



Search Medium

ished in SwiftBlade

hongchai Kolyutsakul

ct 8, 2015 · 6 min read · Listen



Automating iOS app testing and TestFlight deployment with Jenkins

Automate testing and uploading iOS apps to TestFlight! I will talk about detailed configurations (with screenshots!), pitfalls I came across with Jenkins, and alternative tools.

What I need?

Jenkins app —

note: There's a way to [fix this jenkins user issue](#). So you don't actually need Jenkins app anymore.)

Jenkins app is actually not from official Jenkins, it is a 3rd-party script that downloads `jenkins.war` file and runs it as your current user. Originally, the official Jenkins will create a new user named `jenkins`, which causes problems such as Code Signing Identity not found, gems not available, because those are not installed on `jenkins` user account! Going through those setup again would be a pain. So I prefer using my own Jenkins app. Plus, you will get a nice little Jenkins icon in your dock.



Jenkins app, sitting right in your dock!

Jenkins plugins — these are the magics behind the scene. Get these installed:

GitHub plugin

 Über Google anmelden

Melden Sie sich mit Ihrem Google-Konto in Medium an.

Sie müssen sich keine Passwörter mehr merken. Die Anmeldung ist schnell, einfach und sicher.



Weiter

Follow

More from Medium



Abdurrachman — mpj

How to setup iOS CI/CD to TestFlight for free with GitLab and Semaphore CI



Davut in Owl Apps Team

The Fun Way to Visualize Long Builds in Xcode with



Alex "Foxicorn"... in Better Prog...

How To Avoid Merge Conflicts In Xcode Projects

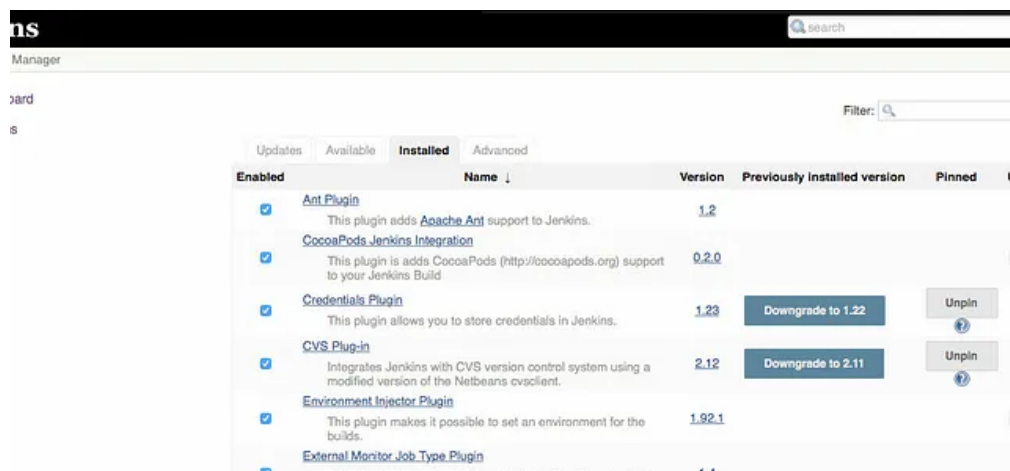


Diego Jimenez in Indie Dev Life

How to apply Clean Swift by example for iOS projects (The easy way)

plugin (yes, you need both Git and Github plugins)
M plugin (if you use rvm to manage your ruby versions)
code integration plugin

[Help](#) [Status](#) [Writers](#) [Blog](#) [Careers](#) [Privacy](#) [Terms](#)
 Text to speech



Jenkins Plugin Manager page. The amount of plugins is massive!

ad/shenzhen — for handling TestFlight upload. There are also some alternatives you can try e.g. KrauseFx/deliver and fastlane/pilot.
 ATE: shenzhen has a newer replacement called gym.)



SHENZHEN

The mighty Shenzhen gem shall now distribute all your IPA files!

aPods — you should have this installed already. If you don't, then learn to use it. It helps managing iOS project dependencies a lot.

