Introdução ao Jmeter para teste de performance

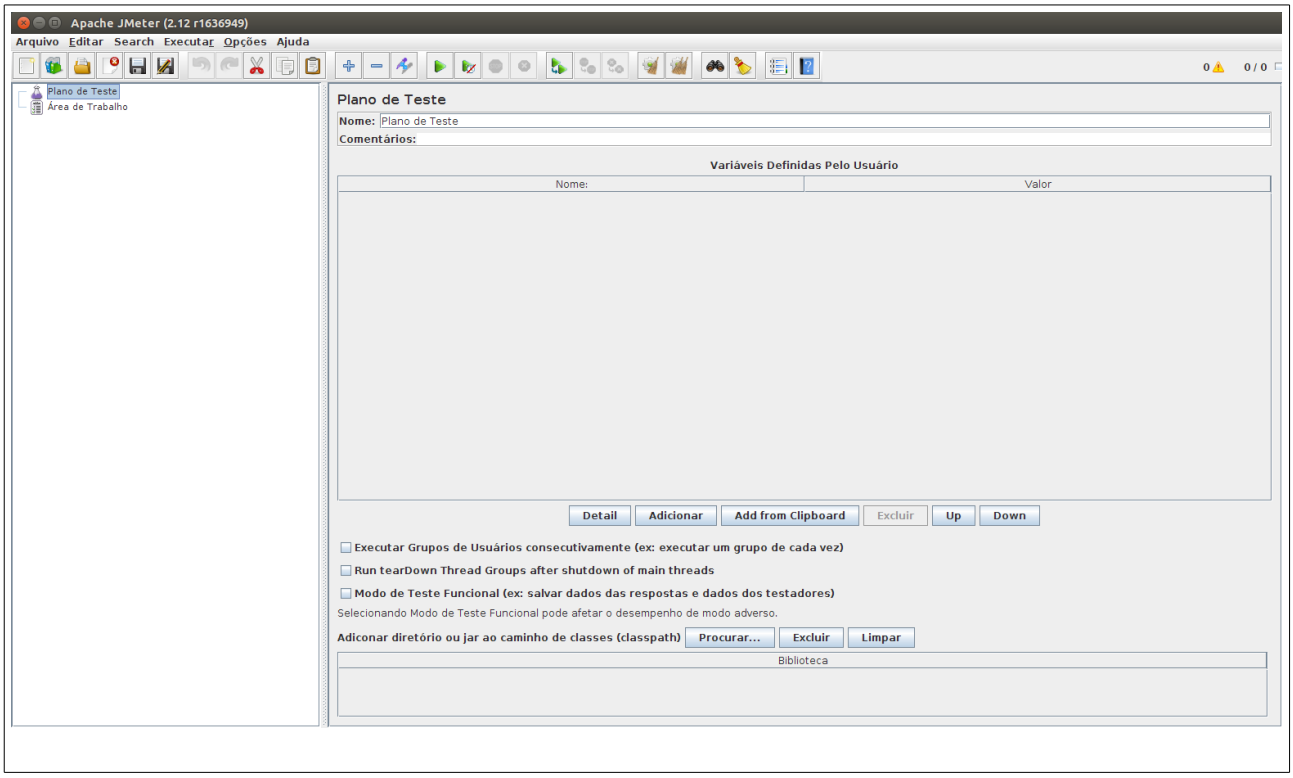
O Jmeter é um programa que serve para fazer testes de performance, carga e stress. Ele é um software livre, sendo parte do projeto Jackarta da Apache Software Foundation.

Neste howto veremos como utilizar o JMeter para fazer um teste de um site.

