

## Step 1 Jenkins Setup for Gmail

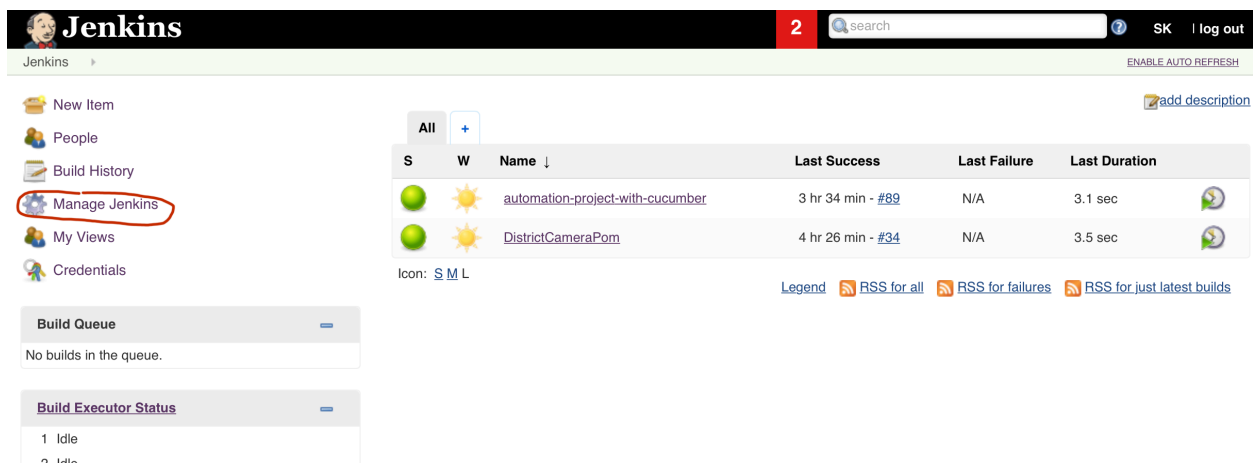
Download & share @ <https://github.com/Salihz/Jenkins-Setup-for-Gmail>

1. Make sure that the IMAP is turned on for the Gmail address you will use.  
Go to this link for how to: <https://support.google.com/mail/answer/7126229?hl=en>
2. Allow Access to your same Google account you will use for Jenkins reporting.  
Go to this link and click "Continue" <https://accounts.google.com/b/0/DisplayUnlockCaptcha>

3. Make sure Email Extension plugin is installed.

Go to: Manage Jenkins → Plugin Manager → Installed Tab

And check if "Email Extension" is installed. If it is not installed, click on "Available Tab" filter the results for "Email Extension" and install it.



The screenshot shows the Jenkins web interface. On the left sidebar, the "Manage Jenkins" option is highlighted with a red circle. The main content area displays a table of build history with columns for status (S), warnings (W), name, last success, last failure, and last duration. Two builds are listed: "automation-project-with-cucumber" and "DistrictCameraPom".

S	W	Name ↓	Last Success	Last Failure	Last Duration
●	☀	<a href="#">automation-project-with-cucumber</a>	3 hr 34 min - #89	N/A	3.1 sec
●	☀	<a href="#">DistrictCameraPom</a>	4 hr 26 min - #34	N/A	3.5 sec

Below the table, there is a legend for RSS feeds: "RSS for all", "RSS for failures", and "RSS for just latest builds".

- New Item
- People
- Build History
- Manage Jenkins
- My Views
- Credentials

**Build Queue**

No builds in the queue.

**Build Executor Status**

1 Idle

2 Idle

## Manage Jenkins

 **New version of Jenkins (2.132) is available for download (changelog).** Or Upgrade Automatically

Warnings have been published for the following currently installed components:


