



VERSION: 1.0

General requirements

To start developing applications using Cartesi Rollups, first make sure that the packages listed in this section are all installed and working in your system. The instructions listed below should be enough to turn a fresh copy of the latest Ubuntu LTS distribution into a fully fledged Cartesi development environment.

SECTION GOAL

- ensure you have all the necessary dependencies for building Cartesi dApps :::

Basic tools

Before installing any package specifically needed for Cartesi Rollups development, we must first make sure that some basic tools are available.

First of all, let's install the `curl` tool for transferring data using a number of supported protocols:

```
sudo apt-get update
sudo apt-get install curl
```

Now, let's also install the `wget` tool for non-interactive download of files from the Web:

```
sudo apt-get install wget
```

Docker and Docker Compose

Docker is a widely used platform for users to build, run, and share applications with containers, and is a great way of distributing the Cartesi Rollups framework and all of its dependencies.

Docker Compose is an additional tool that allows us to start multiple Docker containers simultaneously and set up the communication between them. With Compose, we can locally instantiate a full Cartesi Rollups environment (i.e., all off-chain and on-chain components, including the local "testnet" blockchain itself), and thus test our Cartesi dApps using only your physical development machine.

To install Docker and Docker Compose, you can follow the [official installation instructions](#) from the [Docker website](#). For Ubuntu, you may also refer to [installation instructions that are specific for the platform](#). Below, we reproduce the installation steps that we have verified as working for a fresh Ubuntu installation:

First, we need to set up Docker's repository:

```
sudo apt-get update
sudo apt-get install ca-certificates curl gnupg lsb-release
sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

And then install Docker itself:

```
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Finally, it is a good idea to allow non-root users to manage Docker, so as to avoid having to type `sudo` for every Docker command:

```
sudo usermod -aG docker $USER
newgrp docker
```

After that, you can check if Docker is properly installed by running:

```
docker --version
```

It is recommended that the installed Docker version be at least `20.10.14` in order to adequately build a development environment and execute example dApps made available by Cartesi.

Node.js and NPM

Node.js is a very popular asynchronous event-driven JavaScript runtime, and is often distributed along with the NPM package manager. These are required for running Yarn.

To install Node.js, follow the official instructions. Specifically for Ubuntu, you can perform the steps below:

```
curl -fsSL https://deb.nodesource.com/setup_lts.x | sudo -E bash -  
sudo apt-get install -y nodejs
```

And then test them:

```
node --version  
npm --version
```

Yarn

Yarn is a great dependency management tool, and will be used for adding dependencies to your Cartesi dApps. You can install it by accessing the official installation site. The official instructions for installing on Ubuntu are as follows.

First, configure the repository:

```
curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -  
echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee  
/etc/apt/sources.list.d/yarn.list
```

Then, run:

```
sudo apt-get update && sudo apt-get install yarn
```

And finally, test that Yarn is installed and working property:

```
yarn --version
```

WSL

For Windows users, we recommend using [WSL](#).

Sunodo

[Sunodo](#) is a Rollups-as-a-Service platform and a set of tools for a simplified development of Cartesi dApps. You can install it as described below:

```
import Tabs from '@theme/Tabs'; import TabItem from '@theme/TabItem';
```

You can use Homebrew:

```
brew install sunodo/tap/sunodo
```

Or:

```
npm install -g @sunodo/cli
```

```
npm install -g @sunodo/cli
```

Install [WSL](#) and then run:

```
npm install -g @sunodo/cli
```

If you are using Docker Desktop, make sure to update it to the latest version. If you are using Homebrew to install Sunodo, navigate to *Settings > Resources > File sharing* and add the following path: `/opt/homebrew/Cellar/sunodo. :::`

Tags: [build](#) [dapps](#) [developer](#)

 [Edit this page](#)



[About](#)

[Blog](#)

[Governance & Grants](#)

[Discord](#)

[GitHub](#)

[Telegram](#)

[Twitter](#)

[Medium](#)

[Whitepaper](#)

[Foundation Notice](#)

[Usage Policy](#)

[Youtube](#)

[Reddit](#)

[LinkedIn](#)

[Instagram](#)