1º Passo: O JMeter está disponível para download em:

http://jmeter.apache.org/

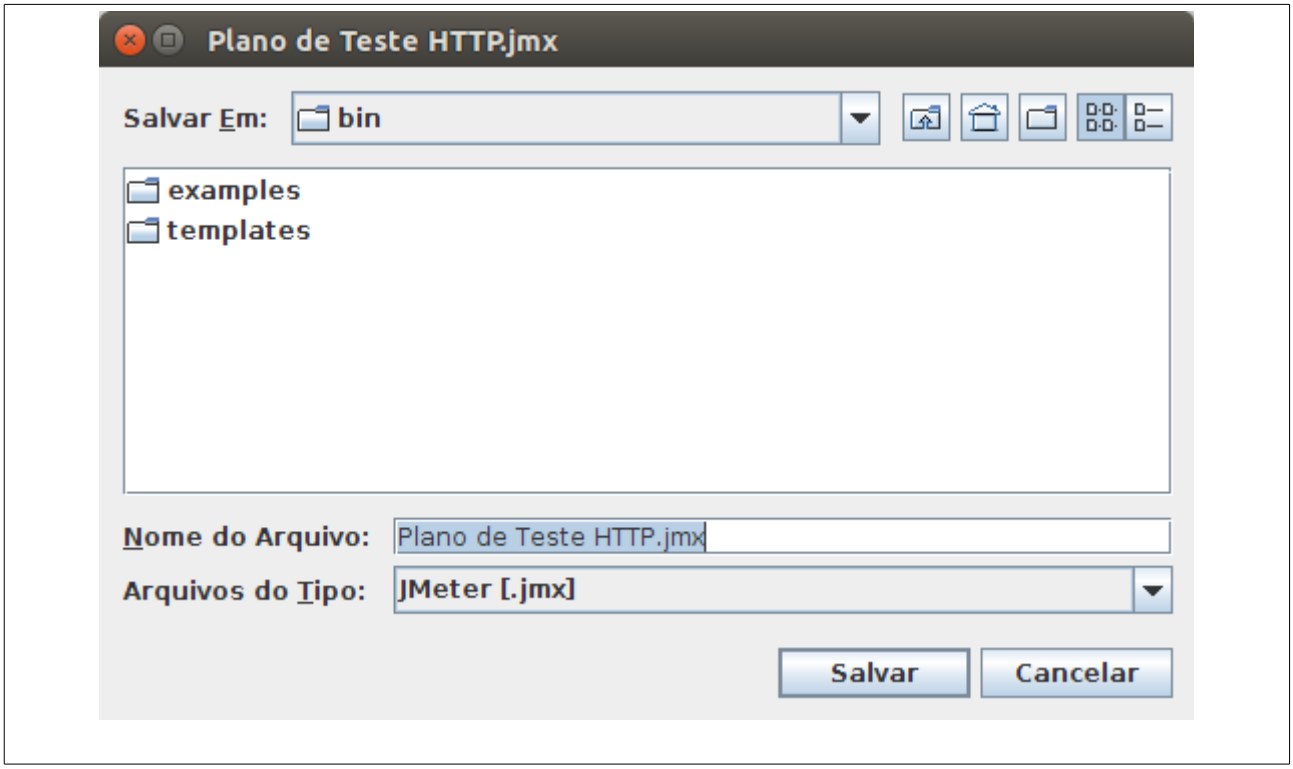
Faça download do JMeter, e descompacte-o. Ao abrir o programa, será exibido a seguinte tela:



2º Passo: O Jmeter possui uma área de trabalho e um plano de teste. Vamos iniciar a criação do plano de teste, que é a base de qualquer teste no JMeter. No plano de teste é possível incluir os recursos de testes.

Insira o nome do plano de teste e, se possível, algum comentário sobre o plano a ser realizado.