- Jenkins 2.74 core and libraries:
  - Multiple security vulnerabilities in Jenkins 2.120 and earlier, and LTS 2.107.2 and earlier
  - Multiple security vulnerabilities in Jenkins 2.88 and earlier, and LTS 2.73.2 and earlier
  - Multiple security vulnerabilities in Jenkins 2.106 and earlier, and LTS 2.89.3 and earlier
  - Multiple security vulnerabilities in Jenkins 2.83 and earlier, and LTS 2.73.1 and earlier
  - Multiple security vulnerabilities in Jenkins 2.94 and earlier, and LTS 2.89.1 and earlier
  - Multiple security vulnerabilities in Jenkins 2.115 and earlier, and LTS 2.107.1 and earlier
- GitHub Branch Source Plugin 2.2.3:
  - Server-side request forgery
- Pipeline: Nodes and Processes 2.13:
  - Incorrect permission checks allow executing builds on agents without Computer/Build p
- Subversion Plug-in 2.9:
  - Users without Overall/Read are able to access lists of user names and node names
- Script Security Plugin 1.33:
  - Arbitrary file read vulnerability
- Credentials Binding Plugin 1.13:
  - Improper masking of secrets in rare circumstances
- Pipeline: Supporting APIs 2.14:
  - Arbitrary code execution due to incomplete sandbox protection
- GitHub plugin 1.28.0:
  - Server-side request forgery
  - CSRF vulnerability and lack of permission checks allows capturing credentials
  - CSRF vulnerability and insufficient permission checks allow capturing credentials
- Mailer Plugin 1.20:
  - Unauthorized users able to send test emails
- Email Extension Plugin 2.58:
  - SMTP password gets transmitted in unencrypted form
- Ant Plugin 1.6:
  - Cross-site scripting (XSS) vulnerability
- JUnit Plugin 1.21:
  - XML External Entity (XXE) processing vulnerability
- Git plugin 3.5.1:
  - Server-side request forgery
  - Users without Overall/Read are able to access lists of user names and node names
- SSH Credentials Plugin 1.13:
  - Arbitrary file read vulnerability with Credentials Binding Plugin 1.13 or newer

Go to plugin manager

Configure which of these warnings are shown

Restore the previous version of Jenkins

Downgrade to 2.71

 **Configure System**

Configure global settings and paths.

Updates	Available	Installed	Advanced		
Enabled					
		Name ↓	Version	Previously installed version	Uninstall
<input checked="" type="checkbox"/>		<a href="#">Ant</a> Adds Apache Ant support to Jenkins	1.6		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Authentication Tokens API Plugin</a> This plugin provides an API for converting credentials into authentication tokens in Jenkins.	1.3		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">bouncycastle API</a> This plugin provides an stable API to Bouncy Castle related tasks.	2.16.2		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Branch API</a> This plugin provides an API for multiple branch based projects.	2.0.11		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Build Timeout</a> This plugin allows builds to be automatically terminated after the specified amount of time has elapsed.	1.18		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Credentials</a> This plugin allows you to store credentials in Jenkins.	2.1.14		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Credentials Binding</a> Allows credentials to be bound to environment variables for use from miscellaneous build steps.	1.13		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Display URL API</a> Provides the DisplayURLProvider extension point to provide alternate URLs for use in notifications	2.0		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Docker Commons</a> Provides the common shared functionality for various Docker-related plugins.	1.8		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Docker Pipeline</a> Build and use Docker containers from pipelines.	1.12		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Durable Task</a> Library offering an extension point for processes which can run outside of Jenkins yet be monitored.	1.14		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">Email Extension</a> This plugin is a replacement for Jenkins's email publisher. It allows to configure every aspect of email notifications: when an email is sent, who should receive it and what the email says	2.58		<div>Uninstall</div>
<input checked="" type="checkbox"/>		<a href="#">External Monitor Job Type Plugin</a> Adds the ability to monitor the result of externally executed jobs	1.7		<div>Uninstall</div>

4. Now we can continue for email set up.

Go to: "Manage Jenkins" → "Configure System"

5. In the "Configuration" page, go to "Jenkins Location" and enter your Gmail address as shown below:

6. At the "Extend E-mail Notification" field, Click on "Advanced" button

7. Complete the "Extend E-mail Notification" as shown below

Then click on the "Default Content Type", choose "Plain Text" or "HTML"

The screenshot shows the 'Extended E-mail Notification' configuration form. The following fields and options are highlighted with red circles:

- Extended E-mail Notification** (Section Header)
- SMTP server**: smtp.gmail.com
- Default user E-mail suffix**: @gmail.com
- ☒ **Use SMTP Authentication**
- User Name**: calmseas100@gmail.com
- Password**: (masked with dots)
- ☒ **Use SSL**
- SMTP port**: 465
- Charset**: UTF-8
- Default Content Type**: A dropdown menu is open, showing 'Plain Text (text/plain)' (selected) and 'HTML (text/html)'.
- ☐ **Use List-ID Email Header**

