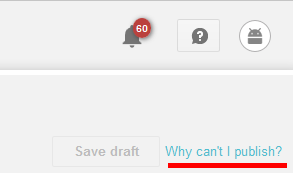
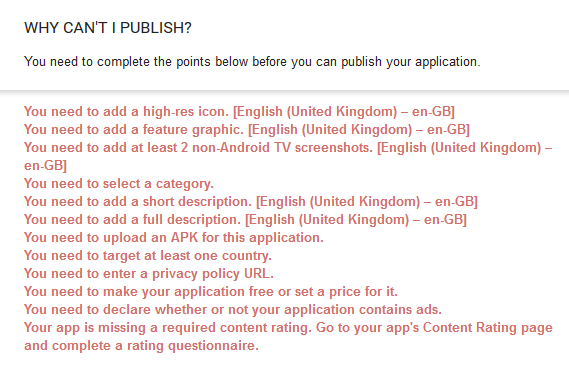
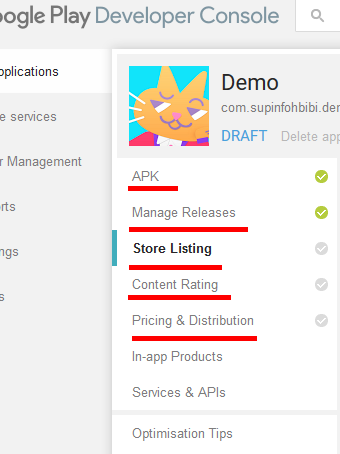
# Create your Android App

In order to publish your app, you will need to upload your app on the Play Store and complete some Google Play settings.



All of this has to configured in the dashboard :



Those are pretty straightforward, and we won’t dwell too much on them.

Still, a few details for the trickiest parts:

* APK : publish your app in beta, and select open testing. We’ll use the “Early Acess” program of Google Play to make sure our apps get some visibility and are played by people all over the world
* Country : target all countries, people are nice you’ll see!
* Privacy Policy: tick the box below the link, we don’t need to worry with those for now
* Content Rating: avoid referencing drugs, promoting violence or doing anything illegal in your app and all should be good!

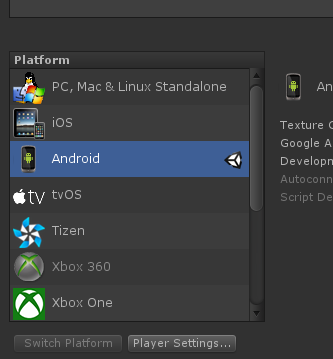
But before all that, you’ll need to develop and upload your game.

All the process to create the APK (the Android App) is described below. You will need to provide a signed build to publish your game and release it to the world !!!

# Unity Setup

## Switch your platform to Android

File > Build Settings...



## Configure an app Bundle Identifier for your game

Edit > Project Settings... > Player > Other Settings > Bundle Idelntifier

Use something like: **com.supinfohbibi.yourgamename**

- use only lower-case letter and digits

- it has to be unique so that no other app in the world can use it

- never change it once you'll have uploaded your first APK

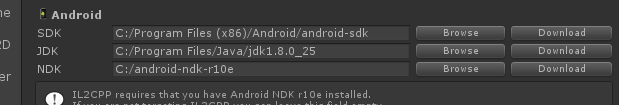


# Android signing

In order to publish your app on the GooglePlayStore, you need to sign it, that is to establish that your are the rightful owner of the app.

## Configure Android for Unity

In Unity, go to Edit > Preferences... > External Tools

Click Download for each package, install it, then configure Unity path to match your installations.

## Create a Keystore

You must create a keystore file that will be used to sign the production builds. You will need to have another Java package installed to create it from a command line.

