

Table of Contents

- Build script** 1
- Requirements** 1
- Commands** 1
 - Complete (start.complete) 1
 - iOS (start.ios) 1
 - Android (start.android) 1
 - Web (start.web) 3
- Updates** 3

Build script

The build script is available in the **creator** directory. The script is [Apache ANT](#) based and it's a simple XML file. It defines several build commands. The default command (`start.complete`) creates output for iOS, Android and Web deployment.

The iOS build command requires macOS but won't fail if your system is not macOS.

Requirements

Java Runtime environment > 8.0

Commands

Complete (`start.complete`)

```
cd creator
./build.sh
```

or

```
cd creator
build.cmd
```

iOS (`start.ios`)

```
cd creator
./build.sh start.ios
```

Android (`start.android`)

```
cd creator
./build.sh start.android
```

For app signing, a keystore will be used. Our build process needs a file with the name **key.properties**. Put this file in the folder `<project>/creator/android`, together with your keystore. It's also possible to put the file in your android project folder. We support both locations.

The **key.properties** will be used to read all relevant information and should contain:

```
keyAlias=<alias>
keyPassword=<keypassword>
```

```
storeFile=<keystore_filename>  
storePassword=<storepassword>
```

To support signing, it's important to change your gradle build in folder `<project>/android/app/build.gradle`. We need following changes:

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

and

```
signingConfigs {  
    release {  
        keyAlias keystoreProperties['keyAlias']  
        keyPassword keystoreProperties['keyPassword']  
        storeFile keystoreProperties['storeFile'] ?  
rootProject.file(keystorePropertiesFile.getParent() + '/' +  
keystoreProperties['storeFile']) : null  
        storePassword keystoreProperties['storePassword']  
    }  
}  
  
buildTypes {  
    release {  
        signingConfig signingConfigs.release  
    }  
}
```

A [complete build file](#) is available in our project.

Web (start.web)

```
cd creator
./build.sh start.web
```

The results will be available in

```
creator/build/install
```

directory.

Updates

To get the latest build script, simply call

```
cd creator
./update.sh
```

The latest [build.xml](#) will be saved in the creator directory.

From:

<https://doc.sibvisions.com/> - **Documentation**

Permanent link:

https://doc.sibvisions.com/flutterui/build_release



Last update: **2023/02/22 07:31**