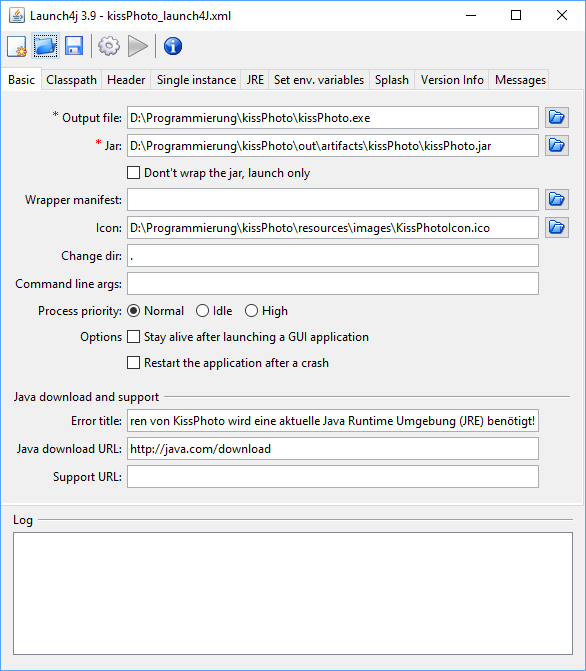
# KissPhoto-Project Settings Doku

2016-11-05

## Git

C:\Users\ikreuz>git config --global http.proxy <http://proxy-sifi.rd.corpintra.net:3128>

## Launch4J

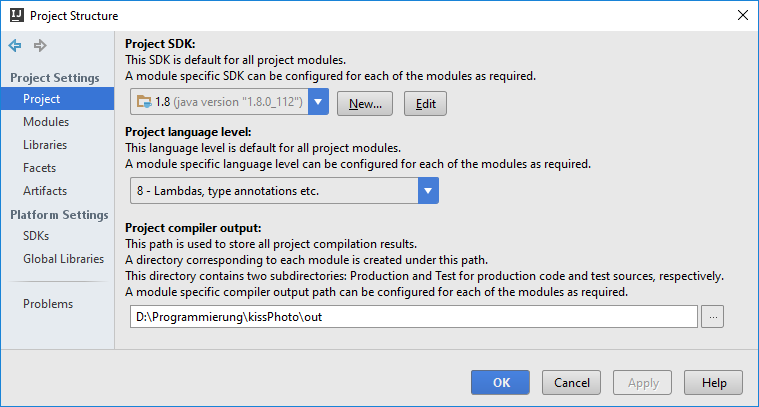


Alle anderen Reiter sind leer. Auch der Splash-Screen wird nicht hierüber eingestellt, sondern stammt aus den Javafx-Einstellungen in IntelliJ.

Achtung: Vor der Anwendung von Launch4J nochmal überprüfen, ob ins Jar ausverwehen eine alte .exe und eine .settings-Datei mit eingepackt wurde: mit 7-Zip öffnen und ggf. rauslöschen! Oder vor dem Build ein Clean durchführen.

