Jenkins项目构建流程

目录

[1.新建Jenkins项目 2](#_Toc495487365)

[2.项目设置 3](#_Toc495487366)

[2.1 项目基础设置 3](#_Toc495487367)

[2.2设置项目代码来源信息 3](#_Toc495487368)

[2.3配置项目触发器 4](#_Toc495487369)

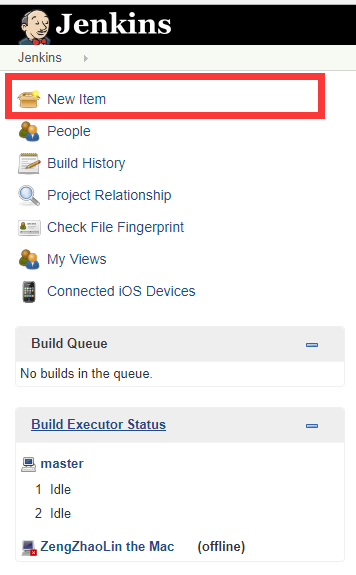
[2.4配置项目构建环境参数 5](#_Toc495487370)

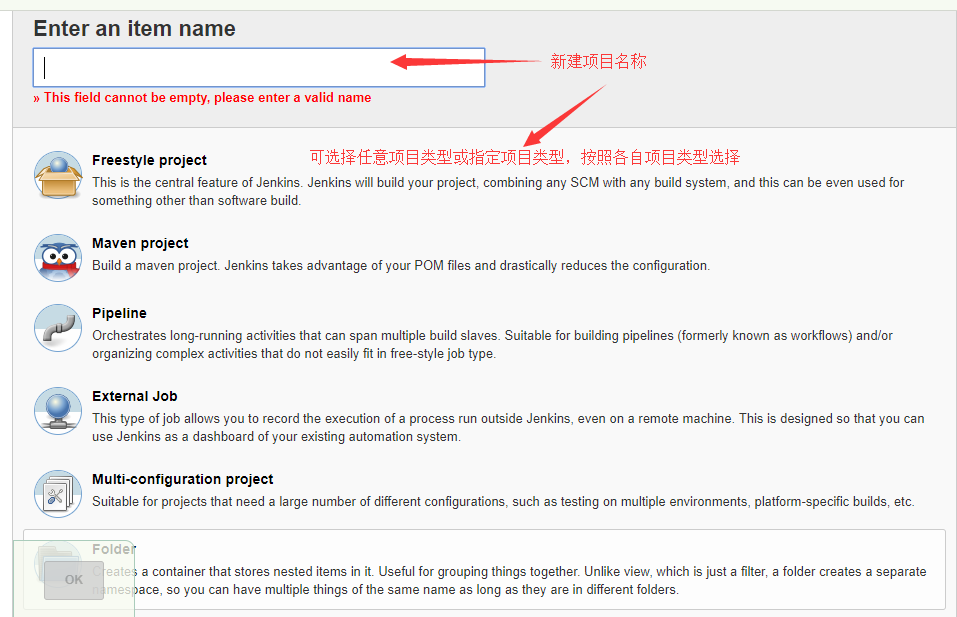
[2.5配置项目构建完成后执行命令 5](#_Toc495487371)

[3.项目保存并构建 5](#_Toc495487372)

[4.项目构建情况 6](#_Toc495487373)

