

NOTE: As far as Epam firewall doesn't allow me to run Jenkins locally I will describe task in a way of instructions.

JENKINS PART:

Prerequisite: Git Plugin, Maven Plugin, Gradle plugin

Build -> Git -> Repositories:

add repository URL: <https://github.com/olgaliubchenko/gitTask>

add Git credentials

Branches to build: ****/tags/***

Build triggers -> Build Periodically -> Poll SCM

Schedule: **0 * * * ***

Build -> Execute shell: **increment-version-and-update-project.sh** (please find in separate file)

Tags:

Precognition:

```
git tag -a v.1.0 -m "Version 1.0"
```

```
git tag -a v.1.1 -m "Version 1.1"
```

You can use command:

```
Git rev-list --tags --date-order
```

Or

```
It show-ref --tags
```

To review tags history

So the in Jenkins:

Build -> Execute shell: **COMMIT_ID=\$(git rev-list --tags --date-order | head -1)**

TAG=\$(git show-ref --tags | grep \$COMMIT_ID | awk - F / '{print \$NF}')

To verify and run job with new version such steps need to be performed

With the latest Git plugin 1.1.14 (that I just release now), you can now do this more easily by simply executing the following command:

Build -> Execute shell:

```
curl http://yourserver/jenkins/git/notifyCommit?url=  
https://github.com/olga1ubchenko/gitTask
```

This will scan all the jobs that's configured to check out the specified URL, and if they are also configured with polling, it'll immediately trigger the polling (and if that finds a change worth a build, a build will be triggered in turn.)

Email notification could be set using Jenkins file (from official Jenkins docs:
<https://www.jenkins.io/doc/pipeline/tour/running-multiple-steps/>)

E.g.

```
pipeline {  
    agent any  
    stages {  
        stage('Test') {  
            steps {  
                sh 'echo "Fail!"; exit 1'  
            }  
        }  
    }  
    post {  
        always {  
            echo 'This will always run'  
        }  
        success {  
            echo 'This will run only if successful'  
        }  
        failure {  
            mail bcc: "", body: "<b>Example</b><br>Project:  
${env.JOB_NAME} <br>Build Number: ${env.BUILD_NUMBER}  
<br> URL de build: ${env.BUILD_URL}", cc: "", charset: 'UTF-8',
```

```
from: ", mimeType: 'text/html', replyTo: ", subject: "ERROR CI:
Project name -> ${env.JOB_NAME}", to:
"roman\_gordiienko@epam.com",
"olga\_liubchenko@epam.com";
} unstable {
    echo 'This will run only if the run was marked as unstable'
}
changed {
    echo 'This will run only if the state of the Pipeline has
changed' echo 'For example, if the Pipeline was previously
failing but is now successful'

}
}
}
```