YIGH288V1

Automation Framework setup in Jenkins

Contents

[Automation Framework Jenkins setup for Keyword XML project 3](#_Toc469404561)

[Automation Framework Jenkins Setup for Java TestNG project 11](#_Toc469404562)

[Jenkins for Build Pipelines (Setup Build triggers for Upstream or Downstream Projects) 14](#_Toc469404563)

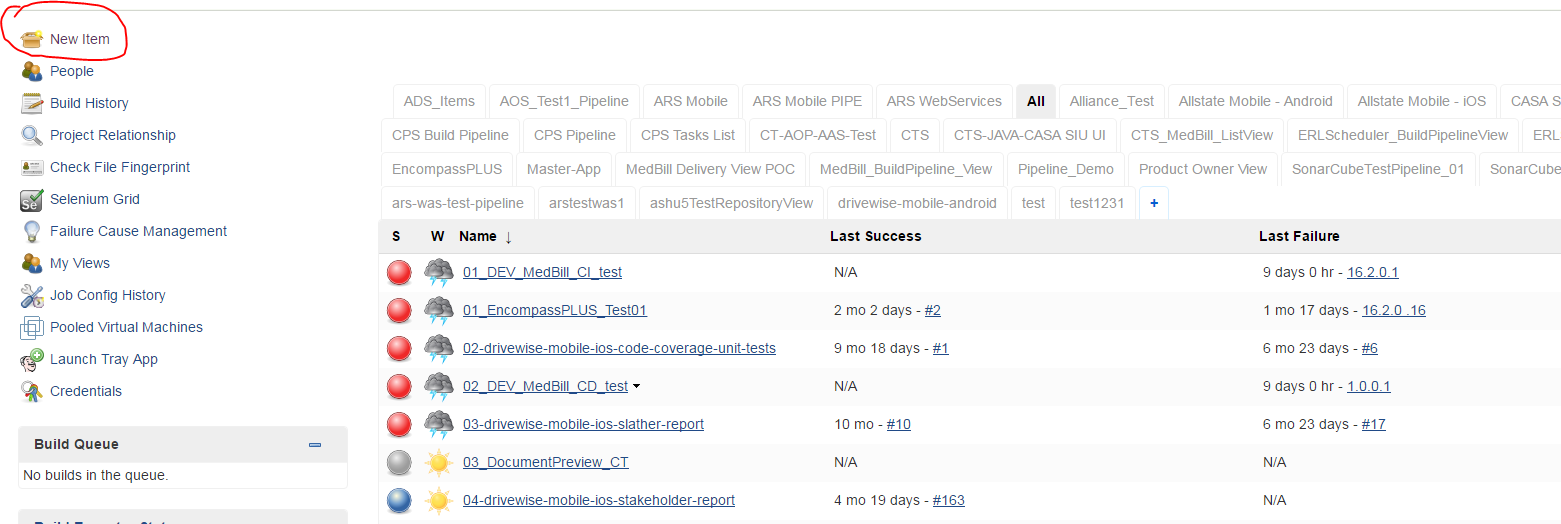
[Build after other projects are built 14](#_Toc469404564)

[Build other projects 14](#_Toc469404565)

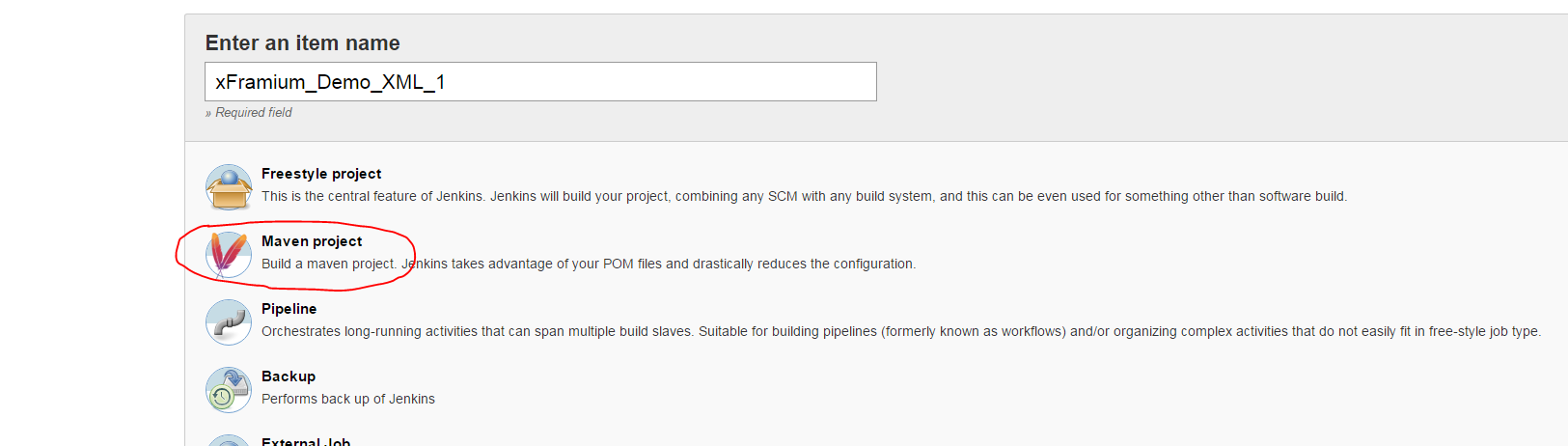
[Build Periodically 15](#_Toc469404566)

# Automation Framework Jenkins setup for Keyword XML project

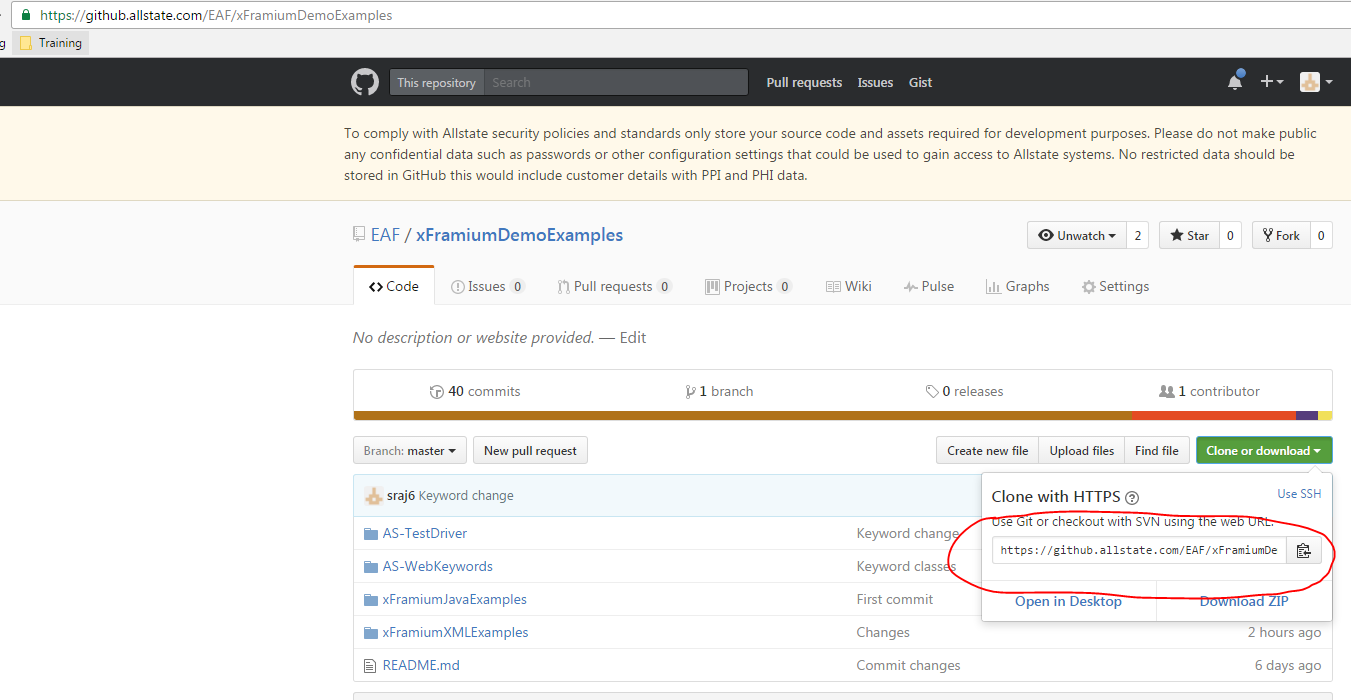
1. Create new Jenkins Job by clicking on new item



