

# **The DDLC Mod Template Guide to Android Mod Making**

Works on Ren'Py 6.99.12.4 and Ren'Py 7.4.5 – Latest Tested Version

Guide Version 1.6

Author: GanstaKingofSA#0235 | u/GanstaKingofSA

# Android Guidelines

---

Last Updated on January 12th, 2022

## Disclaimer

This guide is not affiliated or endorsed in anyway with Team Salvato. Neither Dan Salvato, nor any current or former Team Salvato employees took part in the writing of this guide. This guide and the Android Guidelines here are solely made by Azariel Del Carmen with the assistance of others; which are credited in the [Acknowledgements](#) page of this guide.

## Mods on Android

This guide is not meant to port the mods made by other mod developers to Android without their permission. This guide is meant for you to port your **own** mod project to Android or port other mod developers work with their explicit permission.

As per Team Salvato's Intellectual Property Guidelines, you may not port DDLC itself over to Android, nor release it onto the Google Play Store, Amazon Appstore, Samsung's Galaxy Store, Huawei's App Store, Xiaomi's Mi GetApps, FDroid, Aurora Store, Itch.io, or any other first-party/third-party Android app store. This also applies to DDLC mods. Mods that are released onto these platforms will be reported to Team Salvato.

## Everything Else

These guidelines is not necessarily exhaustive. For questions about what aspects of DDLC may be included in your port, you may contact GanstaKingofSA or Team Salvato for clarification.

Thank you for taking the time in reading the Android Guidelines. This disclaimer is not meant to discourage people to make their mods available on Android or make an Android mod, rather express the desire to make IP compliant and mod developer permitted Android DDLC-related content. We hope to see mod developers make their mods available on Android and see fans of the game enjoy DDLC mods on-the go and enjoy DDLC mods in many ways.

## Contents

---

Android Guidelines .....	2
Disclaimer .....	2
Mods on Android.....	2
Everything Else.....	2
Setting Up Your PC for Android Development .....	4
Installing the Mod Template.....	5
Upgrading Ren'Py/The Template.....	7
Upgrading from the Monika After Story/RationalPi Template (Versions 0.2.0 to 1.1.2) ....	7
Upgrading from Versions 2.1.0 to 2.3.1 (includes 2.3.1-uX versions).....	7
Upgrading from Versions 2.4.0 to 2.4.9 .....	7
Android by following a DDLC guide (Tormuse's Mod Guide).....	7
Android via a tool-made project (DDMM/DDMMaker).....	7
Android under Ren'Py 7.3.5 - 7.4.4 .....	7
Building the Mod for Android .....	8
Notes.....	10
FAQ.....	13
Acknowledgements.....	15

## Setting Up Your PC for Android Development

---

To get started, you will need to setup and have the following.

1. A system with at least 8 GB of RAM.
  - a. Depending on the size of your mod, this number may increase or decrease in size.
  - b. Mods which files (including extracted DDLC assets) total more than 500MB will need more than 4 GB RAM in order to build properly.
2. A fresh copy of DDLC from [DDLC.moe](https://ddlc.moe).
3. DDLC Mod Template Version 2.4.0 or higher from [GanstaKingofSA](#)'s template repository.
  - a. This will ***not*** work on the MAS (Monika After Story)/RationalPi template or by decompiling the game yourself.
  - b. For more information, see [Upgrading from the Monika After Story/RationalPi Template \(Versions 0.2.0 to 1.1.2\)](#) and/or [Android by following a DDLC guide](#).
4. Ren'Py 6.99.12.4 or Ren'Py 7.4.5 (or above) SDK.
  - a. The files to obtain these Ren'Py versions are in *Installing The Mod Template*.
  - b. Ren'Py versions 7.3.5 – 7.4.4 will not work. For more information, see [Android under Ren'Py 7.3.5 – 7.4.4](#).
5. Visual Studio Code + the Ren'Py Extension from [Microsoft](#).
  - a. To install the Ren'Py Extension, go to *Extensions* (Button with 4 boxes on the left side), search *Ren'Py* and install the first extension in the results bar.
6. Eclipse Temurin 8 from [Adoptium](#).
  - a. Building your mod while having Temurin 8 & 11 installed may cause problems. See **Notes** Section D for more information.
7. RPATool from [Shizmob](#)'s repository.
8. Python from [Python.org](https://python.org).
  - a. Make sure to check *Add python to PATH* when installing Python to your system.
9. An android phone.

