

Building my Curriculum Vitae (CV)

Last Update: 2024-10-24 09:52:42 +0100

Author: Paulo Jerônimo (paulojeronimo@gmail.com)

Online version: <https://paulojeronimo.com/cv/README.html>

Table of Contents

1. Introduction	1
2. Prerequisites	2
3. Downloading and extracting the source code	2
4. Developing it locally	2
4.1. Building and viewing the generated files	2
4.2. Stopping the local web server	3
5. Generating all the static formats (HTML, TXT, PDF and DOCX).....	3
6. Building all the files for a specific language (en as a sample).....	3
7. Listing the generated files	3
8. Generate all the files required to publish.....	3
9. Publish all the generated files to GitHub Pages.....	3
10. Deleting the generated files.....	4
11. Other topics	5
11.1. Installing requirements (including ruby, asciidoctor and asciidoctor-pdf).....	5

1. Introduction

This document explains how I build my résumé:

<https://paulojeronimo.com/cv/en/resume.pdf>

<https://paulojeronimo.com/cv/en/cv.pdf>

Its source code is available here:

<https://paulojeronimo.com/cv/source-code.zip>

2. Prerequisites

Currently, I use a Ubuntu machine to build my CV.

It is already in this version (and it can be running in a WSL2 environment):



```
$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:   Ubuntu 22.04.4 LTS
Release:       22.04
Codename:      jammy
```

In this machine, the following softwares are required to build this CV. [Read my procedures to install these requirements.](#)

1. Bash (the script will run on it).
2. `asciidoctor` and `asciidoctor-pdf` (to generate HTML and PDF formats).
3. `w3m` (for TXT generation).
4. `pandoc` (for DOCX generation).
5. `serve` (for local testing. Installed via `npm install -g serve`).
6. Other tools: `git`, `zip`, `rsync`, and `tree`.

3. Downloading and extracting the source code

```
$ curl https://paulojeronimo.com/cv/source-code.zip -o paulojeronimo-cv.zip
$ unzip paulojeronimo-cv.zip
$ cd cv
```

4. Developing it locally

4.1. Building and viewing the generated files

```
$ ENVIRONMENT=development ./build
$ ./build serve
```

Open <http://localhost:1234/en>.

4.2. Stopping the local web server

```
$ ./build serve stop
```

5. Generating all the static formats (HTML, TXT, PDF and DOCX)

```
$ ./build
```

Open [en/index.html](#).

6. Building all the files for a specific language (en as a sample)

```
$ ./en/build
```

7. Listing the generated files

```
$ ./build show-results
```

8. Generate all the files required to publish

```
$ ./build  
$ ./build publish
```

9. Publish all the generated files to GitHub Pages

```
$ ./build  
$ ./build publish-to-gh-pages
```

10. Deleting the generated files

```
$ ./build clean
```

11. Other topics

11.1. Installing requirements (including ruby, asciidoctor and asciidoctor-pdf)

```
$ sudo apt install -y build-essential gpg2 pandoc
```

```
$ gpg2 --keyserver keyserver.ubuntu.com --recv-keys  
409B6B1796C275462A1703113804BB82D39DC0E3 7D2BAF1CF37B13E2069D6956105BD0E739499BDB
```

```
$ \curl -sSL https://get.rvm.io | bash -s stable
```

```
$ source ~/.rvm/scripts/rvm
```

```
$ rvm pkg install openssl
```

```
$ rvm install ruby-3 --with-openssl-dir=$HOME/.rvm/usr
```

```
$ ruby --version  
ruby 3.0.0p0 (2020-12-25 revision 95aff21468) [x86_64-linux]
```

```
$ gem install asciidoctor
```

```
$ gem install asciidoctor-pdf
```

```
$ gem install rouge
```

```
$ asciidoctor --version  
Asciidoctor 2.0.23 [https://asciidoctor.org]  
Runtime Environment (ruby 3.0.0p0 (2020-12-25 revision 95aff21468) [x86_64-linux])  
(lc:UTF-8 fs:UTF-8 in:UTF-8 ex:UTF-8)
```

```
$ asciidoctor-pdf --version  
Asciidoctor PDF 2.3.19 using Asciidoctor 2.0.23 [https://asciidoctor.org]  
Runtime Environment (ruby 3.0.0p0 (2020-12-25 revision 95aff21468) [x86_64-linux])  
(lc:UTF-8 fs:UTF-8 in:UTF-8 ex:UTF-8)
```

References

- <https://rvm.io/>
- <https://github.com/rvm/rvm/issues/5209>