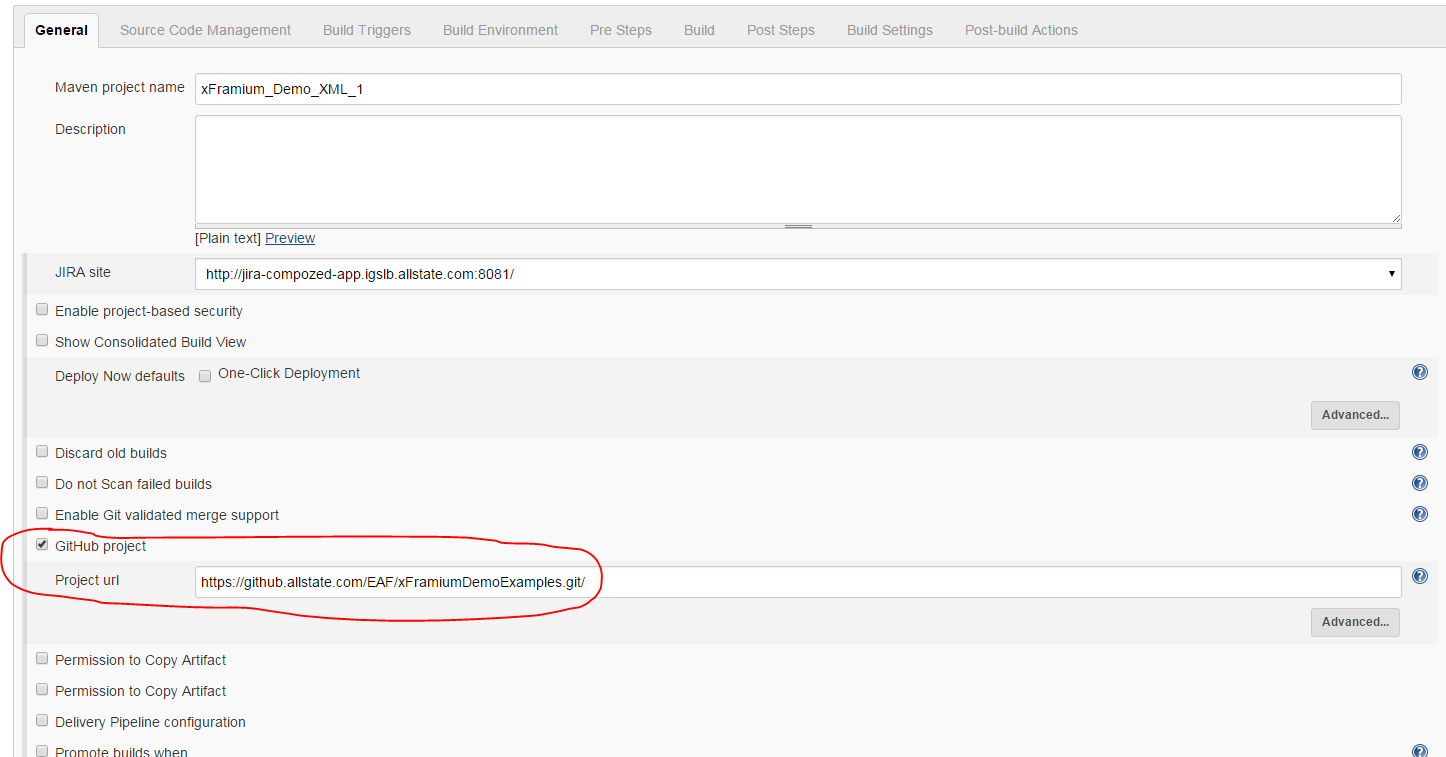
1. Select Maven project and enter the name of the Job



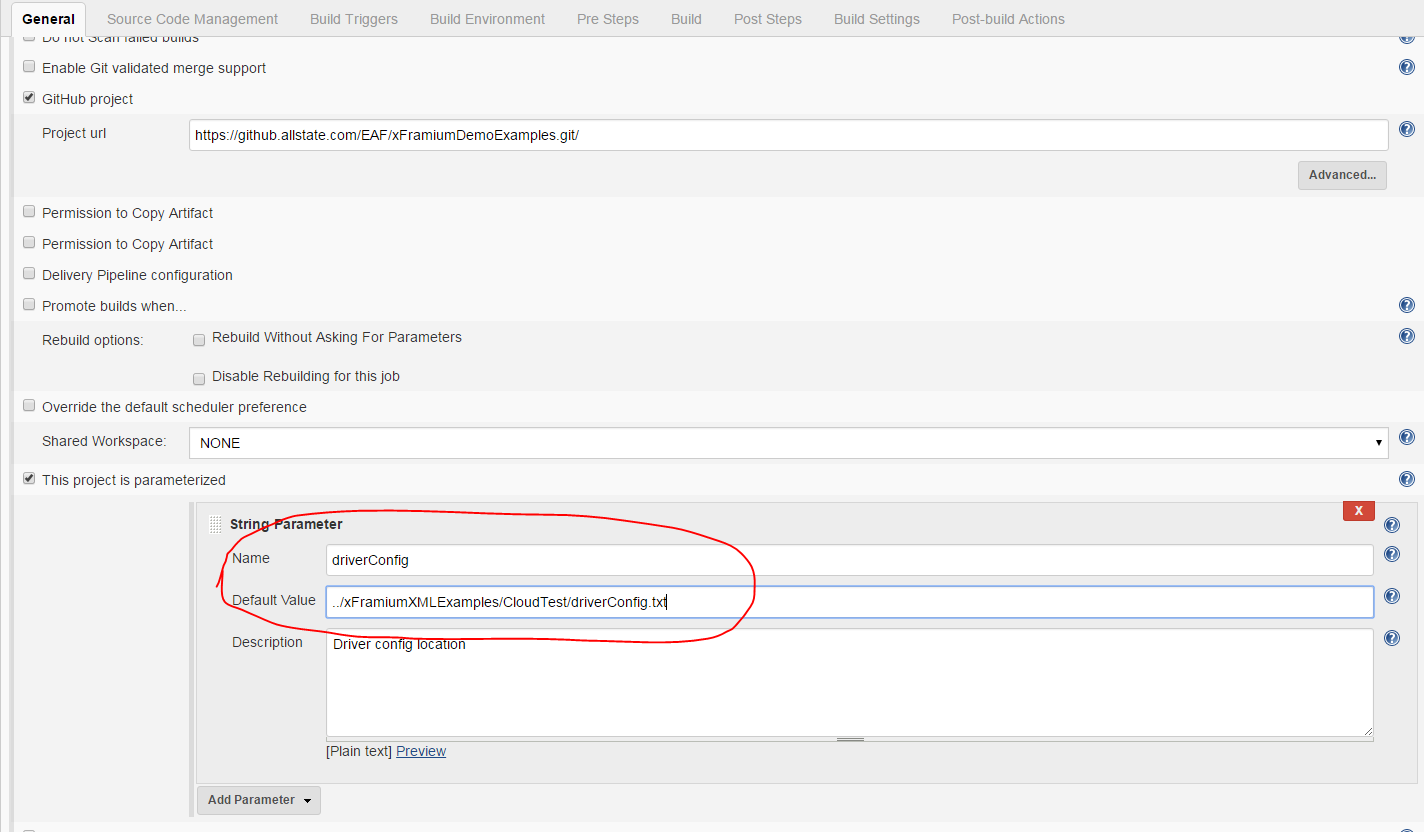
1. Under the general section, select Github Project > Specify Project URL. The GIT Project URL can be accessed from Allstate github site for your GIT repository



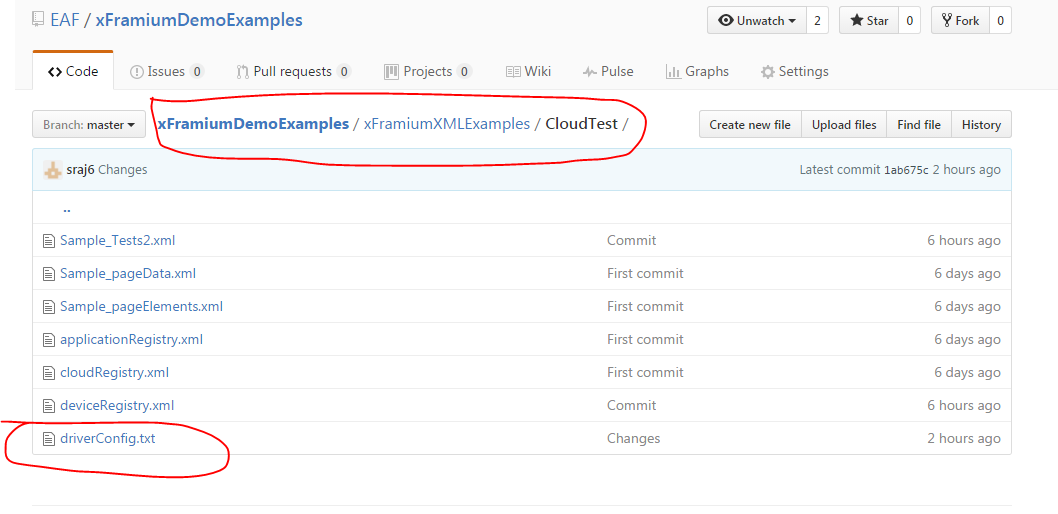
Copy the URL into the project URL folder below



