

Install Prompt Quill llmware from scratch

You need to set up 4 different parts to run Prompt Quill on your local system.

I will explain how you do this on a windows system, the steps are the same if you do it on linux but the details will be different. I don't have a linux system so I can not provide the proper steps, if you got a linux please help and add the linux version install steps to the repo.

1. Docker Desktop
2. Qdrant as the vector store
3. Mongo DB as the llmware backend (you need a client for mongo too)
4. Python to be able to run Prompt Quill

Docker Desktop

First, we start with setting up the docker to run mongo and qdrant. If you know a way to run qdrant outside of docker please let me know.

To run any container in docker you will need to have a version of docker running on your system.

For windows you can download it here:

<https://www.docker.com/products/docker-desktop/>

You do not need to create any account with docker, they just ask you if you understand the license agreement. Say yes and its all good. After the install you have to run the docker desktop from your startmenu. This will startup the docker daemon which is needed to run the docker compose command.

Qdrant

Once the docker desktop is up and running open a command prompt (cmd) and cd to the folder

```
<Your path>\docker\llmware\llmware_qdrant
```

There you run docker compose up -d

This will install and run the two database systems mongo and qdrant.

If this was a success you will see this in the docker desktop

Containers
[Give feedback](#)

Container CPU usage ⓘ
0.53% / 2000% (20 CPUs available)

Container memory usage ⓘ
797.4MB / 15.16GB

[Show charts](#)

Only show running containers

| <input type="checkbox"/> | Name | Image | Status | CPU (%) | Port(s) | Last started | Actions |
|--------------------------|------|----------------------------|---------------|---------|---------|----------------|-------------------------------------|
| <input type="checkbox"/> | > | limware_qd | Running (2/2) | 0.45% | | 39 minutes ago | <div></div> <div></div> <div></div> |

If you open it with the > button it will look like this

Containers
[Give feedback](#)

Container CPU usage ⓘ
0.45% / 2000% (20 CPUs available)

Container memory usage ⓘ
797.4MB / 15.16GB

[Show charts](#)

Only show running containers

| <input type="checkbox"/> | Name | Image | Status | CPU (%) | Port(s) | Last started | Actions |
|--------------------------|------|--|---------------|---------|---|----------------|-------------------------------------|
| <input type="checkbox"/> | ▼ | limware_qd | Running (2/2) | 0.58% | | 40 minutes ago | <div></div> <div></div> <div></div> |
| <input type="checkbox"/> | | mongodb bbc207a8: mongo:5.0.10 | Running | 0.25% | 27017:27017 | 40 minutes ago | <div></div> <div></div> <div></div> |
| <input type="checkbox"/> | | qdrant 680631c0: qdrant/qdrant:latest | Running | 0.33% | 6333:6333 Show all ports (2) | 40 minutes ago | <div></div> <div></div> <div></div> |

Showing 3 items

With this we are done with the docker things, now we start loading the data into the two databases.

The order does not matter which one first, I start with qdrant here.

To get to the qdrant UI open your browser and enter this URL:

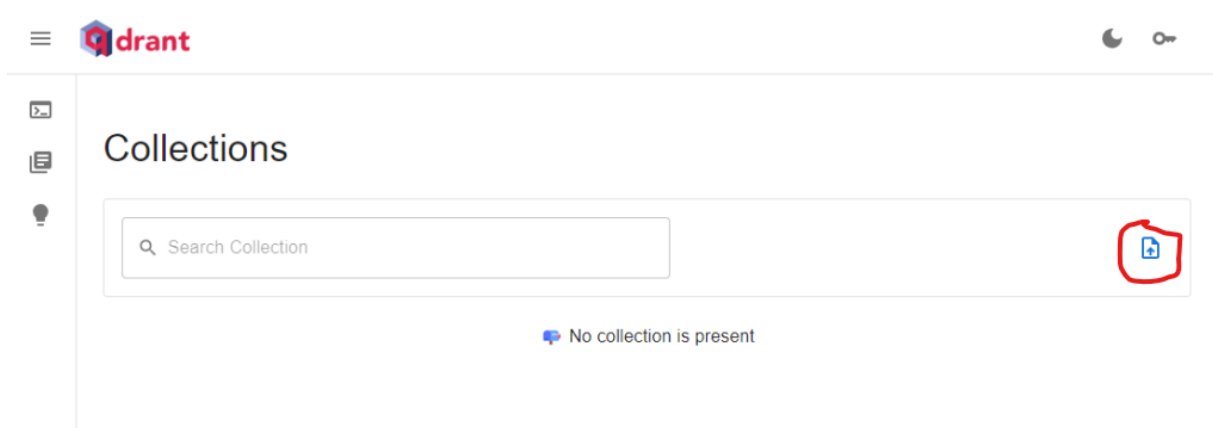
<http://localhost:6333/>

drant

Collections

No collection is present

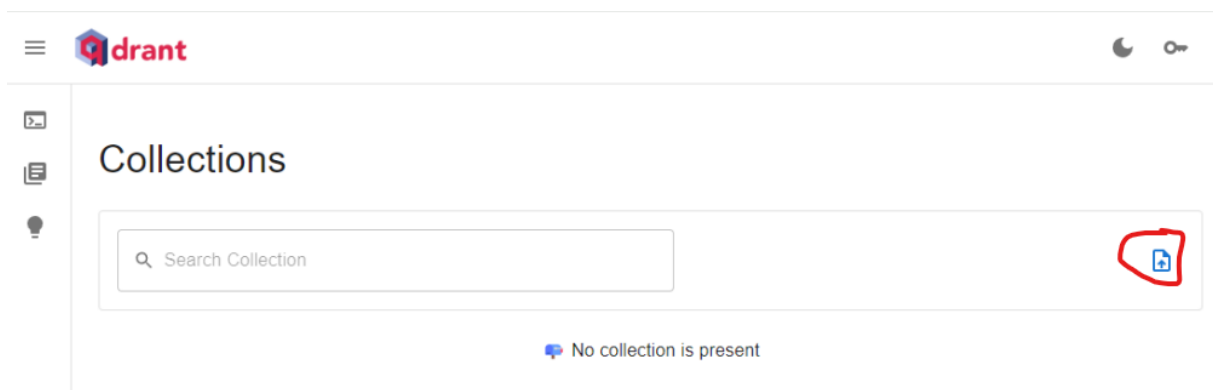
Click the little blue thing on the right



It asks you for a name for the collection to be created.

Enter this name, it has been created by llmware during the embedding so that name is stored within the library information of the llmware mongo data, it important to use that exact name.