8. Go to "Email Notification" field and click "Advanced" button

The screenshot shows the 'E-mail Notification' configuration form. The following fields and options are visible:

- E-mail Notification** (Section Header)
- SMTP server**: (empty field)
- Default user e-mail suffix**: (empty field)
- ☐ **Test configuration by sending test e-mail**
- Advanced...** button (highlighted with a red circle)

9. Complete this field as shown below, then click on "Test configuration" button

The screenshot shows the Jenkins SMTP configuration page. Red circles and rectangles highlight the following elements:

- E-mail Notification** (Section header)
- SMTP server** field with value `smtp.gmail.com`
- Default user e-mail suffix** field with value `@gmail.com`
- ☒ **Use SMTP Authentication**
- User Name** field with value `calmesest10@gmail.com`
- Password** field (masked with dots)
- ☒ **Use SSL**
- SMTP Port** field with value `465`
- Reply-To Address** field with value `calmesest10@gmail.com`
- Charset** field with value `UTF-8`
- ☒ **Test configuration by sending test e-mail**
- Test e-mail recipient** field with value `[redacted]@gmail.com`
- Test configuration** button
- Save** button

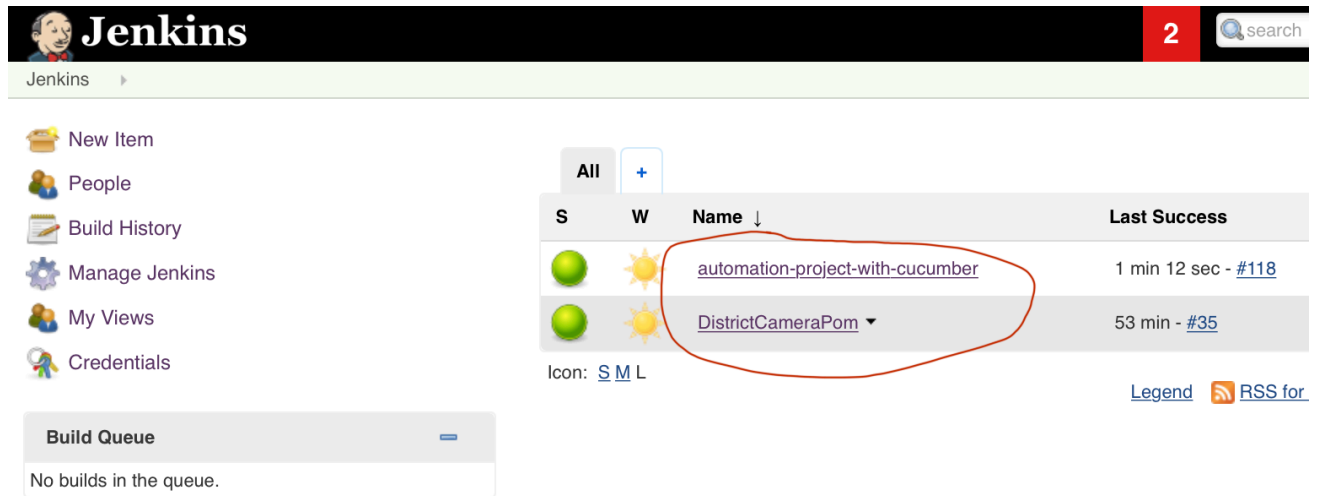
10. After clicking the "Test configuration" button you will see "Email was successfully sent" text as seen below picture. And, an e-mail must be received from Jenkins by the e-mail address you used for test.

The screenshot shows the same Jenkins SMTP configuration page, but with the following changes:

- The **Test configuration** button is now disabled and highlighted with a blue border.
- A message **Email was successfully sent** is displayed below the recipient field, highlighted with a red circle.
- The **Save** button remains highlighted with a red circle.