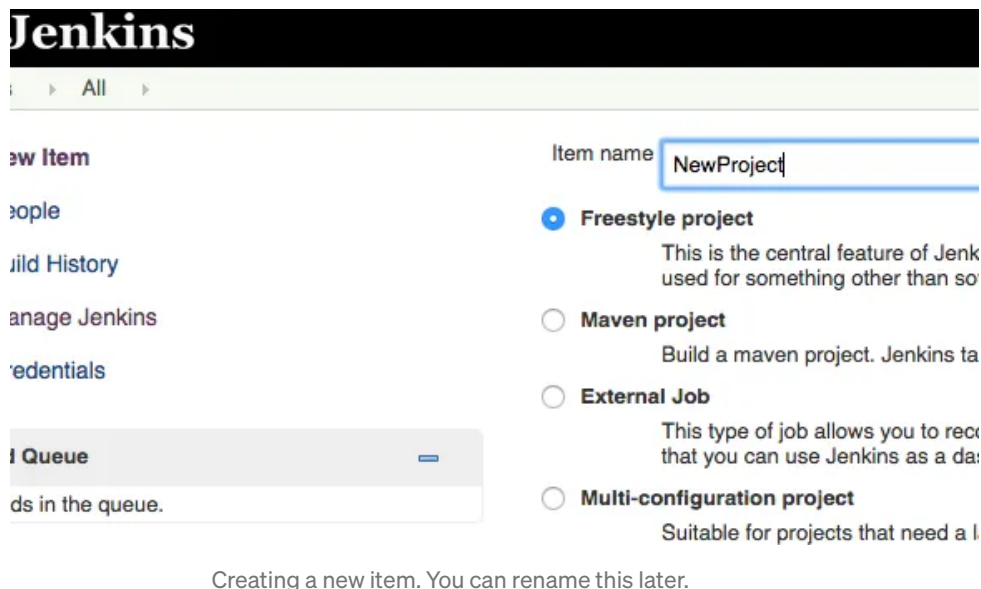
Setting Identities and Provisioning Profiles — set these up on the machine that you will run Jenkins, just like your own development machine. You can also Exporting Your Signing Identities and Provisioning Profiles from Xcode.

Getting set up?

Now that you have all the tools. It's time to tweak all those knobs.

this article will use Jenkins version 1.632.

New item



Jenkins

Item name:

☒ **Freestyle project**
This is the central feature of Jenkins used for something other than so

☐ **Maven project**
Build a maven project. Jenkins ta

☐ **External Job**
This type of job allows you to rec that you can use Jenkins as a da

☐ **Multi-configuration project**
Suitable for projects that need a l

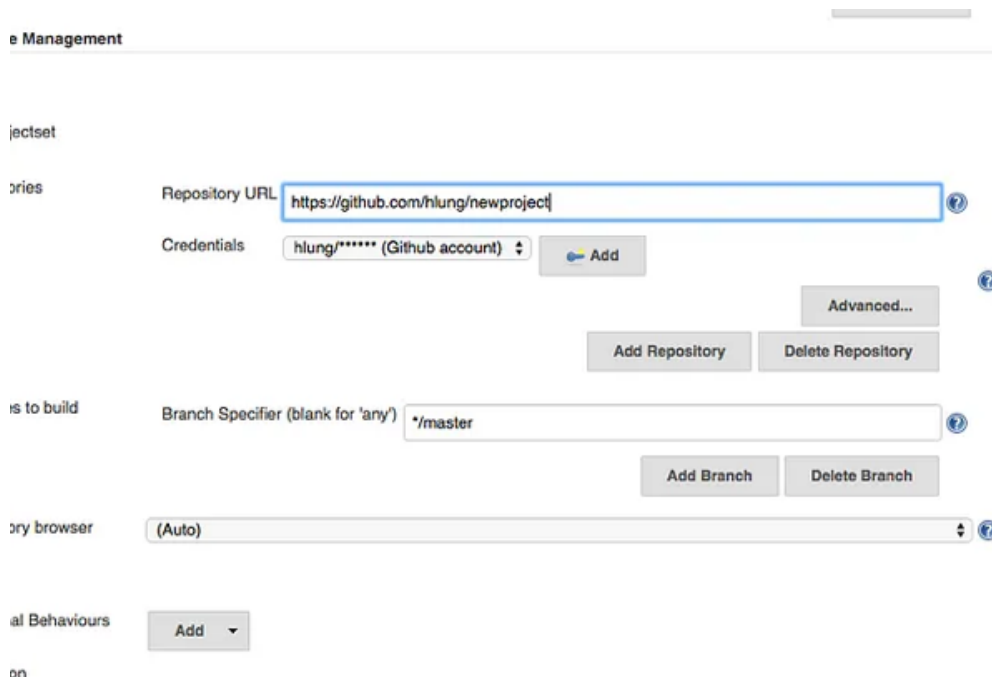
Creating a new item. You can rename this later.

On the left of Jenkins home, click **New Item**

Enter your project name, e.g. "NewProject"

Select **Freestyle project**, and click **OK**

Repository



Repository URL:

Credentials:

Advanced...

Branch Specifier (blank for 'any'):

Repository browser:

Additional Behaviours:

Hook up to your Github repo

In the **Code Management** section, setup Github repo URL, credentials, and the branch you want Jenkins to look at.

