

QGIS-plugins for ProMaDes

How to?

Prof. Dr.- Ing. Daniel
Bachmann

Magdeburg, May 2020

Content

Content	i
List of figures	ii
1 Introduction	3
2 The QGIS-plugins	5
2.1 Objective of the plugins	5
2.2 Tools.....	5
2.3 Advices and Tips	7
3 Summary	10
Literature	11

List of figures

Figure 1-1: Set-up of PROMAIDES	4
Figure 2-1: Dikeline export via QGIS-plugin	5
Figure 2-2: Path to plugin folder of QGIS3	6
Figure 3-1: <i>Hello World</i> plugin.....	8
Figure 3-2: <i>Dikeline Export</i> plugin	8

1 Introduction

The modular designed decision support system PROMALDES (see Figure 2), which is being developed at the Institute of Hydraulic Engineering and Water Resources Management (RWTH Aachen University) and the research group *Flood Risk Management* of the University of applied sciences Magdeburg-Stendal, is a tool for computer-based support in the selection of flood protection measures. The effectiveness of a protective measure is evaluated using risk-based criteria. Additionally, cost criteria (COST module) which evaluate the cost directly caused by the implementation of such a measure, are taken into account. The integral risk is quantified mathematically using the general risk definition:

$$R = \int K(x) \cdot f(x) dx. \quad (1.1)$$

In Equation (1.1), R defines the integral risk, $f(x)$ the probability density function of the random variable X and $K(x)$ defines the consequences resulting from the realisation of random variable x (e.g. a flood event). The model based flood risk analysis comprises three basic analyses (see Figure 1-1):

- Reliability analysis (FPL module): The probability of the failure of flood defence structures, such as dikes or flood walls, is quantified.
- Hydrodynamic analysis (HYD module): The flood event is transformed into hydraulic variables, such as water levels or flow velocities, taking into account the morphological characteristics of the river and the hinterland.
- Consequence analysis (DAM module): The hydraulic variables of a flood event across areas of specific land use are converted into consequences for the people, assets and goods located in these affected areas of the hinterland.

The task of the risk analysis (RISK module) is to combine the results of the named basic analyses into an integrated flood risk (see Figure 1-1) for the analysed system.

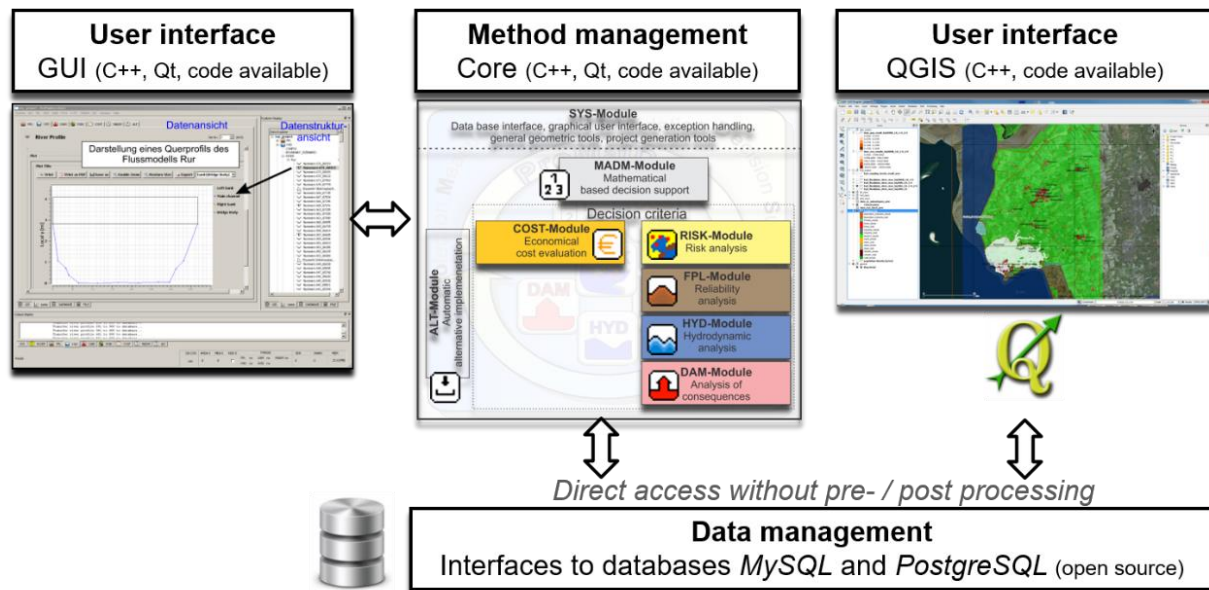


Figure 1-1: Set-up of PROMAIDES

In order to effectively support the design and selection of flood protection measures, the PROMAIDES software package is supplemented with a graphical user interface and a data base interface. The data management system is based on the open source POSTGRES SQL data base, which also provides a close connection to the open source GIS-System QGIS. Thus, input data and result data visualisation and representation as well as a further analysis of the data is directly possible in QGIS.

