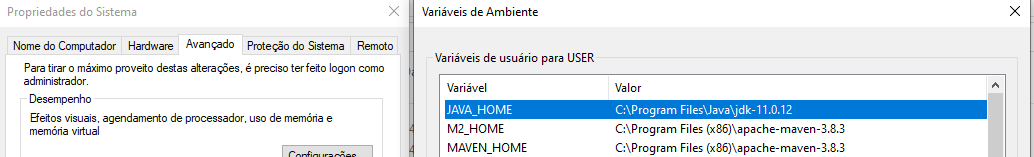
Jenkins X Spring

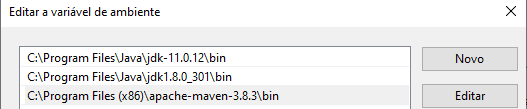
Jenkins:  é um servidor de automação de código aberto. Ele ajuda a automatizar as partes do desenvolvimento de software relacionadas à construção, teste e implantação, facilitando a integração e entrega contínuas.

Spring: é um framework open source para a plataforma Java criado por Rod Johnson e descrito em seu livro "Expert One-on-One: JEE Design e Development".

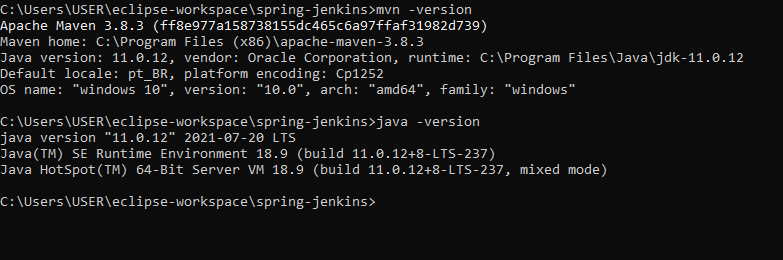
Para que a api se execute corretamente primeiro devemos configurar as variáveis de ambiente com JDK y MAVEN. Jdk se encarga de interpretar y compilar a linguaje java enquanto que maven maneja as dependências que se configuram em el arquivo POM (*é a peça fundamental de um projeto do Apache Maven. Um POM possui as informações básicas de um projeto, bem como as diretivas de como o artefato final deste projeto deve ser construído)*

Se configura JAVA\_HOME e MAVEN\_HOME com os arquivos descargados de oracle



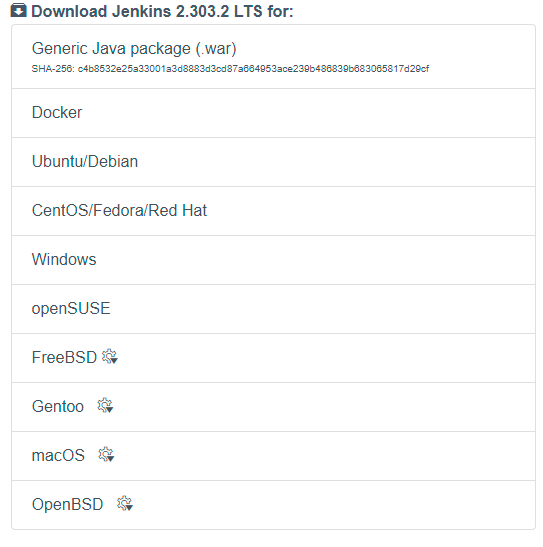


Verificamos que se haja instalado corretamente mediante a linha de comando de Windows