Triggers & Build environments

The screenshot shows the Jenkins configuration interface. Under the 'Triggers' section, the 'Poll SCM' checkbox is checked. The polling interval is set to 'H/5 * * * *', which means every 5 minutes. Below this, a text box shows the last run time as 'Thursday, October 8, 2015 8:40:39 PM ICT' and the next run time as 'Thursday, October 8, 2015 8:45:39 PM ICT'. Under the 'Environment' section, the 'Run the build in a RVM-managed environment' checkbox is checked. Below this, there is a text field for 'RUBY[@GEMSET], such as '1.9.3' or 'ruby@foo' which is currently empty.

Setup polling interval and RVM environment

In the **Triggers** section, tick **Poll SCM** and set value to “H/5 * * * *”. This will trigger a build every 5 minutes. (This is the same interval that Xcode Bot uses.)

In the **Environment** section, tick **Run the build in a RVM-managed environment**. Leave the textfield blank to use the default ruby version. You can check which ruby version is default by using “rvm list” command in your terminal. And make sure your gems are available for that ruby version.

Instead of polling, you can also setup Github webhooks to notify your Jenkins server in realtime when there are new commits. But that requires setting up your server to be visible to the internet, which causes security concerns. So I think polling is good enough.

Shell (part 1)

Go back a bit and click the **Add build step** button > **Execute Shell** and add

o run CocoaPods and run your tests.

r installing pods:

```
stall --verbose
```

r running tests:

print out all env vars for debugging, search for "+ set" in
e output

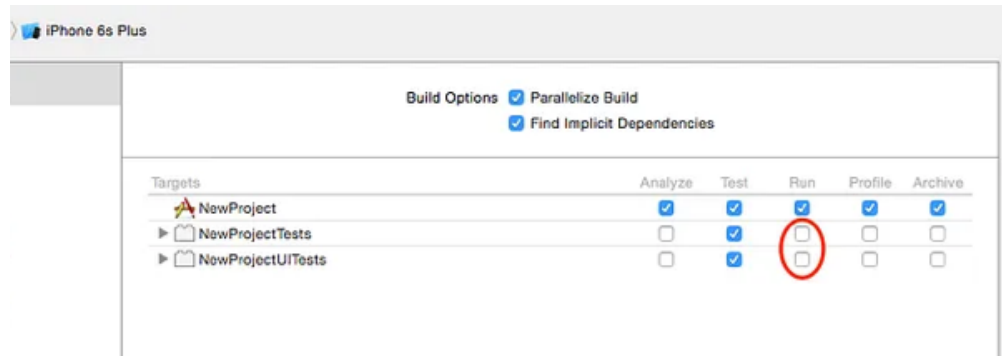
```
N and TEST, if this fails the build will stop
in/xcodebuild clean test \
e "NewProject" \
pace "NewProject.xcworkspace" \
nation 'platform=iOS Simulator,name=iPhone 5s'
```



Scripts make us look professional, isn't it? :P

*There is actually a Jenkins plugin for CocoaPods. But there is some problem
The plugin can't find the correct gem path if you use rvm to install it. So I just
use the shell command instead.*

Important: For tests to work, make sure these checkboxes are unchecked.



Uncheck Run in test targets

Settings

follows. For the name of “Code Signing Identity” you can find it in the certificate.

General settings

Leave empty for all targets

☐ Interpret As Regular Expression

Build? ☒ Yes

This will delete the build directories before invoking the build.

Build results? ☐ Yes

Checking this option will prevent this build step from failing if xcodebuild exits with a non-zero return code.

Archive? ☐ Yes

Checking this option will generate an xcarchive of the specified scheme. A workspace and scheme are also needed for archives

Configuration

This is the name of the configuration as defined in the Xcode project.

Application and build .ipa?

Pattern

A pattern for the ipa file name. You may use \${VERSION} and \${BUILD_DATE} (yyyy.MM.dd) in this string

Directory

The output directory for the .ipa file, relative to the build directory.

Base URL

The base URL to use to create a Manifest.plist. If omitted no Manifest.plist will be generated

General build settings > click “Settings...”

Signing & OS X keychain options

Bundle ID?

Signing Identity

Override the code signing identity specified in the project

Profile

The relative path to the mobileprovision to embed, leave blank for no embedded profile

Keychain?

Code signing & OS X keychain options > click “Code signing settings...”

Code build options

ports? ☐ Yes

na File ?

Only needed if you want to compile for a specific schema instead of a target.

?

Leave empty for default SDK

?

Leave empty for default SYMROOT

lebuild arguments ?

Additional xcodebuild arguments

space File ?

Only needed if you want to compile a workspace instead of a project.

st Directory ?

Relative path within the workspace that contains the xcode project file(s).

st File ?

Only needed if there is more than one project file in the Xcode Project Directory

directory ?