llmware_llmwareqdrant_minilmsbert



Upload a Snapshot

1 Step 1 - Enter a collection name

Can be new or existing

Collection Name

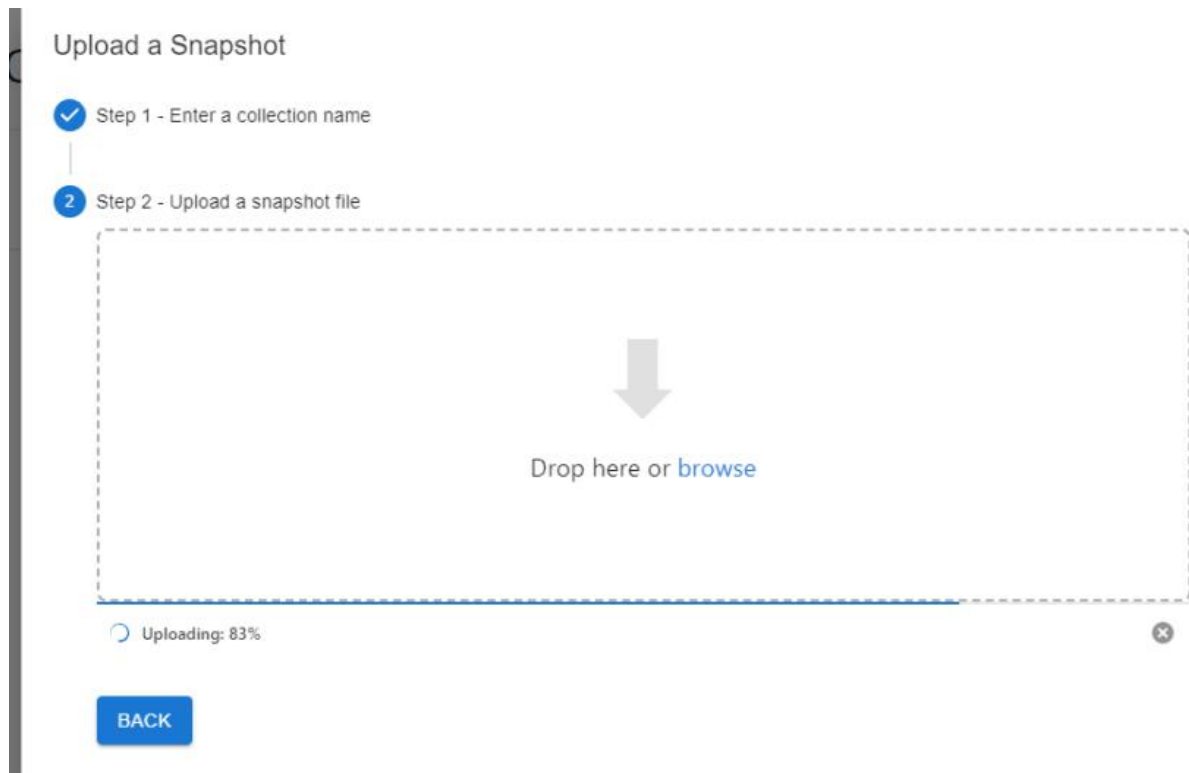
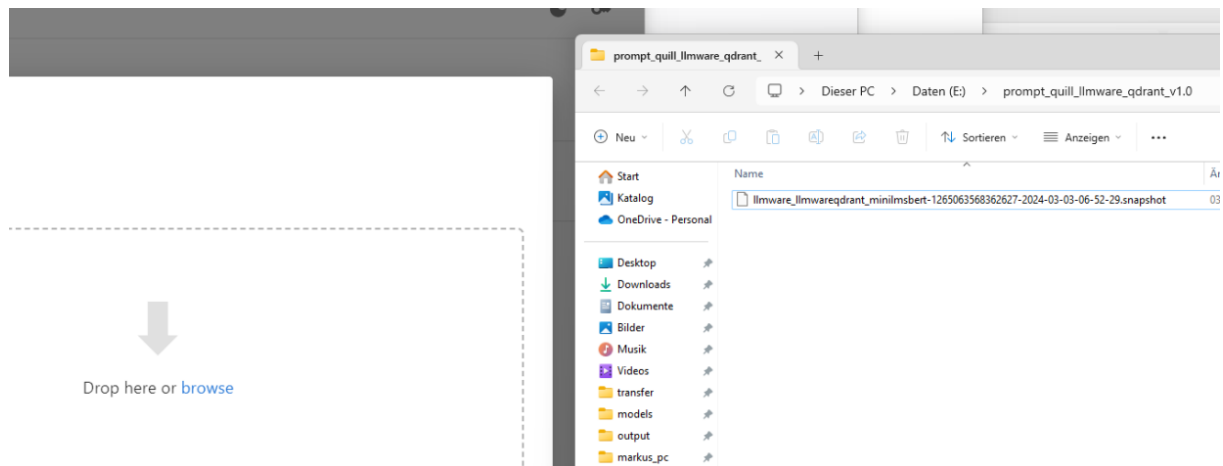
llmware_llmwareqdrant_minilmsbert