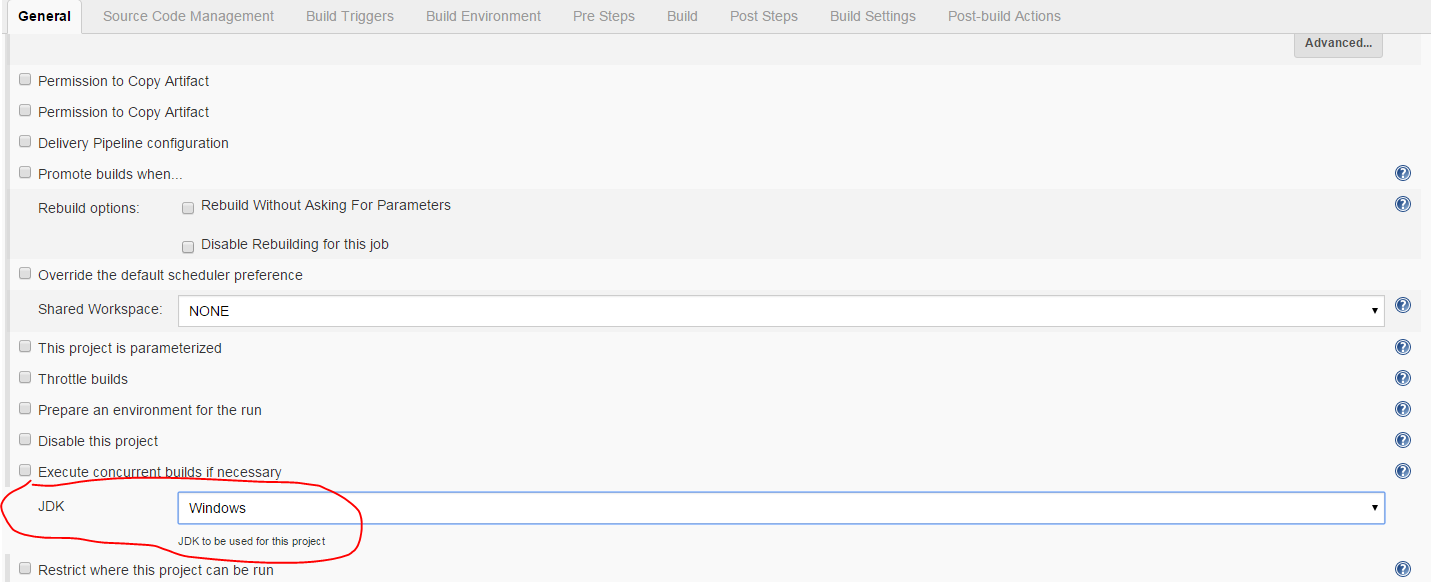
1. Configure the project Parameters as shown below. Select “This project is parameterized” option > Add parameter of type String. Specify the location of your driverConfig.txt



Make sure the folder “xFramiumXMLExamples”, sub folders and driverConfig.txt are available in the specified GIT repository



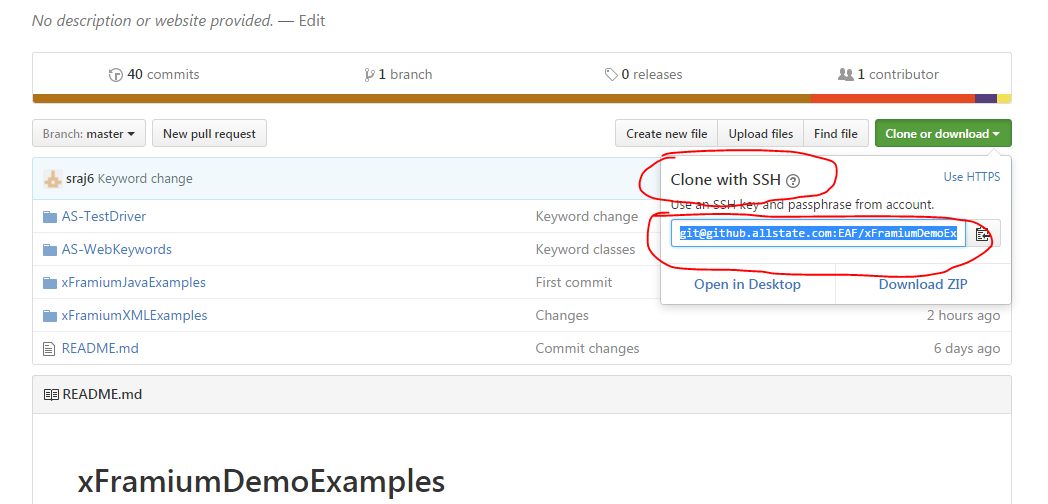
1. Configure JDK to be used for compilation of the build. Select Windows in the list box as shown below.



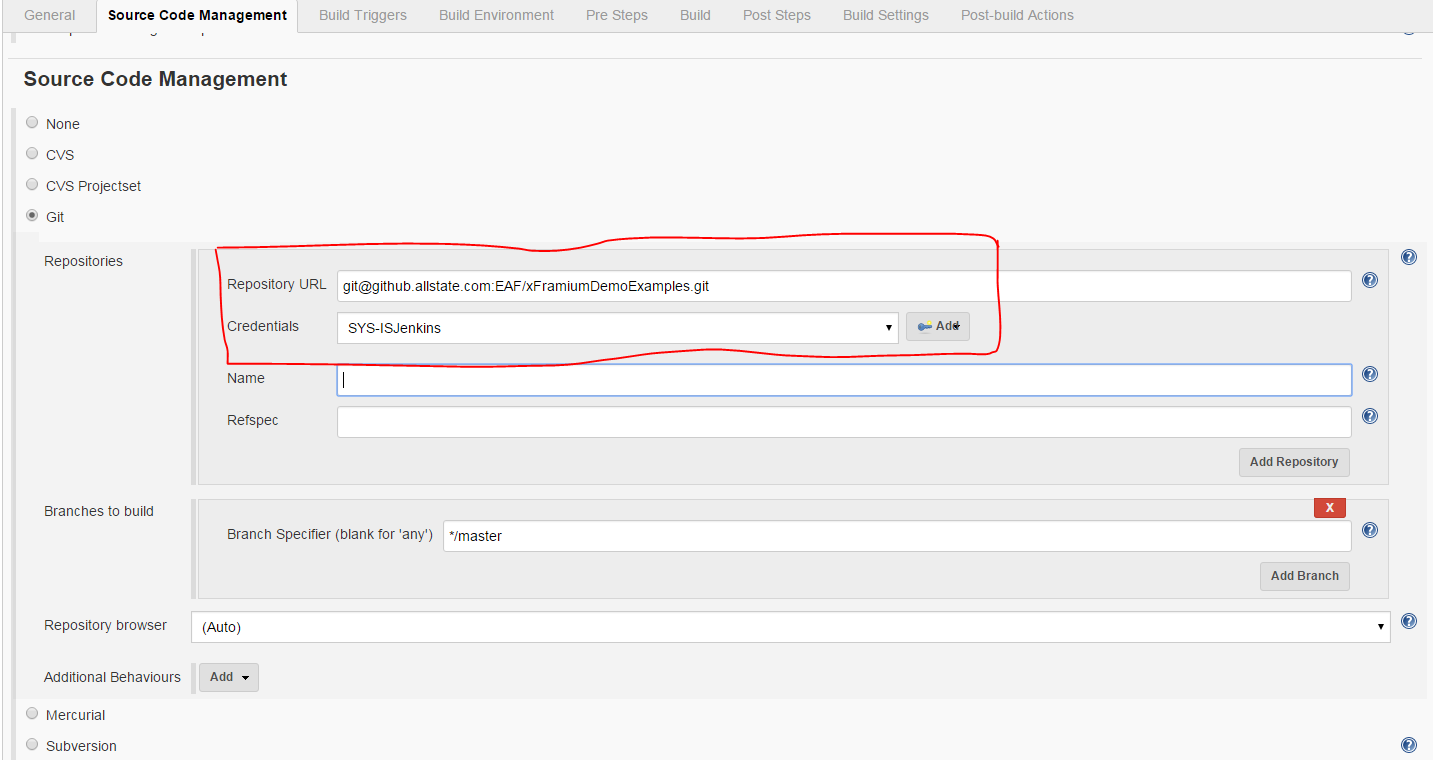
1. To specify where the project/build should be run, please check “Restrict where the project can be run” and mention the slave machine name. If the below screenshot we are using a windows slave machine. If this option is not checked, Jenkins will pick the any available machine



