

Contents

1. CANStudio Installation Guide	1
1.1. File Contents	1
2. Running Via Docker	2
2.1. Building CANStudio	2
2.2. Checking images	2
2.3. Running CANStudio Image	2
2.4. Accessing the Application (Via a Web-browser)	2
2.5. Checking containers	2
2.6. Stopping containers	2
3. Running via Python3	2
3.1. Virtual Environments	2
3.1.1. Creating Virtual Environments	3
3.1.2. Activating Virtual Enviroments	3
3.1.3. Installing required Python3 dependencies:	3
3.2. Running CANStudio:	3
3.3. Accessing the Application (Via a Web-browser)	3

1. CANStudio Installation Guide

The offline version of CANStudio is powered by docker and the files are provided to have host a localized version. If you are not capable of running docker, you may use the requirements.txt along with python3 (> 3.8.10) and run it locally with flask.

1.1. File Contents

Purchasing a license for CANStudio also comes with the hosting files:

```
dist
|-- app/
|   |-- static/
|   |-- templates/
|-- app.py
|-- Copyright Notice.doc
|-- Copyright Notice.pdf
|-- Disclaimer.docx
|-- Disclaimer.html
|-- Disclaimer.md
|-- Dockerfile
|-- docs.pdf
|-- eula.html
|-- EULA.txt
|-- README.txt
|-- requirements.txt
```

|-- running.pdf

2. Running Via Docker

Note: On linux, these commands may need super user privilege.

2.1. Building CANStudio

```
docker build --tag canstudio .
```

2.2. Checking images

```
docker images
```

2.3. Running CANStudio Image

Running on localhost, port 5000

```
docker run -v /dev:/dev --privileged -d -p 5000:5000 canstudio
```

2.4. Accessing the Application (Via a Web-browser)

```
localhost:5000/canstudio
```

2.5. Checking containers

```
docker ps
```

2.6. Stopping containers

The container id is viewed through the ‘docker ps’

```
docker stop <container-id>
```

3. Running via Python3

3.1. Virtual Environments

General Link for Python Virtual Environments: [Python Virtual Environments](#)

Using a virtual environment is recommended for CANStudio. This will ensure that all the dependencies required to run CANStudio will be given through the requirements.txt file. This will also ensure that the installation does not interfere with a python version already installed.

3.1.1. Creating Virtual Environments

Make sure you're inside the directory for "dist". Keep in mind that depending on which version of python you're running, you will need to install the virtualenv package to complete these tasks.

```
python -m venv venv
```

3.1.2. Activating Virtual Enviroments

From Command line, run the following command:

Windows, run:

```
venv\Scripts\activate.bat
```

Unix or MacOS, run:

```
source venv/bin/activate
```

3.1.3. Installing required Python3 dependencies:

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

Note: This should take a some time but will gather all the required modules to ensure that the application will run.

Once complete you may start the CANStudio host and access it via a web-browser.

3.2. Running CANStudio:

Make sure you're inside the directory for "dist".

Running CANStudio on localhost(127.0.0.1) and port 5000

```
flask run --host="127.0.0.1" --port=5000
```

3.3. Accessing the Application (Via a Web-browser)

```
localhost:5000/canstudio
```