



ICT Department

រៀបចំបញ្ចេញ App (release mode) សំរាប់ដាក់ចូល Apple AppStore និង Google Play

Instructor: Oum Saokosal, Master of Engineering in Information Systems, South Korea '2010

Email: oumsaokosal@gmail.com

Phone: 012 252 752 (Telegram)

បន្ទាប់ពីយើងបានបង្កើត App បានហើយ គឺយើងដល់ពេលវេលាដែលត្រូវ release App ដើម្បីដាក់ App Store និង Play Store ដើម្បីអោយគេអាចទាញយក App របស់យើងហើយ។ តែមុននឹង release គឺយើងត្រូវដឹងពីរឿងខ្លះសិន៖

1. Account ក្នុង Google Play Store មានតំលៃ 25\$ (បង់តែម្តង)
2. Account ក្នុង Apple AppStore តំលៃ 99\$ ក្នុងមួយឆ្នាំ (បង់ជារៀងរាល់ឆ្នាំ)
3. ដើម្បី release iOS App គឺជាចាំខាតត្រូវតែមាន MacOS និងមានកម្មវិធី XCode



ដើម្បី release App សំរាប់ Android គឺយើងត្រូវ៖

1. បង្កើត keystore៖

សំរាប់ Mac៖

```
keytool -genkey -v -keystore ~/key.jks -keyalg RSA -keysize 2048 -validity 10000 -alias key
```

សំរាប់ Windows៖

```
keytool -genkey -v -keystore c:/key.jks -storetype JKS -keyalg RSA -keysize 2048 -validity 10000 -alias key
```

2. ពេលដែល run cmd ខាងលើនេះ គឺត្រូវដាក់ password ណាមួយ

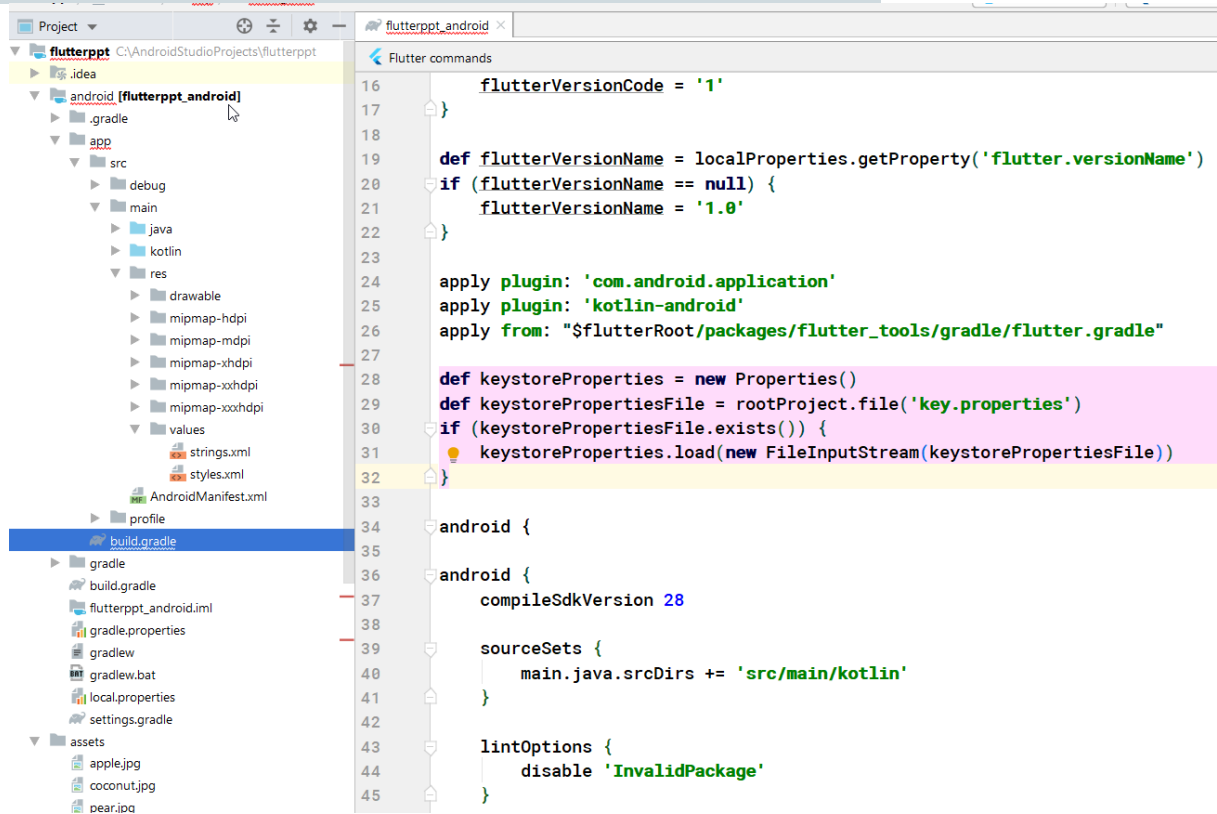
3. បង្កើត file មួយឈ្មោះ key.properties ក្នុង folder <app dir>/android/ ហើយដាក់ កូដខាងក្រោម៖

```
storePassword=<password from previous step>  
keyPassword=<password from previous step>  
keyAlias=key  
storeFile=<keystore location, such as /Users/<user name>/key.jks>
```