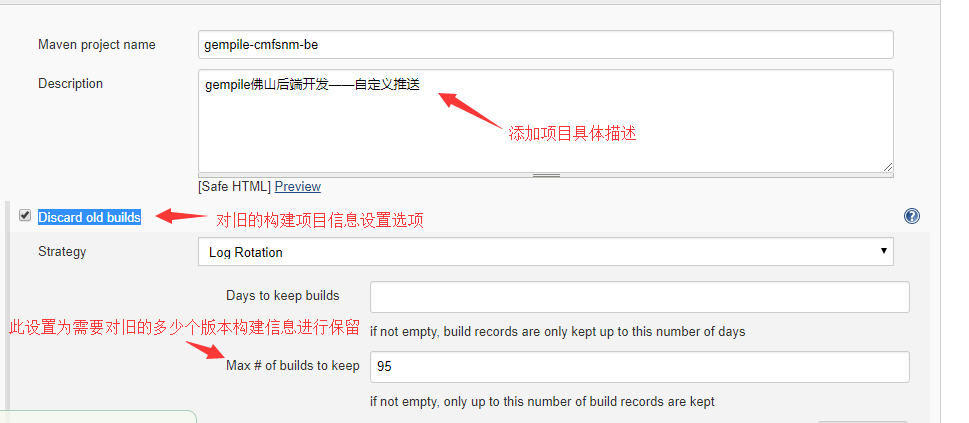
# 1.新建Jenkins项目



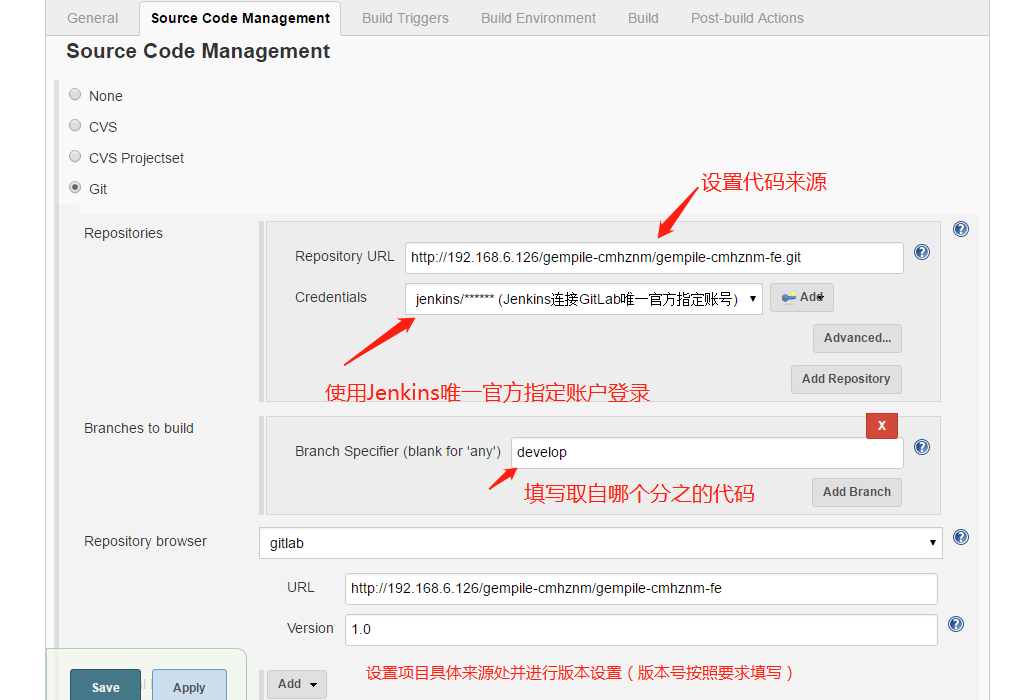


# 2.项目设置

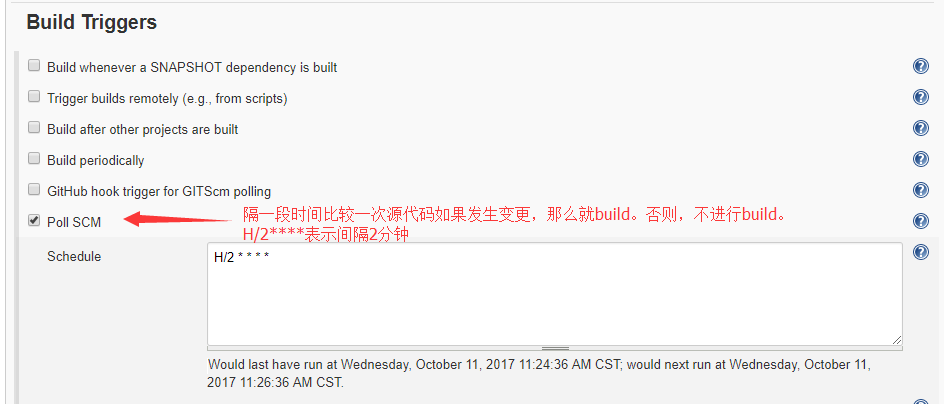
