

Fastlane iOS

Automatic app deployment to TestFlight

This document highlights the step by step process to release the app to Testflight using Fastlane.

1. **Make sure you have the latest version of the Xcode command line tools installed:**

`'xcode-select —install'`

2. **Install Fastlane**

`'brew cask install fastlane'`

3. **Set the path**

`'export PATH="$HOME/.fastlane/bin:$PATH"'`

4. **Navigate your terminal to your project's directory and run:**

`'fastlane init'`

You'll be asked for your Apple ID, and *fastlane* will automatically generate a fastlane configuration for you, based on the information provided.

5. **Make sure the Automatically Manage Signing in Xcode is off. Download and select the provisional profiles for the project in the Xcode from the apple developer portal.**


6. **Add all the app icons. Add this to the Info.plist file:**

`<key>CFBundleIconName</key>`

`<string>AppIcon</string>`

7. **If your app is not using encryption and you don't want to have to answer these questions at the time of submission, you can provide export compliance information with your build in your Info.plist file:**

<key>ITSAAppUsesNonExemptEncryption</key><false/>

8. Create an application on iTunes Connect with the correct bundle identifier.
9. Go to your Project -> Build Settings -> Versioning:
Add 1.0 to the *Current Project Version*
Choose *Apple Generic* for Versioning System
This settings is necessary for the fast lane to automatically increment build number
10. Open the Fastfile from the project root directory:
open *fastlane/Fastfile*
11. Add web hook url from the slack:
 - a. Go to the url: 
 - b. Select *Incoming Webhooks*
 - c. Click on : *Add Configuration*
 - d. Select the Slack channel where you want to post build notifications
 - e. After the channel is selected, click on : *Add Incoming Webhooks Integration*
 - d. Copy the url and paste it on the Fastfile for this key: `ENV["SLACK_URL"]`
12. Add the slack url to the `ENV["SLACK_URL"]` key before any lane is executed:

```
platform :ios do
  before_all do
    ENV["SLACK_URL"] = "https://hooks.slack.com/services/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"
    cocoapods
    # carthage
  end
end
```

After the lane being executed is successful, the selected channel of the Slack will be notified with the given success message.

If the lane being met with some error or exception, the channel will be notified with the error message.

See screenshot below:

```
after_all do |lane|
  # This block is called, only if the executed lane was successful
  slack(message: "App successfully released!")
end

error do |lane, exception|
  slack(
    message: exception.message,
    success: false
  )
end
end
```

13. Before any lane is executed, make sure to add the cocoapods or carthage based on what dependency manager you are using.

14. Step by step explanation of lane beta:

```
lane :beta do
  cert
  sigh(force: true)
  increment_build_number
  # Read from file system
  changelog = File.read("../TestInformation.txt")

  gym(scheme: "TestFastlane",
      workspace: "TestFastlane.xcworkspace",
      include_bitcode: true)

  testflight(changelog: changelog,
             beta_app_feedback_email: [REDACTED],
             beta_app_description: 'dummy app description for fastlane',
             testers_file_path: "../tester_import.csv"
            )
end
```

- a. **cert** - Code Signing Certificate. It is a distribution and development certificates along with the private key
- b. **sigh** - Creates development and distribution Provisional Profiles. By specifying **force: true**, a new provisioning profile is created on each run; this ensures you're always using the correct code signing certificate.
- c. **increment_build_number**: automatically increment the application build number by one
- d. **changelog**: Fetch the changeling or the Test Information from the file system stored on your machine. Give the correct

location of the file.

e. **gym** - *gym* builds and packages the app. It takes care of all the heavy lifting and makes it super easy to generate a signed ipa or app file.

Mention the correct scheme and the workspace(if using workspace).

include_bitcode - Should the ipa file include bitcode or not.

f. **testflight** - to deploy the application to the testflight. Pass the changelog, app description(beta_app_description), beta app email(beta_app_feedback_email), path to a CSV file of testers(testers_file_path) and other parameters as necessary.

15. To deploy the build to the Testflight, go to the project too directory and run:

fastlane name_of_the_lane

Note: In the example mentioned above, we are using a lane called beta: so run fastlane beta(only is you are using beta lane)

Pending Tasks:

1. Add internal testers to Testflight
2. Screenshots
3. Gitlab CI integration