Automating Android Builds Using Jenkins

Created  
by Vineet Kumar <vineet.kumar@sugaldamani.com>

1. Import the code to repository (Git).

2. Login to your Jenkins account

3. Create New Item

4. Enter Desired Project Name

5. Select Freestyle Project -> OK

6. Under General -> Select Restrict where this project can be run.

* Label Expression: master

7. Under Source code management -> Select Desired Repo (Example instructions are for Git Based Repo)

8. Enter SSH Path in Repository URL, eg. git@192.168.126.99:GrpName/App.git

9. Select Add->Jenkins in “Credentials” Row.

10. Select “SSH Username with private key” in “Kind” Row

11. Enter your Git username in “Username” row

12. Select “Enter Directly” in “Private Key” row

13. Enter the private key that was earlier generated

Steps:-

* Open Terminal
* cd ~/.ssh
* cat id\_rsa
* Copy the key and enter it to the “Key” text box row

14. Enter the passphrase for the private key (Passphrase was set when the key was generated earlier using ssh-keygen command)

15. Select Add

16. Select your newly created credential from the drop down in the “Credentials” row (Now the red error message should be resolved).

17. Under Build Triggers, check build when a change is pushed to Gitlab.

18. Under Build Section, “Add Build Step”

19. Select Invoke Gradle Script from drop down list

* Select Use Gradle Wrapper
* Check Make gradlew executable
* Check From Root Build Script Dir
* Tasks: clean test build jacocoTestReport assemble
* Check Force GRADLE\_USER\_HOME to use workspace

20. Under Post-build Actions

* Archive the artifacts, Files to archive: \*\*/\*.apk
* Publish Junit test result report, Test report XMLs: \*\*/build/test-results/\*\*/\*.xml
* Record JaCoCo coverage report:

Path to exec files: \*\*/build/jacoco/\*.exec

Path to class directories: \*\*/build/intermediates/classes

Path to source directories: \*\*/src

Exclusions: \*\*/R.class, \*\*/R$\*.class, \*\*/\*$ViewInjector\*.\*, \*\*/BuildConfig.\*, \*\*/Manifest\*.\*

21. Click On “Save”

22. Click “Build Now”

23. Build is generated.

**Nexus**

Create Repository: Raw (Hosted)

Name: Enter name

Format: raw

Type: hosted

Url: Repository url

Online: check

Blob store: default

**Upload Artifact To Nexus**

Nexus Version: Nexus3

Protocol: HTTP

Nexus URL: 192.168.126.99:8087

Nexus User Name Password

Credentials: Add -> Jenkins -> Kind Username With Password -> Enter Nexus Username & Password

Group id: Enter your desired id

Artifact id: This is the apk name

Type: apk

Repository: Nexus Repository

File: app/build/outputs/apk/appName.apk