Compile Instructions

# MuseScore

The compatible MuseScore source code is included in the repository, it’s under the folder “MuseScore”. It’s mandatory to create a custom Android ndk toolchain.

To compile this libraries are needed:

* libVorbis
* libVorbisFile
* libOgg
* libsndfile
* LAME

This libraries should be compiled under a Linux distribution (wsl also works) and as static libraries (.a) with the exception of lame (mp3) this one needs to be a shared library (so). Once compiled the header files need to be installed into the ndk, under “sysroot/usr/include” then copy the libraries themselves into the project root (the folder created from git) on the folder Libraries/android.

After those libraries have been compiled we can proceed with the MuseScore compile, this also needs to happen in a Linux environment.

Let’s start off with the compile instructions on [its site](https://musescore.org/en/handbook/developers-handbook/compilation/compile-instructions-ubuntu-1410-and-1604lts-git).

Then we need to setup the system environment so cmake crossCompiles:



Once set, we execute this line “cmake . –DisAndroid=1 -DCMAKE\_SYSROOT=%android\_toolchain%/sysroot”

WARNING: You may need to copy the pkgconfig files into your PKG\_CONFIG\_PATH (set earlier).

After it compiles we gather all the static libraries that are produced and we copy them into %project\_root%/Libs/android

# MuseWarp

This Library compiles like every other qmake project, using Qt creator.

So we open MuseWrap.pro on QtCreator, setup an Android kit and hit compile. Once compiled copy the resulting static library (.a) into Libs/android

Done!

# MusikViewer

No hustle here, the same as before. Open MusikViewer.pro set it up and hit run or Build. At this point an Apk is produced.