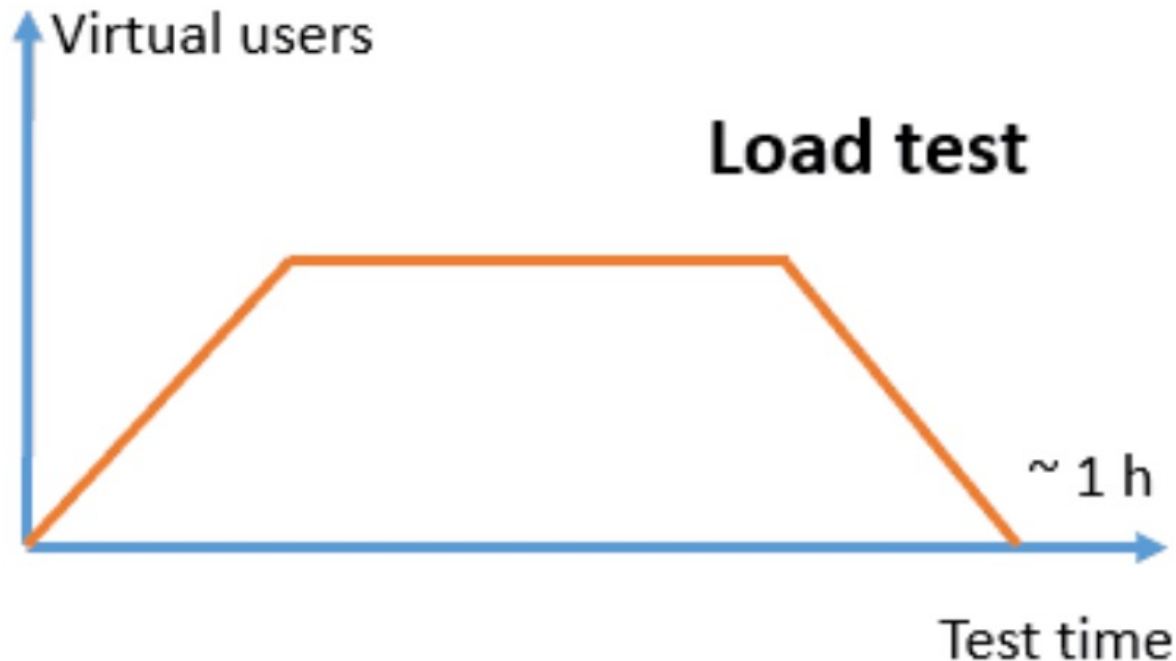
Teste de Pico



Teste de Resistência

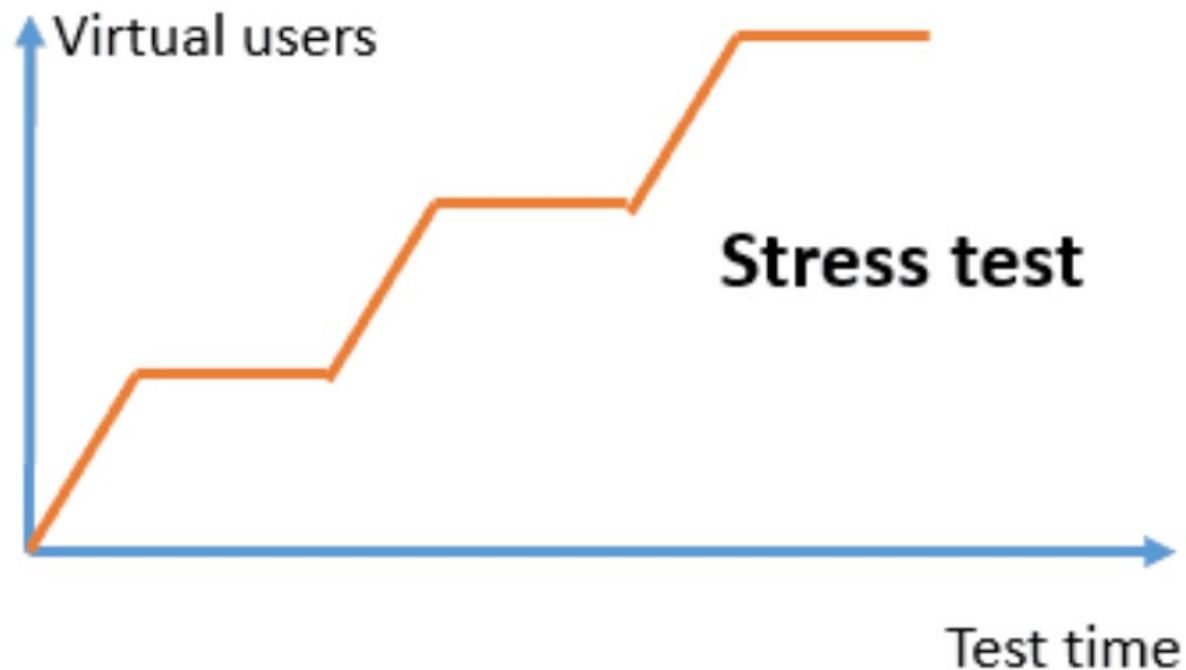


Teste de Carga



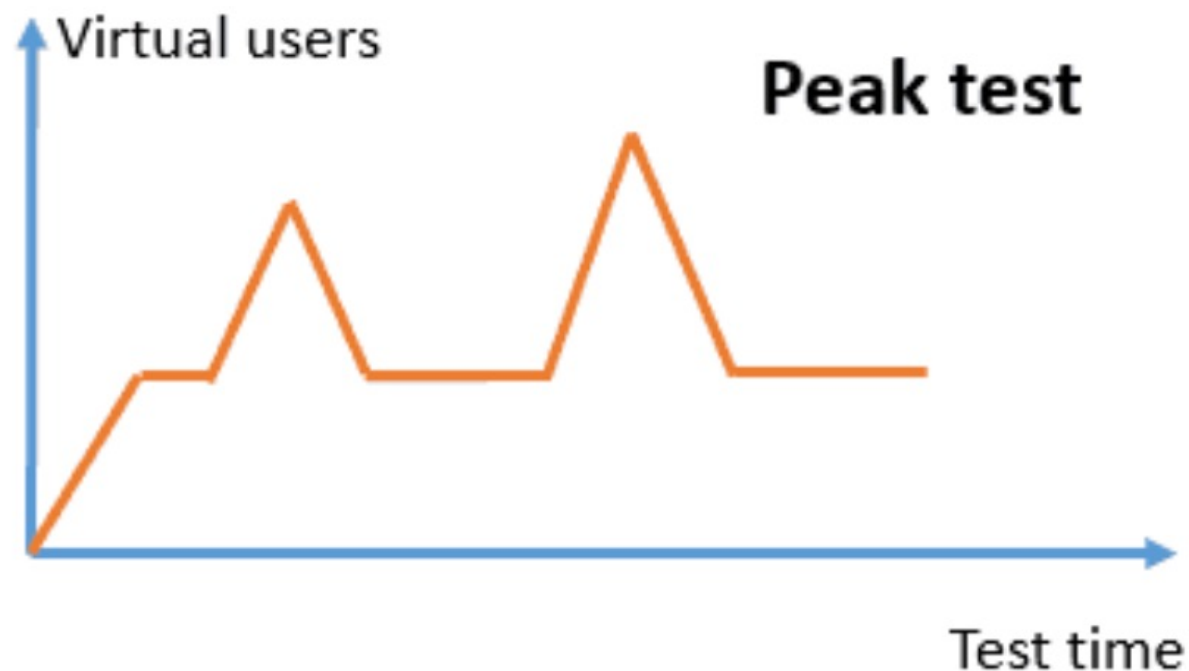
- Técnica de teste de performance que simula uma aplicação a partir de uma **carga esperada** de usuários;
- Projetado para simular as **condições reais de uso da aplicação**;
- Recomendado ser executado durante o **“período razoável” (> 30 min) de tempo**.

