Alistar App para publicación Android

Alejandro Romero edited this page 29 days ago · [1 revision](https://github.com/alejo8591/unipiloto-apm-5/wiki/Alistar-App-para-publicaci%C3%B3n-Android/_history)

**Pages 10**

* [**Home**](https://github.com/alejo8591/unipiloto-apm-5/wiki)
* [**Alistar App para publicación Android**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Alistar-App-para-publicaci%C3%B3n-Android)
* [**Android Platform Guide**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Android-Platform-Guide)
* [**Cheat Sheets**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Cheat-Sheets)
* [**Debugging en Apache Cordova**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Debugging-en-Apache-Cordova)
* [**Git: Bitbucket y GitHub desde la trinchera**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Git:-Bitbucket-y-GitHub-desde-la-trinchera)
* [**Material de Apoyo**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Material-de-Apoyo)
* [**Publicar App en el market place de Android (Google Play Store)**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Publicar-App-en-el-market-place-de-Android-(Google-Play-Store))
* [**Splash Screen en las Apps**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Splash-Screen-en-las-Apps)
* [**Tamaño del Icono del App**](https://github.com/alejo8591/unipiloto-apm-5/wiki/Tama%C3%B1o-del-Icono-del-App)

**Clone this wiki locally**



[**Clone in Desktop**](github-windows://openRepo/https:/github.com/alejo8591/unipiloto-apm-5.wiki)

Quitar plugins innecesarios

Se debe tener cuidado de ajustar los plugins necesarios para desarrollar la App que no deberían estar en Producción. Por ejemplo, es probable que no queremos activar el plugin consola de depuración, por lo que deberíamos eliminarlo antes de generar la versiones de lanzamiento:

**$ cordova plugin rm cordova-plugin-console**

Se debe general el release de la App para Android utilizando Cordova:

**$ cordova build --release android**

Se genera el build con base a la información del config.xml; quien es el que contiene la información de configuración necesaria para las diferentes plataformas.

Se debe ubicar el APK unsigned platforms/android/build/outputs/apk el paquete puede tener el nombre HelloWorld-unsigned.apk. Ahora se debe firmar el APK para poder publicarlo en la consola de desarrolladores de Android o herramientas con HockeyApp. Vamos a generar nuestra clave privada usando el comando keytool que viene con el JDK:

**$ keytool -genkey -v -keystore my-release-key.keystore -alias alias\_name -keyalg RSA -keysize 2048 -validity 10000**

Parecido a la creación del package.json la ejecución del comando; en primer lugar debe crear una contraseña la cual se utilizara cada que vayamos a publicar por primera vez o subir alguna actualización del App . A continuación, contestar el resto de las preguntas en la linea de comandos, cuando termina de llenar la información debe generar un archivo llamado mi-release-key.keystore creado en el directorio en el cual esta trabajando. \*\* Nota: Por favor guardar este archivo pues si lo cambian cada vez que generan un release del App, de seguro lo registraran como App nueva en el market place de Android \*\*

Para firmar el APK sin firmar, ejecutar la herramienta jarsigner que también se incluye en el JDK:

**$ jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore my-release-key.keystore HelloWorld-release-unsigned.apk alias\_name**

Este comando firma el APK. Por último, tenemos que comprimir para optimizar el APK al momento de publicar. Se hace con la herramienta zipalign que se puede encontrar en /path/to/Android/sdk/build-tools/VERSION/zipalign.

**$ zipalign -v 4 HelloWorld-release-unsigned.apk HelloWorld.apk**

Ahora tenemos el binario que depende el nombre del App debería aparecer HelloWorld.apk y que se puede publicar en el market place de Android: Google Play.