

Introduction

`dev-docs-template` is a toolbox for creating development documents with minimal setup so that developers can focus on documentation.

How It Works

`dev-docs-template` is a project with all source codes needed to build an environment for documentation.

It packages the tools below needed to build/edit the documents into `docker images` or `VSCode Remote Container` :

- [VuePress](#) : Static Site Generator (Markdown to html)
- PlantUML VSCode Extension and plantuml-server docker container: Provide PlantUML edit/preview/export function
- [C4-PlantUML](#) : a plantuml library to combines the benefits of [PlantUML](#) and the [C4 model](#) for providing a simple way of `describing and communicate software architectures` – especially during up-front design sessions – with an intuitive language using open source and platform independent tools.
- Draw.io VSCode Extension: Provided complicated diagram edit feature within VSCode.
Recommended to use the svg file (with suffix `.drawio.svg`) so that no export is needed.

Developers can write documents with Markdown, and then uses this toolbox to `exported all markdown documents into a static html site` . Also dev mode of VuePress can help to preview your document sites while documenting.

Advance

To modify the setting or use the full features of VuePress, please check [official VuePress documents](#) .

dev-docs-template

Provide a toolbox/template and examples for easily creating the development documents

Get Started →

Out-of-Box

Minimal setup with markdown-centered project structure helps you focus on writing documents.

VuePress-Powered

Enjoy the dev experience of VuePress.

Rich Diagrams Edit

Support plantuml & draw.io based diagrams edit.

Source Code

Github: <https://github.com/lisy09/dev-docs-template> ↗

This is a project to provide a toolbox/template and examples for easily creating the development documents.

Features

This project provides these features:

- Major documents drafted in Markdown , generated as a single static html sites (with VuePress)

- Support export the document as a single `pdf` file
- Building the docs only depends on `docker` and `Gnu Make`
- Editing the docs depends on
 - `VSCode`
 - `VSCode Remote Container Extension`
 - `Docker engine`
- `PlantUML` in `Markdown` for creating some diagram with `UML`
- Also support `Plantuml` in `.puml` file and generated as image (`png/svg`) file (Recommended)
- Create more complicated diagrams & images by `draw.io` inside `VSCode`
- Several docs template

Easy to Start

```
1  # build all needed docker images
2  make all
3
4  # create a markdown file
5  echo '# Hello dev-docs-template' > docs/README.md
6
7  # build static html files
8  make html
```

sh

Getting Started

This section will help you setup the documentation environment with dev-docs-template from ground up.

Prerequisite

- The environment for build needs to be linux/amd64 or macos/amd64. Windows x64 users can use wsl2 as linux environment.
- The environemnt for build needs [docker engine installed](#)
- The environemnt for edit needs [docker compose](#)
- The environemnt for build needs `GNU make >= 3.8` installed
- Need [VSCode](#) and [VSCode Extension: Remote Development Extension Packs](#) installed.

Quick Start

1. Clone this Repository

Clone this repository as your own repository or a submodule of your monolithic repository.

```
1 | git clone https://github.com/lisy09/dev-docs-template
```

sh

2. Build Docker Images before Editing

To build all docker images locally:

```
1 | make all
```

sh

(Optional) To push built docker images to the remote registry:

```
1 | make push
```

sh

(Optional) To delete built local docker images:

```
1 | make clean
```

sh

Or you can check `./Makefile` for more details.

3. Create the Base Documents

Developers can create the base documents with simply replace `./docs` directory with one example in `./docs_example` , e.g.

```
1  # delete original ./docs directory
2  rm -rf docs
3
4  # copy an example docs
5  cp ./docs_examples/mono-data-science docs
6
7  # delete original ./docs_examples
8  rm -rf docs_examples
9  # modify the line below of .devcontainer/docker-compose.yml to exclude mounting c
10 # - ../${EXAMPLES_RELATIVE_PATH}:${WORKING_PACKAGE_PATH}/${EXAMPLES_RELATIVE_F
```

Developers can also create the base documents from scratch with proper [VuePress directory-structure](#) if familiar enough with VuePress and the source codes of this project.

