# HOW TO BUILD MUJOCO FOR ANDROID

5 March 2025

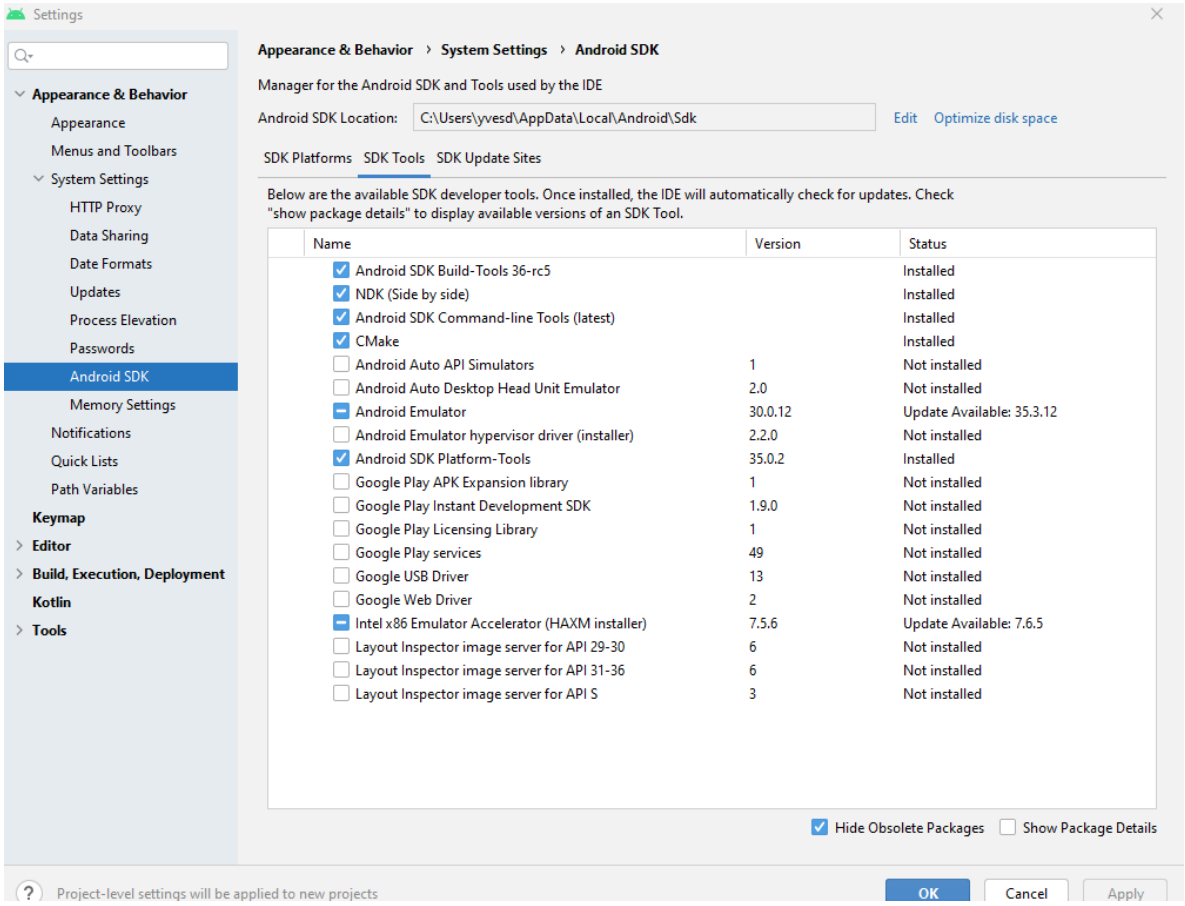
This compilation method has been tested on Mujoco 3.2.6 in the Mujoco Unity plugin.
The compilation method also works for Mujoco 3.3.0.