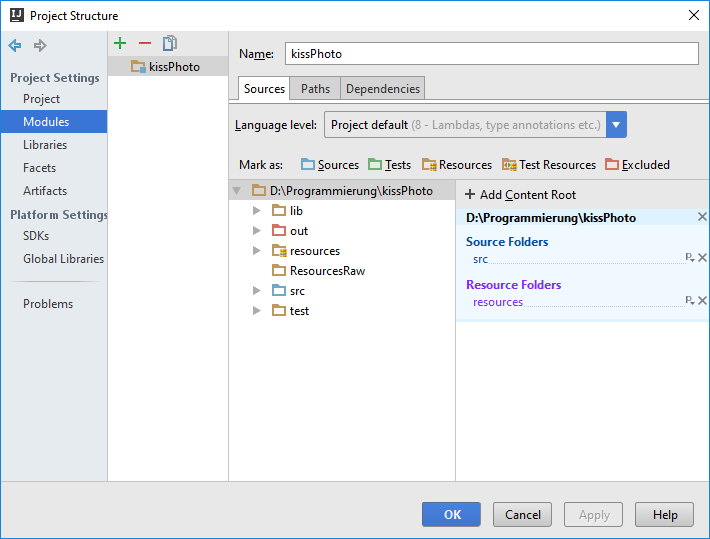
# IntelliJ

Einstellungen unter Files-Project Structure



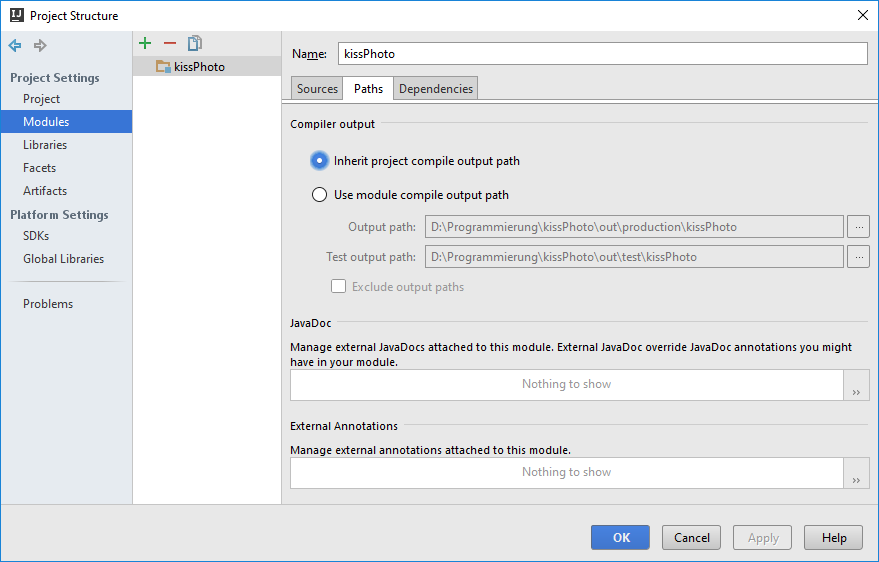
Hier muss das richtige SDK eingestellt werden, sobald es ein neues gibt. Language Level nur bei Bedarf anpassen.

Die alten SDK-Versionen habe ich in der Regel immer gleich gelöscht

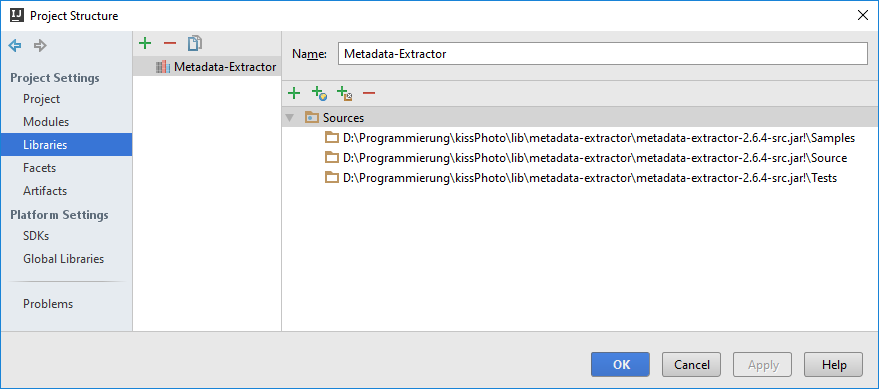


Der Content-Root muss auf KissPhoto stehen, damit z.B. auch die Resourcen auftachen.

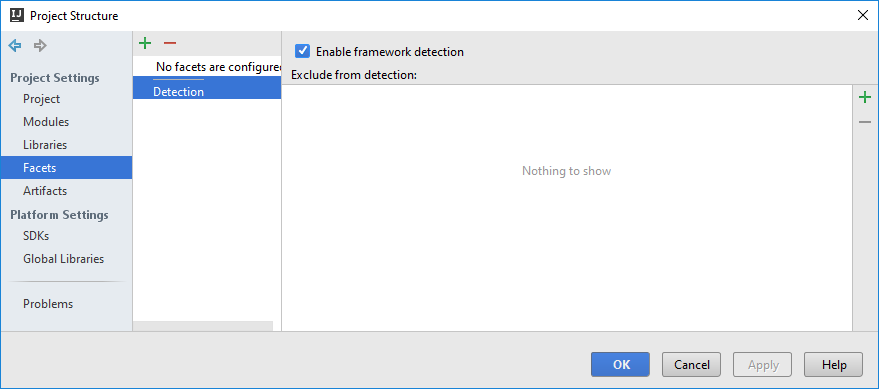
SRC muss als Sources markiert werden und ressources als Ressourcen. Nur dann werden sie auch in die JAR-Datei übernommen. Die anderen Markierungen sind egal.