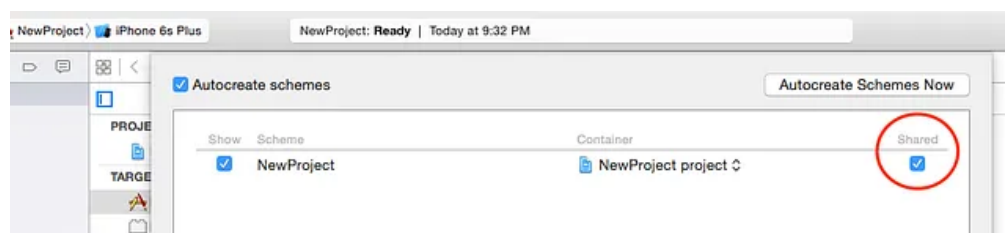
The value to use for CONFIGURATION_BUILD_DIR setting.

ersion number and run avgtool?

Delete

Advanced Xcode build options > click “Advanced build settings...”

nt: For building using scheme to work, make sure your scheme is
This adds the scheme files to git, thus visible to Jenkins.



Check the shared checkbox

nt again: Also make sure in Xcode > Build Settings > Provisioning
is set to “Automatic”.

Shell (part 2)

d build step > Execute Shell and add another script to run
n's ipa command. This will upload your ipa file to TestFlight!

l` is actually your app identifier in App Store. You can find it in connect. Go to My Apps, App Store, App Information, and look in Information section.

```

istribute:itunesconnect \
e "${WORKSPACE}/build/upload.ipa" \
ount yourappleid@gmail.com \
urPa55w0rd \
le-id 1234567890 --upload --verbose

```

shell

```

ipa distribute:itunesconnect \
--file "${WORKSPACE}/build/upload.ipa" \
--account yourappleid@gmail.com -p kkupxwZW82r7iEVJfymX6W8EUBxefEt5 \
--apple-id 1044951517 --upload --verbose

```

See [the list of available environment variables](#)

Delete

The password above is fake btw. No need to find my apple id and use it. :P

g up the configs files

ider, hit cmd+shift+G, and type in this directory path...

```

kins/jobs/<jobname>/config.xml

```

here each job configuration is stored. Just copy it to somewhere else
t up.

after updating Xcode

st updated Xcode to a newer version and then trigger a build, there
n error like this.

ng to the Xcode/iOS license requires admin privileges, please
as root via sudo.

```

    at
nkinsci.plugins.gitclient.CliGitAPIImpl.launchCommandIn(CliGitA
.java:1640)
    at
nkinsci.plugins.gitclient.CliGitAPIImpl.launchCommandIn(CliGitA
.java:1616)
    at
nkinsci.plugins.gitclient.CliGitAPIImpl.launchCommandIn(CliGitA
.java:1612)
    at
nkinsci.plugins.gitclient.CliGitAPIImpl.launchCommand(CliGitAPI
ava:1254)
    at
nkinsci.plugins.gitclient.CliGitAPIImpl$4.execute(CliGitAPIImpl
604)
    ... 12 more
null
ed: FAILURE

```

ans that you have to run Xcode and **accept the terms** in a popup
first. You will have to do this for each new version of Xcode.

Apple ID's password

connect command line don't accept Apple ID's password with **special
characters** like * or & . Your password has to be alphanumeric (e.g. secret123).
<https://github.com/nomad/shenzhen/issues/189>.

code 7 — PackageApplication script

7, you'll have to edit the **PackageApplication script**, AND avoid
this flag in **xcodebuild** ...

```
IGN_RESOURCE_RULES_PATH=$(SDKROOT)/ResourceRules.plist
```

reference: [https://cutting.io/posts/packaging-ios-apps-from-the-command-](https://cutting.io/posts/packaging-ios-apps-from-the-command-line/)

Good tutorials, everywhere

tutorials on the internet are outdated and you may get frustrated by

onsistency. Don't worry, just go back to the origin. Take a look at the **documentation in each plugin's official page**. Most of them has up-to-date, setup steps that are very useful. (Well, one day this tutorial is going to be updated to.)

Account security

Since Jenkins runs as your user, meaning that **anyone who has access to Jenkins can modify jobs to run any scripts on your behalf**. This is a security issue. But no worries, this is why I use Github polling instead of connecting Jenkins to the internet to receive Github hook URL. So only your work can access it. You can also install **Github authentication** to allow only the authorized users manage your configurations.

Useful Tools

Travis CI — *It also provides continuous integration and TestFlight upload too.*

Let me know in the comments if you have any thing to share ;)

Testflight

Jenkins



197



4

Want to read? Reward the writer. Beta

Go to Thongchai Kolyutsakul through a third-party platform to support the writer, letting them know you appreciate their story.

Give a tip

mail whenever Thongchai Kolyutsakul publishes.

 Subscribe

You will create a Medium account if you don't already have one. Review our [Privacy Policy](#) for more information about our practices.