## 2.1 项目基础设置



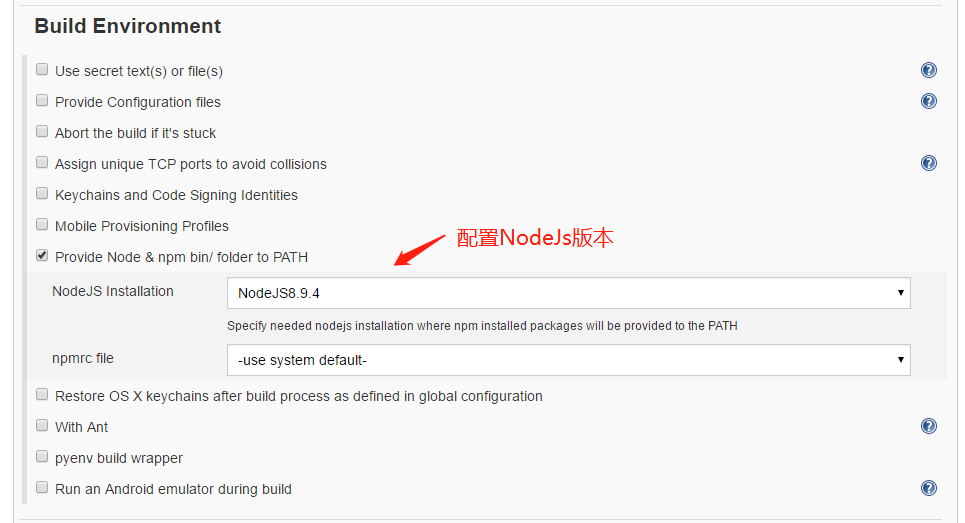
## 2.2设置项目代码来源信息



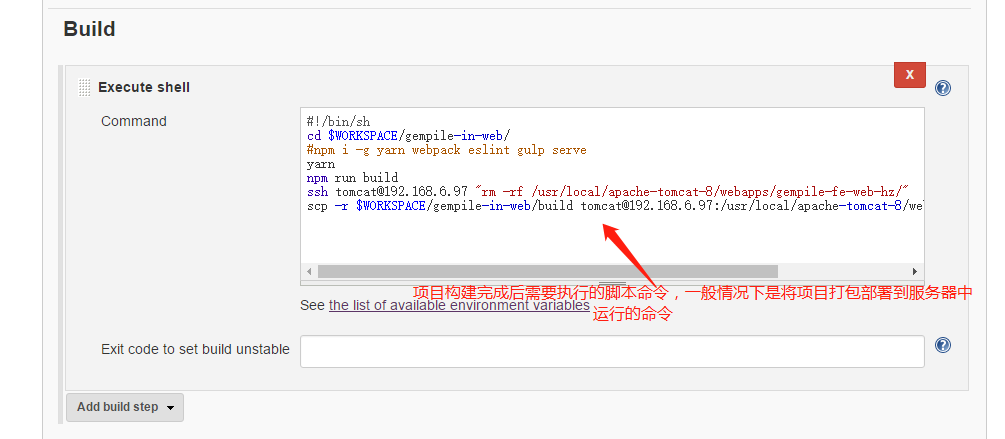
## 2.3配置项目触发器

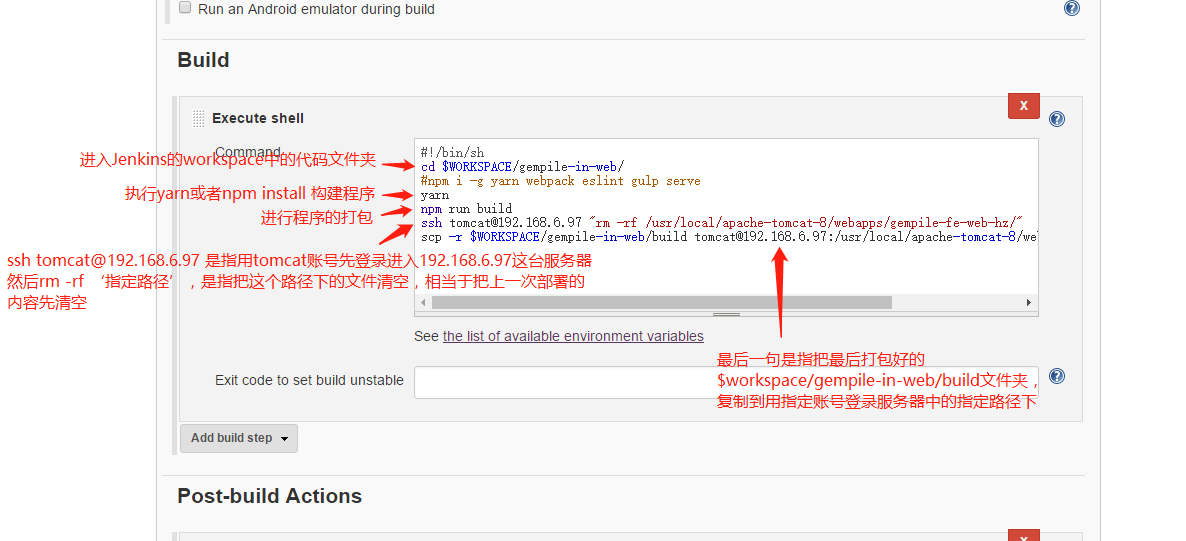


