

Integrando Jenkins com Sonar

Transcrição

O Sonar é uma ferramenta de visualização de dados para facilitar a análise dos dados referentes a qualidade do nosso código. Ele não tem objetivo de analisar o código fonte, assim como fizemos com PMD, por exemplo. Ele só usa os insumos gerados por outros plugins e ferramentas.

Há uma outra aplicação criada pela mesma equipe do Sonar chamada SonarQube Scanner (<http://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner> (<http://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner>)). Neste endereço, podemos quais são as maneiras de adicionar informações no Sonar. Aliás, há uma delas que nos interessa, que é a integração do Sonar com o Jenkins.

Voltando ao Jenkins, na opção "Gerenciar Jenkins", há a opção "Gerenciar Plugins". Nela, podemos clicar em "Plugin ins disponíveis" e filtrar pelo nome `Sonar`. Selecionaremos a opção `SonarQube Plugin`. Depois de selecionando, o Jenkins baixará o plugin para nós.

Com o plugin instalado, vamos para a opção "Gerenciar Jenkins -> Configurar sistema". Nela encontraremos a configuração do Sonar em "SonarQube installations". Clicando no botão `add SonarQube`, ele já adiciona a URL padrão do Sonar, mas ainda assim precisamos dar um nome para essa configuração. Usaremos "SonarQube". Além do que fizemos, precisamos marcar a checkbox "Enable injection of SonarQube server configuration...".

Baixando o SonarQube Scanner pelo Jenkins

Agora, precisamos dizer para o Jenkins o local do Sonar Scanner, o aplicativo que enviará os dados para o Sonar. Para isso, vamos na configuração do Jenkins na opção de configuração de ferramentas. Nele, há a opção do `Sonar Scanner`. Quando clicarmos em "Adicionar SonarQube Scanner", precisamos dizer o nome (podemos escolher qualquer um) e precisamos apontar para o local do nosso sistema de arquivos onde ele está. Só que dessa vez, vamos deixar o Jenkins instalar automaticamente para nós, uma vez que a checkbox desse procedimento já vem marcado como padrão. Depois de alguns instantes o SonarQube Scanner será totalmente baixado.

Configurando mais uma vez nosso Job

Na configuração do nosso Job, em "Ambiente de Build", vamos marcar a opção "Preparar SonarQube Scanner environment". Além disso, precisamos garantir o envio das informações para o SonarQube Scanner. Para isso, em "goals", vamos adicionar o plugin de Maven "Sonar":

```
clean verify sonar:sonar (aqui vem as demais configurações)
```

Agora já podemos realizar o build da nossa aplicação. No lugar de esperarmos os 15 minutos, vamos construir nossa aplicação manualmente. Voltando ao projeto após a construção, vejamos que o Jenkins marcou o projeto integrado ao Sonar, pois existe agora um link para o `SonarQube`. Agora, em um único local, vemos todos os detalhes relacionados a qualidade do nosso código.