## Step 2 Jenkins Setup for Gmail

1. Go to your builds and click the one you prefer to receive an email report from



The screenshot shows the Jenkins dashboard. On the left is a sidebar with navigation links: New Item, People, Build History, Manage Jenkins, My Views, and Credentials. The main area displays a table of builds. The first row is for 'automation-project-with-cucumber' and the second for 'DistrictCameraPom'. The 'DistrictCameraPom' row is circled in red. Below the table, there are links for 'Legend' and 'RSS for'. A 'Build Queue' section at the bottom left shows 'No builds in the queue.'

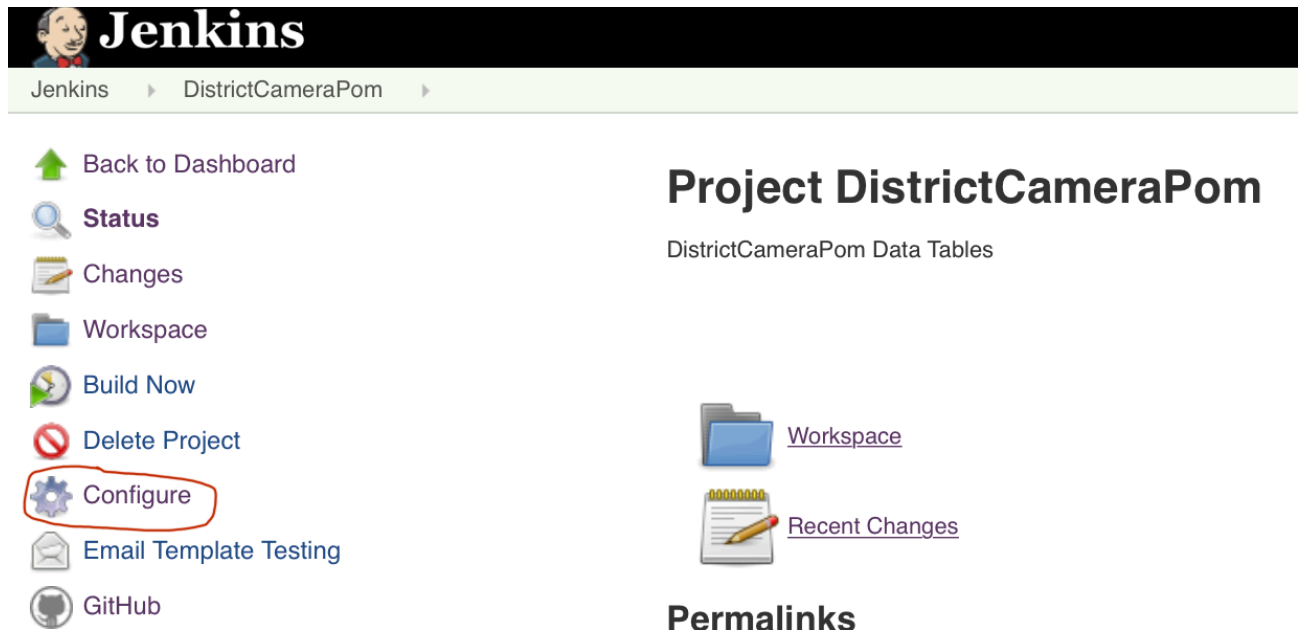
S	W	Name ↓	Last Success
		<a href="#">automation-project-with-cucumber</a>	1 min 12 sec - <a href="#">#118</a>
		<a href="#">DistrictCameraPom</a> ▼	53 min - <a href="#">#35</a>

Icon: [S](#) [M](#) [L](#)

[Legend](#) [RSS for](#)

**Build Queue**  
No builds in the queue.

2. Click on "Configure"



The screenshot shows the Jenkins configuration page for the 'DistrictCameraPom' project. The left sidebar contains links: Back to Dashboard, Status, Changes, Workspace, Build Now, Delete Project, Configure (circled in red), Email Template Testing, and GitHub. The main area is titled 'Project DistrictCameraPom' and includes a link to 'DistrictCameraPom Data Tables'. Below this are links for 'Workspace' and 'Recent Changes'. At the bottom, there is a section for 'Permalinks'.