## 2.4配置项目构建环境参数



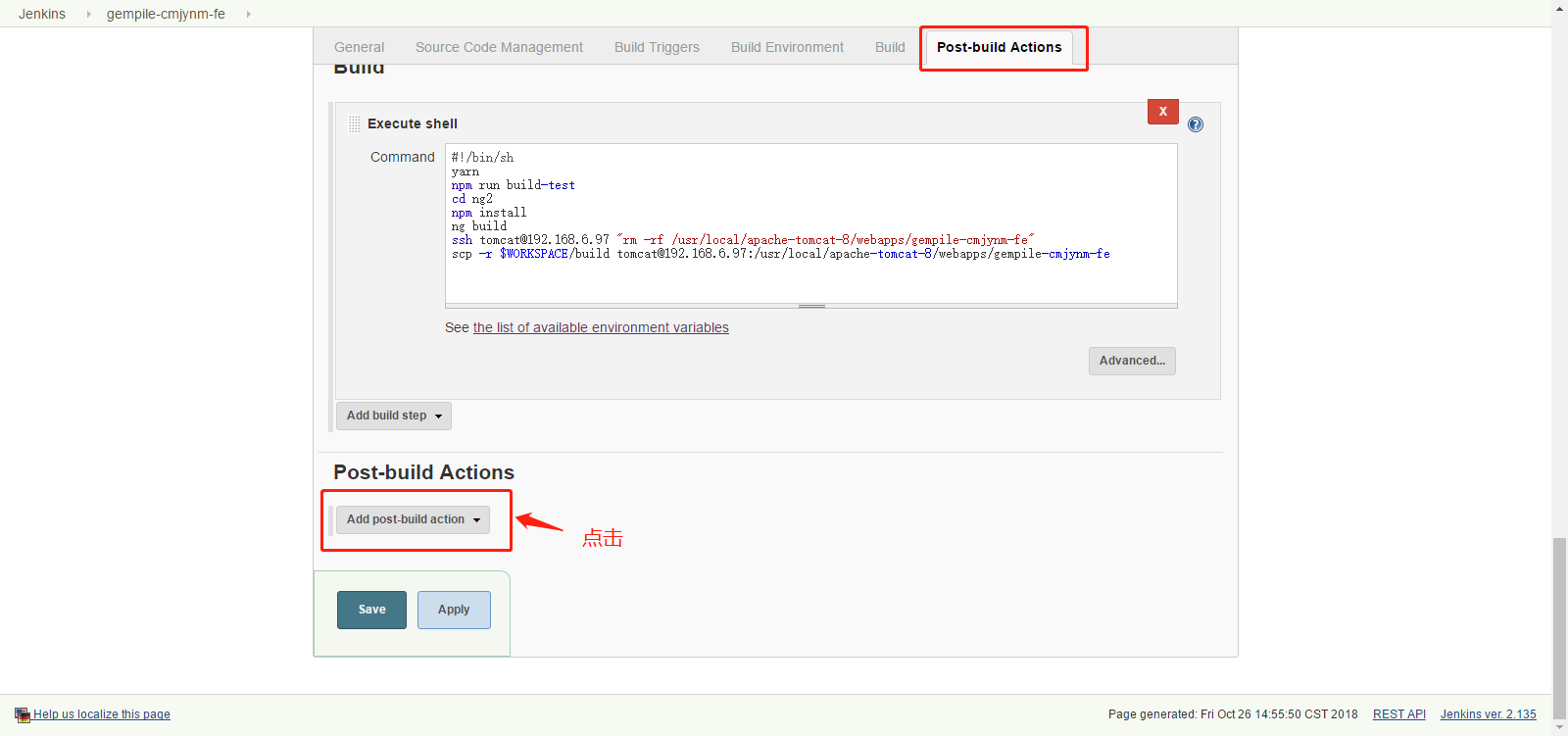
## 2.5配置项目构建完成后执行命令

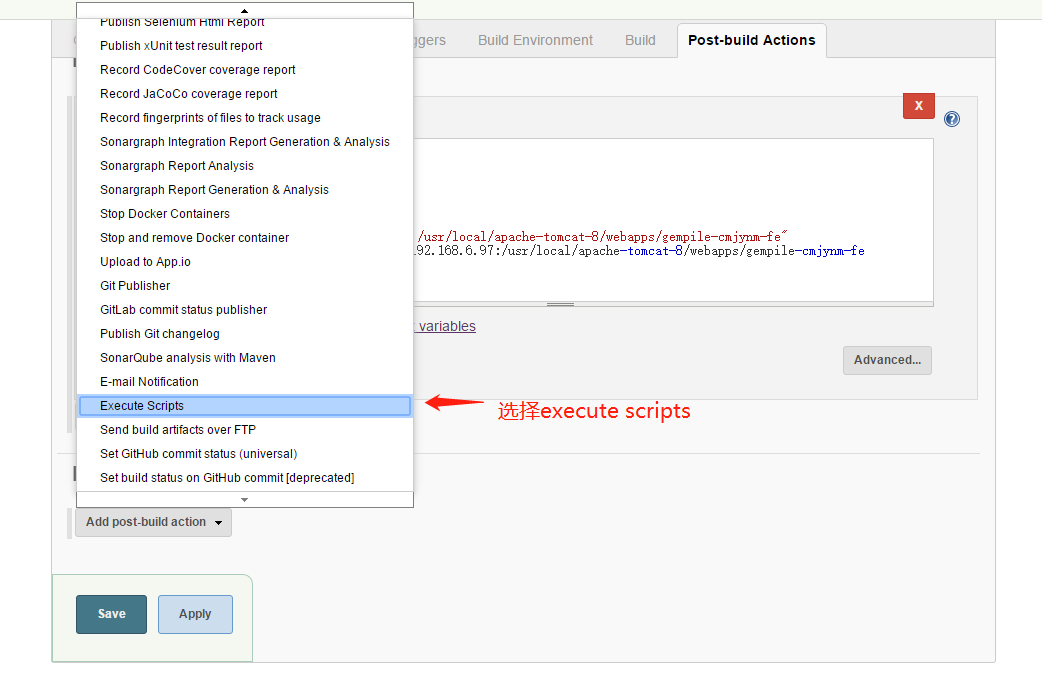


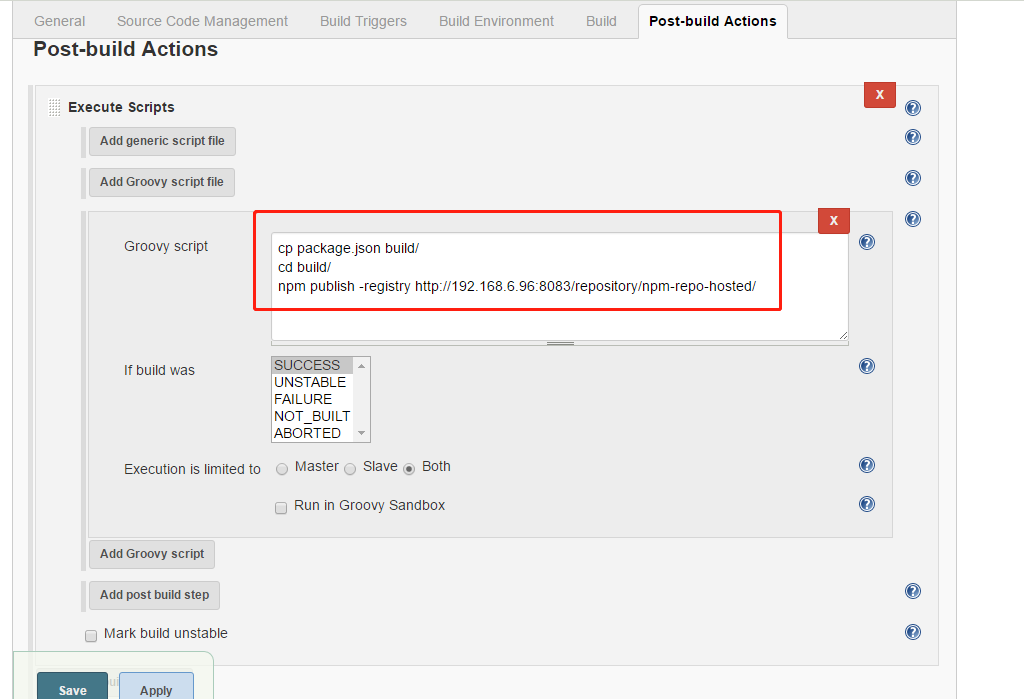