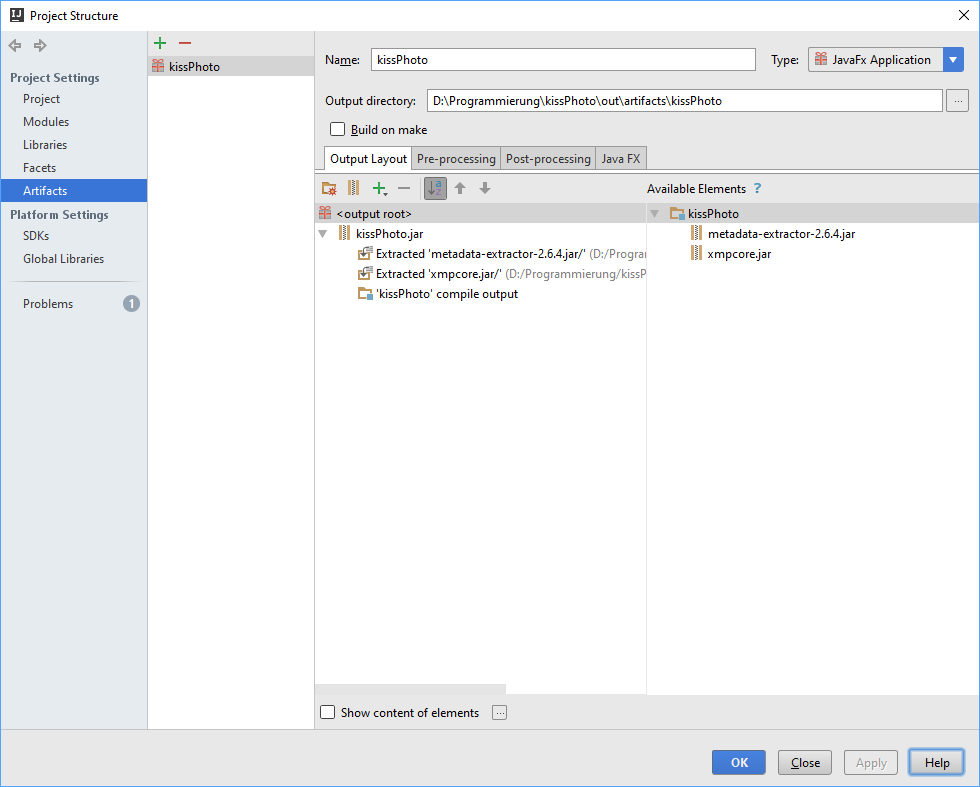
Hier Inherit wählen.