CONTINUE

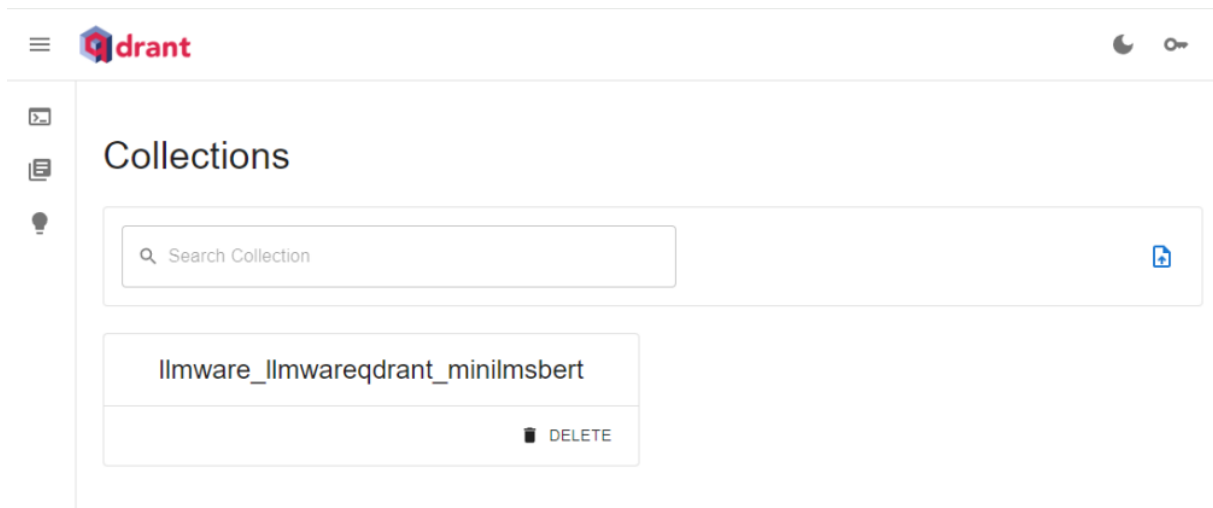
2 Step 2 - Upload a snapshot file

Click on continue

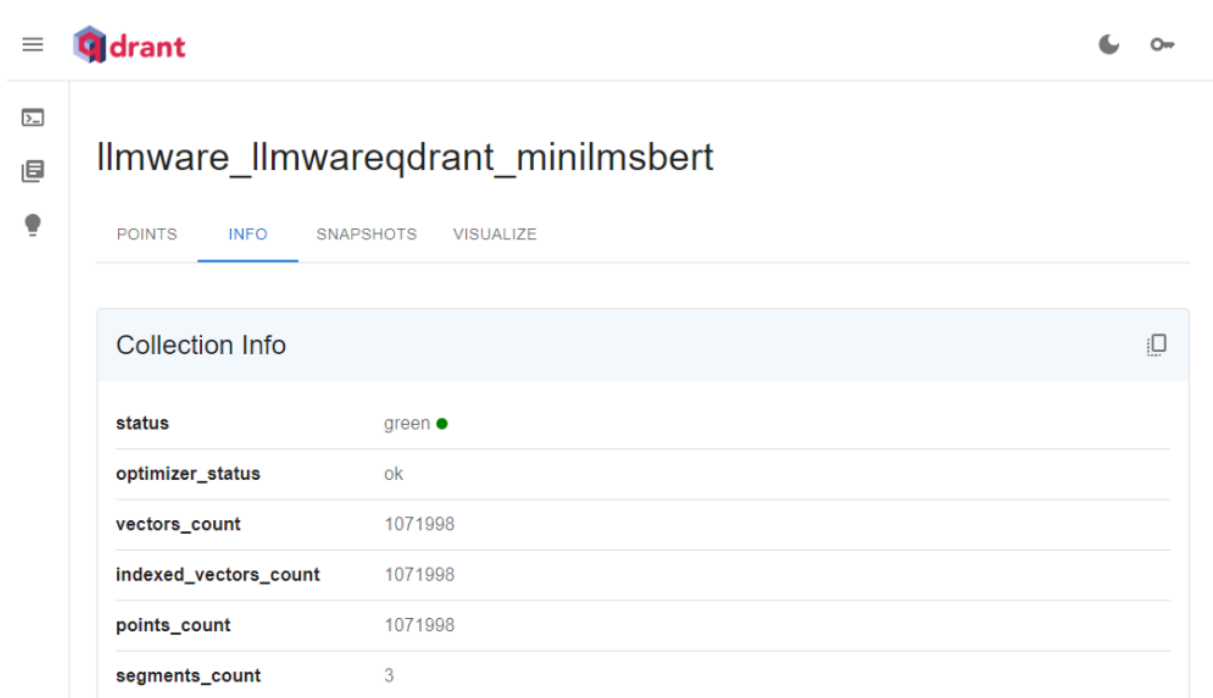
Drag the file from the zip archive to the drop here zone