## Project DistrictCameraPom

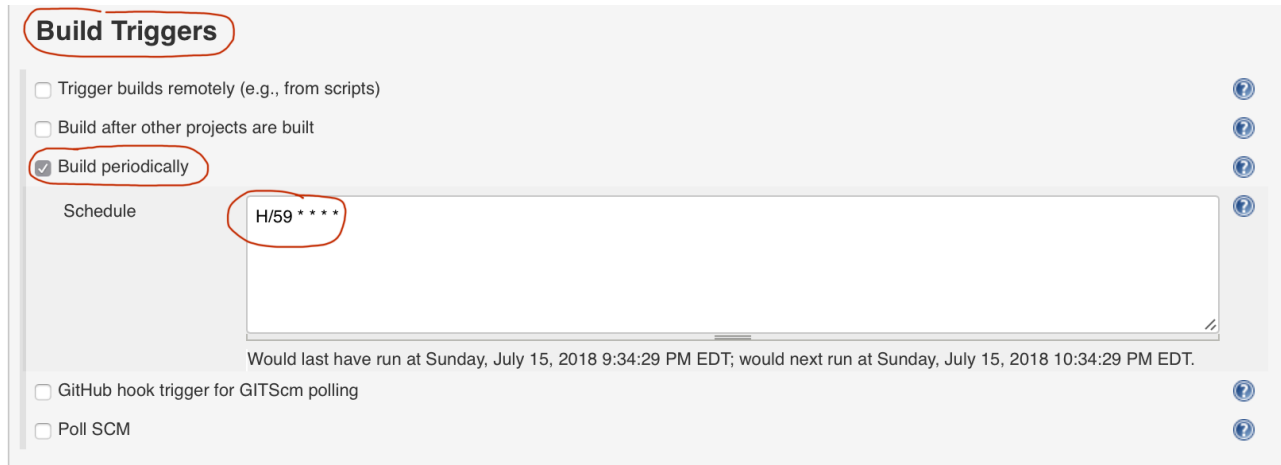
DistrictCameraPom Data Tables


[Workspace](#)

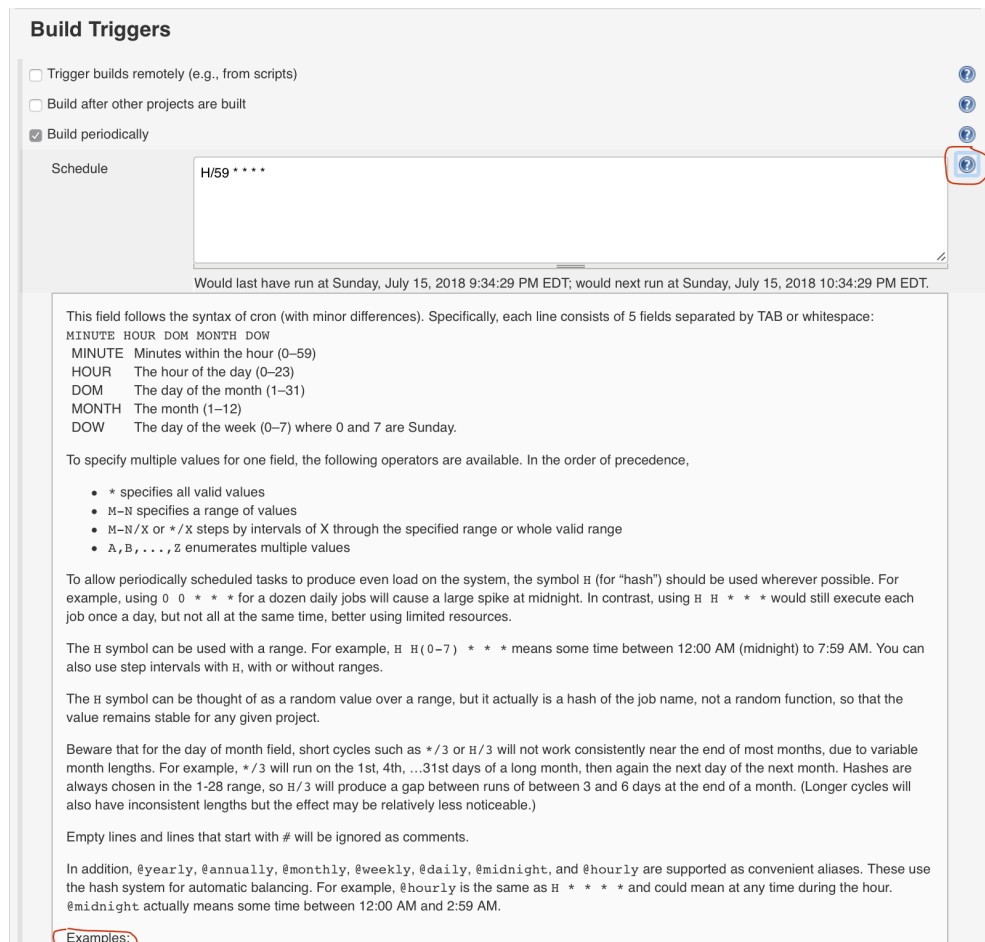
[Recent Changes](#)