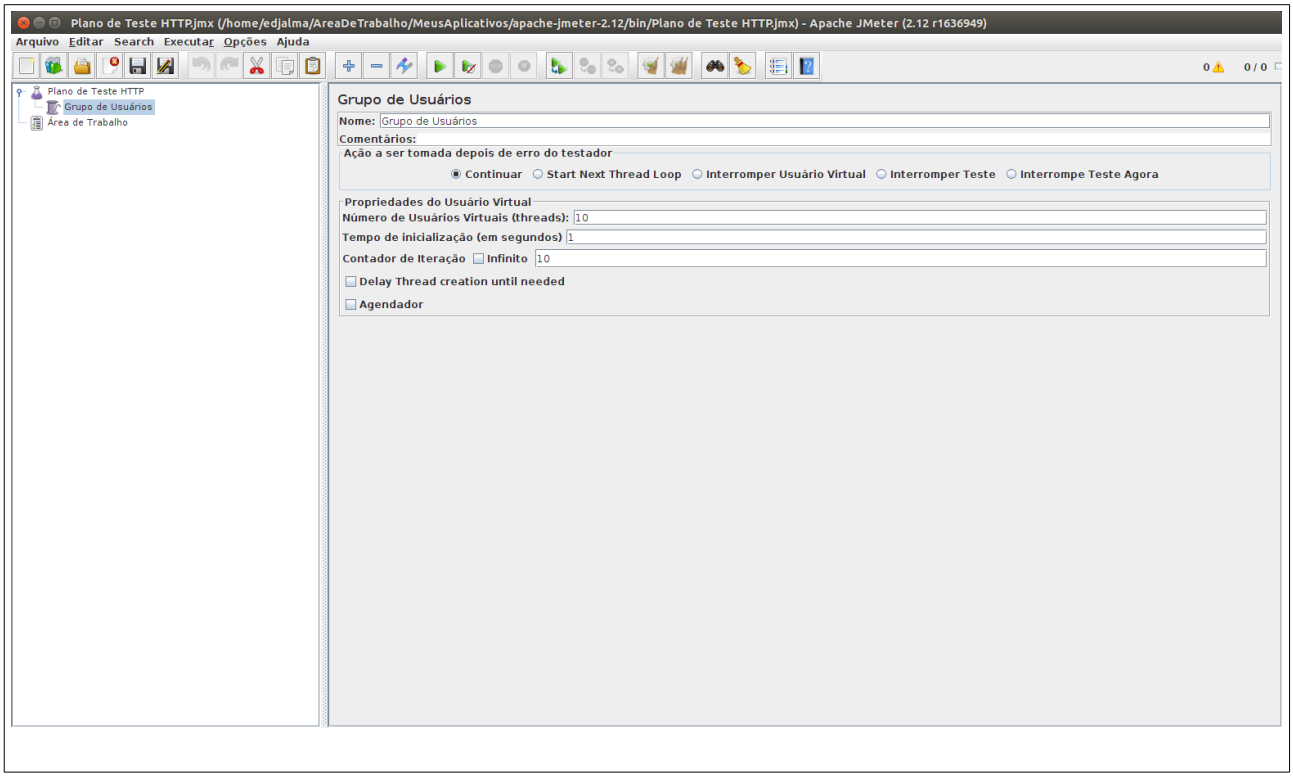
Após a configuração do plano de teste, salve o plano de teste, clicando em arquivo → salvar.



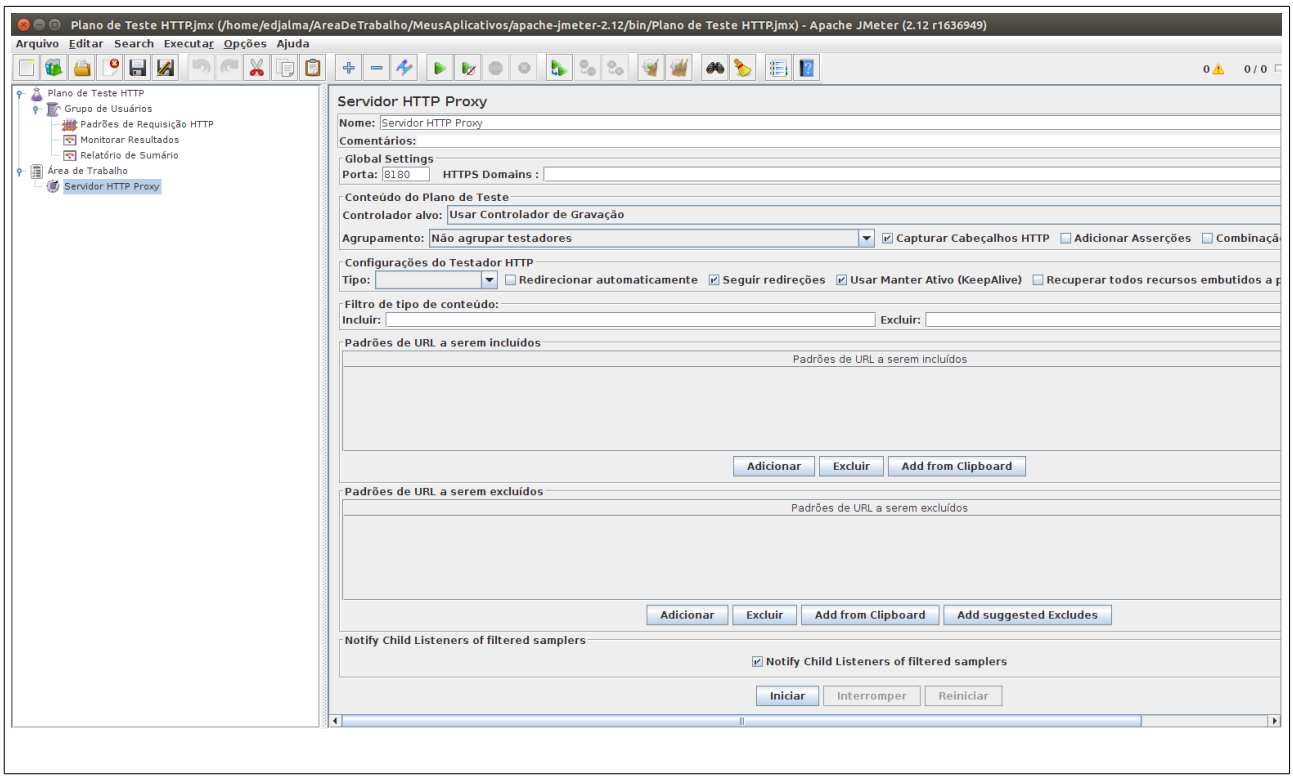
3º Passo: O próximo passo para a criação de um teste é a criação de um grupo de usuários.