Teste de Estresse



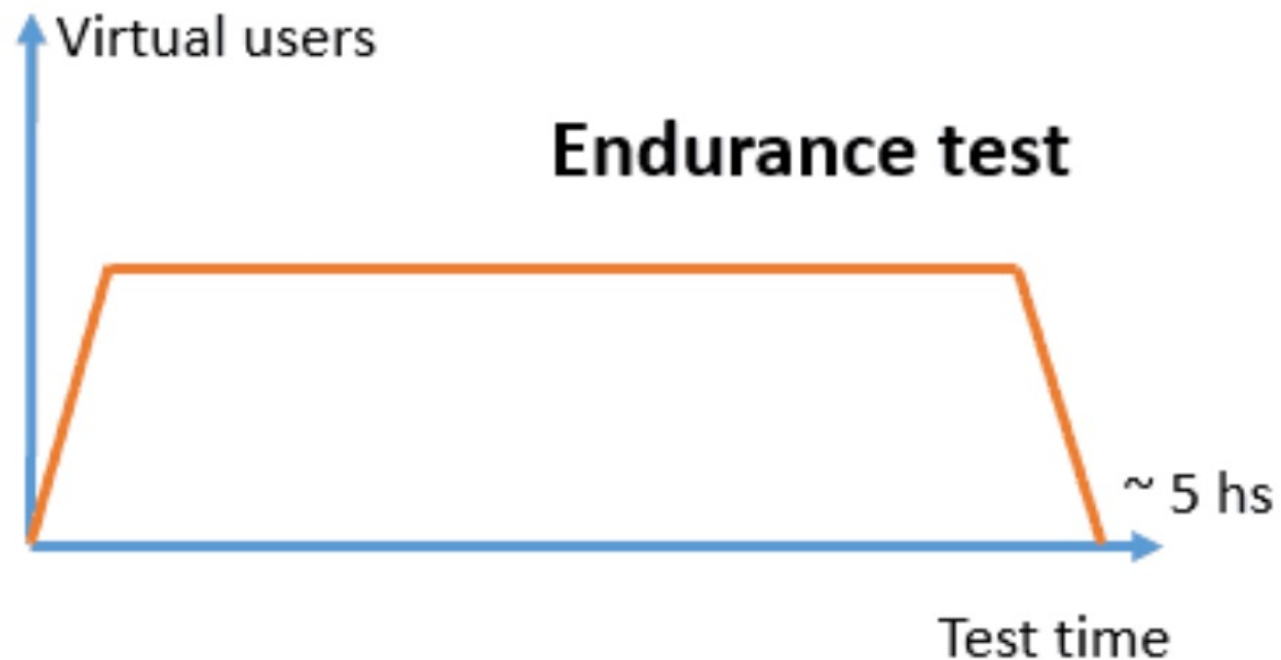
- Técnica de teste de performance onde uma aplicação é simulada a partir de uma **carga extrema de usuários simultâneos** e dos limites esperados;
- Projetado para levar o sistema **além dos seus limites de capacidade**, para avaliar a capacidade do sistema de lidar com as **condições extremas**;
- Útil para **identificar os limites de carga** da aplicação;
- Recomendado ser executado durante um **período curto de tempo (< 30 min)**.

Teste de Pico



- Técnica de teste de performance onde a capacidade de uma aplicação é avaliada a partir de **picos repentinos de carga**;
- Utiliza cenários que simulam **picos repentinos de carga** no sistema, como **picos de tráfego**, **aumento súbito de usuários** ou **grandes volumes de dados**;
- A carga deve ser gradualmente aumentada para avaliar o **comportamento do sistema sob diferentes níveis de carga**;
- Normalmente realizado por um **período curto de tempo (≤ 30 min)**.