## Installing the Mod Template

---

**Note:** If you are upgrading your template to 3.0.X, followed a guide, or used a tool to make your mod project, please read [\*Upgrading Ren'Py/The Template\*](#) before proceeding.

**Note for MacOS Users:** To access the directories inside DDLC.app, right-click on it and select *Show Package Contents*.

1. Download the Ren'Py SDK and place the folder inside somewhere on your desktop.
  1. If you are making a Ren'Py 6 mod, download the Ren'Py 6.99.12.4 SDK [here](#).
  2. If you are making a Ren'Py 7 mod, download the latest [tested](#) version [here](#).
2. Make a folder for your mod inside the extracted Ren'Py SDK folder.
3. Download DDLC from DDLC.moe and follow the instructions for your operating system.
  1. **Windows/Linux**
    - Open *ddlc-win.zip* and double-click the *DDLC-1.1.1-pc* folder. Copy its contents to the folder you made in the Ren'Py SDK folder.
  2. **MacOS**
    - If you have a *ddlc-mac* [folder](#), copy the contents inside it to the folder you made in the Ren'Py SDK folder.
    - If you have a *ddlc-mac* [ZIP file](#), open *ddlc-mac.zip* file and copy its contents to the folder you made in the Ren'Py SDK folder.
4. Download the mod template and follow the instructions for your operating system.
  1. **Windows/Linux**
    - Open *DDLCModTemplate-3.0.X.zip* and copy its contents to the folder you made in the Ren'Py SDK folder
  2. **MacOS**
    - If you have a *ddlc-mac* [folder](#), copy the contents inside it to the folder you made in the Ren'Py SDK folder.
    - If you have a *ddlc-mac* [ZIP file](#), open *ddlc-mac.zip* file and copy its contents to the folder you made in the Ren'Py SDK folder.
5. Download RPATool and follow the instructions for your operating system.
  1. **Windows/Linux**
    - Open the *rpatool-master* [ZIP file](#) and double click the *rpatool-master* folder. Copy *rpatool* to the *game* folder.
  2. **MacOS**

- If you have a *rpatool-master* folder, copy *rpatool* from within the *rpatool-master/rpatool-master* folder to *DDLC.app/Contents/Resources/autorun/game*.
  - If you have a *rpatool-master* ZIP file, open *rpatool-master.zip* and double click the *rpatool-master* folder inside it. Copy *rpatool* to *DDLC.app/Contents/Resources/autorun/game*.
6. Launch your mod by following these steps.
1. Go back to the Ren'Py SDK folder and launch the following for your operating system.
    - **Windows:** *renpy.exe* (*renpy-32.exe* if you are on a 32-bit system)
    - **MacOS:** *renpy.app*
    - **Linux:** Open a Linux terminal in Ren'Py SDK folder and type *./renpy.sh*
  2. Select your mod from the *PROJECTS* tab and click Launch Project.
  3. If the game runs without any errors, you can exit the game and proceed to make your mod and follow the rest of the guide to setting up your mod for Android.
    - If DDLC doesn't start for the first time, try launching the game again.
    - If that still doesn't work, look for any traceback or error.txt files, else report the issue to the GanstaKingofSA repository.

## Upgrading Ren'Py/The Template

---

### *Upgrading from the Monika After Story/RationalPi Template (Versions 0.2.0 to 1.1.2)*

- This update has major code changes to be like DDLC which the Monika-After-Story/RationalPi template lacks; especially the android code needed to run the game on Android.
- It is highly suggested to move all your existing code to the 3.0.X files to retain all DDLC functionality, your past work, and have Android compatibility in your mod.