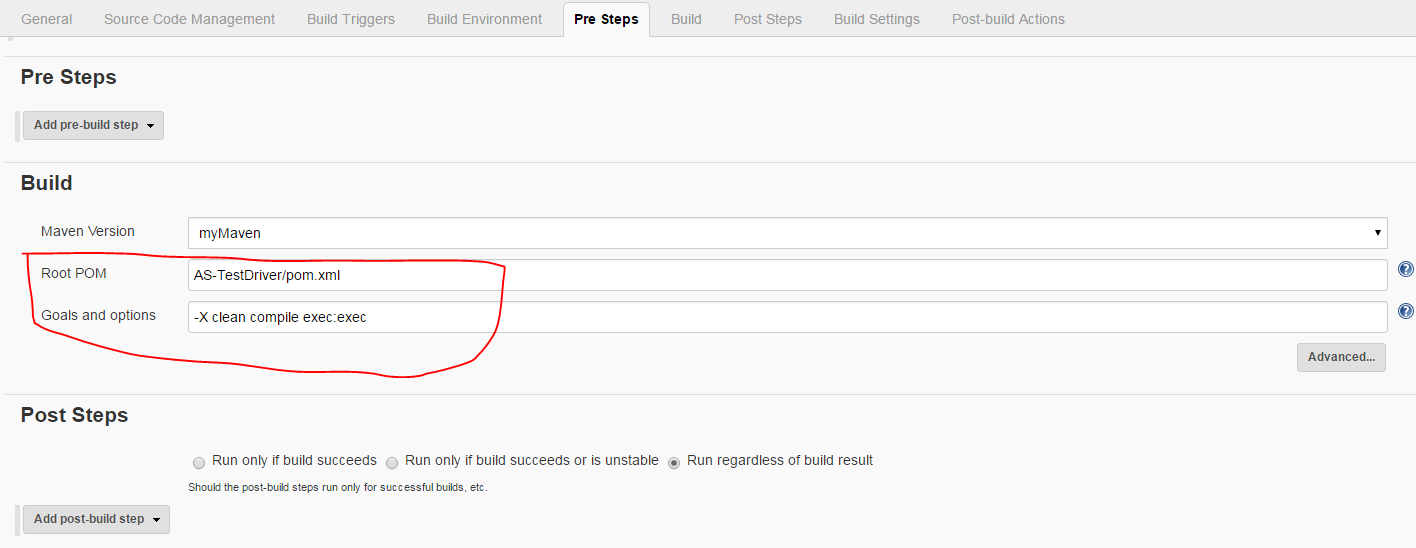
1. Under source code management > Select GIT > Specify repository URL and credentials. URL should be SSH clone URL of GIT repository. Credentials specified should have super user access. Select user “SYS-ISJenkins”



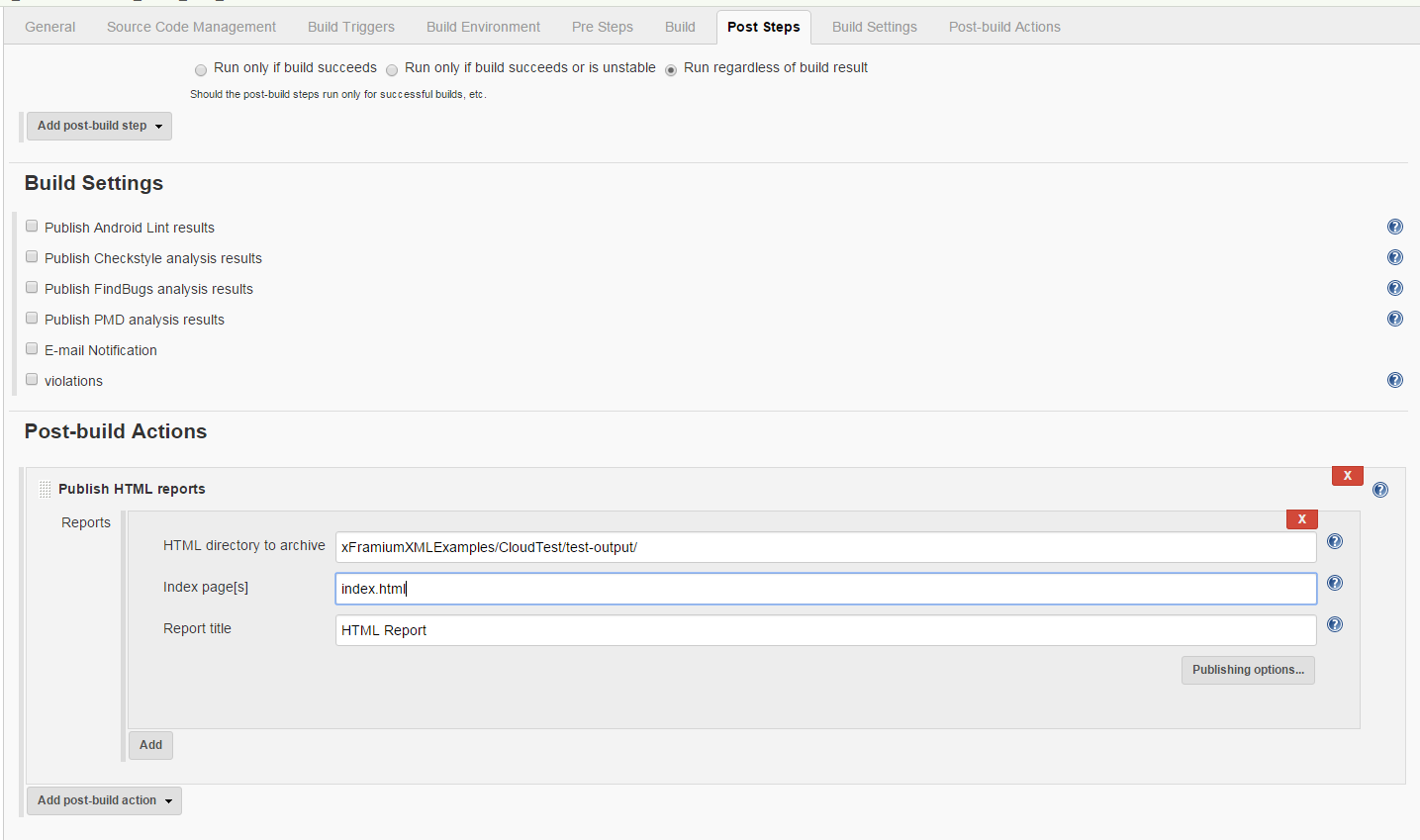
Other parameters such as “Branches to build” and “Repository browser” should be the default values as shown below



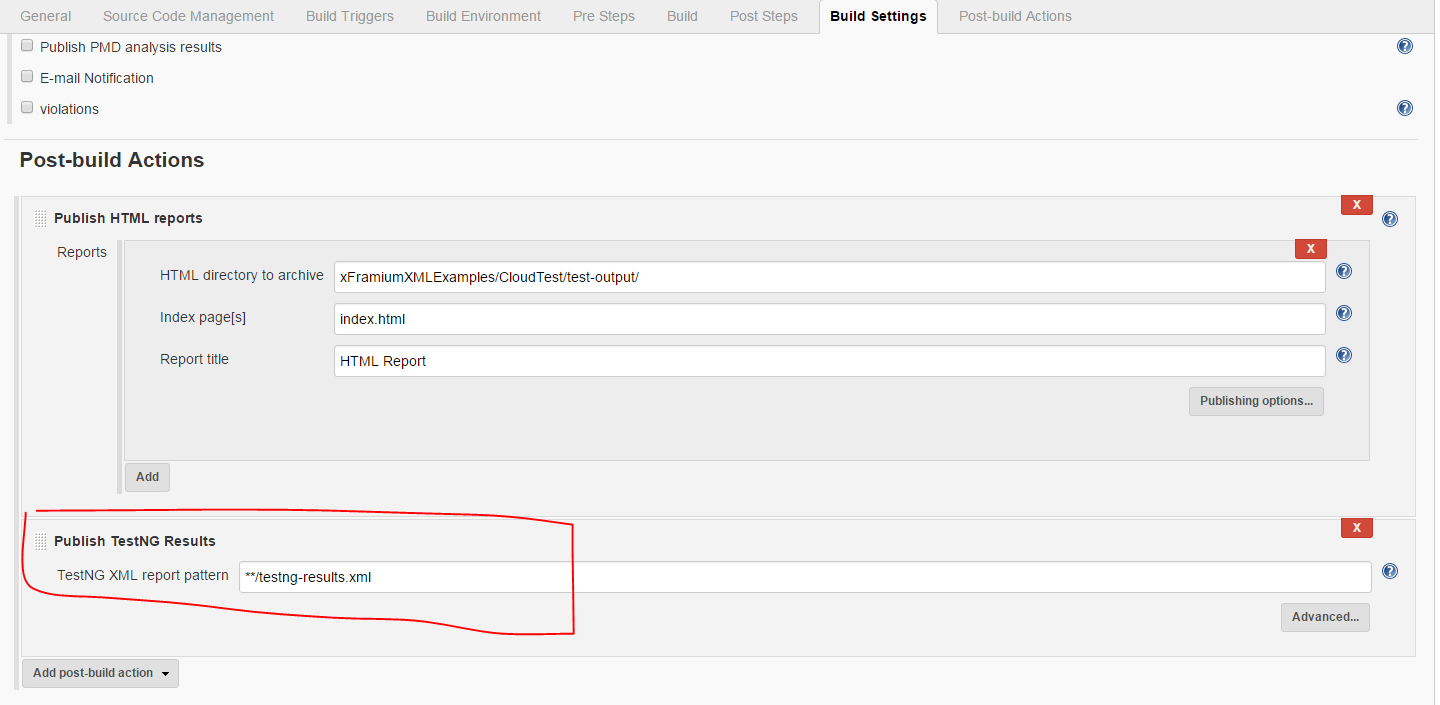
1. Configure prebuild steps. Specify the path of POM.xml and maven build command. Clean removes all the files generated by previous build in the workspace