Detailed information about the PROMAIDES decision support system and the theoretical fundamentals of procedure implementation are provided by Bachmann (2020).

2 The QGIS-plugins

2.1 Objective of the plugins

A model-based risk analysis requires a model set-up for the different base analyses (see chapter 1). For example, to perform a hydrodynamic analysis a hydrodynamic model for the area under investigation has to be set-up in advance. The same applies for the other mentioned analyses.

Setting up models can be a quite time-consuming work. In case of flood risk analysis (and their base analyses) GIS-systems are the logical choice, because there is always a strong relation to spatial data as digital elevation models (DEM), land use models etc.

To facilitate the model set-up of PROMAIDES a collection of QGIS-plugins has been developed. For example a 1d-dikeline-file (see Figure 2-1), required for a hydrodynamic analysis, can be easily extracted by the plugin *Dikeline Export*, a given Line-shapefile and a DEM. For the development of the plugins PYTHON in combination with the QTDESIGNER (for the GUI elements) is used.

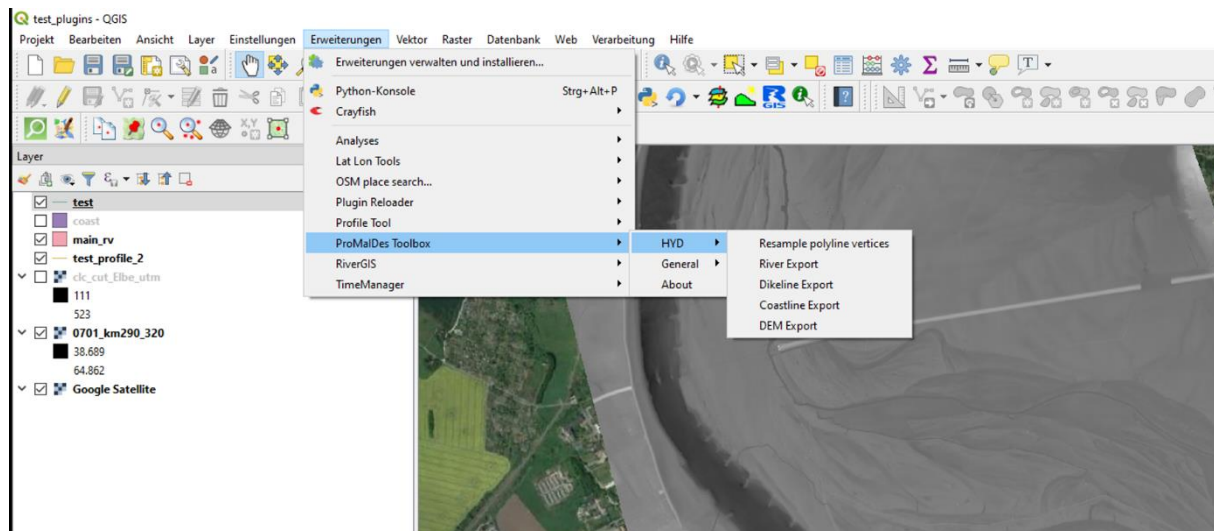


Figure 2-1: Dikeline export via QGIS-plugin

2.2 Tools

QGIS

Please use QGIS-version 3.x! (<https://www.qgis.org/de/site/forusers/download.html>); you should be familiar with QGIS!

QTDesigner

For the development of element the QTDesigner is very helpful (<https://build-system.fman.io/qt-designer-download>). First information how to use it, you will find also under the given link. With the QTDesigner you can open and edit ui-files; you will find them in the provided *promaides_gis_tools* folder. A manual editing of the ui-files is in general not required. All is done via the QTDesigner.

QGIS-plugins for ProMalDes

In the folder *promaides_gis_tools* the plugins for PROMAIDES are stored. You have to copy this folder in your plugin folder of your QGIS3 installation. The path is quite difficult to find. In my case the path is shown in Figure 2-2. Be aware, the folder *AppData* is a hidden folder, you have to activate “Show hidden folders”.