Clique com o botão direito no plano de teste. Um menu pop-up será exibido. Clique em Adicionar → Threads (users) → Grupo de usuários.

Um grupo de usuários serve para configurar quantas pessoas (virtuais) serão utilizadas no teste, o tempo de execução do teste e a quantidade de interações dos processos. No nosso teste vamos determinar que temos 10 usuários virtuais, o tempo de inicialização será 1 (tempo em segundos) e as interações (quantidade de vezes que será executado o teste) determinaremos 10.

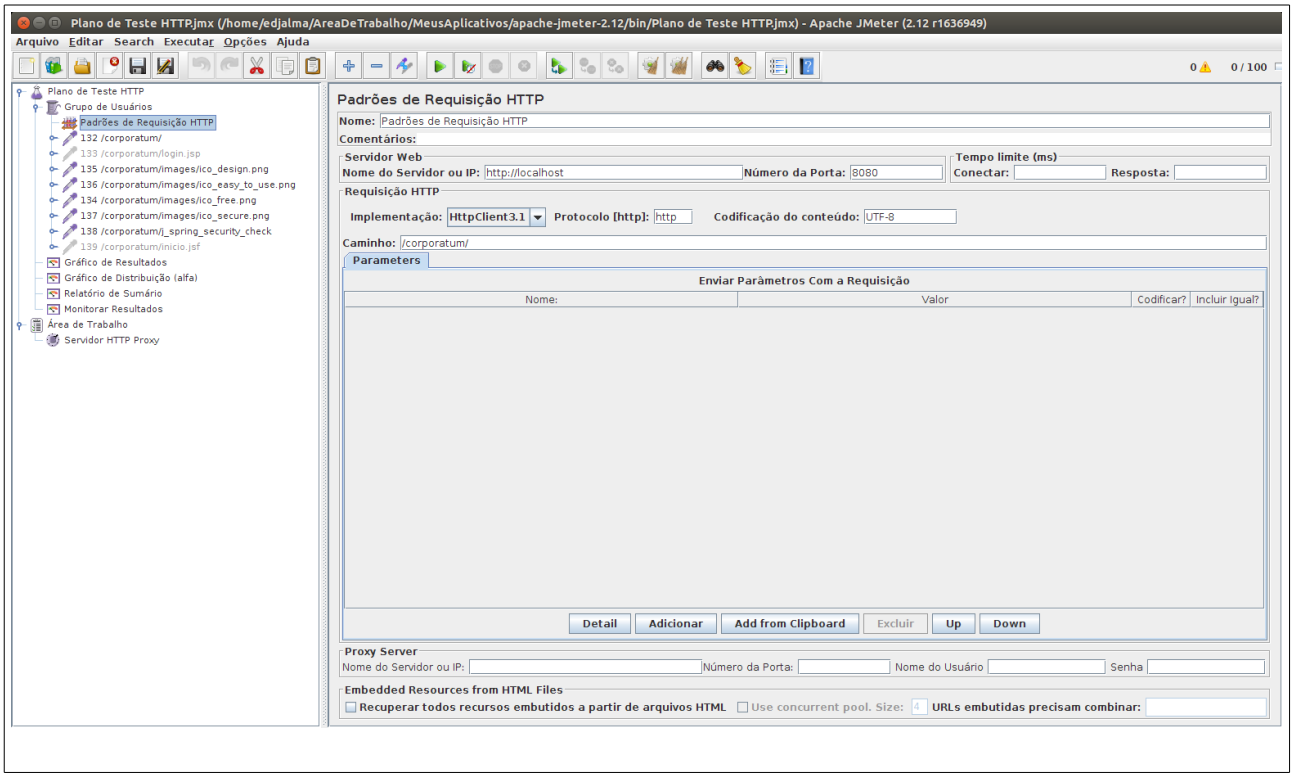


4º Passo: O próximo passo para efetuar um teste é adicionar ao grupo de usuários os elementos de teste. Clique com o botão direito do mouse no grupo de usuários Adicionar → Elemento de Configuração → Padrão de Requisição HTTP. O padrão de requisição Http serve para capturar as requisições Web via Proxy. Neste caso o responsável