### *Upgrading from Versions 2.1.0 to 2.3.1 (includes 2.3.1-uX versions)*

- Due to the Android code, some functionality will require a transfer of code to 3.0.X.
- I suggest transferring the code you have done to the 3.0.X files to retain your past work and have Android compatibility in your mod.

### *Upgrading from Versions 2.4.0 to 2.4.9*

- Templates above 2.4.5 shouldn't be affected too much as the Android code in the template has just been optimized and fixed slightly.
- If you do plan to upgrade, please note the following.
  - `package_name` in `options.rpy` has become obsolete in favor of environment paths.
  - `script-poemgame`, `splash`, `script`, `options`, and `definitions.rpy` have changed a bit with its Android code; similar to `options.rpy`.

### *Android by following a DDLC guide (Tormuse's Mod Guide)*

- Some DDLC guides will have you download some decompiled scripts or show you how to decompile DDLC with RPATool.
- This, however, will not make your mod work on Android as it needs Android specific code for the game to run properly.
- Rely on the mod template files itself for your Android mod and this guide to help you make your mod work on Android.

### *Android via a tool-made project (DDMM/DDMMaker)*

- If you made your project in DDMM/DDMMaker, you should be fine following this guide as the tool uses the correct mod template.
- Any future tools made may not work with this guide's steps, it's provided mod template and such.

### *Android under Ren'Py 7.3.5 – 7.4.4*

- Due to BinTray shutting down, Ren'Py versions older than 7.4.5 will not be able to compile mods on Android anymore.
- There is no solution to resolve this problem apart from upgrading the SDK to 7.4.5 for now.

## Building the Mod for Android

---

Read the steps carefully before doing anything. This section assumes you only followed the steps in **Installing The Mod Template** or have a mod finished with a Android template.

**Note for DDMM/DDMMaker Users:** When following this guide, refer to Ren'Py Launcher as DDMM/DDMMaker.

1. Before doing anything, make sure that your files are classified in *options.rpy* like the one below. This is needed so your files are included for your PC mod and Android mod.
  - `build.classify("game/shakespeare.pdf", "mod all")`
2. Open Terminal, Command Prompt or PowerShell in the game folder and run the following commands. Make sure you **backup** your mod files before you run them or they might get overridden with DDLC's own files.
  - `python rpatool -x audio.rpa`
  - `python rpatool -x fonts.rpa`
  - `python rpatool -x images.rpa`
    - If you get a *Python was not found* and not a '*python*' is not recognized error, see **Notes** Section 7 for more information.
    - If you get a '*python*' is not recognized error, then Python is not installed properly. Refer to **Setting up the Template** Section 8 for more information.
    - **Keep the RPA files in the game folder!** Don't worry, the RPAs won't be included in the final APK release.
3. Start Ren'Py Launcher and select your mod from the Projects Sidebar, then click *Android*.
  - You will be asked to download RAPT (Ren'Py Android Platform Tools). Select *Yes* and it will download it. It will restart Ren'Py Launcher after it installs.
4. Go back to the *Android* section and click *Install SDK and Select Keys*.
  - Accept the Android Terms and say *Yes* to setting up a key and to backing it up. Set your organization name to your name.
    - If the *Install SDK and Select Keys* is grayed-out, you are either missing Temurin 8 or misconfigured it. Refer to **Setting Up The Template** Section 6 along with **Notes** Section 4 for more information.
5. Click *Configure* and follow the steps below to setup the mod configuration for Android.
  - For Full Name, set it as your mod name.
  - For Short Name, set it to your mod name's abbreviation.
  - For Package Name, use *com.* then your name with a period, then the build name of your mod.
    - Example: *com.yourname.buildname*