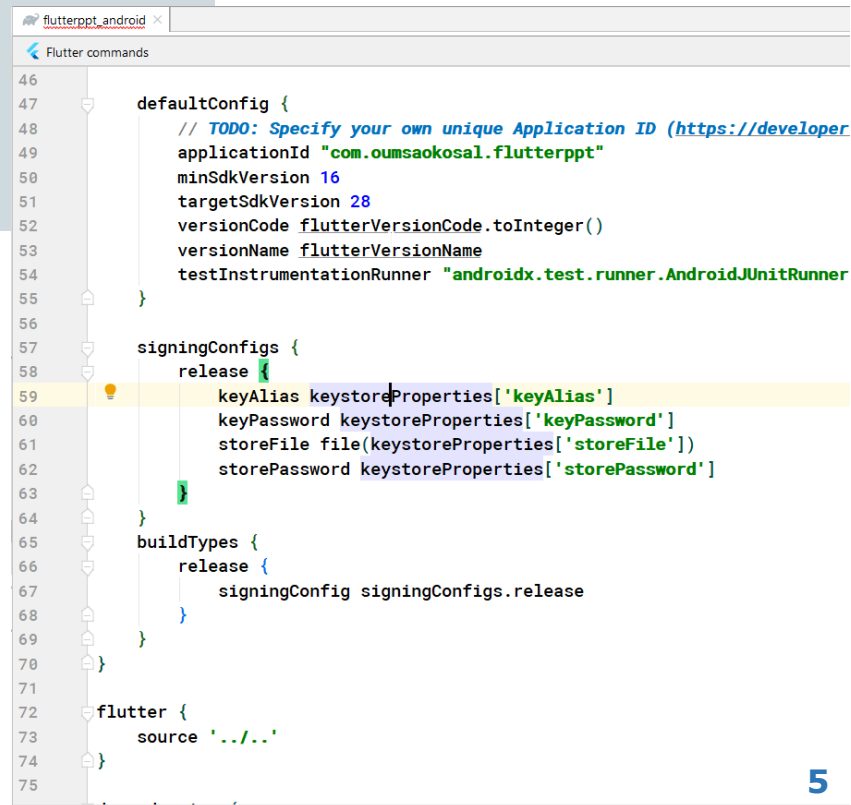
4. ចូលក្នុង <app dir>/android/app/build.gradle ហើយដាក់៖

```
def keystoreProperties = new Properties()
def keystorePropertiesFile = rootProject.file('key.properties')
if (keystorePropertiesFile.exists()) {
    keystoreProperties.load(new
    FileInputStream(keystorePropertiesFile))
}
```