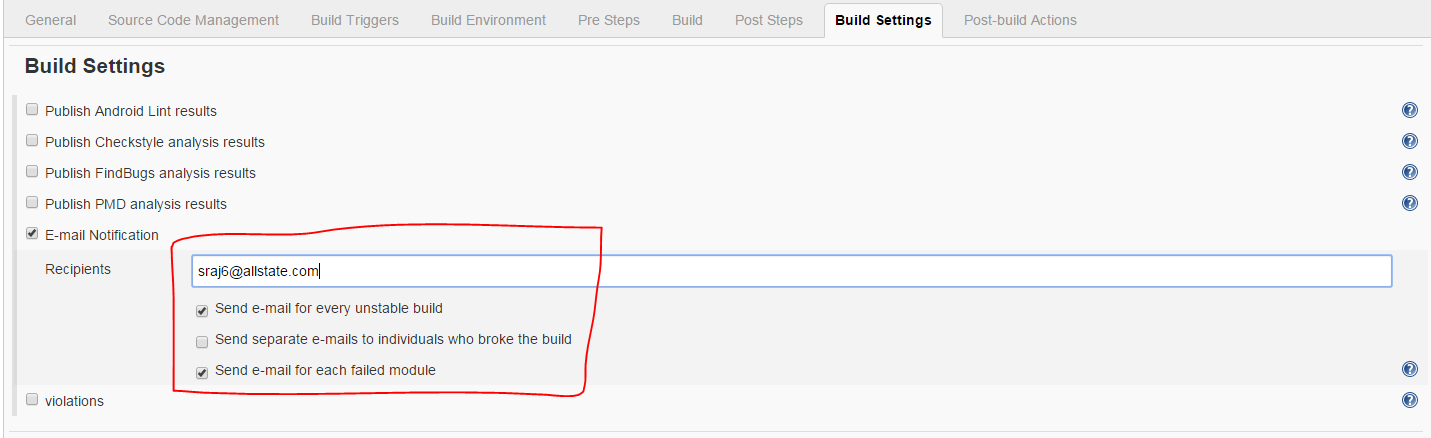
1. In post build actions > Configure reports by selecting “Publish HTML reports”. Specify HTML Directory to archive



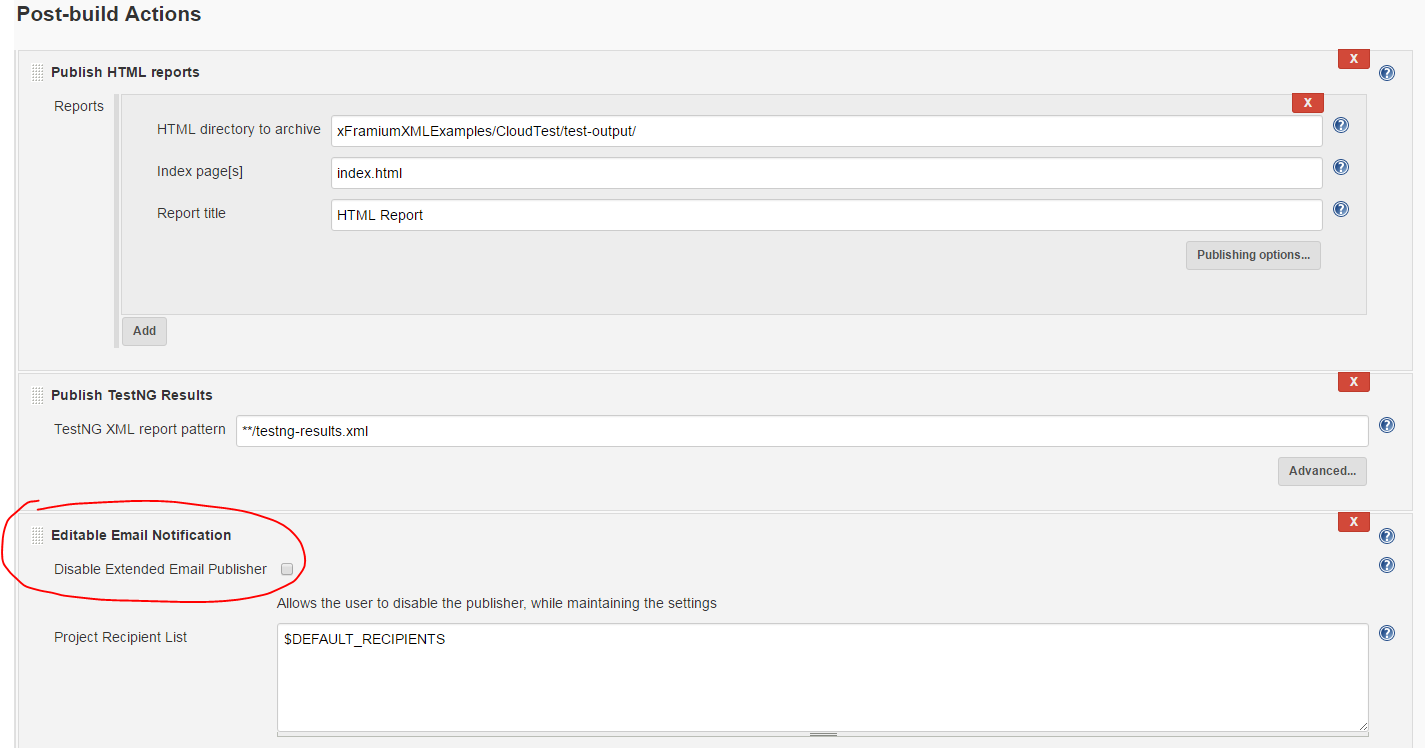
Test NG results can be added by selecting “Publish TestNG Results” option



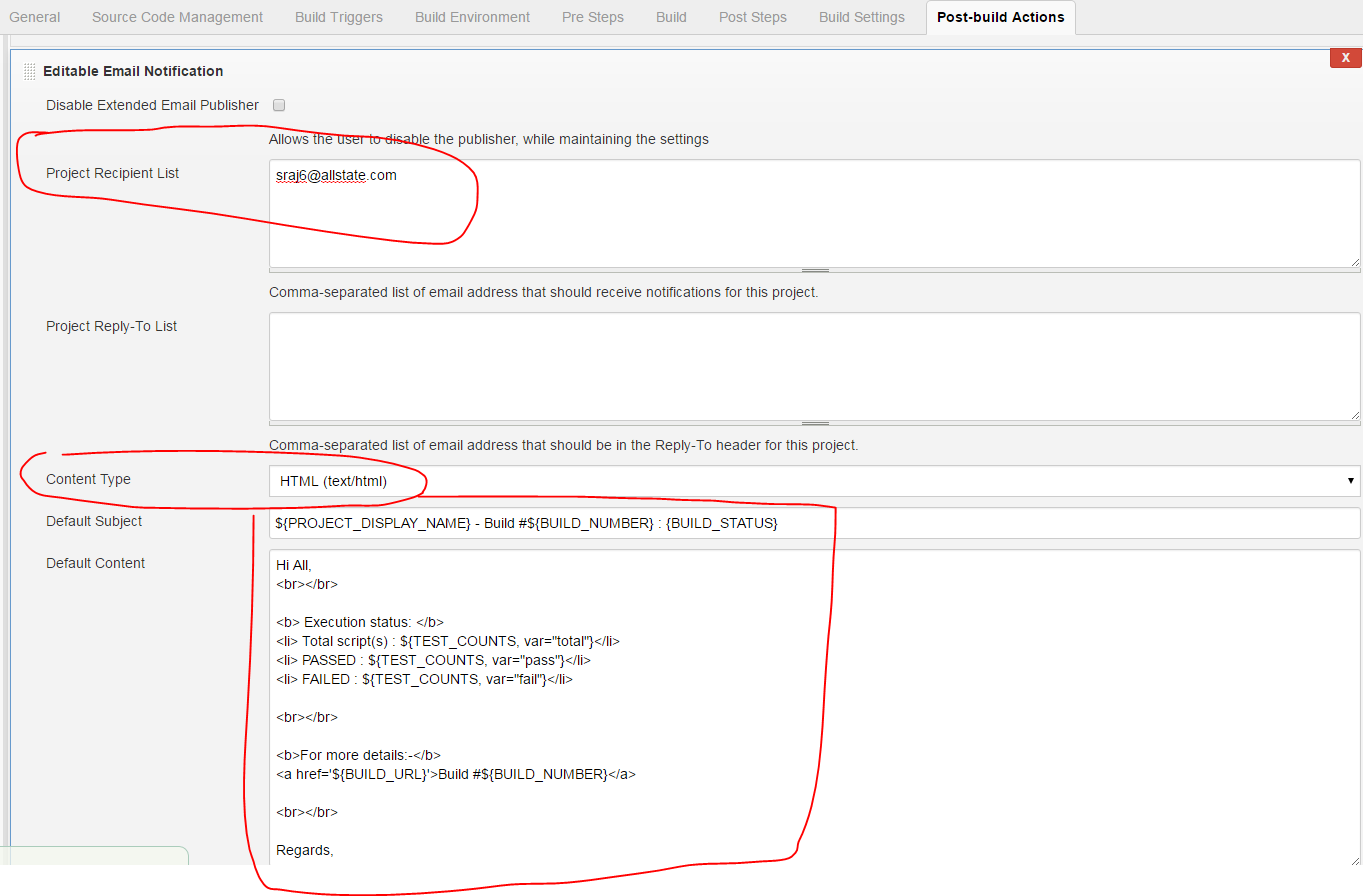
1. Email configuration can be done by selecting “Email notification” option in Build settings. Note this will just send a default email body with a link to the job executed