pelo Proxy é o próprio Jmeter. É assim que o Jmeter consegue criar o plano de teste, pois por ele é passado todas as requisições que serão transmitidas ao Server. Configure uma porta qualquer para ser o Proxy (como se você estivesse criando um proxy) .



5º Passo: O próximo passo é adicionar os Listeners/Ouvintes que visam fornecer acesso aos resultados dos testes que foram realizados pelo JMeter. Clique com o botão direito sobre o “Plano de Teste”, e insira os ouvintes que desejar como gráfico, relatório e monitor de resultados. Eu adicionei aqui os Listeners: Gráfico de Resultados, Relatório de Sumário

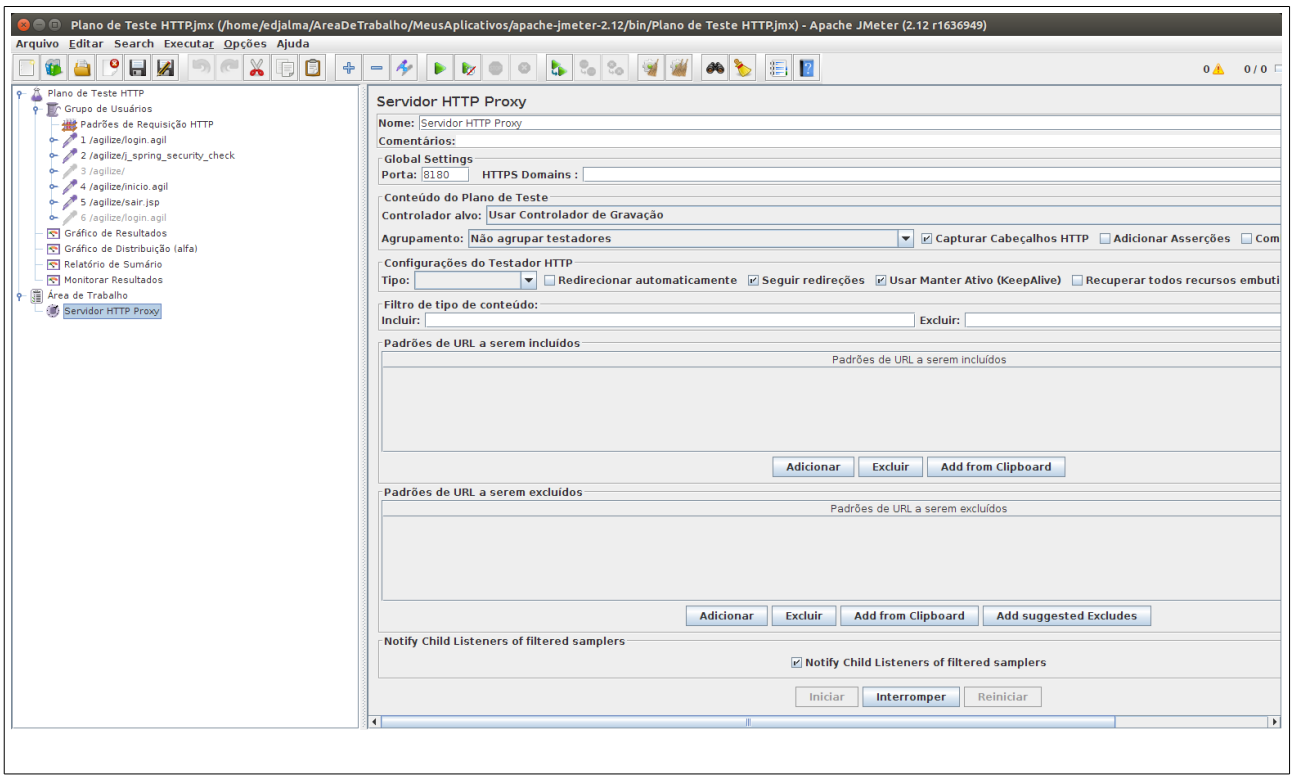
6º Passo: Após configurarmos as configurações do teste, vamos criar o Proxy e abrir o site desejado. Clique na área de trabalho com o botão direito → Adicionar → Elementos que não são de teste → Servidor HTTP Proxy