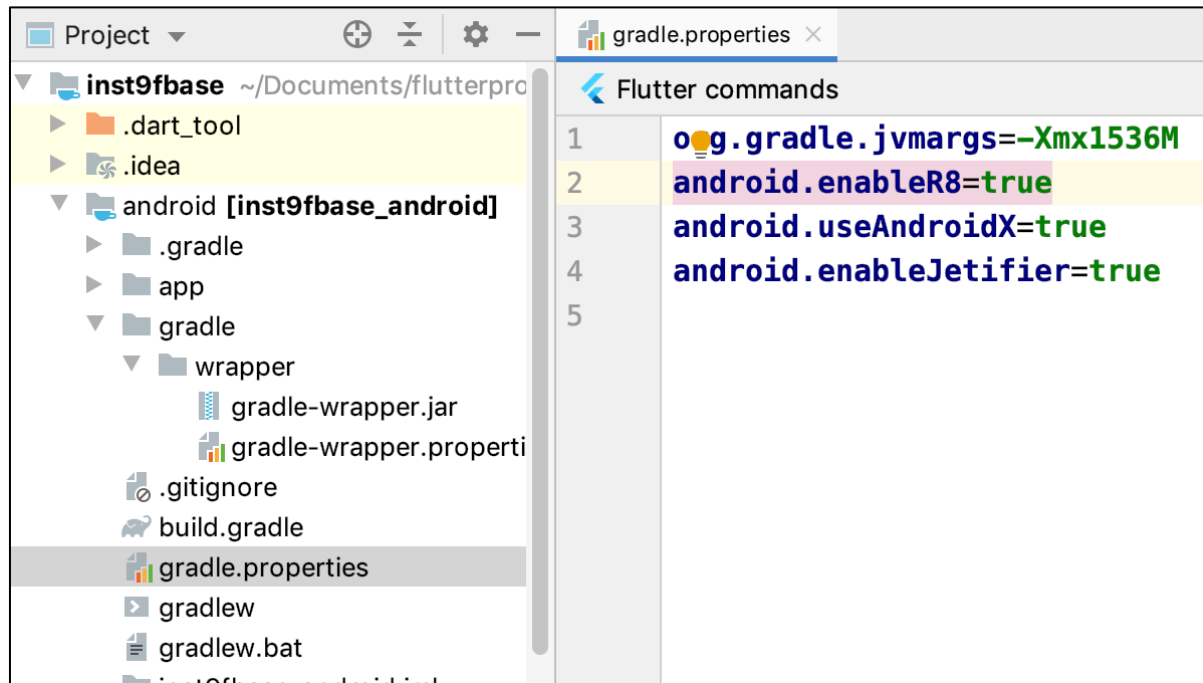
5. ក្នុង file ដដែលនេះ គឺថែមកូដខាងក្រោម៖

```
signingConfigs {
    release {
        keyAlias keystoreProperties['keyAlias']
        keyPassword keystoreProperties['keyPassword']
        storeFile file(keystoreProperties['storeFile'])
        storePassword keystoreProperties['storePassword']
    }
}
buildTypes {
    release {
        signingConfig signingConfigs.release
    }
}
```



```
46
47
48     defaultConfig {
49         // TODO: Specify your own unique Application ID (https://developer
50         applicationId "com.oumsaokosal.flutterptp"
51         minSdkVersion 16
52         targetSdkVersion 28
53         versionCode flutterVersionCode.toInteger()
54         versionName flutterVersionName
55         testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
56     }
57
58     signingConfigs {
59         release {
60             keyAlias keystoreProperties['keyAlias']
61             keyPassword keystoreProperties['keyPassword']
62             storeFile file(keystoreProperties['storeFile'])
63             storePassword keystoreProperties['storePassword']
64         }
65     }
66
67     buildTypes {
68         release {
69             signingConfig signingConfigs.release
70         }
71     }
72
73     flutter {
74         source '../..'
75     }
76 }
```

6. ចូលទៅឆែកមើលថាតើមាន `android.enableR8=true` នៅក្នុង `gradle.properties` ដែលវាមានប្រយោជន៍យ៉ាងខ្លាំង ក្នុងការការពារកុំអោយគេបើកមើលកូដយើងឃើញ៖



```

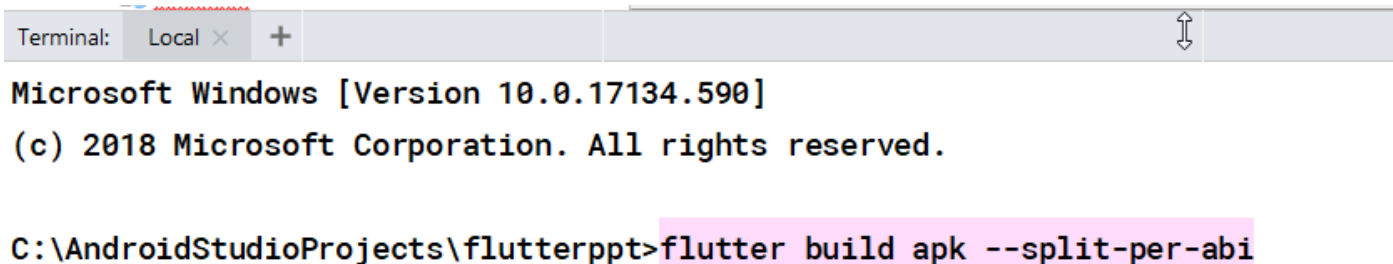
Project ▾
  inst9fbase ~/Documents/flutterpro...
    .dart_tool
    .idea
    android [inst9fbase_android]
      .gradle
      app
      gradle
        wrapper
          gradle-wrapper.jar
          gradle-wrapper.properties
        .gitignore
        build.gradle
        gradle.properties
        gradlew
        gradlew.bat
        inst9fbase_android.iml
  