### Permalinks

3. At "Build Triggers" field, check "Build periodically" box and enter "Schedule"  
In this example Jenkins will build every other 59 minutes (H/59\*\*\*\*)



If you click on  mark next to "Schedule" field you can access the syntax and examples.



**Build Triggers**

☐ Trigger builds remotely (e.g., from scripts)

☐ Build after other projects are built

☒ Build periodically

Schedule

Would last have run at Sunday, July 15, 2018 9:34:29 PM EDT; would next run at Sunday, July 15, 2018 10:34:29 PM EDT.

This field follows the syntax of cron (with minor differences). Specifically, each line consists of 5 fields separated by TAB or whitespace:

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
- N–M specifies a range of values
- N–M/X or \*/X steps by intervals of X through the specified range or whole valid range
- A,B,...,Z enumerates multiple values

To allow periodically scheduled tasks to produce even load on the system, the symbol # (for "hash") should be used wherever possible. For example, using 0 0 \* \* \* for a dozen daily jobs will cause a large spike at midnight. In contrast, using # # \* \* \* would still execute each job once a day, but not all at the same time, better using limited resources.

The # symbol can be used with a range. For example, # # (0–7) \* \* \* means some time between 12:00 AM (midnight) to 7:59 AM. You can also use step intervals with #, with or without ranges.

The # symbol can be thought of as a random value over a range, but it actually is a hash of the job name, not a random function, so that the value remains stable for any given project.

Beware that for the day of month field, short cycles such as \*/3 or #/3 will not work consistently near the end of most months, due to variable month lengths. For example, \*/3 will run on the 1st, 4th, ...31st days of a long month, then again the next day of the next month. Hashes are always chosen in the 1-28 range, so #/3 will produce a gap between runs of between 3 and 6 days at the end of a month. (Longer cycles will also have inconsistent lengths but the effect may be relatively less noticeable.)

Empty lines and lines that start with # will be ignored as comments.

In addition, @yearly, @annually, @monthly, @weekly, @daily, @midnight, and @hourly are supported as convenient aliases. These use the hash system for automatic balancing. For example, @hourly is the same as # \* \* \* \* and could mean at any time during the hour. @midnight actually means some time between 12:00 AM and 2:59 AM.

Examples:

External sources for Schedule:

<https://www.lenar.io/jenkins-schedule-build-periodically/>

4. Go to "Post-build Actions" field and enter the "Project Recipient List" can be multiple recipient (click on question button to see how)

**Post-build Actions**

**Editable Email Notification** X ?

Disable Extended Email Publisher ☐ ?

Allows the user to disable the publisher, while maintaining the settings

Project Recipient List  ?

Comma-separated list of email address that should receive notifications for this project.

This is a comma separated list (optional whitespace after commas) of email addresses that should receive emails. May contain references to build parameters. To CC or BCC someone instead of putting them in the To list, add cc: or bcc: before the email address (e.g., cc:someone@example.com, bcc:bob@example.com).

5. If you like to attach the "Build Log" to the email click on "Attach Build Log" button and choose from drop down (see next two pictures)

Attachments

Can use wildcards like 'module/dist/\*\*/\*.zip'. See the [@includes of Ant fileset](#) for the exact format. The base directory is [the workspace](#).

Attach Build Log

Content Token Reference

Advanced Settings...

Add post-build action ▼

Attachments

Can use wildcards like 'module/dist/\*\*/\*.zip'. See the [@includes of Ant fileset](#) for the exact format. The base directory is [the workspace](#).