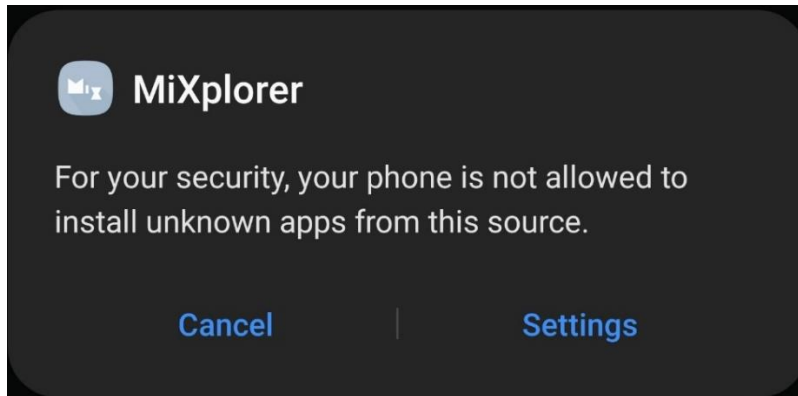


- For Application Version, set it to the version number of your mod.
  - For Version Code, leave it as is.
  - **(For Ren'Py 7)** For RAM Allocation to Gradle, leave it as-is.
    - If you run into *java.lang.OutOfMemoryError* when building your mod, return to *Configure* and increase this number.
    - Remember to not exceed your computer's RAM size. Refer to **Notes** Section 5 for more information.
  - For Display, set it to *landscape orientation*.
  - For app store in-app purchasing, select *Neither*.
  - Select *No* to creating an expansion APK.
  - **(For Ren'Py 6)** For version of Android, select *Android 4.0*.
  - Select *No* to allowing the mod to access the internet.
  - **(For Ren'Py 7)** Select *Yes* to automatically update your project.
6. Select *Build Package* and your mod should compile to a APK. You may then sideload it to your phone and test it for errors before release.
- For Ren'Py 6, your mod will be compiled with the following naming scheme: *buildname-X.X.X-release.apk*
  - For Ren'Py 7, your mod will be compiled and output 4 APK files. For universal compatibility between different phones, use the APK that follows this naming scheme: *com.yourname.buildname-XXXXX-universal-release.apk*
    - If you encountered a *java.lang.OutOfMemoryError* while building your mod, refer to **Notes** Section 5 for more information.
7. After you tested that your mod works as is, your port should be complete.
- Make sure to tell players that they must turn on *Unknown Sources* on their phones temporarily to install your mod and to tap *Install Anyways* if they get a Google Play Protect warning. Refer to **Notes** Sections 1-3 for more information.

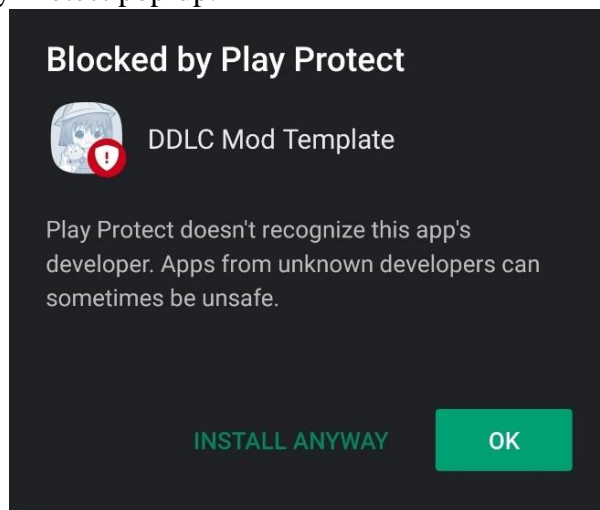
## Notes

---

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



- To fix this, tap *Settings* and turn on Unknown Sources.
    - This step will differ from device to device. Consult Google for more information on turning on unknown sources for your device.
  - Android will warn you that installing apps from unknown sources may be malicious and some devices allow you to turn this on temporarily which is fine. Select either *OK* or *Install Just Once* then select *Install*.
2. When installing your mod either via a file manager or by downloading off Chrome, etc. you may get this Play Protect pop-up.