To build a customized email, add “Editable email Notification” under Post build actions



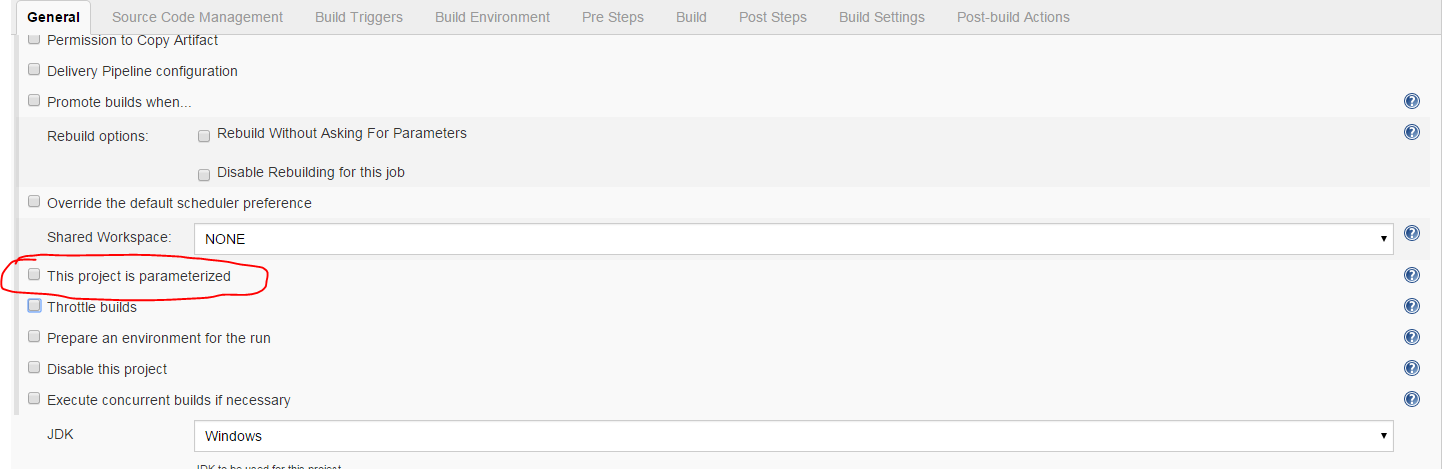
A sample email body is provide below. Ensure that the content type is HTML (text/xml) if using the attached format.

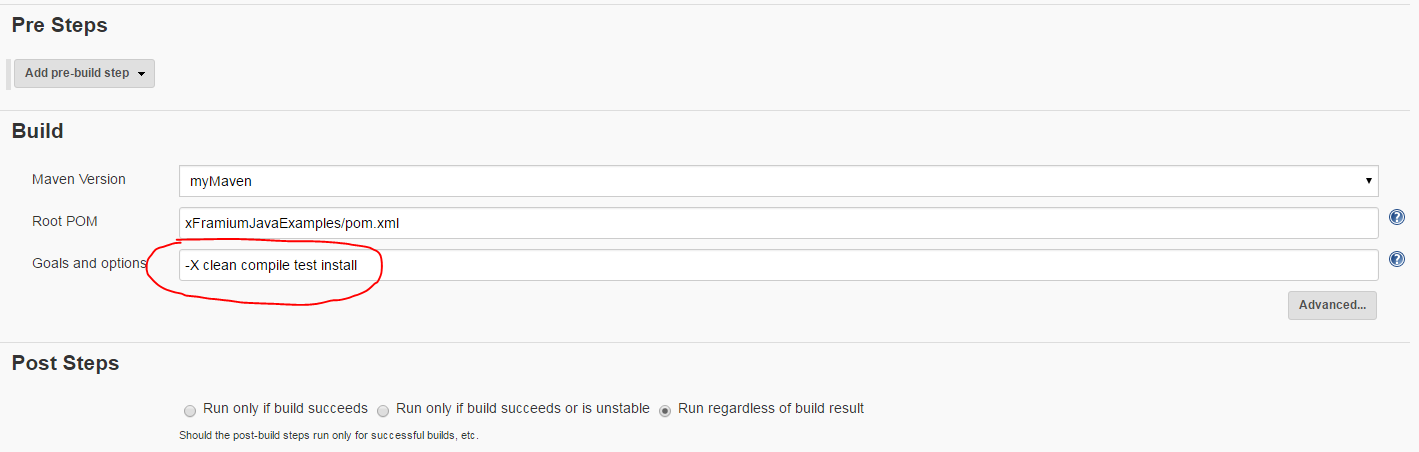
# Automation Framework Jenkins Setup for Java TestNG project

The steps to configure Java framework will be the same with few additional changes as documented below:

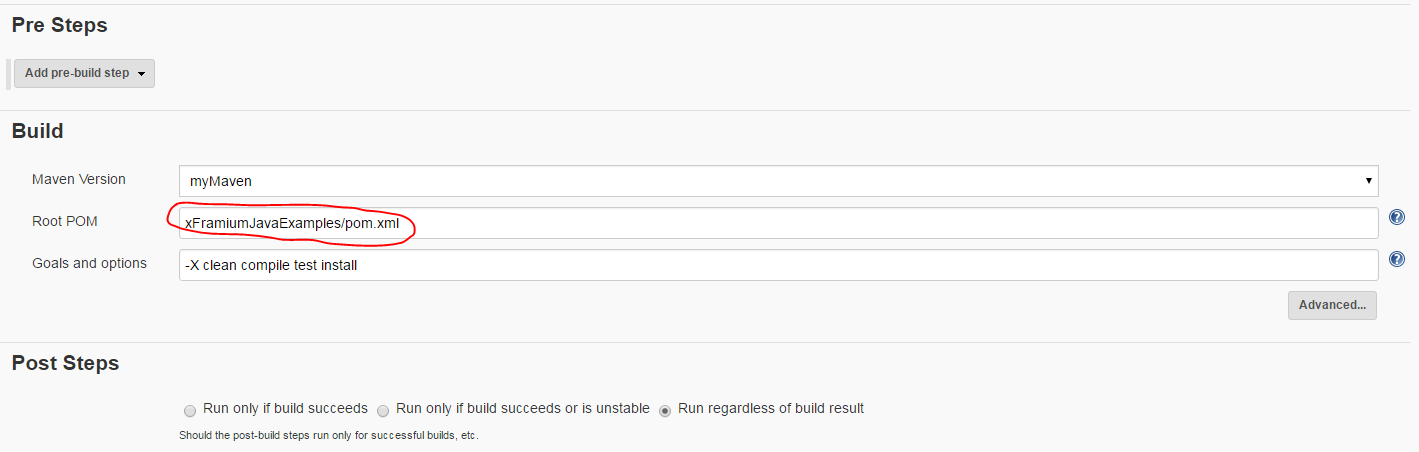
1. The project parameterization is not required for Java Framework configuration. Ensure option “This project is parameterized” is unchecked as shown below

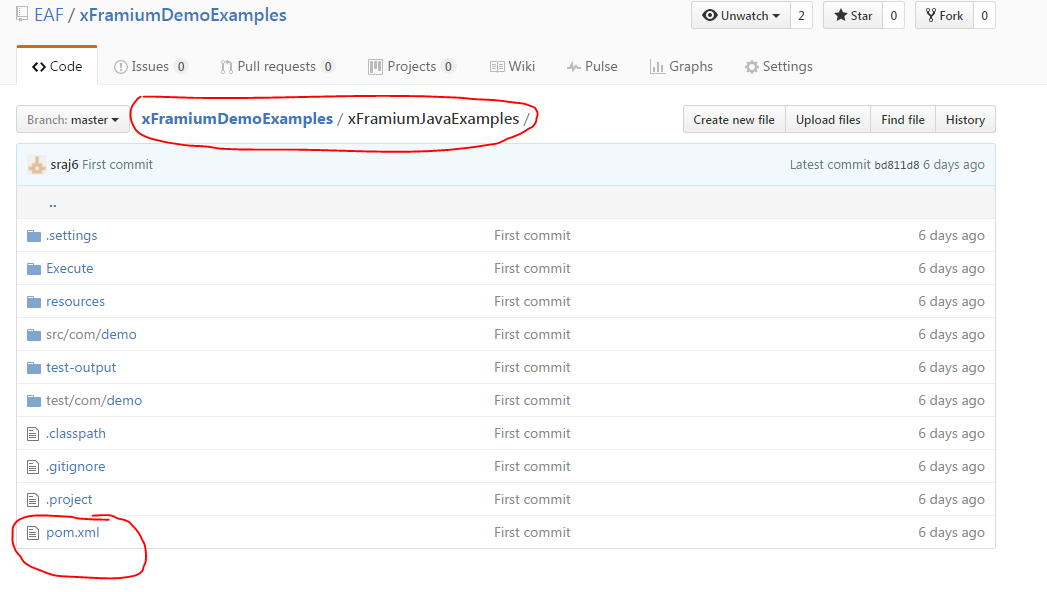


1. Maven execution command is different from XML Keyword test project. Please ensure below Maven execution command “-X clean compile test install” is configured for Pre-build step