Attach Build Log

Content Token Reference

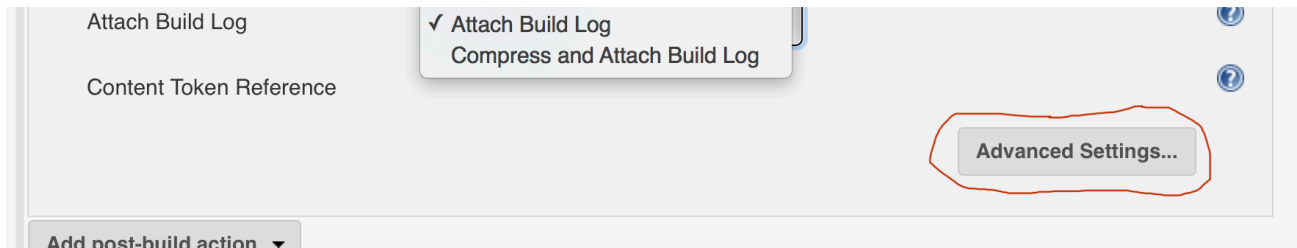
Do Not Attach Build Log  
✓ Attach Build Log  
Compress and Attach Build Log

Advanced Settings...

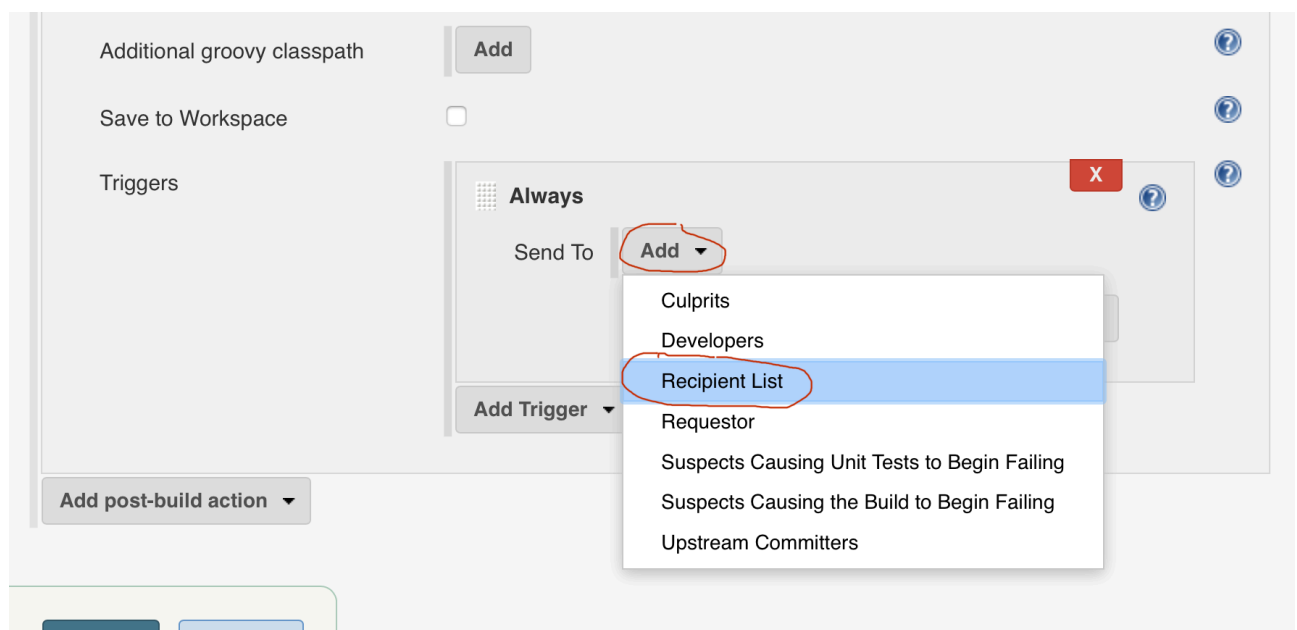
Add post-build action ▼



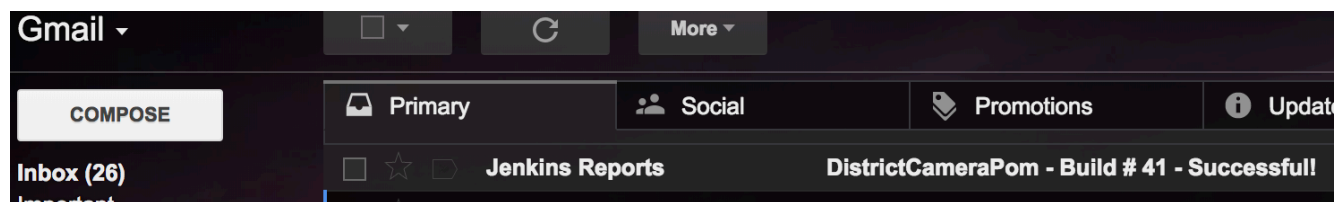
6. After done with attachment click on "Advanced Settings" button