# 配置nexus到Jenkins中

在jenkins中配置post step，如果build成功，execute shell执行如下命令，发布到nexus上：





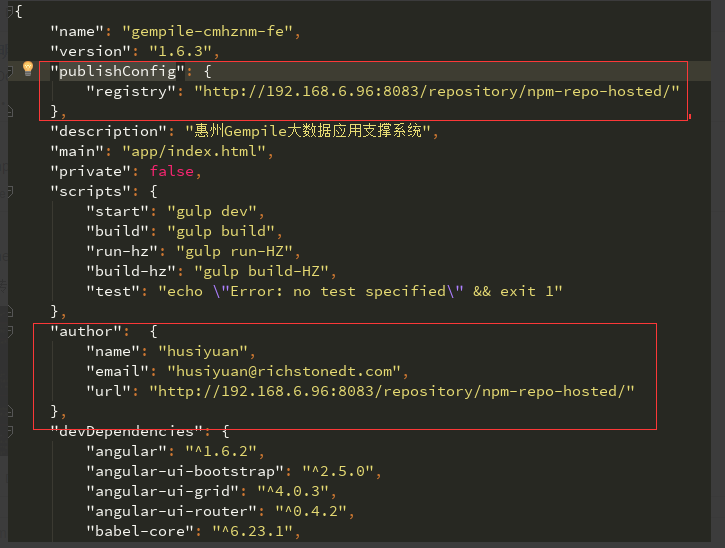


至于为什么要跳到build文件夹里执行npm publish是因为如果直接在跟目录文件夹执行npm