## 1. Android Studio

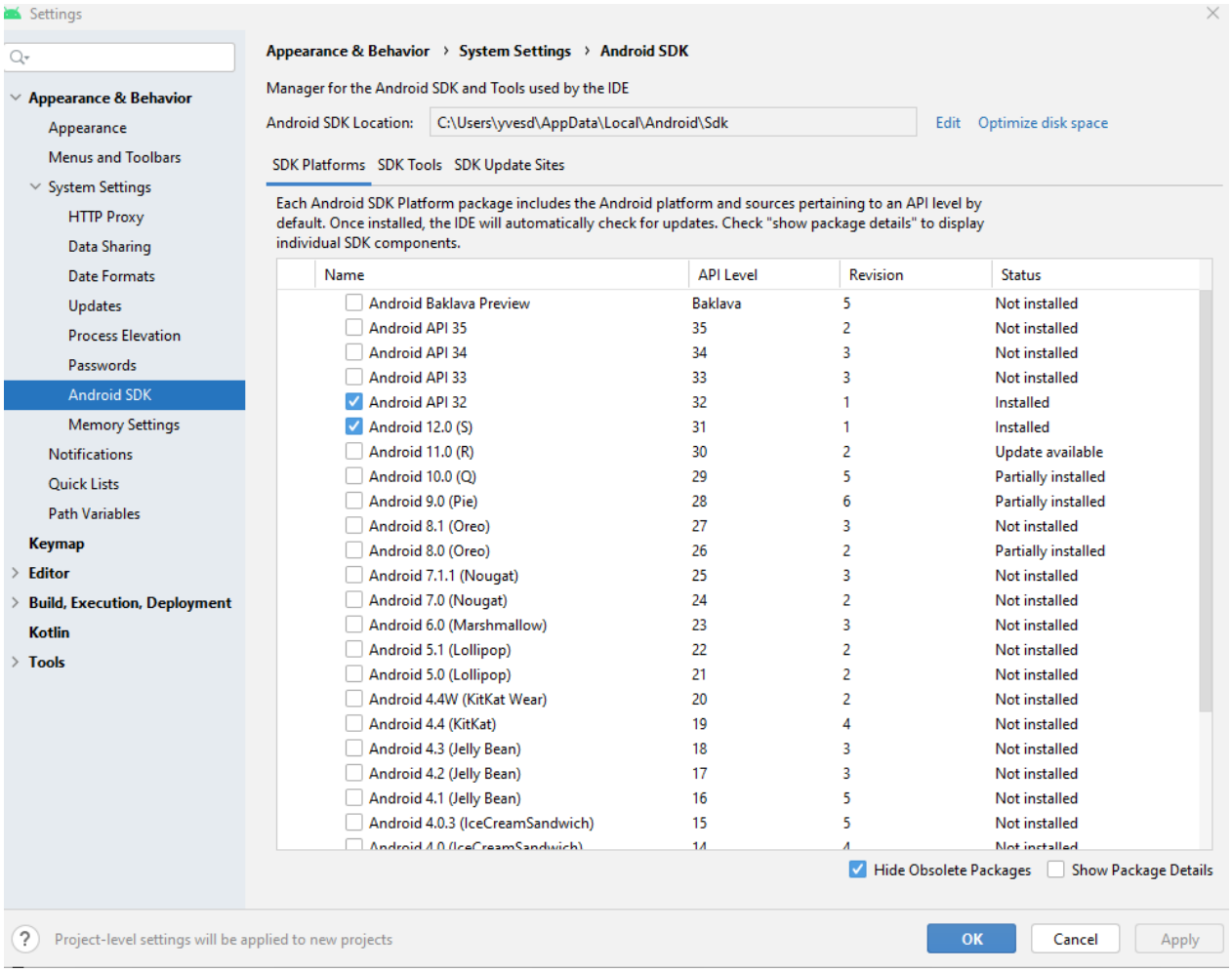
I use cmake and NDK from Android Studio as a build context on Windows. I guess you can use your own cmake install if you have one and directly download the Android NDK.

In my case I use the Android Studio to install everything. It was quite straightforward. It should not be too difficult to install the components directly if you don't want to install Android Studio. In this documentation, I'll show you how to do it using Android Studio.

1) Install Android Studio
2) In Android Studio go to Tools -> SDK Manager and select SDK Tools
3) Verify that these tools are installed (not all are required, but this is the configuration I used):



In the SDK Platform check that Android 31 (Android 12.0) is installed (that is what I used to build the library, but I guess other versions should work too):




4) Note the path of the Android SDK Location (displayed at the top of the window above). You'll need it later.
5) Press ok to install all the components
6) Add the "cmake" tool to your PATH environment variable. Use the Android SDK Location to find the path. For instance, in my case: C:\Users\USERNAME\AppData\Local\Android\Sdk\cmake\3.31.6\bin\

## 2. Ninja

a. Install Ninja from here: https://github.com/ninja-build/ninja/releases
b. Add the path where ninja.exe is installed to your PATH environment variable.

## 3. Prepare Mujoco

a. Clone the Mujoco repository
b. Go to the Mujoco folder and create a build folder

## 4. Fix the Mujoco Source Code for Android

Open and modify the “engine_util_errmem.c” file in mujoco\src\engine this way:

a. First, go to the mju_writeLog function and add a preprocessor test for ANDROID (where__APPLE__ is also tested):

#if defined(_POSIX_C_SOURCE) || defined(__APPLE__) || defined(__STDC_VERSION_TIME_H__) **|| defined(__ANDROID__)**

b. Then go to the mju_error_raw function and replace add “(void)” in front of the getchar() function, so it looks like:

(void)getchar();

**Quick explanation:**

- The __ANDROID__ preprocessor should be correctly handled for localtime
- The call to getchar() is flagged because its return value is marked with the attribute warn_unused_result, which means the compiler expects you to check or explicitly ignore its result. Since warnings are treated as errors (-Werror), the build fails if you don’t do this change.

# 5. Build Mujoco

1) Open a command line
2) Go to your build folder in Mujoco
3) Check your Android SDK path, and check the NDK version that is installed.
4) Configure cmake using ninja with this command line (update the NDK path to your Android SDK path and NDK version):

```
cmake -G "Ninja" ^
-DCMAKE_TOOLCHAIN_FILE=C:\Users\yvesd\AppData\Local\Android\Sdk\ndk\29.0.13113456\build\cmake\android.toolchain.cmake ^
-DANDROID_ABI=arm64-v8a ^
-DANDROID_PLATFORM=android-31 ^
-DMUJOCO_BUILD_SIMULATE=OFF ^
-DMUJOCO_BUILD_EXAMPLES=OFF ^
-DMUJOCO_BUILD_TESTS=OFF ^
-DMUJOCO_TEST_PYTHON_UTIL=OFF ..
```

5) You may have some warnings, but you should not have any errors.
6) Once the configure is done, simply type “ninja” in your command line to build the libraries.
7) If everything worked as expected, you should have your files generated in the build/lib folder