MESA Designer Installation and Deployment Guide

Distribution

This document is distributed under the CC-BY-4.0 License. You may modify and share it under the restriction that you have to attribute its origin to the iGEM Munich 2025 InkSight Team.

Introduction

This document is a guide for installing and deploying iGEM Munich 2025's MESA Designer Tool in common environments. If you require assistance or a specialized deployment contact us at team@igem-munich.com. This document is **not** a **Usage Guide**. For this purpose please visit our wiki's software page and follow the outlined steps in the Usage section or watch our provided videos.

Additional Information

This section includes information which may be especially useful for beginners or non-technical users.

Opening the Terminal / Command Line

The terminal on macOS / Linux and the Command Line (cmd) on Windows. Is an application which allows you low-level access to your system without having to use a GUI. This installation guide will use this application to greatly speed up the installation process. Opening this application is different on Systems, listed are the most common methods.

Windows:

- Press `Windows Key + R`, type `cmd`, and press Enter
- Or search for "Command Prompt" or "PowerShell" in the Start menu

macOS:

- Press `Command + Space`, type "Terminal", and press Enter
- Or find it in Applications > Utilities > Terminal

Linux:

- Press `Ctrl + Alt + T`
- Or search for "Terminal" in your applications menu

Installation Guides

Various installation guides exist, each with upsides and downsides. We will go through common deployment options in the following sections. If you wish to get started using MESA Designer as quickly as possible and do not care about modifying the source code or programmatic integration, we recommend using a **Pre-Built Docker Image**.

Containerized Deployment

Docker deployment ensures consistency across environments and greatly simplifies installation. This also enables easy integration with existing environments via the use of docker compose scripts. This modular, containerized approach minimizes resource usage - a research group might deploy only the API on their compute cluster while running the web interface on a local workstation.

Pre-Built Docker Images (Recommended for Beginners):

Requirements

- Docker: latest version. You can download and install this according to the official installation instructions.

Steps

1. You can view pre-built docker images on dockerhub and find a preferred version (we recommend the latest).
   Note: For most versions there are different containers available: **all**, **webapp** and **api**. These can be identified by their respective names ending in -all, -webapp and -api respectively. You can download (pull) the latest official version from dockerhub by running this command from a commandline or terminal while docker is running on your system:

```
docker pull aeneastews/mesa-designer:latest
```

2. The image can be started by running the following command from a commandline or terminal while docker is running on your system:

```
docker run -p 8501:8501 -p 8000:8000 aeneastews/mesa-designer:latest
```

3. You can now access the web application at http://localhost:8501/ and the API at http://localhost:8000/

Production Deployment / Custom Build:

For institutional deployment, we provide easily adaptable dockerfiles for orchestrated multi-container deployment with reverse proxies, SSL termination, and automated backups. The software runs efficiently on standard cloud platforms such as AWS EC2, Google Cloud Platform or Hetzner Cloud. The environments are also compatible with standard academic computing environments including everything from simple machines to SLURM clusters.

Requirements

- Docker: latest version. You can download and install this according to the official installation instructions.

Steps

1. Download or clone the Dockerfile from the repository. This can be done by navigating to the Munich 2025 iGEM GitLab repo and downloading the **Dockerfile** from the list of files. You can then adapt the file to your specific needs or use the default settings for an easy deployment. However, in the latter case we recommend using a pre-built image.
2. Move the downloaded **Dockerfile** to a directory of your choice and ensure it is named **"Dockerfile"**
3. Start docker on your system if not already running. You can navigate to the same directory using a commandline or terminal and run the following command to compile the docker image:

```
docker build -t mesa-designer .
```

4. This image can be started by running the following command from a commandline or terminal while docker is running on your system:

```
docker run -p 8501:8501 -p 8000:8000 mesa-designer
```

5. You can now access the web application at http://localhost:8501/ and the API at http://localhost:8000/

Local Installation

For researchers requiring local installation due to data sensitivity or customization needs, we provide the option of local deployment. This approach enables complete access, customizability and data privacy by removing the need for cross-device communication thus enabling increased security and protecting user's data.

Requirements

- Python: version=3.13.x. You can download and install this according to the official installation instructions. It may already be installed, you can check this by running `python --version` or `python3 --version` from a commandline or terminal. If the result is anything other than `Python 3.13.x # Note: x represents a flexible number` then you likely do not have the correct version of Python installed.
- Git: This is recommended for cloning the repository, though not strictly necessary. This will, however, enable easy updates to future versions. You can download it according to the official installation instructions. This may already be installed on your system. You can check this by running `git --version` from a commandline or terminal. If the result is anything other than `git version x.x.x` you will probably have to install it.

Steps

1. Download or clone the repository:
   If you elected not to download git (not recommended), you have to download the zip archive from Munich 2025 iGEM GitLab repo by clicking the Code dropdown and selecting the zip-archive option.
   Assuming you downloaded git (recommended), run these commands from a commandline or terminal:

```
git clone https://gitlab.igem.org/2025/software-tools/munich.git
cd munich
```

2. Running the tool requires a few dependencies. To keep these separated, you should create a python virtual environment.
   This can be done by running this command from a commandline or terminal.
   Note: You should do this inside the munich directory which was downloaded during the previous step:

```
python -m venv .venv
```

3. To install the required packages in the virtual environment, it has to be activated first.

```
# On Windows run this command from the munich directory:
.\.venv\Scripts\activate

# On Mac OS or Linux run this command from the munich directory:
source ./.venv/bin/activate
```

The required packages will be automatically installed and setup via this command on both systems:

```
pip install -r requirements.txt
```

4. The MESA Designer tool is based on validated experimental data from multiple databases. This data can be automatically set up using the included setup.py script. Please run it using:

```
python setup.py
```

5. Since the system comes with two available services, you can start these individually to your liking. Starting the webapp can be achieved by running the following command from a commandline or terminal:

```
# On Windows run this command from the munich directory:
streamlit run .\app\main.py

# On Mac OS or Linux run this command from the munich directory:
streamlit run ./app/main.py
```

6. Starting the API does not require the webapp to be running. However, if you wish to start both at the same time, we recommend doing so in a new commandline or terminal window. From the munich directory run the following command in a new commandline or terminal window:

```
uvicorn api.main:app
```

7. You can now access the web application at http://localhost:8501/ and the API at http://localhost:8000/

Python Package Index (PyPi) Package

For easy integration with custom projects, the python package provides extended features, low-level control and complete documentation. This method is intended to be used by developers seeking to integrate MESA functionality into their project or extend the MESA Designer Toolkit.

Requirements

- Python: version>=3.10. You can download and install this according to the official installation instructions. It may already be installed, you can check this by running `python --version` or `python3 --version` from a commandline or terminal. If the result is anything other than `Python 3.13.x # Note: x represents a flexible number` then you likely do not have the correct version of Python installed.
- biopython: compatible version with your own project. You can check it out on and install it from PyPi.

Steps

1. Having activated your desired installation python environment, run:

```
pip install mesa-designer
```

Next Steps

In order to get started using MESA Designer, you can head to iGEM Munich 2025's Wiki and take a look at the usage guides. We provide videos as well as step-by-step instructions.