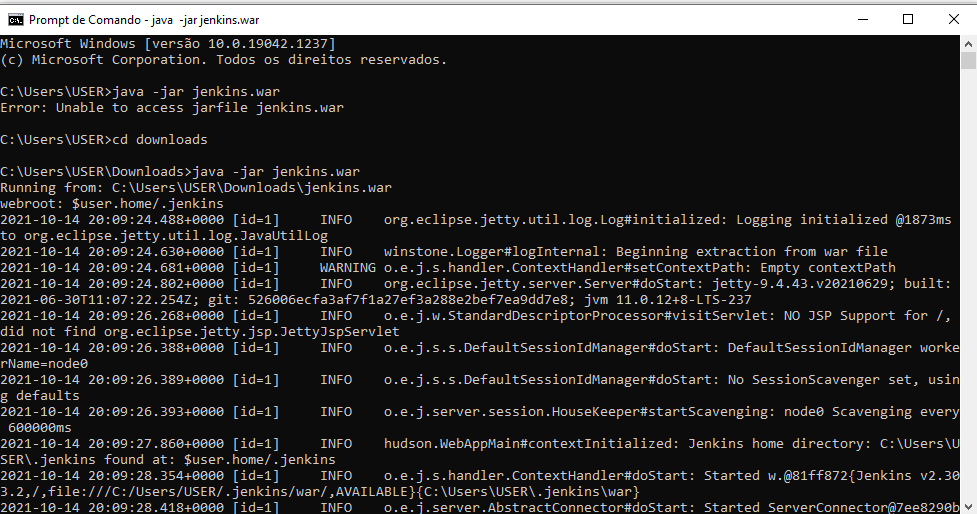


Descarregamos Jenkins da página oficial, existem várias opções, o instalador para Windows y o executável de java .war (Jenkins e uma api desenvolvida em java)



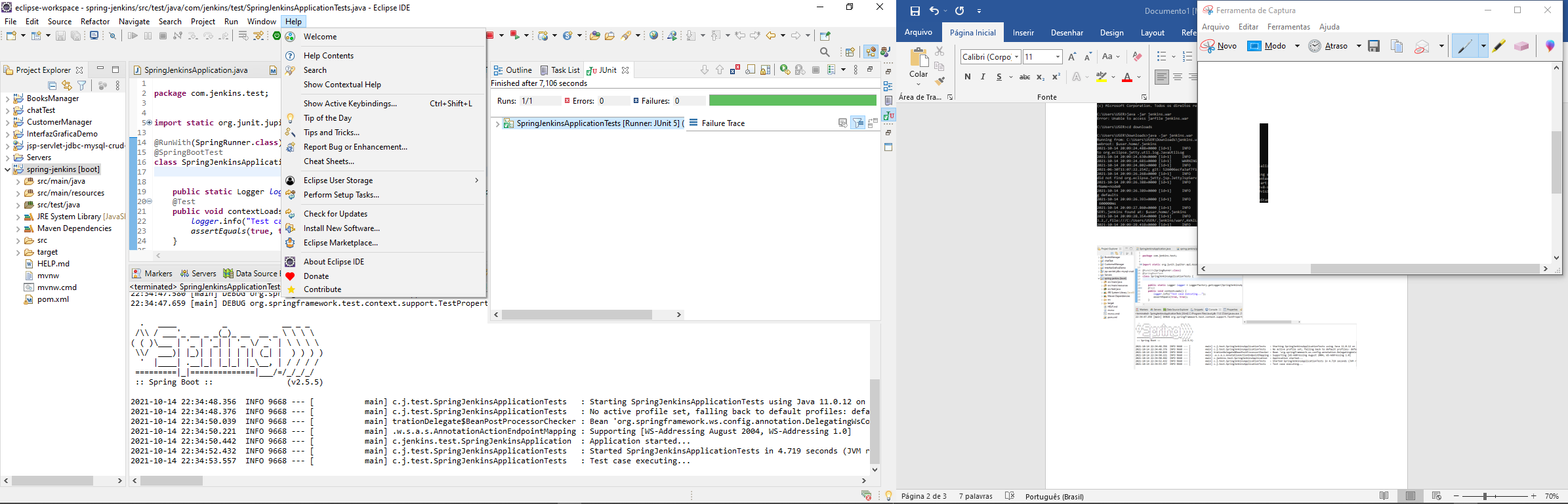
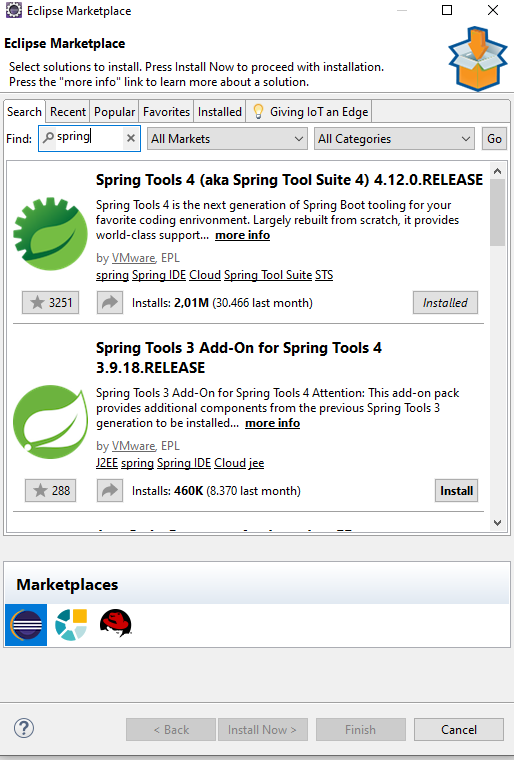


