

Tipos de APK:

debug.apk.

- **Debug APK**: Este APK se genera por defecto y está destinado a pruebas durante el desarrollo. Contiene información de depuración y no está optimizado para su distribución.
- **Release APK**: Este APK está optimizado para su distribución y se debe usar para publicar tu aplicación en Google Play Store u otras plataformas.

Construir una Debug APK:

Abre tu proyecto en Android Studio.
 En la barra de herramientas de Android Studio, haz clic en Build > Select Build Variant y selecciona debug
 Build Variants : —

 Module Active Build Variant
 :app debug (default)

 En la barra de herramientas de Android Studio, haz clic en Build > "Rebuild Project" (Reconstruir proyecto). Esto compilará tu proyecto y generará una APK.
 Una vez que la compilación haya finalizado, puedes encontrar la APK en la siguiente ubicación dentro de tu proyecto: app/build/outputs/apk/debug/app-

Puedes copiar esta APK a tu dispositivo Android para probarla o ejecutarla en un emulador.

Construir una Release APK:

- 1. Abre tu proyecto en Android Studio.
- 2. Asegúrate de que la configuración de compilación sea "release". Debes configurar esto en tu archivo build.gradle del módulo de la aplicación. Busca la sección buildTypes y asegúrate de que release esté configurado correctamente.

Ejemplo:

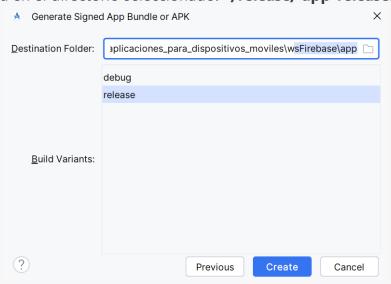
3. En la barra de herramientas de Android Studio, haz clic en Build > Select Build Variant y selecciona release

- 3. En la barra de herramientas de Android Studio, haz clic en Build > "Rebuild Project" (Reconstruir proyecto). Esto compilará tu proyecto y generará una APK.
- 4. Una vez que la compilación haya finalizado, puedes encontrar la APK en la siguiente ubicación dentro de tu proyecto:
 - app/build/outputs/apk/release/app-release-unsigned.apk.
- 5. Esta APK debe ser firmada digitalmente antes de ser distribuida en la tienda de aplicaciones o compartida con otros usuarios. Firmar la APK garantiza la autenticidad de tu aplicación. Para firmarla, puedes utilizar las herramientas de firma de Android Studio o una herramienta externa como jarsigner.

Firmar una Release APK:

1. En Android Studio, haz clic en Build > "Generate Signed Bundle / APK" Seleccionar "APK" y haz clic en "Next".			
	Smaller download size On-demand app features Asset-only modules Learn more APK Build a signed APK that you c ?	o for upload to app stores for the follow an deploy to a device	Cancel
 Si no tienes ningún almacén de claves, pulsa en "Create new", sino elige "Choose existing" En el caso de pulsar en "Create new" rellenar los campos del formulario y pulsar OK. 			
A Generate Signed App Bundle or APK	nerate Signed App Bundle or APK X		×
<u>M</u> odule	v	Key store path: _dispos	ititivos_moviles\almacen de claves\android_apk.jks
Key store path		Password:	Confirm:
Create new Choose existing Alias: key0			
Key store <u>p</u> assword		Password: •••••	
Key alias		<u>V</u> alidity (years):	5 \$
Key pass <u>w</u> ord		Certificate	
Remember passwords		<u>First and Last Name:</u>	Santa Arévalo
? Previous	Next Cancel	Organizational Unit:	Departamento Informática
		Organization:	IES Maestre de Calatrava
		City or <u>L</u> ocality:	Ciudad Real
		State or Province:	Ciudad Real
		Country Code (XX):	13005
			OK Cancel
Si ya tienes un almacén de clave, pulsa en "Choose Existing", selecciona tu almacén			
de claves y rellena los campos de contraseña del almacén de claves y de la clave que			
vas a usar. Después, pulsa en Next.			
A Generate Signed App Bundle or APK X			
Module □ FirebaseEjemplo.app ∨			
Key store path a_dispositivos_moviles\almacen de claves\android_apk, jks			
Create new Choose existing			
Key store password			
Key alias key0			
Key password			
Remember passwords Previous Next Cancel			Cancel

3. Ahora te pedirá el directorio donde quieres generar tu APK firmada. También, debes seleccionar "release" y tras pulsar "Create", verás como se ha generado la apk firmada en el directorio seleccionado: "/release/ app-release.apk"



Antes de firmar la APK:

- -. Define la versión de tu aplicación en el archivo build.gradle (Module: app), en las propiedades de defaultConfig { versionCode y versionName }. Conforme vayas liberando versiones, ves aumentando el versionCode/ versionName para que tu app se actualice.
- -. Asegúrate de que tu aplicación sea compatible con las versiones de Android que desees admitir.