publish的话会把整个源码上传而并不是把打包好的可直接运行的代码上传到nexus

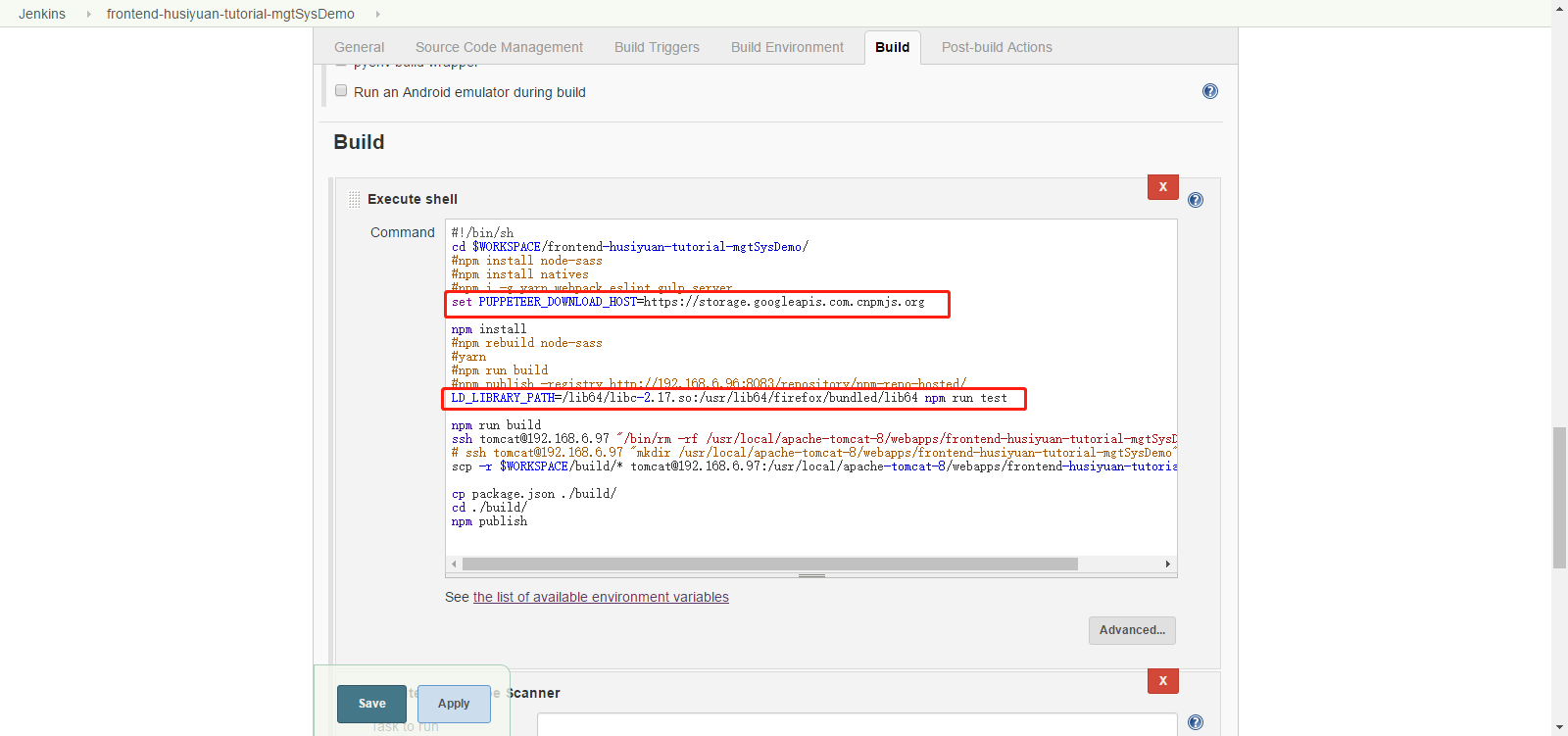
所以需要把package.json这个配置文件复制一份放到打包好的代码文件夹中，再跳到里面执行代码发布，就能只上传打包好的代码包了。