Para executar Jenkins e importante desativar qualquer antivírus que possa identificar Jenkins como um arquivo malicioso y executamos em a línea de comando na pasta descarregada



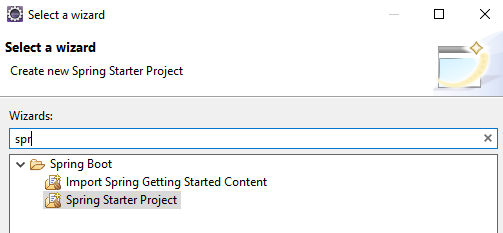
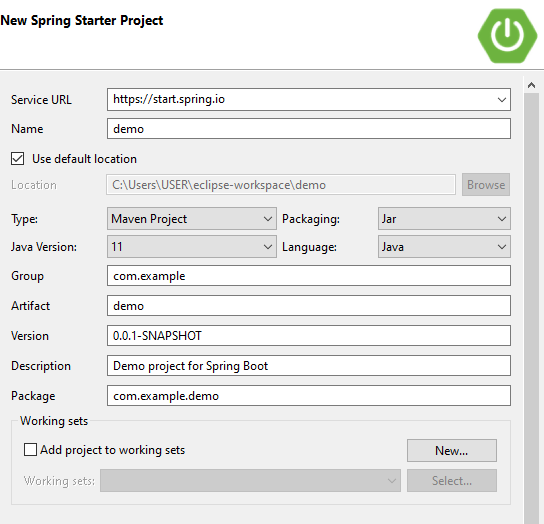
Vamos a testar uma api java-spring dentro do ambiente de desenvolvimento de eclipse, existem várias opções uma de elas e o próprio ambiente de spring-boot, initializer de spring.io o intellij. as instruções de como configurar eclipse se encontram a continuação y as dependências dependeram de cada projeto que se deseje testar

1. 2)