7. Click on "Add" and select "Recipient List" (because we set "Recipient List" in the step 4 above) and click on "Apply" and/or "Save" button(s)



8. After first triggered or manually build you should receive an email report. Open the email.



9. To view the console output results, click on the link (see console Output pic below)

To view the build log, click on the attachment or go to above mentioned link.

## DistrictCameraPom - Build # 41 - Successful!



Inbox x



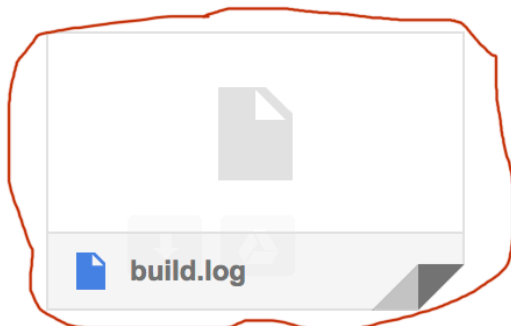
### Jenkins Reports

to me ▾

🔗 10:4

DistrictCameraPom - Build # 41 - Successful:

Check console output at <http://localhost:8080/job/DistrictCameraPom/41/> to view the results.



Click on console output then click on "View as plain text" to see build log

**Jenkins** 2

Jenkins > DistrictCameraPom > #41

[Back to Project](#)  
[Status](#)  
[Changes](#)  
**[Console Output](#)**  
[View as plain text](#)  
[Edit Build Information](#)  
[Delete Build](#)  
[Git Build Data](#)  
[No Tags](#)  
[Previous Build](#)

Timestamps [View as plain text](#)  
☒ System clock time  
☐ Use browser timezone  
☐ Elapsed time  
☐ None

to download or share this document go to: [github.com/Salihz](https://github.com/Salihz)

### Console Output

```
22:49:38 Started by user SK
22:49:38 Building in workspace /Users/sk/.jenkins/workspace/DistrictCameraPom
22:49:38 > git rev-parse --is-inside-work-tree # timeout=10
22:49:38 Fetching changes from the remote Git repository
22:49:38 > git config remote.origin.url https://github.com/Salihz/DistrictCameraPOM.git #
22:49:38 Fetching upstream changes from https://github.com/Salihz/DistrictCameraPOM.git
22:49:38 > git --version # timeout=10
22:49:38 > git fetch --tags --progress https://github.com/Salihz/DistrictCameraPOM.git
+refs/heads/*:refs/remotes/origin/*
22:49:38 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
22:49:38 > git rev-parse refs/remotes/origin/origin/master^{commit} # timeout=10
22:49:38 Checking out
22:49:38 Commit message: "Cukes Update"
22:49:38 > git config core.sparsecheckout # timeout=10
22:49:38 > git checkout -f
22:49:38 > git rev-list
# timeout=10
22:49:38 No emails were triggered.
22:49:38 [DistrictCameraPom] $ /usr/local/Cellar/maven/3.5.0/bin/mvn -f
/Users/sk/Documents/workspace/DistrictCamera/pom.xml test "-Dcucumber.options=--tags @smok
22:49:40 [INFO] Scanning for projects...
22:49:40 [INFO]
22:49:40 [INFO] -----
22:49:40 [INFO] Building DidtrictCamera 0.0.1-SNAPSHOT
22:49:40 [INFO] -----
22:49:40 [INFO]
22:49:40 [INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ DidtrictCam
22:49:41 [WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.
```

Links:

<https://jenkins.io>

<https://wiki.jenkins.io/display/JENKINS/Home>

<https://issues.jenkins-ci.org/secure/Dashboard.jspa>