7º Passo: A seguir, configure o seu navegador para utilizar o Proxy através do ip 127.0.0.1 e da porta 8180 (127.0.0.1:8080).

8º Passo: Volte ao JMeter e, no servidor Proxy, clique no botão **Iniciar**.

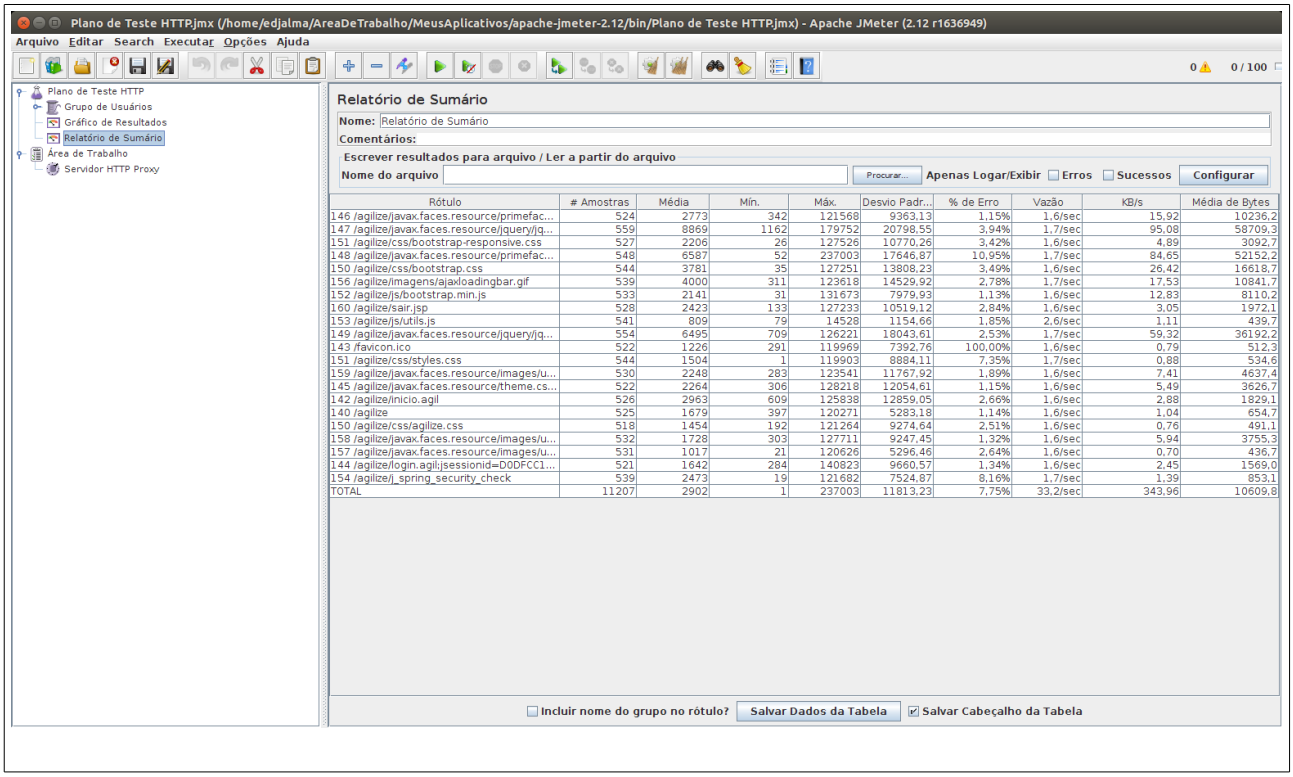
9º Passo: Pronto! Agora basta acessar a url desejada, efetuar os passos que o JMeter deverá repetir. Isso possibilitará estimar o tempo e a carga de trabalho. Ao concluir, volte no Jmeter e clique em Interromper. Note que ao lado esquerdo da tela irão ser exibidas as tarefas executadas, como na imagem a seguir:



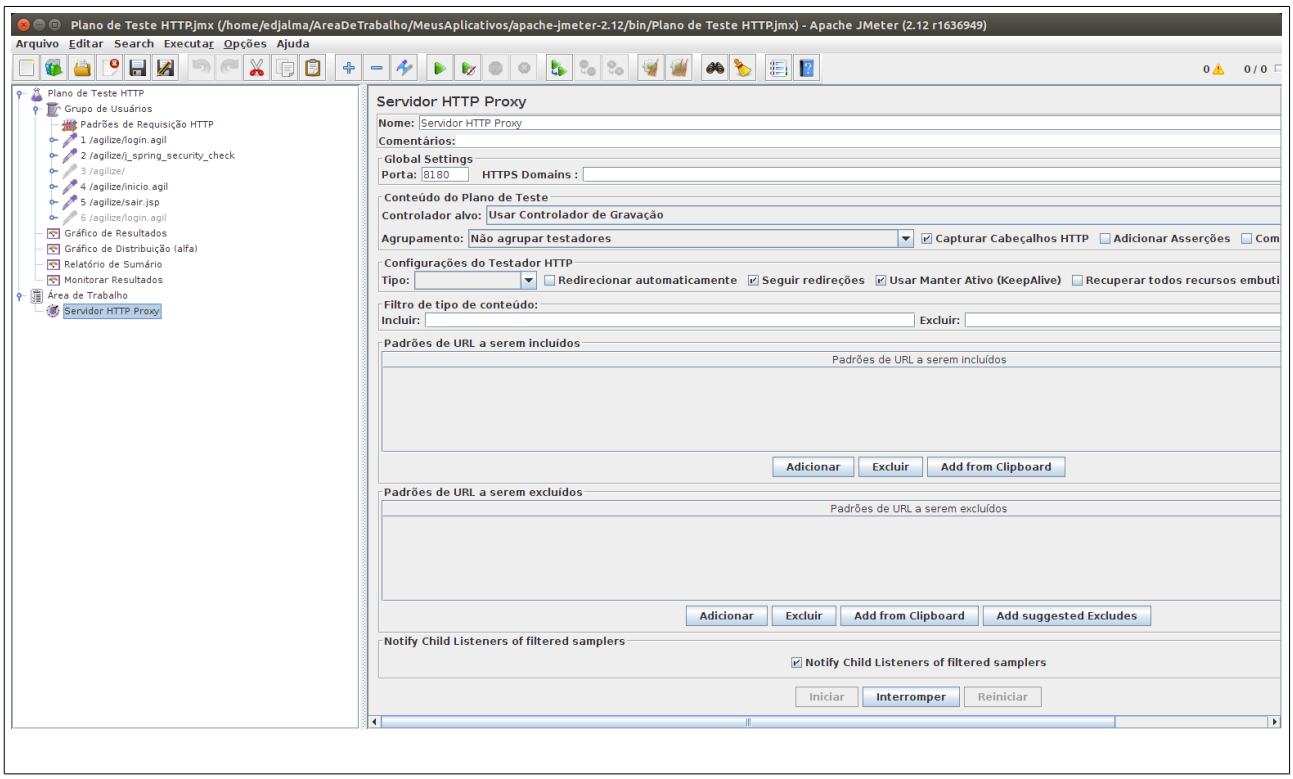
Agora clique no teste e no menu Executar → Iniciar.