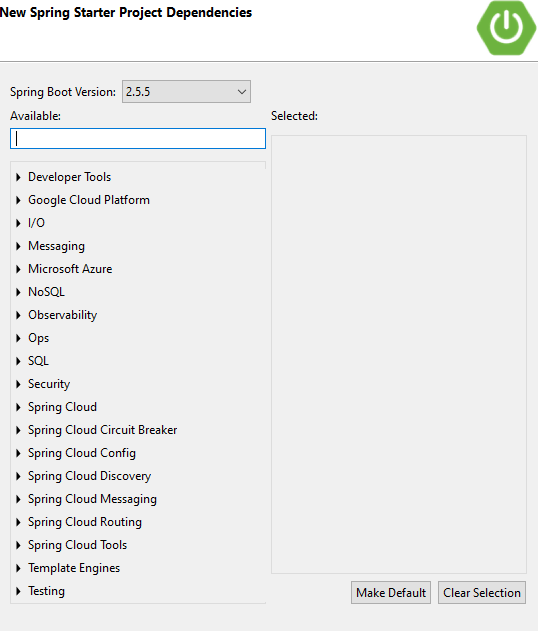




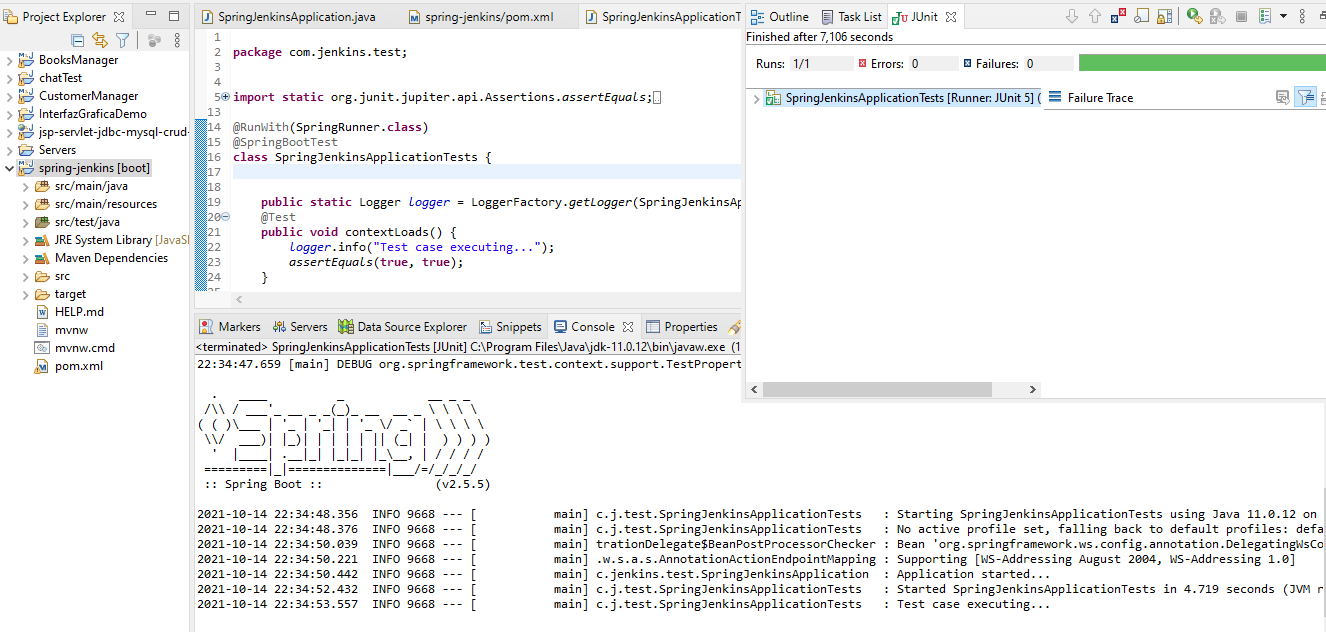
4)



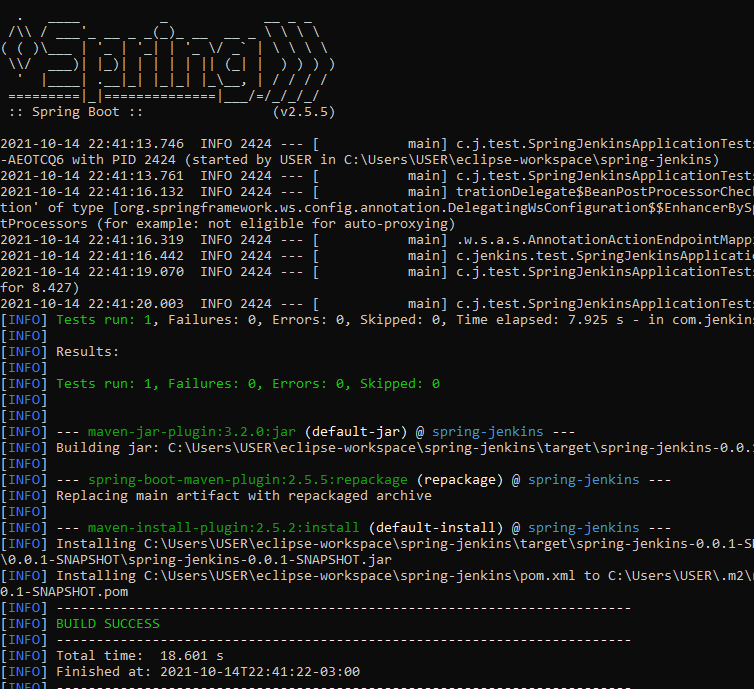
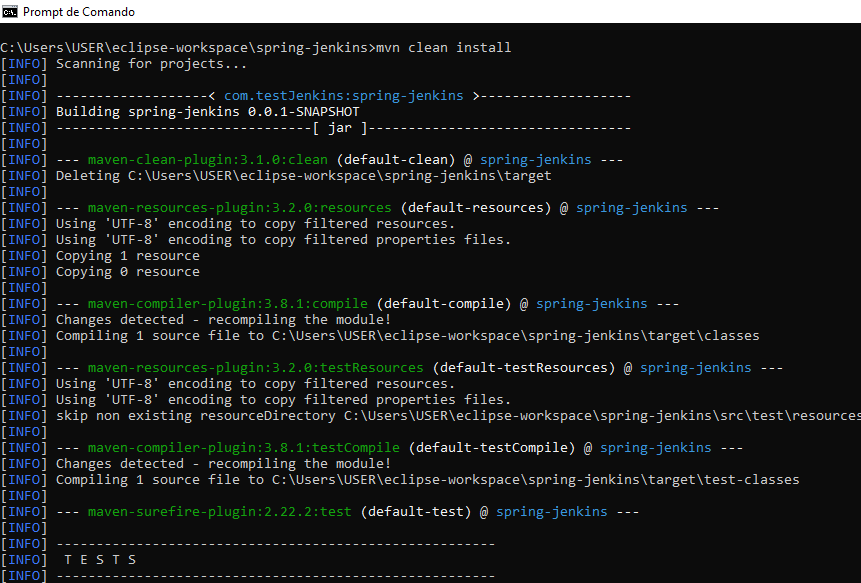
5)



Um exemplo de um teste executado dentro de eclipse

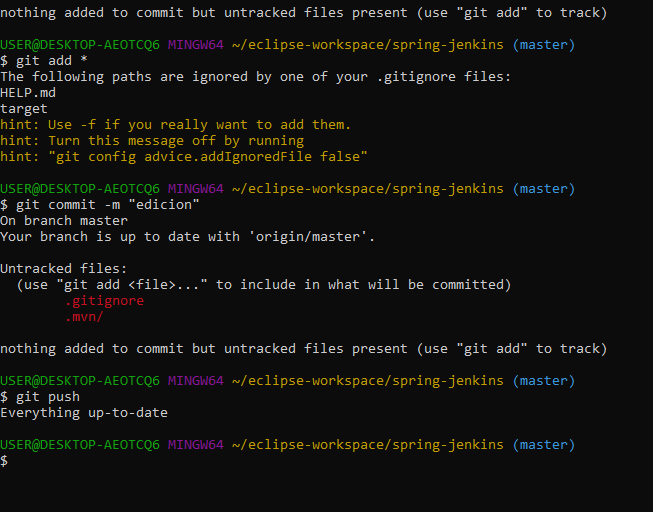
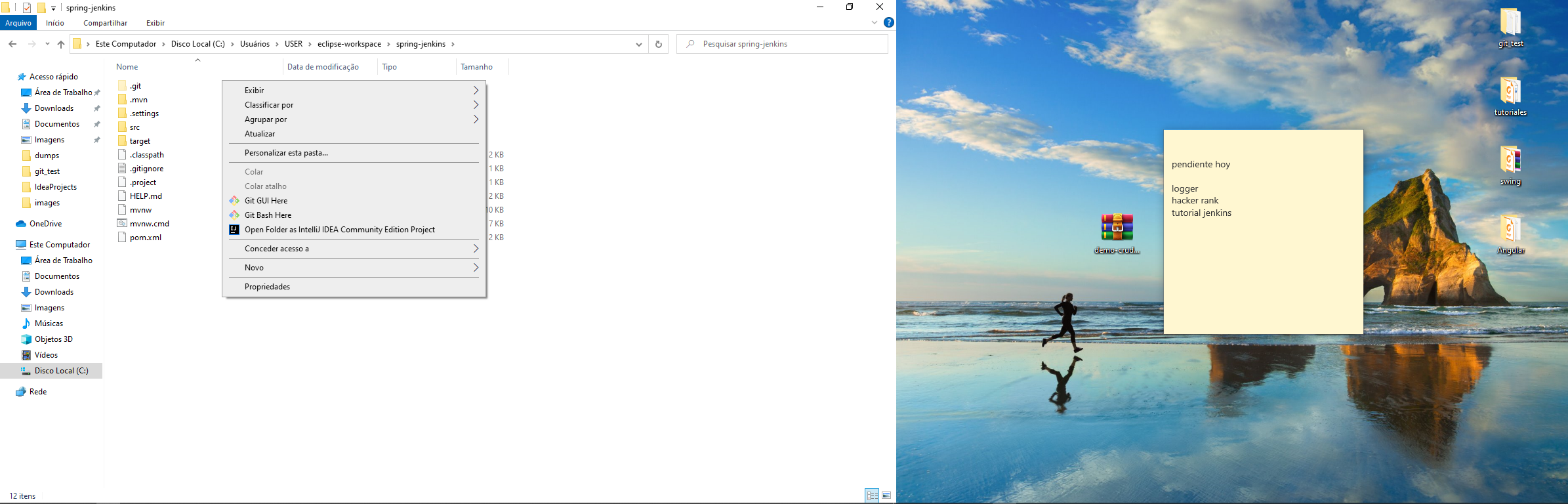