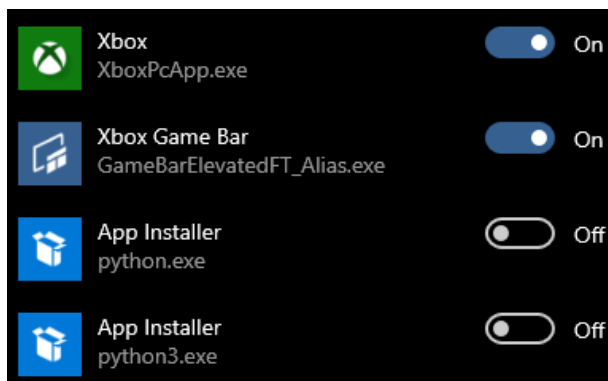


- This is Google detecting that you are someone they do not recognize as a developer and think the app may be malicious.
- Tap Install Anyway and the installation should proceed as normal.

3. While running the game you might get a pop-up that asks whether to send your mod's APK file to Google for analysis.
  - You can choose either option presented as this is just a Google Security Message.
4. If Eclipse Temurin OpenJDK 8 and 11+ are installed on the same computer, they may cause an error in Ren'Py in which it cannot build your mod due to outdated code.
  - Make sure that only Temurin 8 is installed (re-install it if Temurin 11+ was uninstalled).
  - If you need Temurin 11+ on your system, to build your mod on another device or on a virtual machine with Temurin 8 installed.
5. Sometimes when building your mod, you may run into the following error: *Caused by: java.lang.OutOfMemoryError*.
  - This error is caused due to Java using all the space it can use for building the mod which is 3 GB.
  - **Easy Fix**
    - Go back to the *Android* section and click *Configure*, then follow the prompts and increase the RAM allocation number.
  - **Advanced Fix**
    - Go to *rapt/project* and open *gradle.properties* using Visual Studio Code. Change the three in *-Xmx3g* to a different number.
    - Make sure to not exceed the total amount of RAM there is on your system. I recommend 5-8 as the Gradle allocation size but it depends per system.
6. When executing the RPATool commands in Windows, you might get this error.
 

*Python was not found; run without arguments to install from the Microsoft Store, or disable this shortcut from Settings > Manage App Execution Aliases.*

  - To fix this, go to Windows Search and type *Manage App Execution Aliases*.
  - Turn off App Installer for *python.exe*. Re-open Command Prompt or Powershell and type the commands in again.



7. In some rare instances, RPATool may not work if you run the commands with Python 3 and Python 2 installed.
  - To fix this, do *py -2* instead of *python* and try typing the commands again with it.

## FAQ

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

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

Q. A bunch of my are broken on Android.

- 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 the name have to be *com.yourname.buildname*? It looks cheezy and bland.

- No. You can change it to be whatever (except the *buildname* part), but it must follow the Android naming scheme of *x.y.buildname* **and** be all lowercased.
- Consult the other folder naming patterns 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?

- 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's own files against Team Salvato's IPG?

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

Q. What is the difference between this and JoiPlay?

- 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; which is known to break many mods and games.
- 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 read, written to or modified in *Android/data/com.yourname.buildname* folder like the character files?

- The mod template files shows some examples on how these files work in the game under Android. Below is a example of code that writes files into the Android directory.
  - This code below shows an example of writing *CAN YOU HEAR ME.txt* to the game's Android directory.

```
# This python statement writes a file from within the game to the game folder
# or to the Android/data/[modname]/files/game folder.
python:
    if renpy.android:
        try: renpy.file(os.environ['ANDROID_PUBLIC'] + "/hxppy thxughts.png")
        except: open(os.environ['ANDROID_PUBLIC'] + "/hxppy thxughts.png", "wb").write(renpy.file("hxppy thxughts.png").read())
    else:
        try: renpy.file(config.basedir + "/hxppy thxughts.png")
        except: open(config.basedir + "/hxppy thxughts.png", "wb").write(renpy.file("hxppy thxughts.png").read())
```

This FAQ is not an exhaustive list of questions for Android DDLC modding development. For further questions, ask in the *#mod\_help* channel or see the *#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>

## Acknowledgements

I want to acknowledge the following people for their assistance, either voluntary or in-voluntary.

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.

My past projects to optimize the mod template with better code.

You for reading this whole guide.

Weiss Schnee for support. (Weiss: 😊)