Após o teste realizado, acesse Ouvintes e verifique os resultados. Como exemplo, temos os ouvintes a seguir:

Resultados em tabelas:



Relatório agregado:



Com estes e outros ouvintes é possível ver a velocidade do seu site, o tempo de resposta e a quantidade de usuários suportados.

ATIVIDADE A SER DESENVOLVIDA

1. Instalar e abrir o Selenium IDE;
2. Gravar um script com os passos que você deseja automatizar na forma de um teste.
    1. Para executar este passo escolha um projeto do qual já fizemos em aulas anteriores, suba-o com o Tomcat dentro do eclipse.
    2. Para criar um teste na forma de um programa Java, você deve inicialmente capturar as ações relacionadas ao uso do trecho do sistema que você quer testar, para depois exportar para o formato Java. Por exemplo, para criar um Procedimento de Teste para o login no sistema, devo gravar as ações de login nesse sistema.
    3. A linguagem utilizada pelo Selenium é simples e intuitiva. Ao clicar no campo “comando” podemos ver a lista de comandos disponíveis.
3. Exportar o teste para Java
    1. Após criar o teste no Selenium, é necessário exporta-lo para Java. Com isso teremos as mesmas ações capturadas anteriormente, porém utilizando uma API Java para execução de comandos.
4. Criar um Procedimento de Teste Genérico
    1. Após termos o arquivo Java, gerado a partir do script com a captura das ações do teste, precisamos criar um Procedimento de Teste que possa ser utilizado para testar todas as situações envolvidas nessa parte do sistema. Isso incluir testar o “caminho feliz”, assim como todas as possíveis exceções que possam ser geradas a partir da execução da funcionalidade.
    2. Você pode por exemplo criar vários logins corretos, com o intuito de verificar o nível de acesso de cada usuário, assim como tentativas de login utilizando algum valor inválido, que causa a exibição de mensagens de erro.
5. Criar os casos de teste automatizados
    1. A partir desse procedimento de teste é possível fazer uma série de verificações. Conforme já mencionado, as verificações estão associadas ao “caminho feliz” da funcionalidade, assim como os prováveis casos excepcionais, e que normalmente causam a emissão de alguma mensagem ao usuário. Assim, faz parte do processo de teste pensar nesses casos, para em seguida automatizar sua execução.
    2. É interessante salientar que a execução dos testes via selenium exige que uma instância da classe DefaultSelenium seja criada.
        1. @BeforeClass
        2. public static void iniciaBrowser() throws Exception {
        3. browser = new DefaultSelenium("localhost",
        4. 4444, "*firefox", "http://localhost/**<<suaAPP>>**/");
        5. browser.start();

6. login = new LoginProcedimentosDeTeste(browser);
7. }

6. Execução do Teste
    1. Os testes criados com o Selenium são testes que utilizam o JUnit, de forma similar aos testes de unidade. Sua execução acontece da mesma forma que os testes de unidade, clicando com o botão direito do mouse em cima da classe e solicitando sua execução via JUnit. No entanto, existe uma diferença: para que o teste execute, é necessário que exista uma instância do Selenium RC executando na mesma máquina que contém o sistema a ser testado.
    2. O Selenium RC é a parte do selenium responsável por executar cada um dos comandos existentes nos procedimentos de teste. Assim, quando se solicita a digitação de um texto em um comando de uma tela, na verdade emitimos essa solicitação ao Selenium RC, que por sua vez se comunica com o navegador, via funções JavaScript, e executa o comando solicitado.
    3. O Selenium RC é uma extensão do selenium e precisa ser iniciado antes da execução do teste. A iniciação dele pode ser feita a partir da execução do jar com o servidor: java –jar selenium-server.jar. Caso o selenium rc não esteja ativo, a execução dos comandos não será executada com sucesso. O Selenium Rc pode ser obtido a partir do seguinte endereço:

        http://selenium-rc.openqa.org/download.jsp.

7. Chamar o instrutor e demonstrar o teste funcionando.