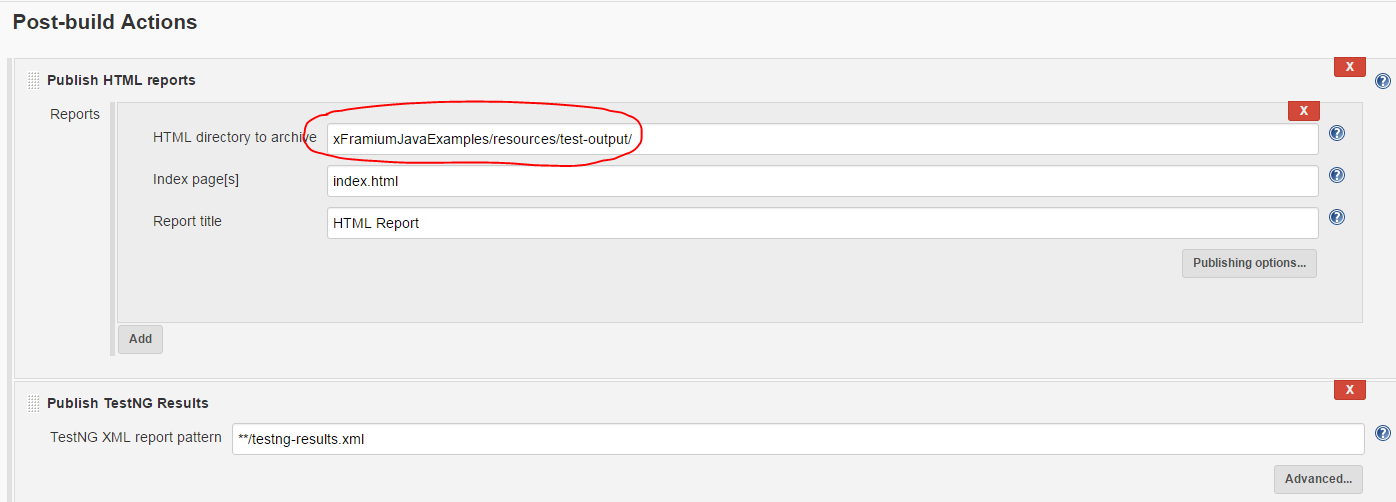


1. Ensure the Root POM in pre-build step configuration is pointing to Java project POM.xml (Do not point to AS-TestDriver project used for XML driven approach)

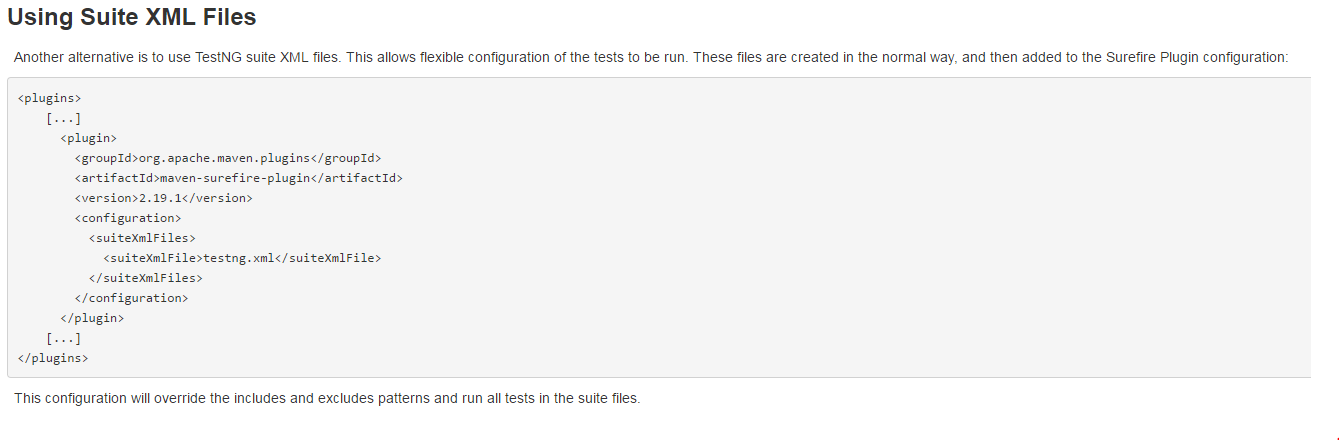




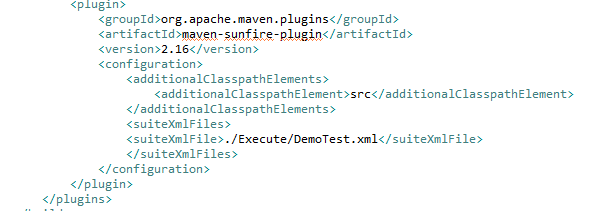
1. HTML report path should be pointing to test-output folder of the Java Framework project



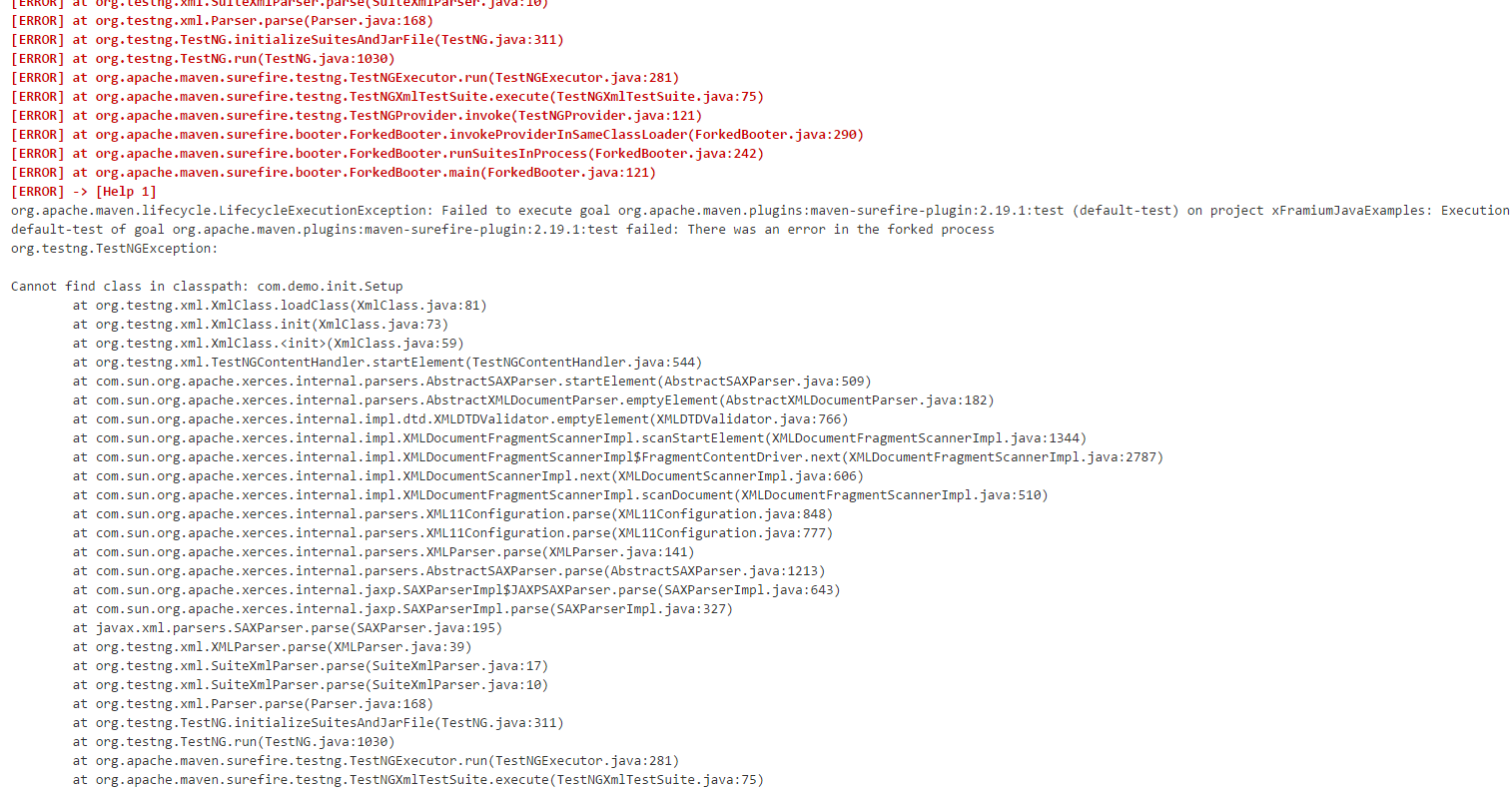
1. Use Maven sunfire plugin to execute TestNG Suite XML Files.