Y também compilamos mediante o comando *mvn clean install,* donde se verifica que todo este funcionando corretamente dentro das configurações do sistema atual

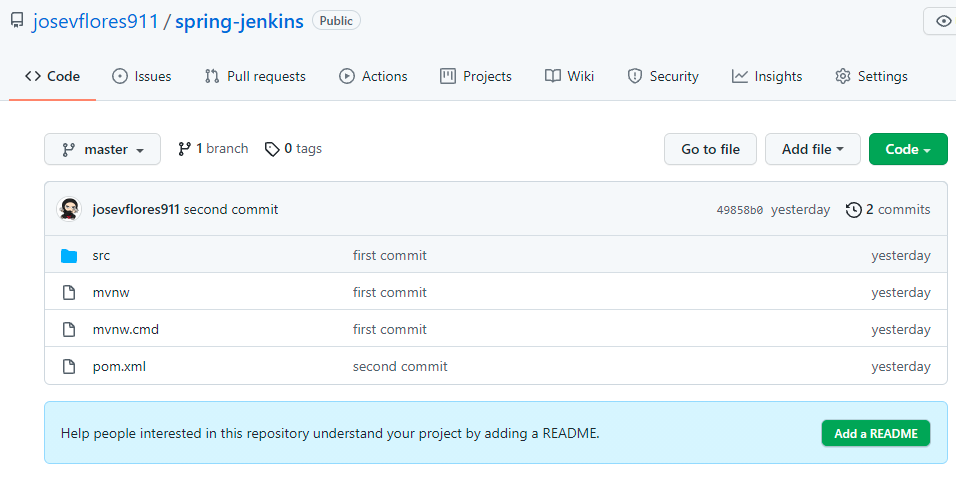


Jenkins trabalha em conjunto com um repositório em este caso se usara github y se subira a informação usando git a a carpeta donde se encontra a api Spring