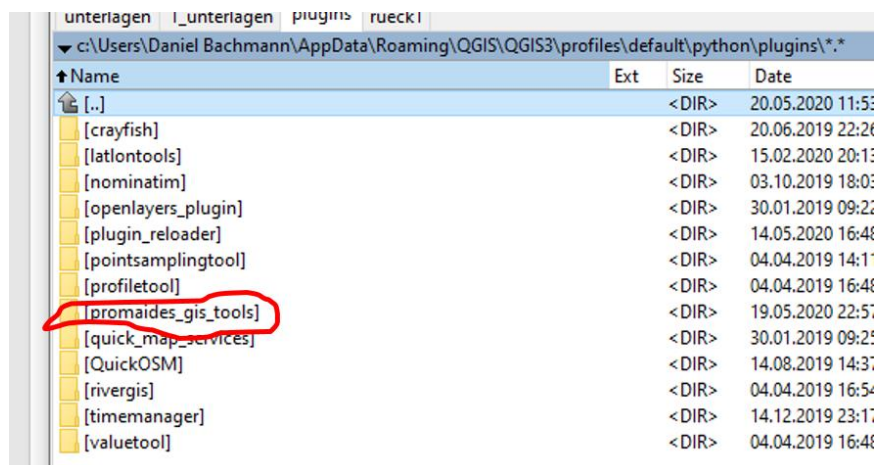


Figure 2-2: Path to plugin folder of QGIS3

After a restart of QGIS you should find the existing QGIS-plugins for ProMalDes under *Extensions/ProMalDes toolbox*.

It is important that you take these files in the folder ProMalDes-plugin folder as example; A general advice: just make a copy of a given file, rename it and change the code.

Plugin development in QGIS

For QGIS I suggest to install following additional plugins (via Extensions/install extensions):

- *QuickMapServices*: it is not required for plugin development, but you get easily a background map (e.g. googleSatellite, googleMaps etc.)
- *Plugin Reloader*: It is very helpful, because you can easily reload your plugin if you change the code

- *First Aid*: I never used it, but it sounds very interesting to facilitate QGIS-plugin development (<https://www.lutraconsulting.co.uk/blog/2016/06/12/introducing-first-aid-plugin/>)

Opening the *python-console* in QGIS (Extensions) can be also helpful, because a output via a simple print-statement in the python-code is shown here, if the plugin is launched.

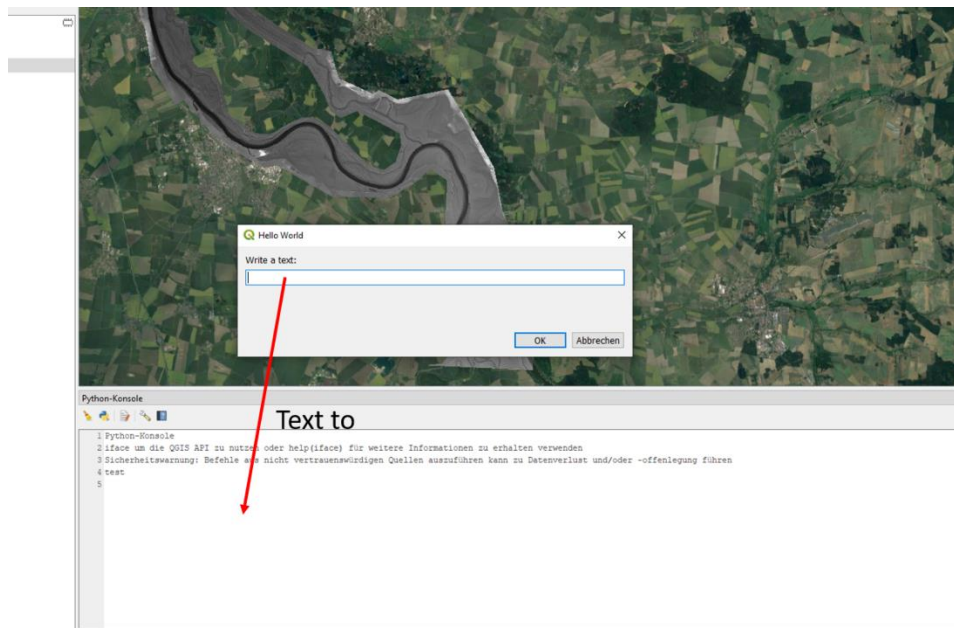
For the python development I used the *PyCharm-IDE* from JetBrains. The Community-version is freely available (<https://www.jetbrains.com/pycharm/download/#section=windows>). In the following link (<https://silverspringenergy.com/using-pycharm-as-an-ide-for-qgis-3-plugin-development-2/>) you will find a good description how to set-up a good working environment in PyCharm for QGIS-plugin development; it is not required (I have never tried it yet), but I think it is helpful. However, if you favour another Python-IDE, e.g. Spyder, just use it.