```

```

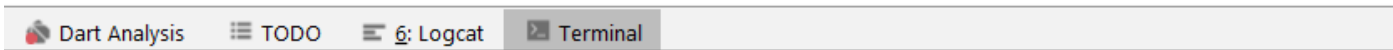
gradle.properties
Flutter commands
1  org.gradle.jvmargs=-Xmx1536M
2  android.enableR8=true
3  android.useAndroidX=true
4  android.enableJetifier=true
5
  
```

7. ចូលក្នុង terminal ហើយវាយកូដ៖

```
flutter build apk --split-per-abi
```



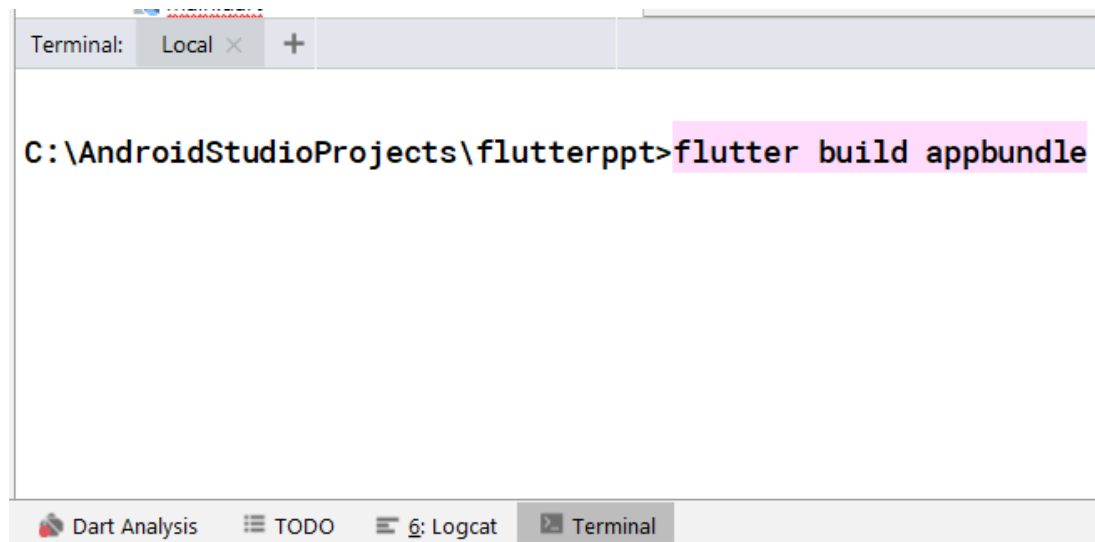
```
Terminal: Local x +  
Microsoft Windows [Version 10.0.17134.590]  
(c) 2018 Microsoft Corporation. All rights reserved.  
  
C:\AndroidStudioProjects\flutterppt>flutter build apk --split-per-abi
```



បញ្ជាក់ កូដខាងលើគឺប្រើសំរាប់ release App ជាក់ចូលទូរសព្ទប៉ុណ្ណោះ។ ដើម្បី release App ជាក់ចូល Play Store យើងត្រូវប្រើ App Bundle (សូមមើល slide បន្ទាប់)។

8. ពេលដែលយើងចង់ដាក់ App ចូលក្នុង Play Store គឺវាទាមទារអោយយើង release App ជា App Bundle។ យើងអាច run cmd ខាងក្រោម ដើម្បីបង្កើត App Bundle៖

```
flutter build appbundle
```



```
Terminal: Local x +  
  
C:\AndroidStudioProjects\flutterppt>flutter build appbundle
```

