

TECHNISCHE HOCHSCHULE ULM -  
SOFTWARE PROJECT

# SCRUMMY 1.0

Detailed Guide

HOW TO RUN THE APPLICATION  
ON YOUR WINDOWS PC

## **Disclaimer**

This software was created by a group of 8 people, with intentions of only achieving a high grade at a university course. There is no purpose of our team to sell this software

Most of the content in this document is created from scratch, though the things for example, pictures or information that were taken from various websites are referenced at the last page of this document, to respect the copyrights policies.

If you have any questions regarding the build process, please reach me out on my email:

[aqeemu01@thu.de](mailto:aqeemu01@thu.de)

Department of Computer Science Technische  
Hochschule Ulm  
Germany

## **Steps to run the code on your PC:**

**Required Steps for having a successful build:**

1. QT 6.6.0
2. MySQL – Drivers
3. Python Drivers
4. Steps for creating the build

# STEP 1:

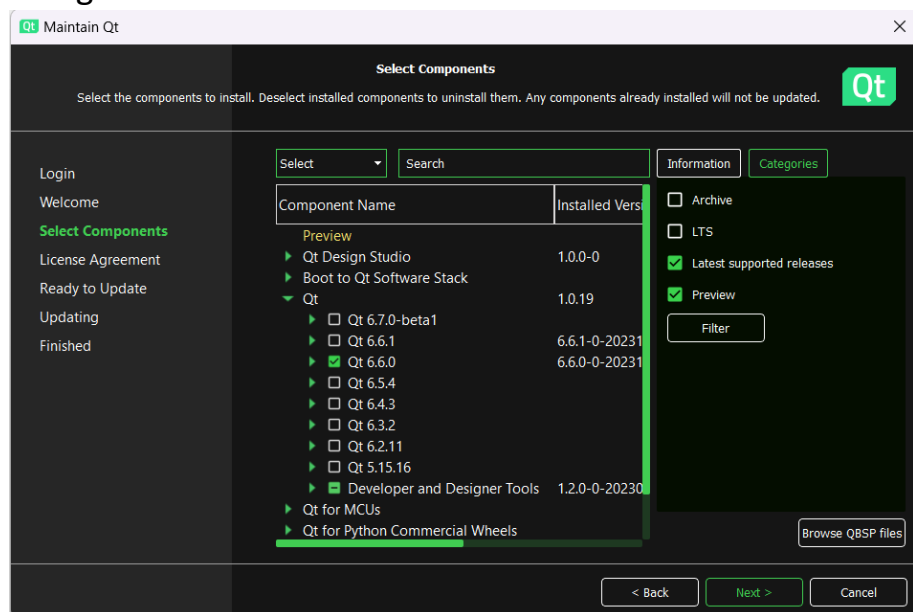
## QT 6.6.0

- a. First you need to create an account to get a license from the QT company providers, if your university is recognized on there system, then for students you can access the product for free for a limited time. Link to their registration forum:

[Try Qt | Develop Applications and Embedded Systems | Qt](#)

- b. After creating an account, you will can either download their Maintenance tool or from this link download this specific version of QT which is version 6.6.0, as our MSQl-Drivers provided on Git is version specific

- i. Using the Maintenance tool



- ii. Using the old achieve files from QT Official Website:

[Index of /archive/qt/6.6/6.6.0/single](#)

- c. Make sure to note down your installation path that you decided on your PC, as later we need to add a few files in some folders