注意：

* 每次发布前需要配置package.json文件中的version，如果Nexus上已存在同一版本的话，将会发布失败。
* 在package.json文件中，如果设置了private属性为true的话，将不会发布到Nexus上，所以需要设置为false或去除此属性。
* 在package.json中记得配置author属性和description描述



# 配置sonar到Jenkins中



set PUPPETEER\_DOWNLOAD\_HOST=https://storage.googleapis.com.cnpmjs.org

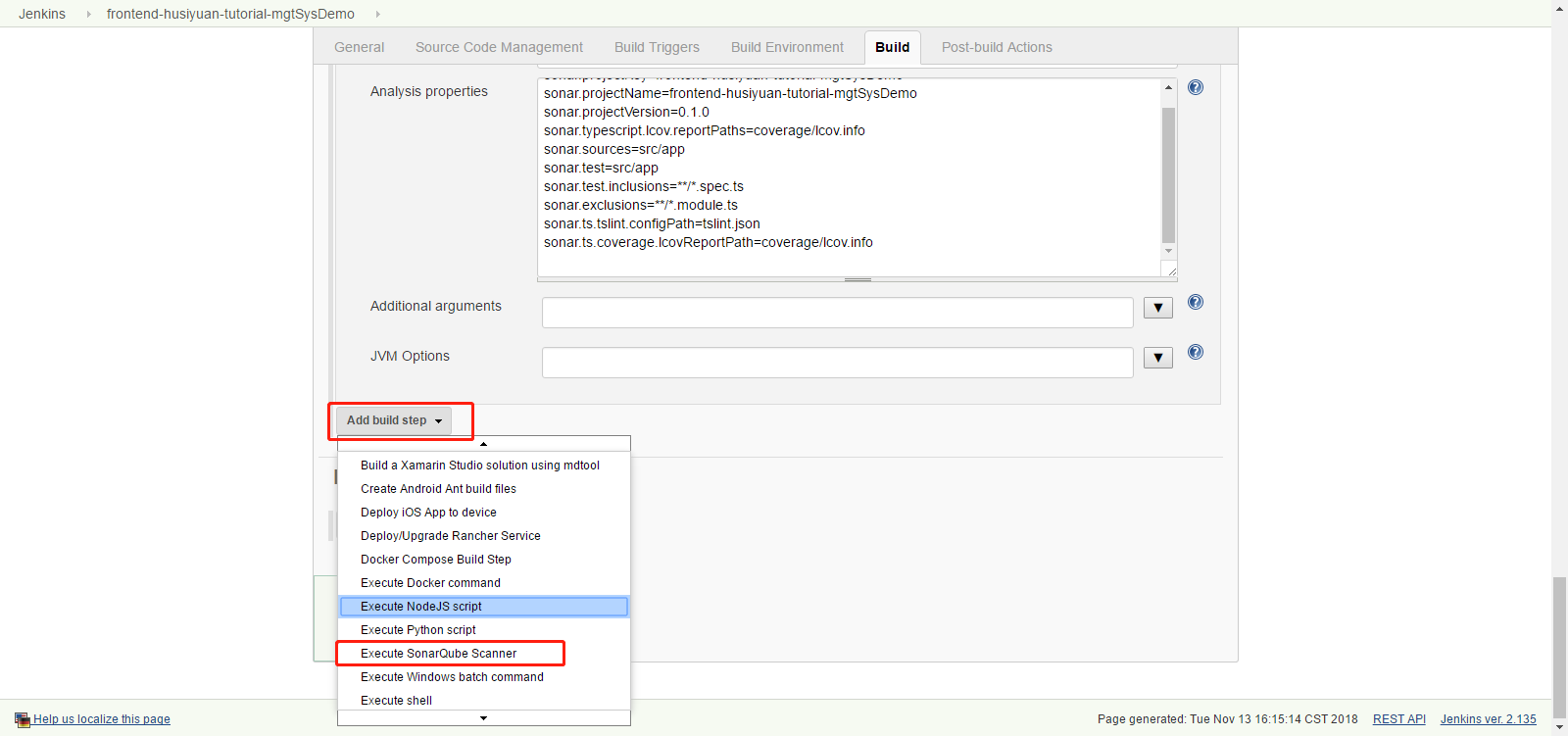
是为了修改puppeteer库的下载地址，该库的目的是在后台启动Headless Chrome用于运行无界面的单元测试