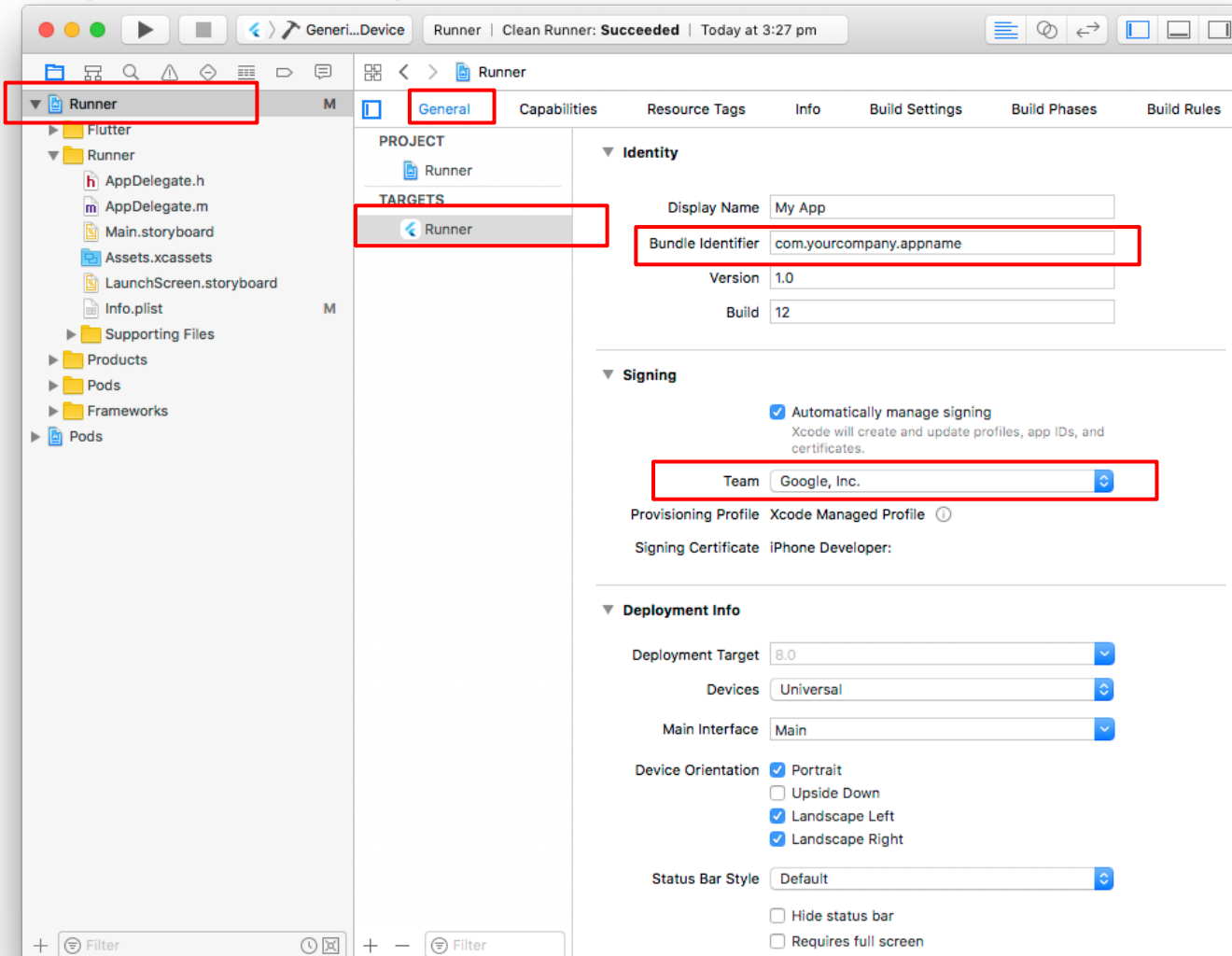
(សូមមើលវីដេអូ <https://flutter.dev/docs/deployment/android> សំរាប់ព័ត៌មានលំអិត)