After the upload it will look like this



You can click the collection and go to the info tab and you should see this



This way you know the data is there.

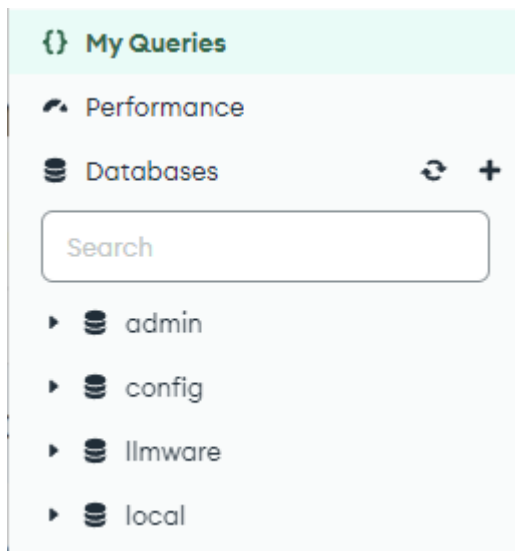
MongoDB

Next step is to load the data into mongo. For this we need a mongo client, you can use any client you like the most. I found MongoDB Compass to work for me.

This you can download here:

<https://www.mongodb.com/try/download/compass>

After the install you must create a database called llmware



To create a database, click the + sign right of databases

×

Create Database

Database Name

llmware

Collection Name

library

☐ **Time-Series**

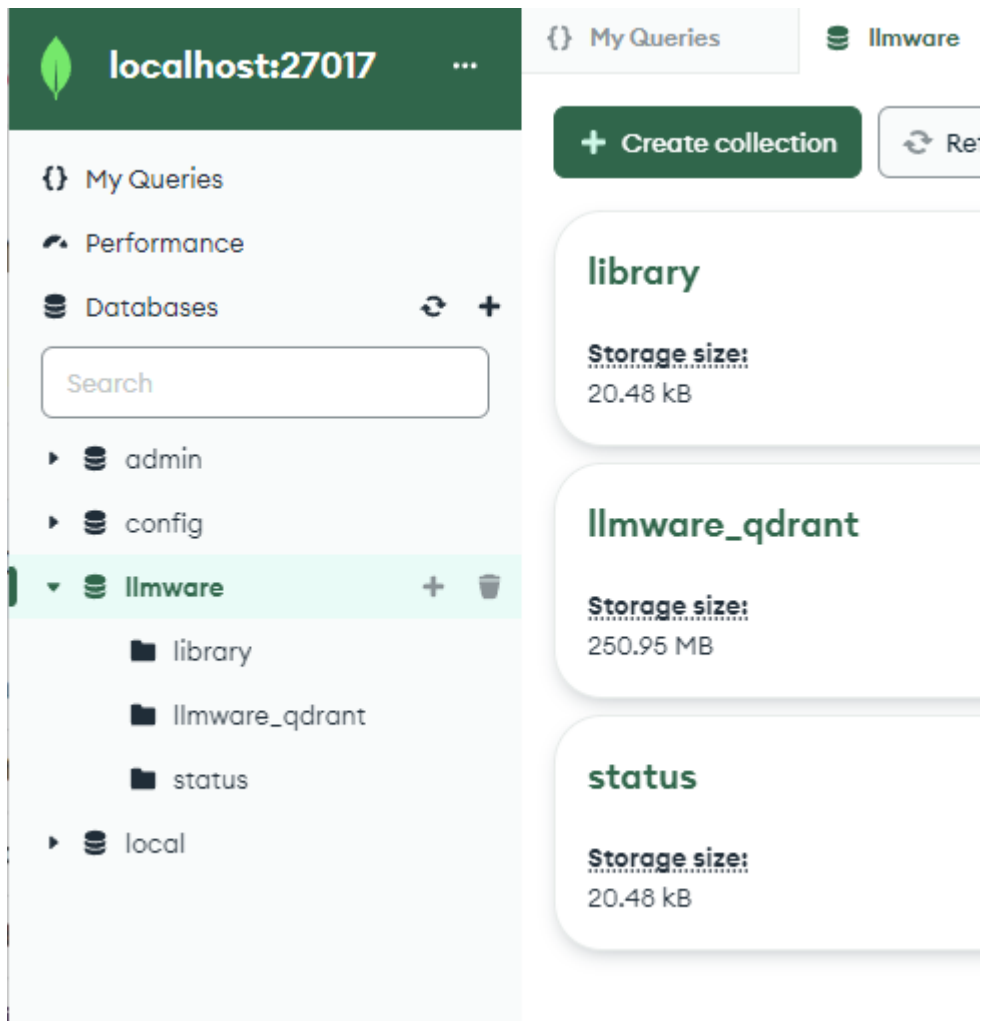
Time-series collections efficiently store sequences of measurements over a period of time. [Learn More](#)