* Install Java Runtime Environment

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

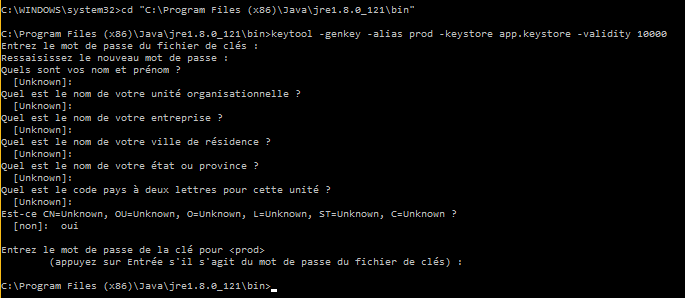
Download and install the latest **Java Runtime Environment (JRE)**



Depending wether you installed the x86 ou x64 version, this will be installed in something like:

* C:\Program Files (x86)\Java
* C:\Program Files\Java

1. Open a DOS command window in Administrator mode:
   1. Open Windows menu
   2. Type "**cmd"** to search the programm
   3. Right click on it
   4. **Execute as Administrator**
2. Go to your JRE bin installation directory where is stored the keytool.exe program, use the "cd" command and be sure to use the " symbol:
   * **cd "C:\Program Files\Java\jre1.8.0\_71\bin"**
3. Enter the following command line:
   * **keytool -genkey -alias prod -keystore app.keystore -validity 10000**
   * Enter a password (be sure to remember it, you will need it later)
   * It is not required to fill Name/Company/Country/...
   * When asked about the last confirmation enter **"yes"** or **"oui"**
   * You're not required to use a different password for "prod"



This process created the file **app.keystore** in the directory where you input the command. “prod” is just an alias name that will be used for the signing process, you will need it in Unity to sign your app.

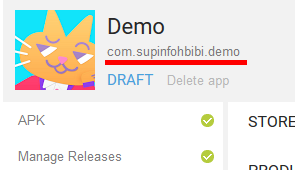
Copy the file to your Unity project, in **Assets/Plugins/Android**.

## Create a signed Unity build

* Go in **Edit > Project Settings > Player > Android** tab > **Publishing Settings** section
* **Browse Keystore** > Select the keystore file you just created
* Input the keystore password and press Enter
* Select the **“prod**” alias
* Input the alias (same than the keystore) password and press Enter
* Create your Android build using **File > Build Settings … > Build**

## Upload your APK

This APK can be uploaded on the GooglePlay console to initialize the Store process:

* <https://play.google.com/apps/publish/>
* Select your app
* Go to the **APK** tab
* In Beta Testing Upload the signed APK you just created
* Once the APK has been uploaded, you can now see your bundle identifier displayed in the dashboard

Then, select Open Beta Testing, tap on Publish in the top right corner and if you have completed all preliminary settings, your app should be available on the store in a few hours!

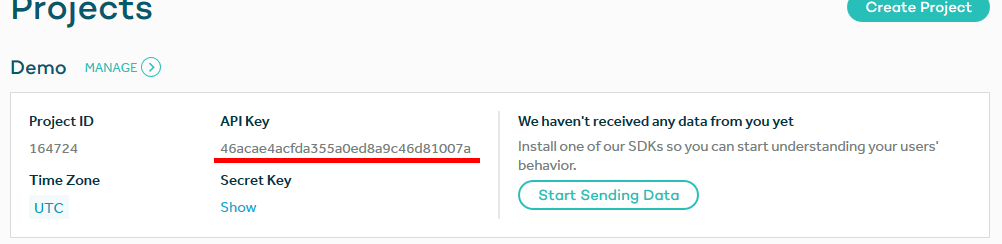
# Amplitude

<https://amplitude.com/home>

This is the very same Analytics tool we use on all our games.

They have a pretty amazing free tier, and will help you monitor your app’s usage and performances.

Once created, use the Amplitude **Project API Key** to initialize Amplitude in your Unity code.



You can either use the AmplitudeHelper documented in our demo code, or download the Amplitude Unity plugin from GitHub:

<https://github.com/amplitude/unity-plugin>

Events studies:

<https://analytics.amplitude.com/org/22841/chart/new>