មុនដំបូងគេបំផុត ដើម្បីអាច Release App សំរាប់ AppStore គឺយើងត្រូវតែមានម៉ាស៊ីន Mac មួយ ដែល run MacOS ហើយត្រូវមាន XCode ព្រមទាំងមាន iPhone និង iPad ពិតប្រាកដ៖

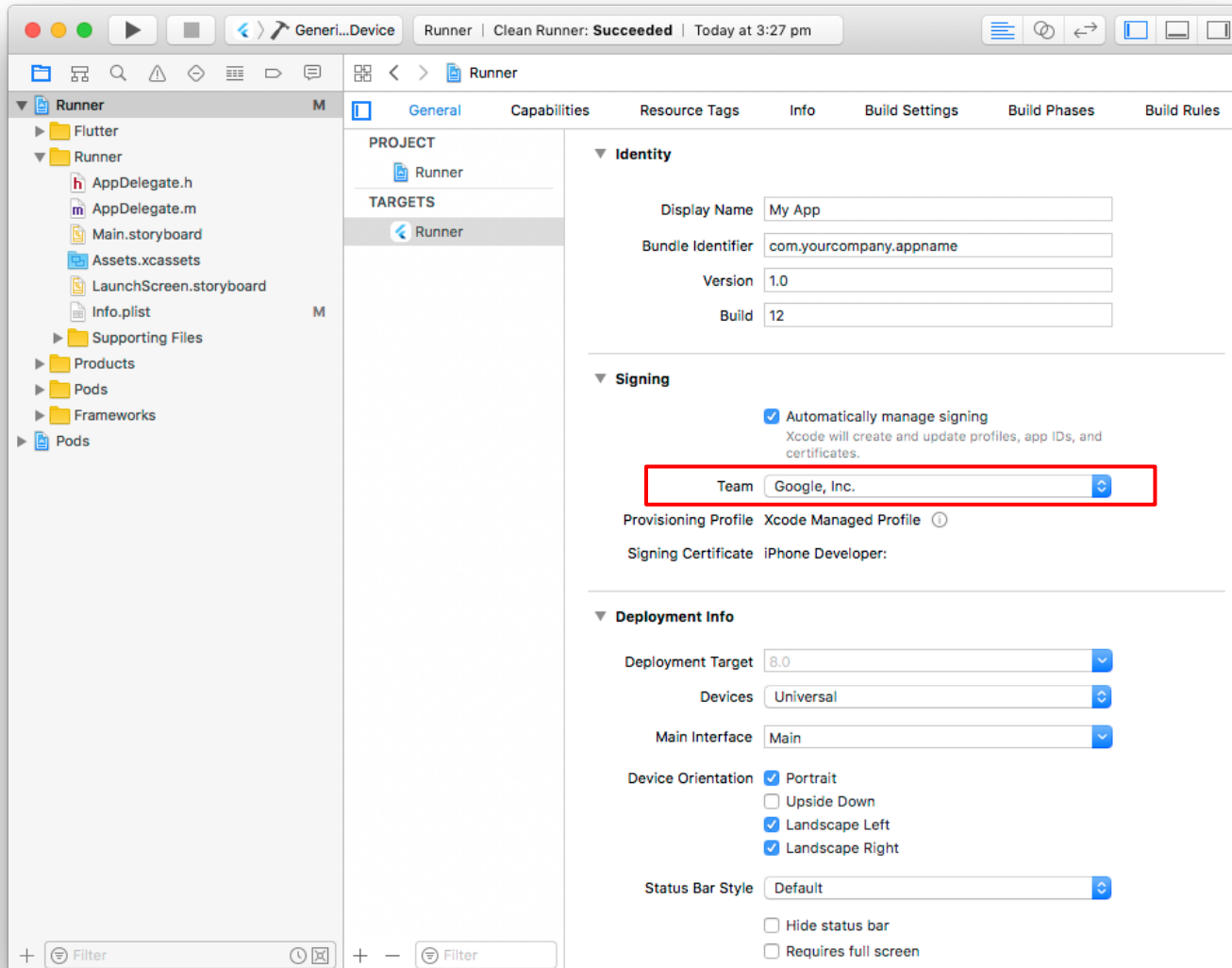
1. ត្រូវបង្កើត Apple ID សិន បើមិនទាន់មាន
2. ត្រូវទិញ Apple Developer Account សិន តម្លៃ 99\$ ក្នុងមួយឆ្នាំ (ត្រូវបង់ថ្លៃតាម Visa Card ឬ Master Card)