## STEP 2:

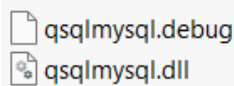
### MY-SQL Drivers

1. Download this folder from our Git Repo:

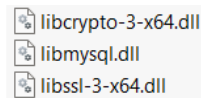
The compressed file known as

“qsqlmysql.dll\_Qt\_SQL\_driver\_6.6.0\_MinGW\_11.2.0\_64-bit\_SE.zip”

in the same place where this document is saved has the files that you need to extract and you will have these files:



qsqlmysql.debug  
qsqlmysql.dll












libcrypto-3-x64.dll  
libmysql.dll  
libssl-3-x64.dll

2. Copy these files in the following folders separately in your QT installation directory
  - i. QT\_InstallationFiles\6.6.0\mingw\_64\lib
  - ii. QT\_InstallationFiles\6.6.0\mingw\_64\bin
  - iii. QT\_InstallationFiles\6.6.0\mingw\_64
  - iv. QT\_InstallationFiles\6.6.0\mingw\_64\plugins
  - v. QT\_InstallationFiles\6.6.0\mingw\_64\plugins\sqldrivers

For example: This is how you need to copy all those files to all of the required folders indicated in the 2nd point above

### QT\_InstallationFiles\6.6.0\mingw\_64\plugins\sqldrivers

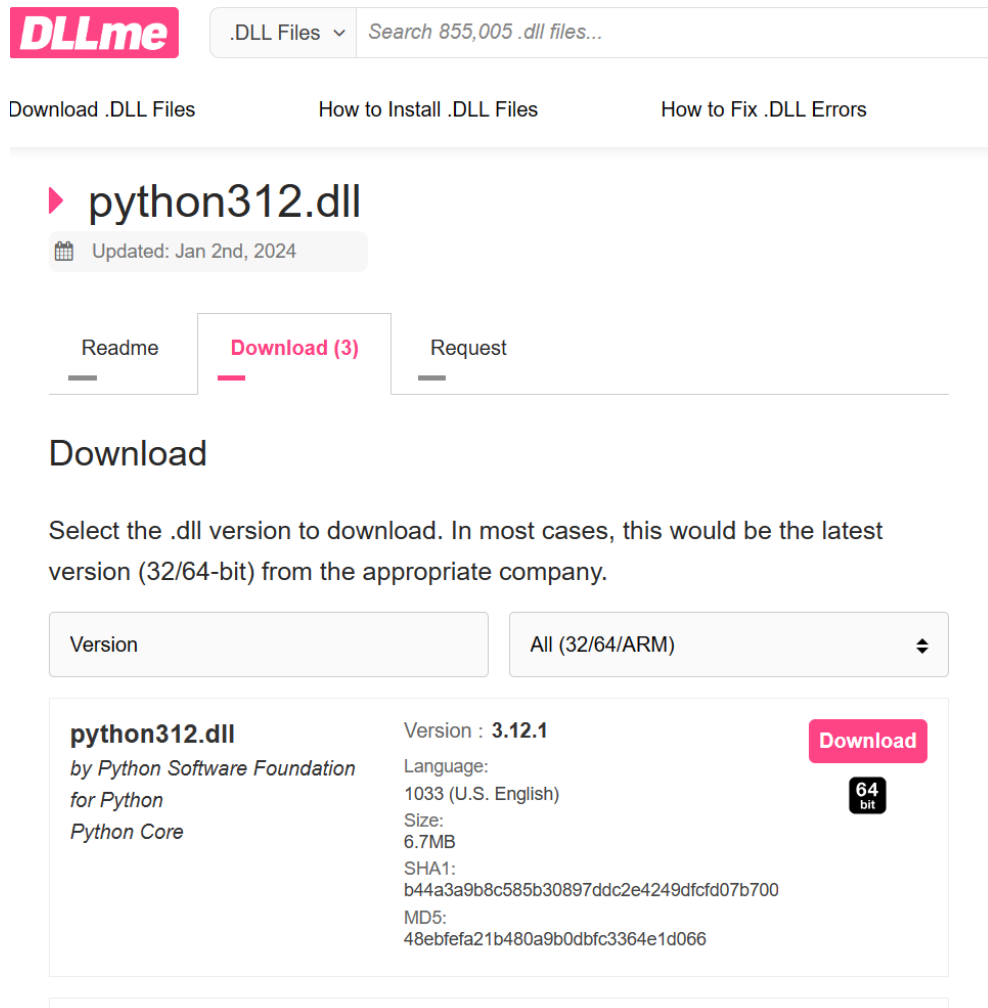
Name	Date modified	Type	Size
 libmysql.dll	12/10/2023 15:00	Application exten...	6.816 KB
 libmysql.lib	12/10/2023 15:00	Object File Library	31 KB
 mysql.h	12/10/2023 12:45	H File	33 KB
 qsqlite.dll	30/09/2023 16:11	Application exten...	1.343 KB
 qsqlmimer.dll	30/09/2023 16:11	Application exten...	93 KB
 qsqlmysql.debug	14/11/2023 09:05	DEBUG File	1.764 KB
 qsqlmysql.dll	14/11/2023 09:05	Application exten...	174 KB
 qsqlodbc.dll	30/09/2023 16:11	Application exten...	129 KB
 qsqlpsql.dll	30/09/2023 16:11	Application exten...	99 KB