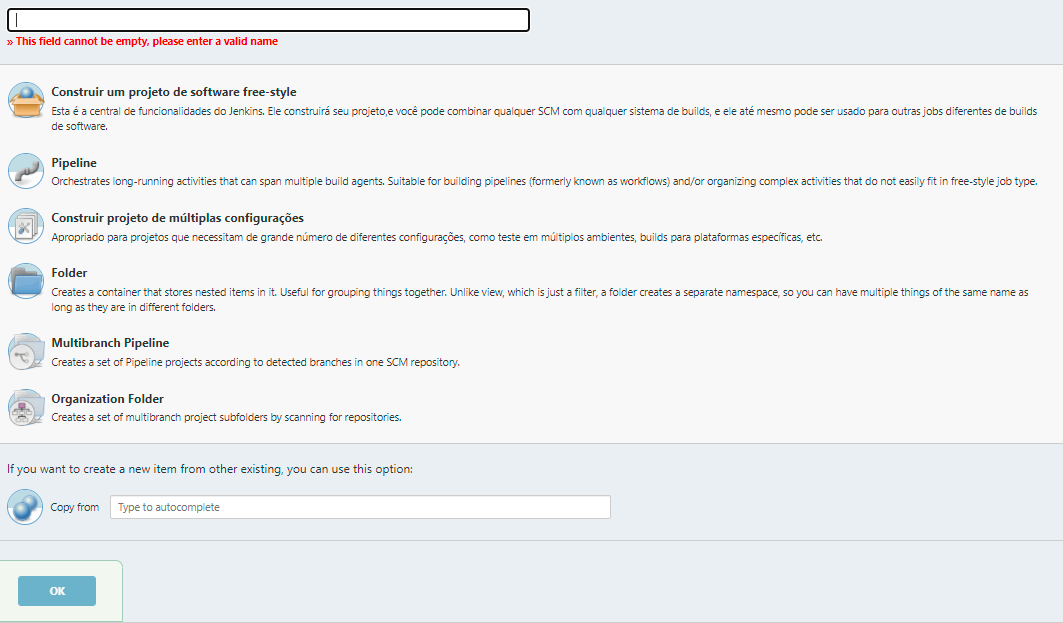
1. 2)



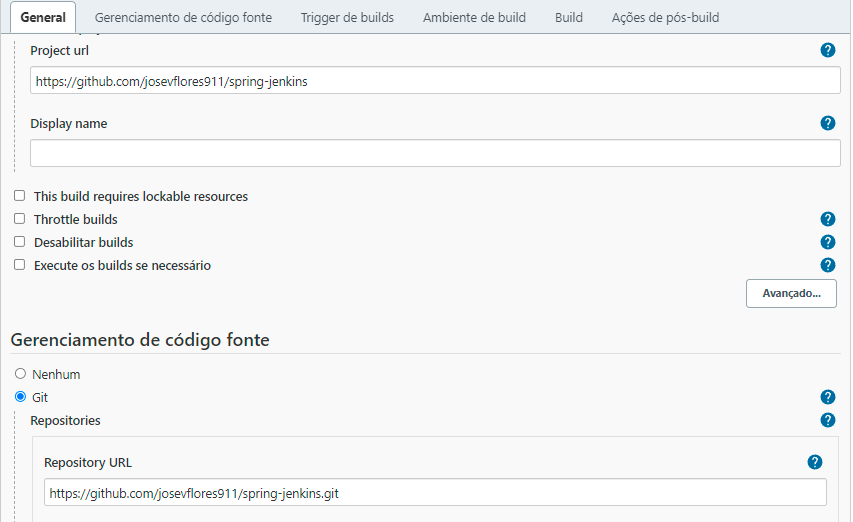
3)