4. Start the Remote Containter Environment in VSCode

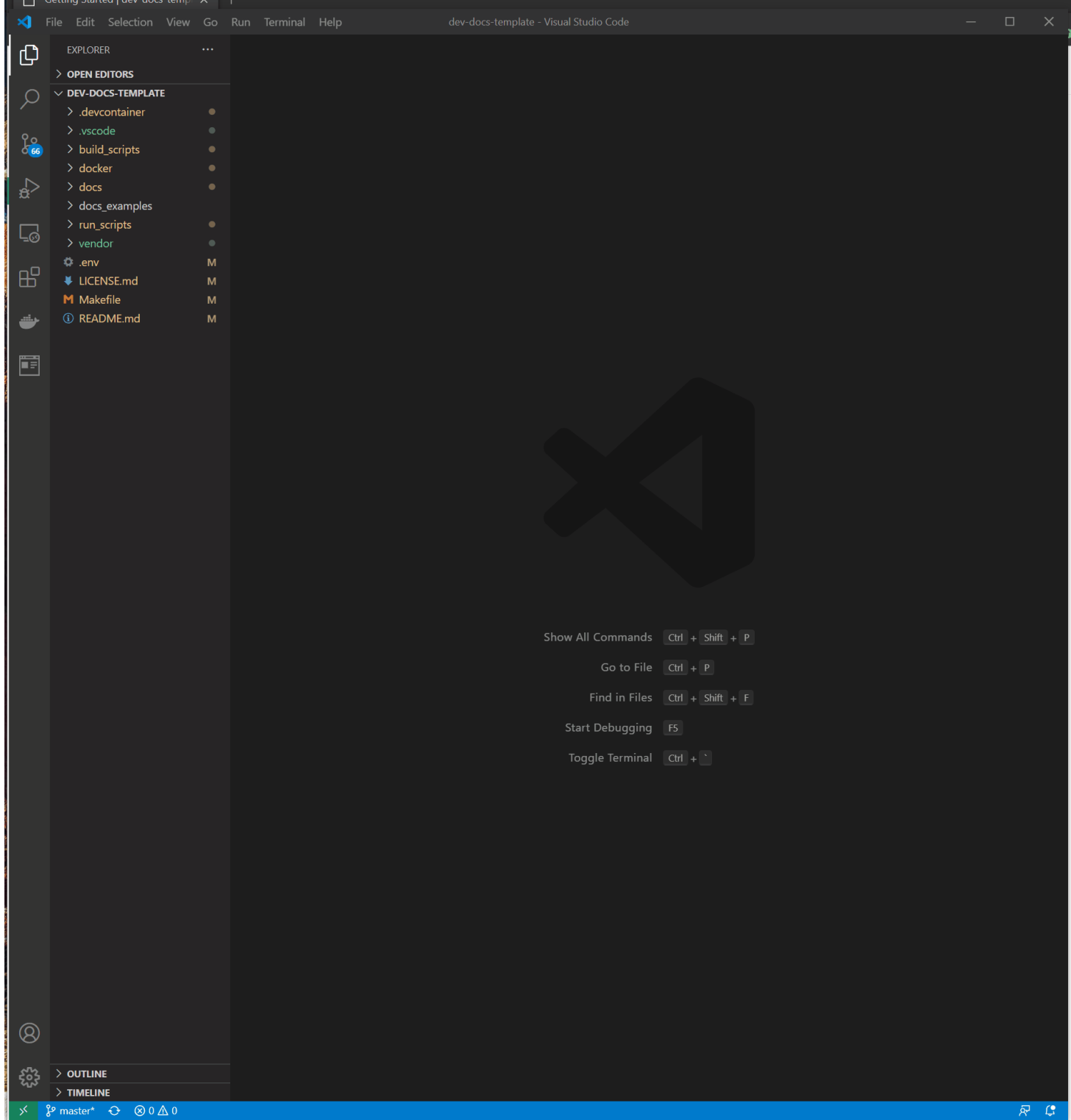
Run in `vscode` and execute `Remote-Containers: Open Folder in Container` and select the root directory of this module. Then VSCode will build a dev-container and run it based on `./devcontgainer/evcontainer.json` . The whole directory will be mounted to `/workspace` .

5. Start VuePress Dev Server for Realtime Preview

Run this command in the remote container.

```
1  make dev-local
```

Then a VuePress dev server will be started and listened at `$VUEPRESS_DEV_PORT` port value defined in `./.env` , default 8080) And you can check the docs in a browser with the url `localhost:$VUEPRESS_DEV_PORT``.



Build VuePress Docs in VSCode Remote Container

Run this command in the remote container.

```
1 | make build-local
```

sh

The built result is in `$DOCS_RELATIVE_PATH/.vuepress/dist` (defined in `./.env`, default `docs`)

dev-docs-template

📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 / 📄 📄 📄 📄 📄

📄 📄 📄 →

📄 📄 📄 📄

📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄

📄 📄 VuePress

📄 📄 📄 📄 📄 VuePress 📄 📄 📄 📄 📄 📄 📄 VuePress 📄 📄 📄 📄 📄

📄 📄 📄 📄 📄 📄 📄

📄 📄 📄 Plantuml 📄 draw.io 📄 📄 📄 📄 📄

📄 📄

Github: <https://github.com/lisy09/dev-docs-template> 📄

📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 📄 / 📄 📄 📄 📄 📄

📄 📄 📄 📄 📄 📄 📄 📄 📄 📄

```
1 # build all needed docker images
2 make all
3
4 # create a markdown file
5 echo '# Hello dev-docs-template' > docs/README.md
6
```

```
7 # build static html files
8 make html
```

MIT Licensed | Copyright © 2021-present lisyo9.thu@gmail.com