## STEP 3:

### Python Drivers

1. For some user if they have not installed any python based applications on their system, then they just need to one additional step to make sure that the build process executes seamlessly
  - a. From the following website or the missing .dll file is also available on our repo:
    - i. Website link: [python312.dll : Free .DLL Download. - DLLme.com](https://python312.dll : Free .DLL Download. - DLLme.com)
    - ii. The compressed file known as “python312.zip” in the same place where this document is also having the same files

Below you can see from a screenshot where is the file located on the website, please download the 64-Bit Version, if you have a 64-Bit based Operating System



The screenshot shows the DLLme website interface. At the top, there is a search bar with the text ".DLL Files" and a search button labeled "Search 855,005 .dll files...". Below the search bar, there are three navigation links: "Download .DLL Files", "How to Install .DLL Files", and "How to Fix .DLL Errors". The main content area displays the file "python312.dll" with a "Download (3)" button. Below this, there is a "Download" section with a table of available versions. The table has two columns: "Version" and "All (32/64/ARM)". The first row shows the file "python312.dll" by Python Software Foundation for Python Python Core, with version 3.12.1, language 1033 (U.S. English), size 6.7MB, and SHA1 hash b44a3a9b8c585b30897ddc2e4249dfcfd07b700. The MD5 hash is 48ebfefa21b480a9b0dbfc3364e1d066. A "Download" button is visible next to the file details.

Version	All (32/64/ARM)
<b>python312.dll</b> by Python Software Foundation for Python Python Core	Version : 3.12.1 Language: 1033 (U.S. English) Size: 6.7MB SHA1: b44a3a9b8c585b30897ddc2e4249dfcfd07b700 MD5: 48ebfefa21b480a9b0dbfc3364e1d066

## STEP 4:

### Creating the Build

(Additional Steps Regarding Git are also explained here)

1. Once you have cloned the project on your PC, navigate to this folder  
Path:
  - a. softwareproject\code\Staging\_Environment\Recovered\Staging\_Environment\_Recoverd
  - b. Now from QT not manually navigate to the same path indicated in part a. and open this file **"CMakeLists.txt"**, which will load up the project.

## Summary

You might find some alternative ways to build the project on your Windows based PC, these were the steps taken by almost every member in our team to execute the source code on their systems, to preform the further advancements of the project itself.

### Team Members :

Muazzam Bin Aqeel – (**Scrum Master** + Full Stack Developer)

Wesley – (**Product Owner** + Full Stack Developer)

Aida Zhumiyeva – (**Quality Architect** + Front-End Developer)

Cosmin – (**Software Architect** + Full Stack Developer)

Keti Hazbiu – (**Software Architect** + Full Stack Developer)

Ruth Jesulayomi Mautin Amosu – (**Software Architect** + Full Stack Developer)

Ebtehal Aldheeshy – (**Software Architect** + Back-End Developer)

Citlalin Reyes Soto – (**Software Architect** + Back-End Developer)



## References

1. [Try Qt | Develop Applications and Embedded Systems | Qt](#)
2. [python312.dll : Free .DLL Download. - DLLme.com](#)
3. [Welcome to Python.org](#)