3. ចូលក្នុង XCode -> ក្នុង TARGETS ចុចយក Runner -> Bundle Identifier



4. ចុចត្រង់ team ហើយវិស័យកិ developer account របស់យើង៖



5. បើសិនជាយើងមិនបានទិញ Apple Developer Account ទេ គឺយើងអាចត្រឹមតែ test app ជាមួយនឹងម៉ាស៊ីន iPhone ឬ iPad ពិតបាន។ យើងអាចវាយ cmd ខាងក្រោម៖

```
flutter build ios --release
```

បញ្ជាក់ cmd នេះគឺអាច run បានតែក្នុងម៉ាស៊ីន Mac ដែលមាន XCode ប៉ុណ្ណោះ

6. បង្កើត App ថ្មីមួយក្នុង App Store Connect

បញ្ជាក់ លុះត្រាតែយើងមាន Apple Developer ទើបយើងអាចប្រើ App Store Connect បាន

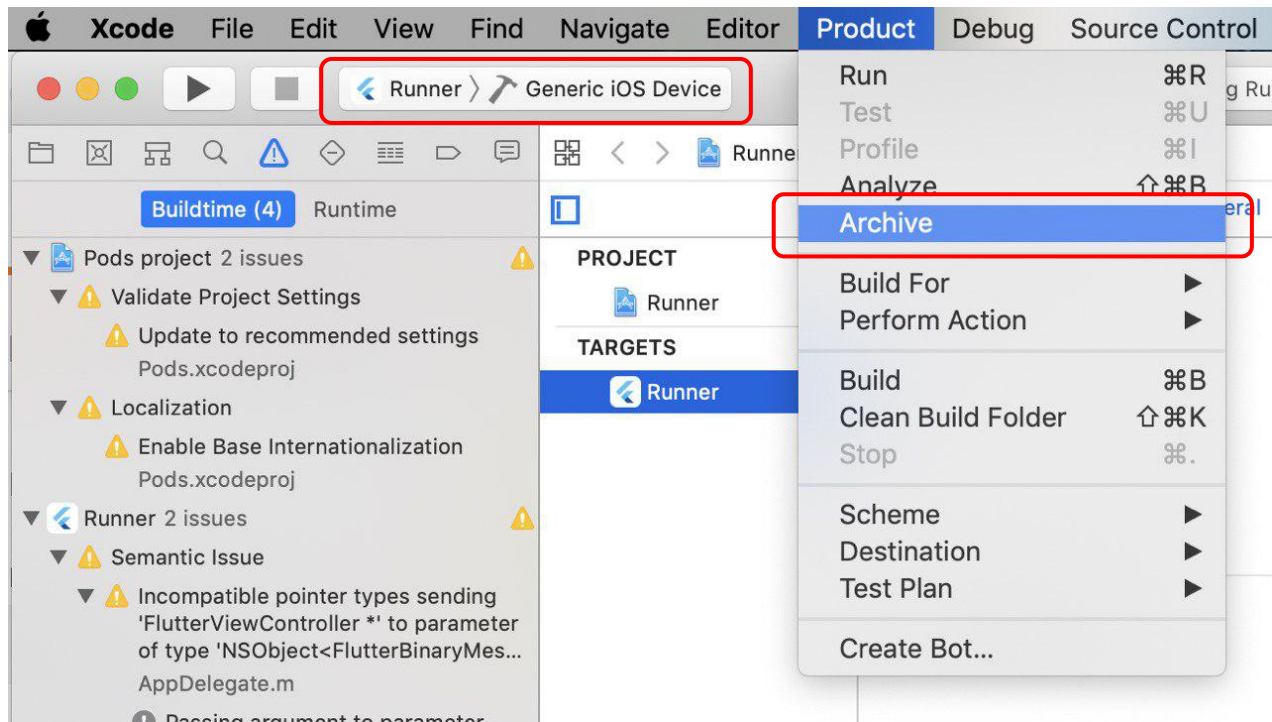
New iOS App

Name ?	Version ?
<input type="text" value="Mag+ New Demo App"/>	<input type="text"/>
Primary Language ?	SKU ?
<input type="text" value="English"/>	<input type="text" value="DemoApp123"/>
Bundle ID ?	
<input type="text" value="Choose"/>	