Teste de Resistência

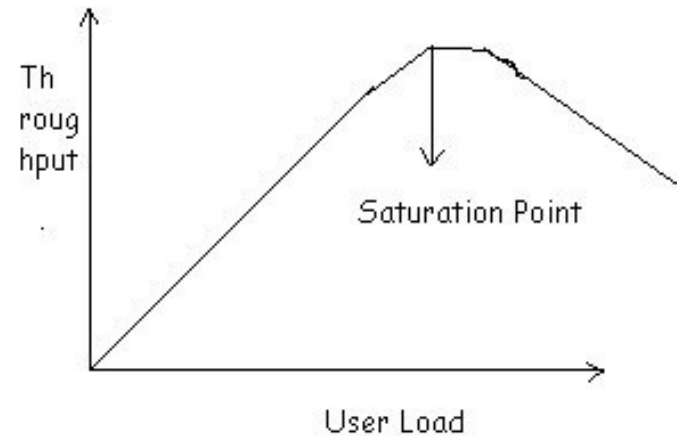


- Técnica de teste de performance onde a aplicação é avaliada a partir de uma **carga constante e prolongada** de acessos simultâneos;
- A intensidade do teste de resistência é mantida constante ao longo do tempo para avaliar a **capacidade do sistema de lidar com carga de trabalho constante**;
- É realizado por um **período prolongado de tempo**, geralmente **várias horas ou até mesmo dias**.

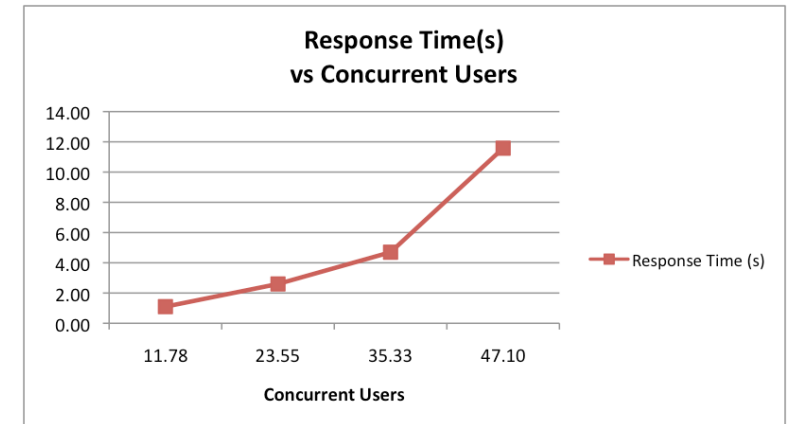
Conceitos Importantes



Thinktime



Ponto de saturação



**Tempo de resposta
(média e percentil)**



Ferramentas

Quais as principais ferramentas disponíveis para conduzir Testes de Performance?



Ferramentas



LOCUST



Apache Jmeter



- Ferramenta *open-source* escrita em Java;
- Ferramenta *multi-Plataforma*, podendo ser executada no Windows, Linux e Mac OS;
- Ferramenta com *interface gráfica* para construção de testes de performance;
- Fornece um *relatório completo* com os resultados do teste de performance;
- Permite a criação de *scripts para automatizar* a execução de testes de performance.

Locust



LOCUST

- Ferramenta *open-source* escrita em Python;
- Desenvolvido para permitir a *distribuição dos testes de carga em vários servidores*;
- Ferramenta simples e fácil de usar, com uma *interface de usuário intuitiva*;
- Permite a criação de testes de carga com o uso de *scripts Python*;
- Fornece uma interface de usuário com gráficos em tempo real para *monitorar o comportamento da aplicação durante o teste*.