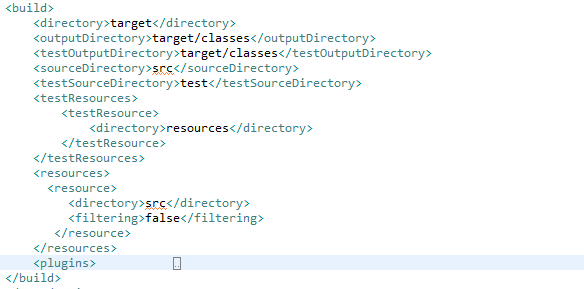
Add this section in the POM.xml file in Java project and commit changes to GIT repository



1. Set the source & class directory path in POM.xml. If not set the below exception will be displayed while running Jenkins job



The above error can be resolved by making the below changes in POM.xml

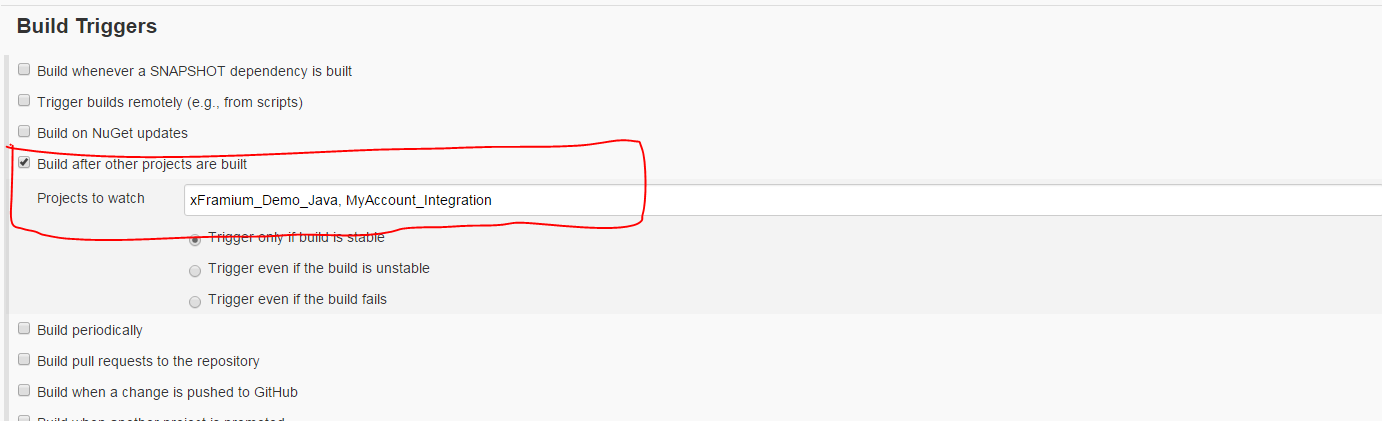


# Jenkins for Build Pipelines (Setup Build triggers for Upstream or Downstream Projects)

Job chaining in Jenkins is the process of automatically starting other job(s) after the execution of a job. This approach lets you build multi-step build pipelines or trigger the rebuild of a project if one of its dependencies is updated

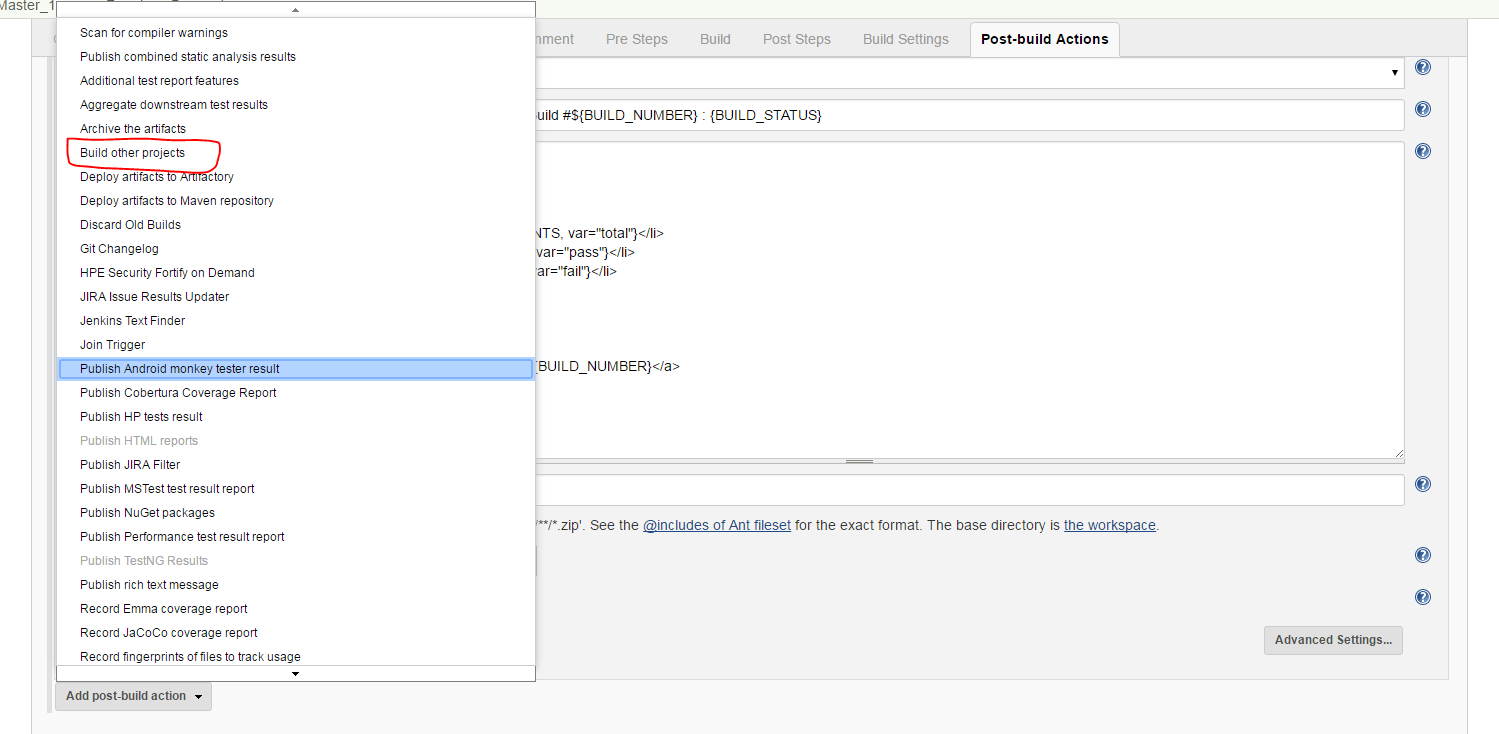
## Build after other projects are built

This option under “Build triggers” allows you to run the current job based on the outcome of upstream projects. E.g. if you want to trigger your “System tests” Job based on the outcome of “Integration tests”. Multiple projects can also be specified as shown below



## Build other projects

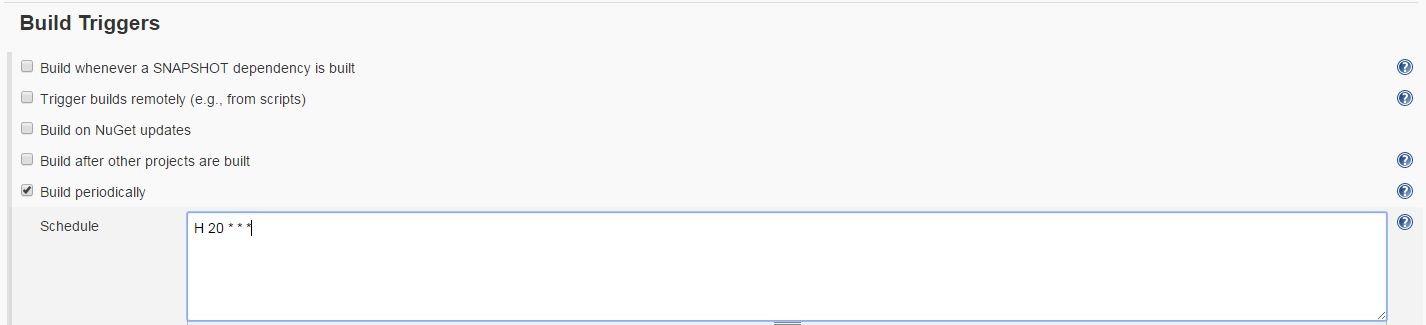
Jenkins has a built-in feature to build other projects (downstream projects) based on the outcome of current Project/Job. It is in the Post-build Actions section, specify the projects that have to be built after this project (you can trigger more than one). So whenever project A is built you will trigger the building of project B. You can also specify the conditions when the other jobs are built (e.g. when project “Regression Test” is built trigger Jenkins project job “Performance Test”)





## Build Periodically

This option under “Build Triggers” allows to trigger current build periodically based on the schedule configuration



Set the build time in following format

This consists of 5 fields separated by TAB or whitespace:

**Syntax:** MINUTE HOUR DOM MONTH DOW

**MINUTE**: Minutes within the hour (0–59)

**HOUR**: The hour of the day (0–23)

**DOM**: The day of the month (1–31)

**MONTH**: The month (1–12)

**DOW**: The day of the week (0–7) where 0 and 7 are Sunday

To specify multiple values for one field, the following operators are available. In the order of precedence,

\* specifies all valid values

M-N specifies a range of values

M-N/X or \*/X steps by intervals of X through the specified range or whole valid range

A,B,...,Z enumerates multiple values

In the above screenshot, the test would run every day of the month, specifying option “H H \* \* \*” would still execute each job once a day, but not all at the same time, better using limited resources

For more details refer to below description:

