DDLC Mod Template Guide to Android Mod Making

Works on Ren'Py 6.99.12.4, 7.3.5 and higher.

Guide Version 1.2
Author: GanstaKingofSA#0235 | u/GanstaKingofSA

For DDLC Mod Template Versions: 2.4.0 and higher

Disclaimer: This guide is not endorsed by Team Salvato. Modding should be done when you have completed DDLC. This is also not meant to port mods without author permission. Do not port other people's mods without their permission or release mods on Google Play, Galaxy Store, etc.

DDLC Mod Template Link Download:
https://github.com/GanstaKingofSA/DDLCModTemplate2.0/releases

Contents

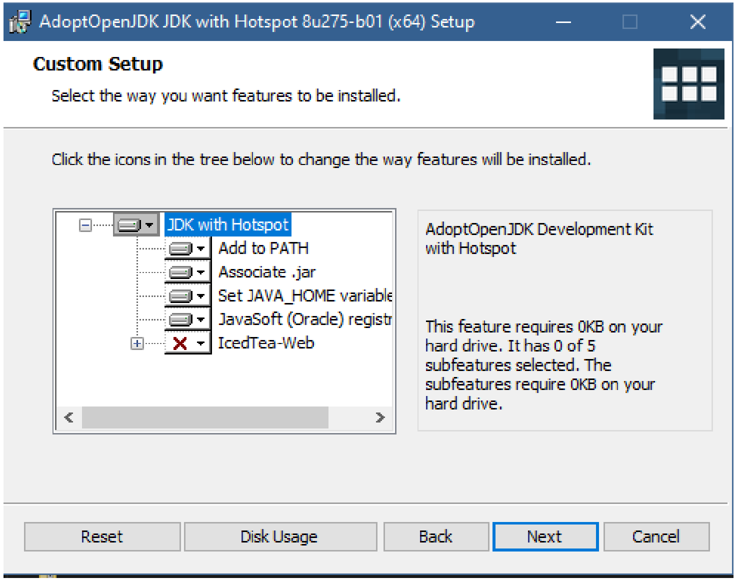
Setting Up The Template

To setup the template on a fresh copy of DDLC to use the Android feature, you'll need the following.

NOTE: To access the contents of **DDLC.app**, right-click DDLC in MacOS and click **Show Package Contents**.

1. A **new install** of DDLC from either https://ddlc.moe or from Steam (only tested on the DDLC.moe copy)
2. **DDLC Mod Template Version 2.4.0** or higher
    a. Please note this is **only** hosted on https://github.com/GanstaKingofSA/DDLCModTemplate2.0/releases.
    b. Extract the template to the folder where **DDLC.exe** [**DDLC.sh** for Linux] is in and accept any replaces.
        i. MacOS/OS X users will have to access this by right via **DDLC.app/Contents/Resources/autorun**
3. **renpy-6.99.12.4-sdk only** or **renpy-7.3.5-sdk**
    a. If you are using Ren'Py 6 (6.99.12.4)
        i. Download **renpy-6.99.12.4-sdk** by downloading it off https://renpy.org/release/6.99.12 . Download either **7z exe**, **dmg**, or **tar.bz2** for your respective OS and extract it somewhere.
    b. If you are using Ren'Py 7 (7.3.5)
        i. Download **renpy-7.3.5-sdk** by downloading it off https://renpy.org/release/7.3.5 and click Download Latest Version. Download either **7z exe**, **dmg**, or **tar.bz2** for your respective OS and extract it somewhere.
4. **Visual Studio Code (VSC)**
    a. It is recommended to install the Ren'Py extension in VSC. To do so go to Plugins (Button with 4 boxes on the left side), search Ren'Py and install the Ren'Py extension to VSC.
5. **AdoptOpenJDK 8** or **Java JDK 8**
    a. If you don't have Java JDK 8 installed, install AdoptOpenJDK 8 instead on https://adoptopenjdk.net/?variant=openjdk8&jvmVariant=hotspot

b. Install everything except IcedTea-Web via the AdoptOpenJDK installer.



AdoptOpenJDK JDK with Hotspot 8u275-b01 (x64) Setup

Custom Setup

Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

- JDK with Hotspot
  - Add to PATH
  - Associate .jar
  - Set JAVA_HOME variable
  - JavaSoft (Oracle) registr
  - ☒ IcedTea-Web

AdoptOpenJDK Development Kit with Hotspot

This feature requires 0KB on your hard drive. It has 0 of 5 subfeatures selected. The subfeatures require 0KB on your hard drive.

   i. Note, building with AdoptOpenJDK 11 installed with AdoptOpenJDK 8 may cause problems. If it doesn't let you build your mod for Android, uninstall AdoptOpenJDK 11 and re-install AdoptOpenJDK 8.

6. **RPATool**
   a. You can obtain RPATool on https://github.com/Shizmob/rpatool/archive/master.zip . Place the rpatool file inside the rpatool-master ZIP -> rpatool-master folder into your mod's game folder.
7. **Python 2.7**
   a. You can download Python on https://www.python.org/downloads/release/python-2718/
8. (Optional) **An android phone** with USB Debugging **on** or **Android Studio** for debugging errors, testing, etc.
   a. If you are using your own android phone:
      i. To turn on **USB debugging**, go to **Settings** -> **About** -> **Software information** and tap **build number** around 5-8 times repeatedly until it says Developer mode has been enabled. Then go back to **Settings** -> **Developer options** and scroll down till you see USB debugging and turn it on.
         1. A prompt may appear saying do you want to trust your computer for USB debugging. Say Yes to it.
         2. This step may differ from device to device. Consult your phone's manufacturer, XDA Developers or Google for more information on turning on developer options.
   b. If you prefer to use Android Studio instead:
      i. Make sure to have Android Virtual Device checked during setup and install everything (including Intel HAXM or the AMD equivalent). Next,

create a project, go to **Tools** -> **AVD Manager** and click the play button to start a virtual android phone for debugging use.

Installation Steps

1. Open the mod template ZIP file and drop its contents from the ZIP to the DDLC base directory where **DDLC.exe** [DDLC.sh for Linux] is located. Accept any replaces.
    a. MacOS/OS X users will have to access this directory via **DDLC.app/Contents/Resources/autorun**
2. Open the rpatool-master ZIP file and double click the rpatool-master folder inside the ZIP. Copy **rpatool** to the game folder in your DDLC mod.
    a. MacOS/OS X users will have to access this directory via **DDLC.app/Contents/Resources/autorun/game**
3. Run DDLC via DDLC.exe, DDLC.app, or via the Linux Terminal by typing ./DDLC.sh
4. The game should run, and you can proceed to make your mod and follow the rest of the guide to setting variables and building your mod for Android.

If you are upgrading your template to 2.4.X, please read this section carefully.

Upgrading from MAS Versions 0.2.0 to 1.1.2 – This update like the original 2.X.X updates have major changes to the code to be like DDLC which MAS' template lacks and especially with the android code for this update. It is highly suggested to move all code you edited to the 2.4.X files to obtain all DDLC functionality, your code, and Android functionality in your mod or follow how 2.4. is formatted and follow this guide still.

Upgrading from Versions 2.1.X to 2.3.1 – Due to the Android code, a bit of some functionality will require a transfer of code as well to 2.4.X. This changes portions of the *poem game*, *splash*, *script* additions, *options* additions, *definition* additions and *gui* code edits to make DDLC work on a mobile device. I suggest transferring the code over or follow how 2.4.X is formatted and follow this guide still.

Building the Mod for Android

Installation Steps:

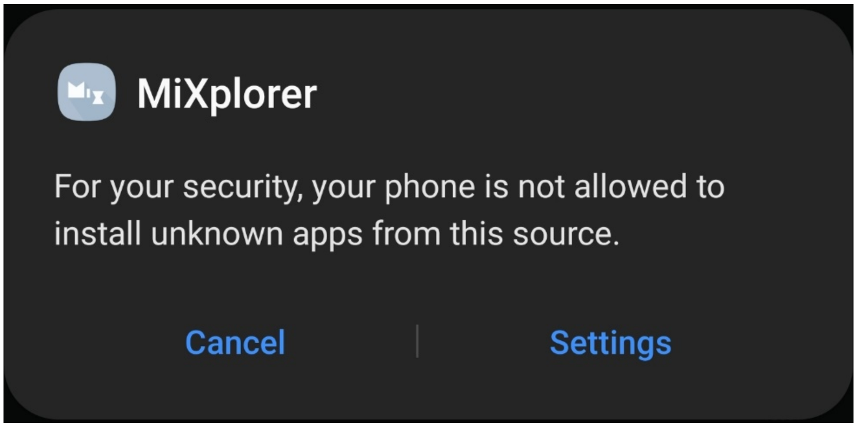
1. Open Terminal, Command Prompt or PowerShell in the game folder and run the following commands
    a. `python rpatool -x audio.rpa`
    b. `python rpatool -x fonts.rpa`
    c. `python rpatool -x images.rpa`
        i. This part must be done as for now Android does not support RPAs. You must extract DDLC's files (except scripts.rpa) to load its files with your mod files. Make sure you BACKUP your mod files before running this else they might get overridden.
        ii. **Keep the RPA files in the game folder!** At minimum remove the archive check in *splash.rpy* if you must or Ren'Py will not compile your mod on since it will give the DDLC archive error. Don't worry, the RPAs won't be included in the final APK release.
2. Start Ren'Py Launcher and select your mod from the Projects Sidebar, then click **Android**
    a. You will be asked to download RAPT (Ren'Py Android Platform Tools). Select Yes and it will download RAPT. It will restart Ren'Py Launcher after it installs.
3. Go back to the **Android** section and click **Install SDK and Select Keys**. It will download the Android SDK and other SDK items. Accept the Android Terms and say yes to setting up a key and to backing it up. Set your organization name to your name.
    a. If the Install SDK and Select Keys is absent and you get that you need to download a Java JDK, you are either missing Java JDK 8 or AdoptOpenJDK8 or misconfigured AdoptOpenJDK8.
    b. Refer to *Setting Up The Template* Step 5b for more information on AdoptOpenJDK8 or Java JDK 8 along with issues with AdoptOpenJDK8 or Java JDK 8 with AdoptOpenJDK11 or Java JDK 11.
4. Click **Configure** and follow the steps below to setup the mod configuration for Android
    i. For Full Name, set it as your mod name.
    ii. For Short Name, set it to the abbreviation of your mod name.
    iii. For Package Name, use *com.* then your name with a period and the build name of your mod. For example, if your name is Monika and your build name for your mod is MAS in *options.rpy* (under *build.name*), it will be formatted like this: *com.monika.mas*
        1. Make sure to change the sdc portion in *options.rpy* to match what was made in package name.
    iv. For Application Version, set the version number of your mod.
    v. For Version Code, leave it as is.
    vi. Set display, set it to landscape orientation.
    vii. For app store in-app purchasing, select Neither.
    viii. Select No to creating an expansion APK.

        ix. (For 6.99.12.4) For version of Android, select Android 4.0.
        x. Select No to allowing the mod to access the internet.
        xi. (For 7.3.5) Select Yes to automatically updating the project.

5. Select **Build Package** and your mod should compile to a APK to be used on Android. You may then sideload it to your phone and test it for errors or enjoy it on the go. See *Notes* however for some notices that need to be address.
    a. If you want to test it on a virtual phone or your own phone, make sure either your virtual phone is on, or your phone is connected to your Computer with USB debugging **on**, select **Build Package & Install** and launch the app once Ren'Py finishes building and test your mod.
    b. For Ren'Py 6, your mod will be compiled with the following name format. Here is an example with mod name *Cookie Club* and version number *2.3.0*
        i. *CookieClub-2.3.0-release.apk*
    c. For Ren'Py 7, your mod will be compiled and output 4 APK files. For universal compatibility between different phone CPU models, use the following name format APK. Here is an example with author name *Natsuki*, mod build name *mlhnstlc* and version number *1.0.0*
        i. com.natsuki.mlhnstlc-10000-universal-release.apk
6. After you tested that your mod works as is, ZIP up your APK or upload it to Google Drive, Mega, etc. and your port should be complete.
    a. Make sure to add that the user must turn on Unknown Sources on their phones temporarily and of a Google Play Protect warning that may occur and to tap **Install Anyways**. In addition, Google will ask to send the APK over to them. This does not break the IPG so it's their choice to select Send or Don't Send. In my opinion, just tap Don't Send and you should be good to go. Refer to **Notes** for more details.
    b. Refer to *Notes* Section 5, if you encounter a *java.lang.OUTOFMEMORYError* when building your mod again for testing purposes.

Notes

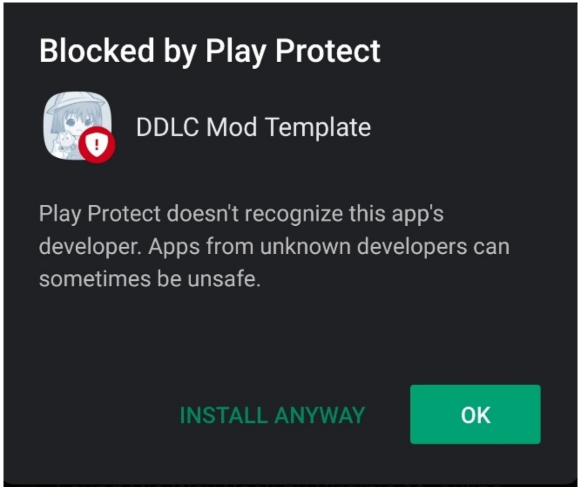
1. When sideloading your mod either via a file manager or by downloading off Chrome, etc. you may get this pop-up by Android or something similar.



For your security, your phone is not allowed to install unknown apps from this source.

To fix this, tap Settings and turn on Unknown Sources. Android will warn you that installing apps from unknown sources may be malicious and some devices allow you to turn this on temporarily for just this one install which is fine. Select either OK or Tap Install Just Once then OK and tap Install. This step will differ from device to device. Consult your phone's manufacturer, XDA Developers or Google for more information on turning on unknown sources

2. Additionally, you may get this pop-up by Google.



Play Protect doesn't recognize this app's developer. Apps from unknown developers can sometimes be unsafe.

This is Google detecting that you the mod developer are someone they don't recognize and think the app may be malicious. Tap Install Anyway and the installation should proceed as normal.

3. While running the game you might get a pop-up to appear that asks whether to send your mod's APK to Google for analysis. This is your choice/player's choice to send it or not. This doesn't violate Team Salvato's IPG as this is a Google Security Message.
4. If AdoptOpenJDK8 or Java JDK 8 and AdoptOpenJDK11 or Java JDK 11 are installed, they may cause an error in Ren'Py in which it can't build properly due to outdated code. Please make sure that only AdoptOpenJDK 8 is installed (re-install it if JDK 11 is uninstalled) and if you need JDK 11 as well, to re-install it after building or build it in another device/virtual machine.
5. Sometimes when building your mod again to fix bugs and try the port again, you may run into an error saying *Caused by: java.lang.OUTOFMEMORYError*. This error is caused due to Java already using the maximum space it can use for building the mod which is 4 GB of RAM. A solution to this is to reboot your PC after a change is made and re-build the mod for Android, try it again, and repeat doing so until your mod works on it.

FAQ

Q. Where are all the game/character files located now?

A. They are located in the APK itself, however some files that can't be loaded from it are copied over to the *Android/data/com.[yourname].[modbuildname]* for deleting files, adding files, etc.

Q. A bunch of my scripts is broken. Why is it the case now?

A. Ren'Py on Android runs differently than on PC. There are different logical patterns needed in Python in Android and such to run properly on all devices. Refer to the template comments or the Ren'Py Documentation for more information.

Q. Does it have to be *com.[yourname].[modbuildname]* to build the mod? It looks cheezy and bland.

A. No. You can change it to be whatever (except the *modbuildname* part), but it must follow the Android naming scheme of *x.y.[modbuildname]* **and** be all lowercased. Consult the other folder naming pattern in the Android/data folder on how to properly name an Android APK data folder.

Q. My GUI elements are all broken. What gives here?

A. The GUI loaded is of DDLC but modified to work on mobile devices. The code for it is located at the bottom section of *gui.rpy*. Change the variables there to adjust your GUI to how it is in-game on PC.

Q. Isn't distributing the APK with DDLC files against Team Salvato's IPG?

A. No. Team Salvato acknowledges that this is a limitation on making DDLC mods available on Android and allows it for the time being, however they will keep an eye on how these APK files are distributed and Android modding may be revoked at any time by Team Salvato.

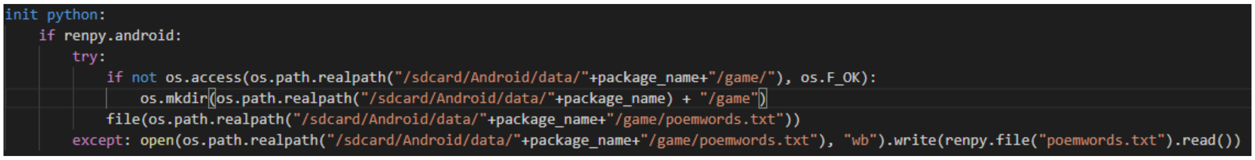
Q. What is the difference between this and JoiPlay?

A. This runs natively on Android rather than emulating a computer. This means it is prone to less bugs and crashes than running mods on JoiPlay. Any errors that appear are either the mod itself being packaged incorrectly or not coded to run differently formatted code for Android.

Q. How would I add my files that need to be run to the *Android/data/com.[yourname].[modbuildname]* folder like *poemwords.txt*?

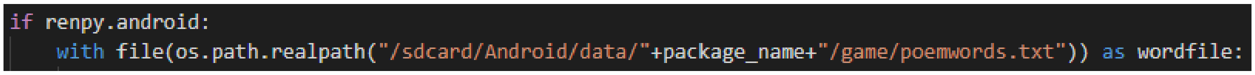
A. Firstly you must make an *init python:* statement in your script to do this.

Make a try statement and then run a check to see if the folder where you store the file exists in the *Android/data/com.[yourname].[modbuildname] folder*. If it doesn't, make one in the folder. Afterwards write the file using *file(* and write the file you want to add to the folder you just made. If it causes a exception/error then make Ren'Py write the file you want to that folder. Here is an example with the code to add *poemwords.txt* to it so *script-poemgame.rpy* can work properly. You may refer to this for other files needed to be written to it.



```
init python:
    if renpy.android:
        try:
            if not os.access(os.path.realpath("/sdcard/Android/data/"+package_name+"/game/"), os.F_OK):
                os.mkdir(os.path.realpath("/sdcard/Android/data/"+package_name) + "/game")
            file(os.path.realpath("/sdcard/Android/data/"+package_name+"/game/poemwords.txt"))
        except: open(os.path.realpath("/sdcard/Android/data/"+package_name+"/game/poemwords.txt"), "wb").write(renpy.file("poemwords.txt").read())
```

Also, if you intend to read the file, make sure also there is a check like this example which points to where poemwords.txt will be located in.



```
if renpy.android:
    with file(os.path.realpath("/sdcard/Android/data/"+package_name+"/game/poemwords.txt")) as wordfile:
```

This FAQ is not an exhaustive list of questions for Android DDLC modding development. For further questions, ask in the *#mod_help* channel or *#mod_faq* channel on the DDMC Discord linked here or contact me in DM for any analysis of what might be wrong. Be advised that not many members will know about Android modding for DDLC and support for it is limited. Explain in depth the issue and someone will try to assist you.

DDMC Discord Invite: https://discord.gg/Gx7m9xE

Acknoledgements to

Tom Rothamel i.e. PyTom for Ren’Py and Ren’Py Android functionality

Dan Salvato for making DDLC and allowing us to mod it to our hearts content that is IP compliant.

You for reading this whole guide.

Weiss Schnee for support.