LD\_LIBRARY\_PATH=/lib64/libc-2.17.so:/usr/lib64/firefox/bundled/lib64

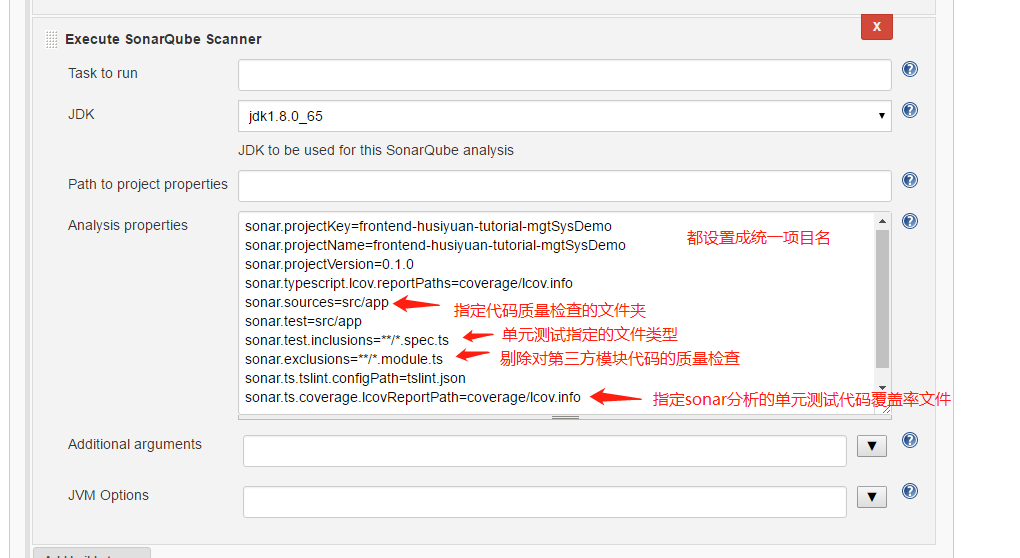
是为了动态得给npm run test这条命令配置动态链接库的路径集合，目的同上

## 启动SonarQube Scanner

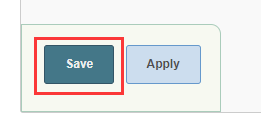
在Build模块中，点击 Add build step，然后选中Execute SonarQube Scanner

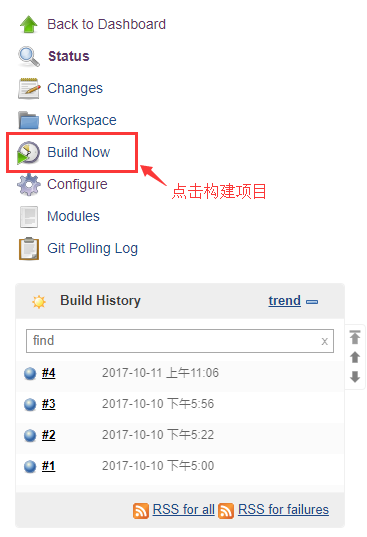


修改当中配置：



# 3.项目保存并构建





# 4.项目构建情况



# 5.注意事项

1、设置环境参数时，Build选项中的Root POM和Goals and options 选项需要根据不同项目的pom路径的不同与项目构建打包命令的不同进行填写。

2、Execute Shell的Command中根据不同的项目打包和部署方式的不同进行设置填写，注意war包重命名部署至tomcat后，url访问路径也需要随之改变。