2.3 Advices and Tips

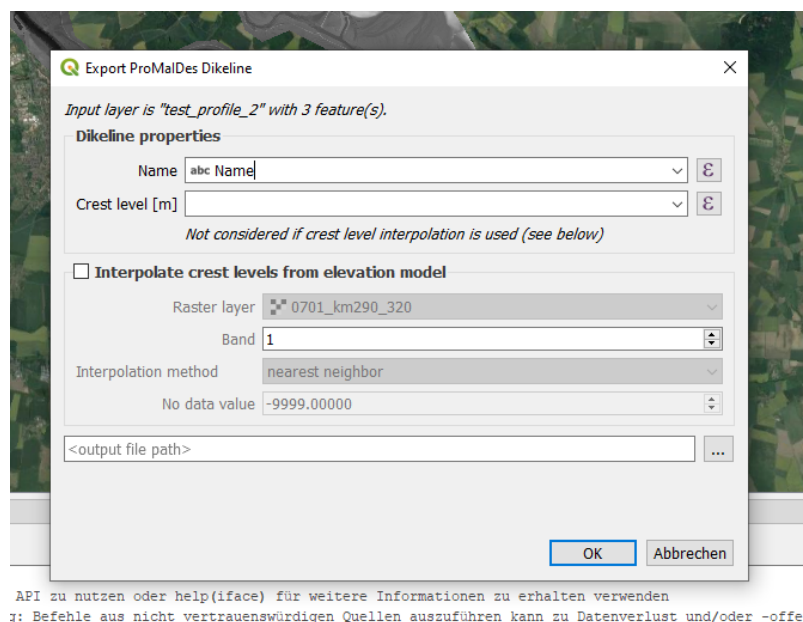
In the following I just summarizes some advices and tips for you:

In the first stage of the internship, get familiar with the information which I provided:

- Make a time schedule (e.g. in Excel) and share it with me; we can discuss it
- Analyse the existing QGIS-plugins for PROMALDES; the Python code, the ui-files etc. These are very helpful examples; you do not have to write all new; use the examples!
- I wrote a simple plugin *Hello world*. You will find it under the *ProMaIDes toolbox/general*. Just write a text in the text label; this text will be printed to the Python console of QGIS (see Figure 2-3). Run this plugin and analyse the code of this plugin. Find out how the files are connected with each other in the plugin folder.

Figure 2-3: *Hello World* plugin

- Test the *HYD/Dikeline Export* plugin. Take the DEM, which is provided, for testing. You have also to create a Line-shape file with a mandatory text-field “Name”. Just create 2 lines with some vertices for testing; the lines should be completely on the given DEM. Mark this line layer and start the plugin (see Figure 2-4); the rest should be self-explaining. Try to analyse also the available code!

Figure 2-4: *Dikeline Export* plugin

In the second stage try to make some small changes in the code (Python-IDE) and in the ui-files (QTDesigner). Maybe start with the *Hello world*-plugin. Just to see how it works. Keep always a safe copy of the original files. Use the *Plugin Reloader*, after code-changes!

In the third stage try to start with the first plugin. It is not a very complex task, but very good task to get familiar with the plugin development. A lot of examples are given already; not all is always required but parts of the existing codes are helpful. Use also copy-paste! For example, after analysis of the *Hello World*- and the *Dikeline Export* plugin, try to find out: which parts do I need from these plugins for my new plugin? Try things out, make errors, solve the errors; this is normal, but in my experience very good for your learning process!

During development process, please use comments in the code. I have to admit, the guy, who wrote the existing plugins, didn't used a lot of comments. But normally a good code consists of 1/3 code and 2/3 comments!

Be also creative, think what is good for the user: for example, in the existing plugins no comments are written to the output file; comments in PROMALDES-files are marked also with "#". But maybe it would be nice for the user to have a generation date, the name of the shape-file etc. in the comments.

3 Summary

This document provides the first information for writing QGIS-plugins for PROMAIDES. Further information please contact me via daniel.bachmann@h2.de.

I hope you enjoy writing QGIS-plugins for PROMAIDES.

Regards,

Daniel Bachmann

Literature

Bachmann, D. (2020): ResearchGate project: PROMAIDES.

<https://www.researchgate.net/project/ProMaIDes-Protection-Measure-against-Inundation-Decision-Support>; [Last access: 19.05.2020].