Alle externen Libraries hier einbinden



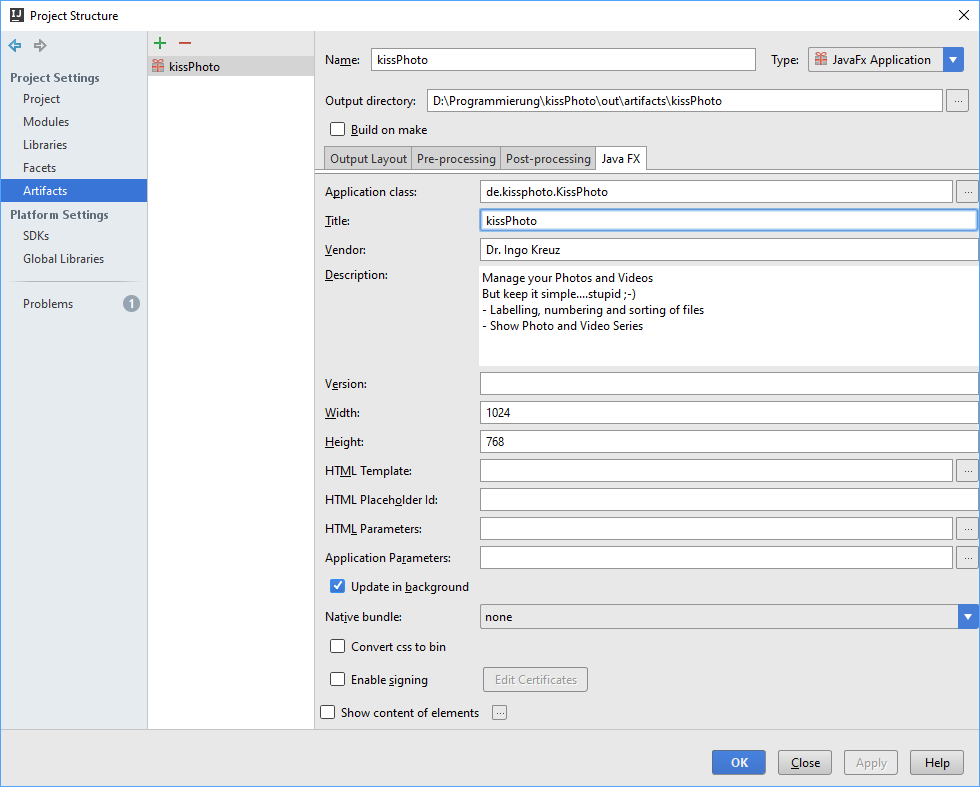
Hier habe ich nichts eingestellt. Wird wohl für Android-Apps gebraucht…



Die Artifacts-Seite beschreibt, was ins JAR-File gelinkt wird. Im Compile-Output sind bereits auch die Resourcen mit enthalten.

Die Libraries müssen per Rechtsklick (von der rechten Spalte aus) in das Jar „extrahiert“ werden. (alternativ, kann man die Jars auch später über 7Zip einfach dazufügen)…wird kleiner…

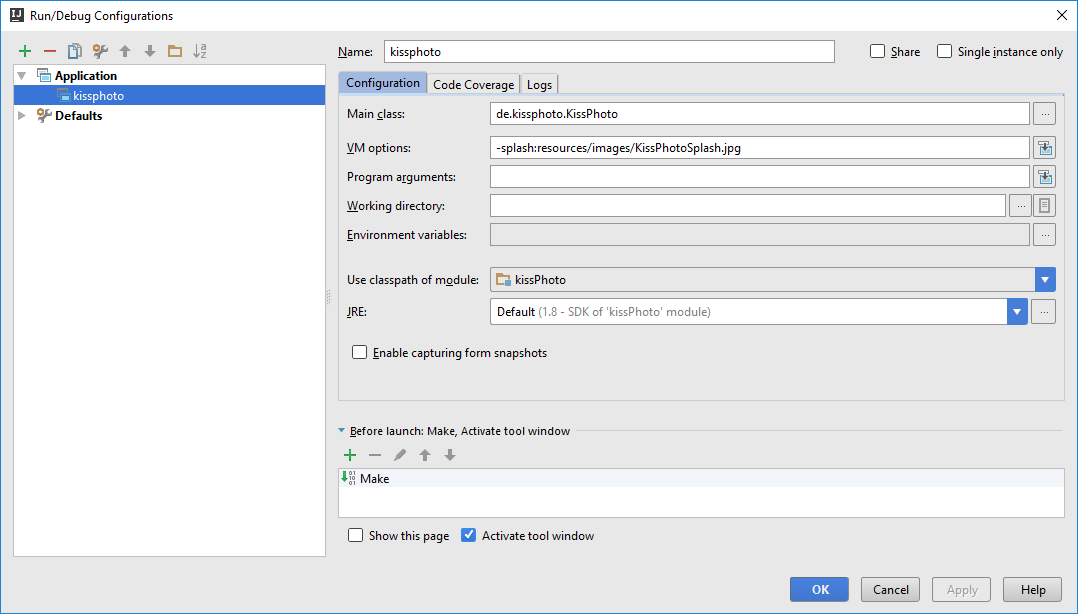
Bei Pre-processing und Post-Processing habe ich nichts eingestellt.



Hier sind die JavaFX-Einstellungen

## Run Settings

Menü: Run-Edit Configuration



Hier habe ich den Splashscreen eingebunden