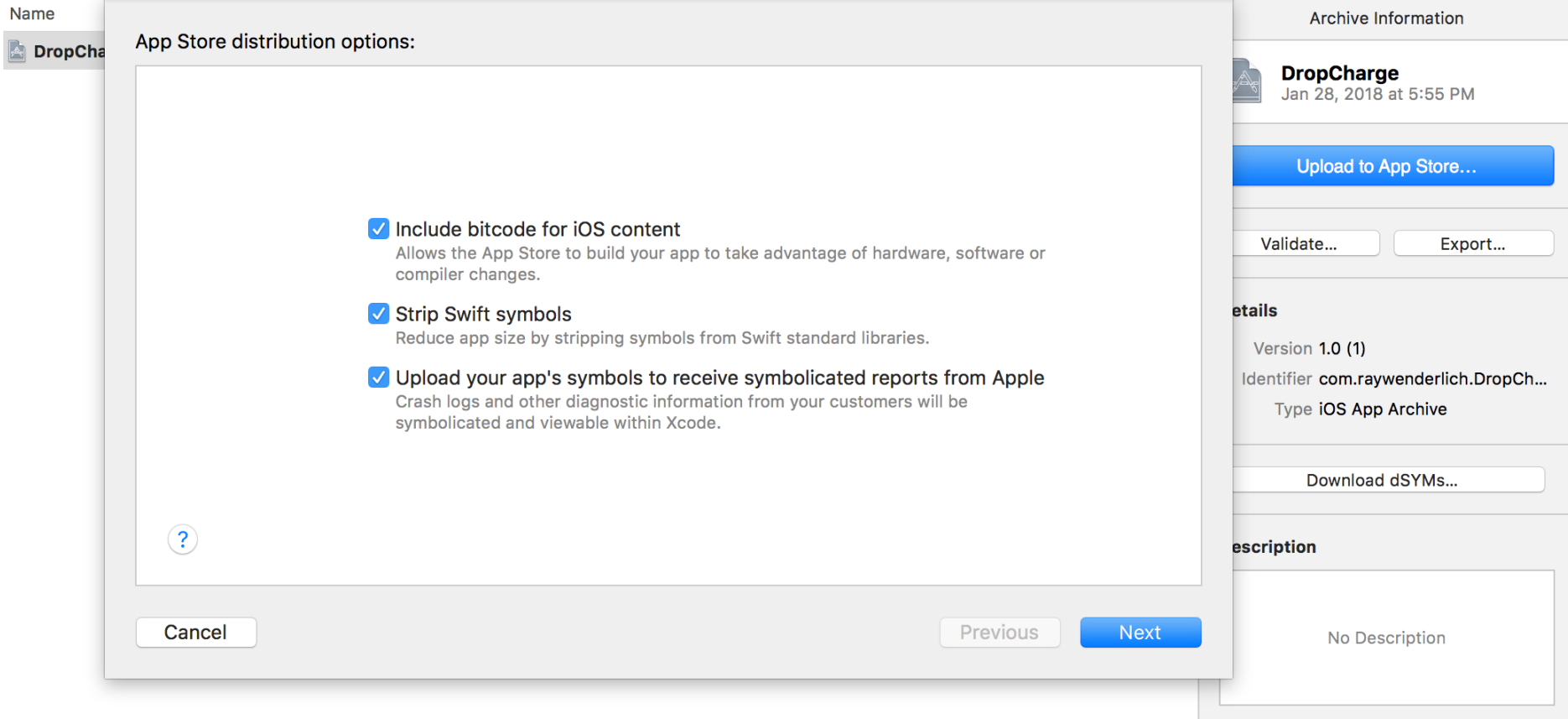
Register a new bundle ID on the [Developer Portal](#).

CancelCreate

7. បន្ទាប់ពីយើងបានបង្កើត App ក្នុង App Store Connect រួចហើយ គឺអាច release App ពី XCode ទៅបាន ដោយចុច **Product > Archive** នៅក្នុង XCode៖
(បញ្ជាក់ ត្រូវចុចយក Generic iOS Device ជាមុនសិន មុនចុចយក Archive)



8. រួចចុច next, ហើយ ចុចលើ Upload to App Store បាន និងរង់ចាំទាល់តែវាចប់។



សូមមើល៖

https://developer.apple.com/documentation/xcode/preparing_your_app_for_distribution