> Additional preferences (e.g. Custom collation, Capped, Clustered collections)

Cancel

Create Database

In there you need to have three collections



You can create them by clicking the create collection button

Once this is done, we can start loading the data.

The data to be loaded you will find inside the mongo_data.zip


Name ^


- llmware.library.json
- llmware.llmware_qdrant.json
- llmware.status.json


To add the data, you click on the library and there the ADD DATA button


llmware.library


[Documents](#) [Aggregations](#) [Schema](#) [Indexes](#) [Validation](#)


Filter 


 ▼

Type a query: { field: 'value' } or [Generate query](#) 

 **ADD DATA** ▼

 **EXPORT DATA** ▼

 **UPDATE**

 **DELETE**

Repeat this for all three collection's and you are all set and ready for the last step.

Setup a python environnement. This can be done in many ways and so I leave that to you to decide. Just have a local python and nothing more or run a conda environment or or or... it is your decision to make :P

Once you got the python up and running just go to the folder with the llmware codes and run
pip install -r requirements.txt

This will install llmware and the qdrant client that is needed.

Once this is done its time to celebrate and to start your new most loved toy Prompt Quill for the first time.

Prompt Quill

python prompt_quill_ui_qdrant.py

Since it is the first time you run it, it will download the needed LLMs and this might take a while.

After that it should look close to this:

```
C:\Users\user\miniconda3\envs\prompt_work\python.exe E:\prompt_work_all\llmware_pq\prompt_magic_ui_qdrant.py
ggml_init_cublas: GGML_CUDA_FORCE_MMQ: no
ggml_init_cublas: CUDA_USE_TENSOR_CORES: yes
ggml_init_cublas: found 1 CUDA devices:
  Device 0: NVIDIA GeForce RTX 3090, compute capability 8.6, VMEM: yes
Using cache from 'E:\prompt_work_all\llmware_pq\gradio_cached_examples\19' directory. If method or examples have changed since last caching, delete this folder to clear cache.

Running on local URL: http://127.0.0.1:7860


To create a public link, set 'share=True' in 'launch()'.
```

Open your browser got to <http://localhost:7860> and start playing and create fantastic prompts.

Chat

Character

Model Settings



Prompt Quill

Chatbot

Retry

Undo

Clear

Enter your prompt to work with

Submit

Examples

A fishermans lake

night at cyberpunk city

living in a steampunk world

Use via API

Mit Gradio erstellt

I hope it was not too hard, I know it's not a one-click install but that's a thing for a later release.