- Ferramenta *open-source* escrita em Go;
- Desenvolvido para permitir a **distribuição dos testes de carga em vários servidores**;
- Permite a criação de testes de carga com o uso de **scripts JavaScript**;
- Fornece uma série de **relatórios e gráficos** para analisar os resultados dos testes de carga e desempenho;
- Fornece recursos de **monitoramento em tempo real** para avaliar o comportamento da aplicação.



LoadRunner



- Ferramenta comercial desenvolvida pela empresa Micro Focus;
- Monitora em tempo real o desempenho da aplicação em teste, permitindo a identificação de gargalos em tempo real;
- Oferece ferramentas avançadas de análise de resultados;
- Permite a automatização de testes de desempenho e a reutilização de scripts;
- Pode ser usado em ambientes de teste distribuídos.

....
Isso não é tudo!





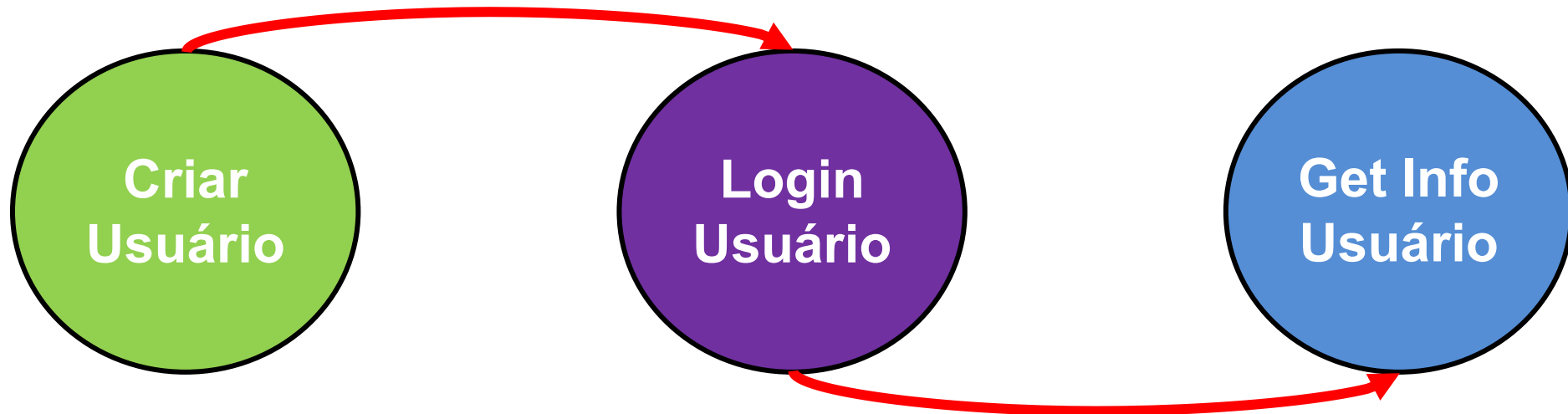
Prática

Aprendendo como implementar Testes de Performance a partir das aplicações disponíveis

Objetivo

“Demonstrar a *implementação* e *execução* de um teste de performance usando uma *ferramenta open-source* em uma *API pública* servindo como uma *base para aprendizado* do teste de performance.”

■ ■ ■ ■ Cenário



Implementação





Erros mais comuns

Quais os erros mais comuns que Analistas de QA e Desenvolvedores cometem ao conduzir Testes de Performance?



Erros mais comuns

- 1 – Importância do *Think time*;
- 2 – Cenários de testes irreais;
- 3 – Atenção ao ambiente de teste;
- 4 – Falta de monitoramento;
- 5 – Interpretar tempo de resposta pela média;
- 6 – Não validar o *response body* ou algum dado esperado.

* CINCO ERROS COMUNS AO PLANEJAR E EXECUTAR TESTES DE PERFORMANCE





■ ■ ■ ■

Obrigado!!!

\$ git clone <https://github.com/misaelljr/PalestraTestePerformance.git>

■ ■ ■ ■