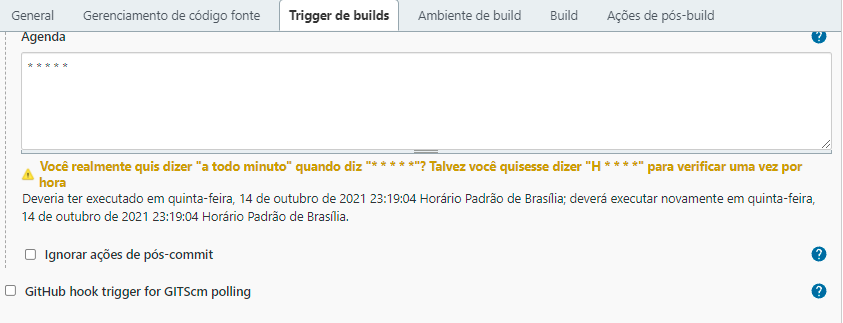
Criamos um novo projeto

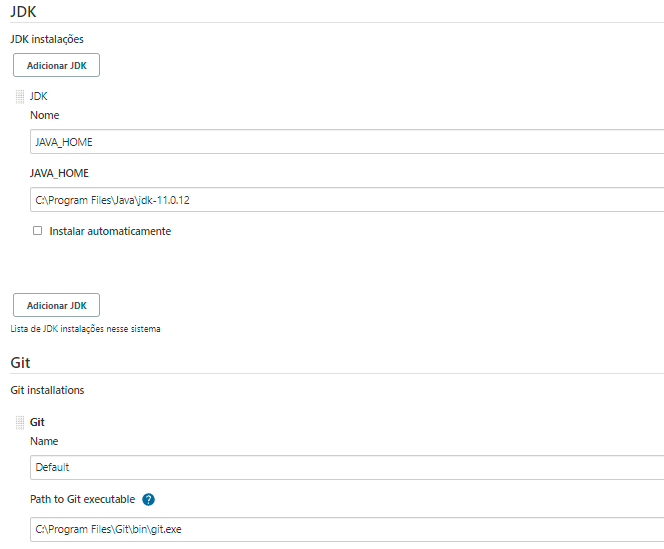
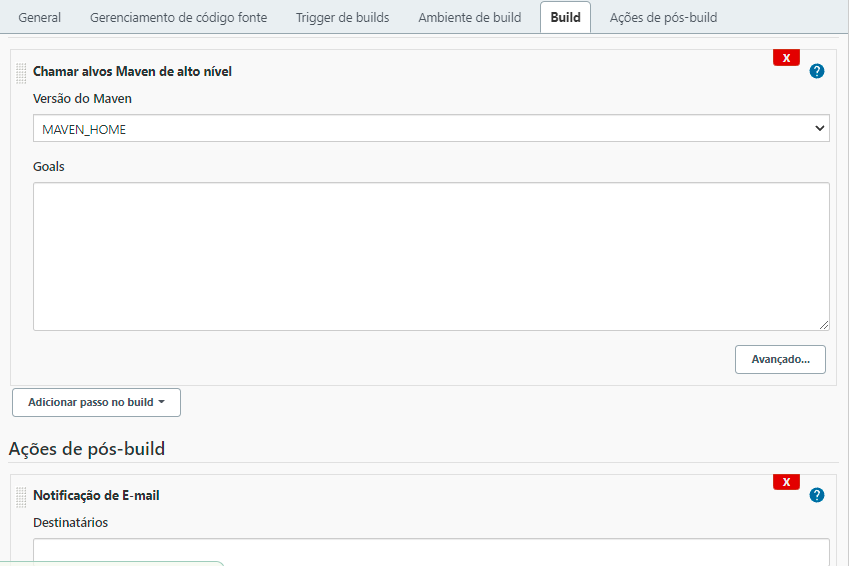


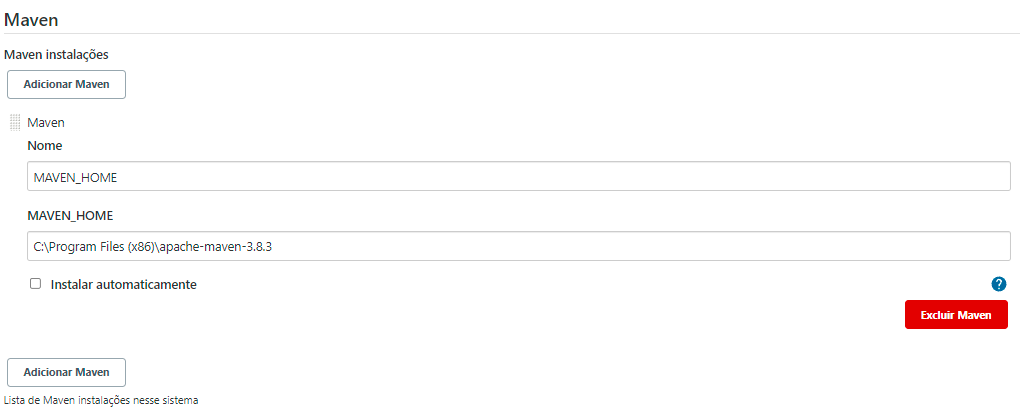
Configuramos com a informação de nosso repositório



Selecionamos a frequência com a que será feita o teste







Jenkins sugere java 11 para su ejecucion pero internamente permite executar diversas versões de java hasta el 9 (al momento de editar este tutorial)