Published at https://arkit.co.in

# How to Install Magento 2 on Linux/Ubuntu

This tutorial will consider the installation of Magento 2.4.2 Open Source, the latest version of the platform. Bear in mind that Magento 2.3, another popular version right now, has different system requirements as well as the installation itself.

## Magento 2.4 vs Magento 2.3

The setup process has changed. While in Magento 2.3 it was possible to deploy an online store from the browser through the graphical interface, in 2.4 you can only use the command line.

After launching an online shop we recommend **converting a Magento website to a PWA.**

## Step 1: Preparation

Before installing Magento 2 on Linux/Ubuntu, you should take care of the appropriate environment. There are several main software components for your future online shop. Let's deal with them.

1. First and foremost, you need to **update Ubuntu packages**.

Here is the command:

```
sudo apt update
```

2. Then choose **a web server**. In the example below, we'll use NGINX (1.x), but you can choose Apache as well (2.4).

Installing the web server:

```
sudo apt install nginx
```

Checking the status after installation:

```
vagrant@m2:~$ sudo service nginx status
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2021-05-03 14:17:09 UTC; 32s ago
     Docs: man:nginx(8)
 Main PID: 18035 (nginx)
    Tasks: 2 (limit: 2361)
   CGroup: /system.slice/nginx.service
           ├─18035 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
           └─18037 nginx: worker process

May 03 14:17:09 m2 systemd[1]: Starting A high performance web server and a reverse proxy server...
May 03 14:17:09 m2 systemd[1]: nginx.service: Failed to parse PID from file /run/nginx.pid: Invalid argument
May 03 14:17:09 m2 systemd[1]: Started A high performance web server and a reverse proxy server.
```

Magento provides a standard web server configuration file that is located in the root of the store and called `nginx.conf.sample`. It can be used as a web server configuration file for your store.

Published at https://arkit.co.in/how-to-install-magento-2-on-linux/

3. Set up **PHP** (7.4)

- Let's add the repository to Ubuntu:

```
sudo LC_ALL=C.UTF-8 add-apt-repository -y ppa:ondrej/php

printf "y\n" | sudo apt-add-repository ppa:brightbox/ruby-ng
```

- Then install PHP from previously added repositories:

```
sudo apt install php7.4 php7.4-common php7.4-gmp php7.4-curl
php7.4-soap php7.4-bcmath php7.4-intl php7.4-mbstring php7.4-
xmlrpc php7.4-mysql php7.4-gd php7.4-xml php7.4-cli php7.4-zip
php7.4-fpm
```

- You can check the version of the installed PHP with the command `php -v`:

```
vagrant@m2:~$ php -v
PHP 7.4.18 (cli) (built: May  3 2021 11:26:48) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.18, Copyright (c), by Zend Technologies
```

4. Set up **Composer** (2.0.13)

Here is the command for the version check:

```
root@b1a99403ec03:/var/www# composer --version
Composer version 2.0.13 2021-04-27 13:11:08
```

5. Choose a **database server**: we'll go with MariaDB (10.4), but MySQL also fits (8.0).

- Firstly, install MariaDB:

```
sudo apt-get install mariadb-server mariadb-client
```

- Then run this command:

```
sudo mysql_secure_installation
```

The following screenshot showcases that the installation was successful and you can check the version:

```
vagrant@m2:~$ mysql --version
mysql  Ver 15.1 Distrib 10.1.47-MariaDB, for debian-linux-gnu (x86_64) using readline 5.2
```

- To connect to the database we use the following command:

```
mysql -u root -p *
```

- Create a database:

```
CREATE DATABASE mg2;
```

- Create a new user:

```
CREATE USER 'mg2'@'localhost' IDENTIFIED BY 'YOUR_PASSWORD';
```

- Give the user rights:

```
Grant mg2 user to mg2 database:

GRANT ALL ON mg2.* TO 'mg2'@'localhost' IDENTIFIED BY
'YOUR_PASSWORD' WITH GRANT OPTION;
```

- Finally, do the privilege update and exit:

```
FLUSH PRIVILEGES;

EXIT;
```

1) **Elasticsearch** (7.x)

2) **Magento** 2.4.2

3) Optional: if additional **caching** is required, you can also install Redis Cache (6.0), for instance.

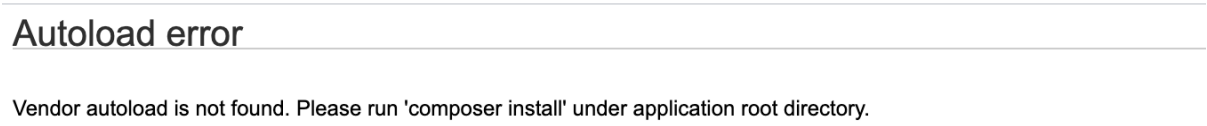
## Step 2: Installation

You can launch the process in three ways:

1. take Composer;
2. clone the Magento 2 GitHub repository;
3. or download the archive from the Magento website.

We'd recommend Composer as a bit more convenient and slightly faster option.

If you start the installation not via Composer, you may see the following error while trying to launch the shop with the URL. This means that there are not enough modules (there is no autoload file). You need to run composer install, as shown in the screenshot.



Before the start of the installation, such a screen appears:

So, we can start to assemble the online store. In the command line, we write the name of the database that we will use, the login, password, other admin data, currency, localization, time zone, and so forth. Everything in one large command:

```
php bin/magento setup:install --base-url=http://m2.local/ --db-
host=mysql --db-name=mg2 --db-user=mg2 --db-password=YOUR_PASSWORD -
-admin-firstname=Magento --admin-lastname=User --admin-
email=magento.user@example.com --admin-user=admin --admin-
password=admin123 --language=en_US --currency=USD --
timezone=America/Chicago --use-rewrites=1 --search-
engine=elasticsearch7 --elasticsearch-host=m2.local --elasticsearch-
port=9200
```

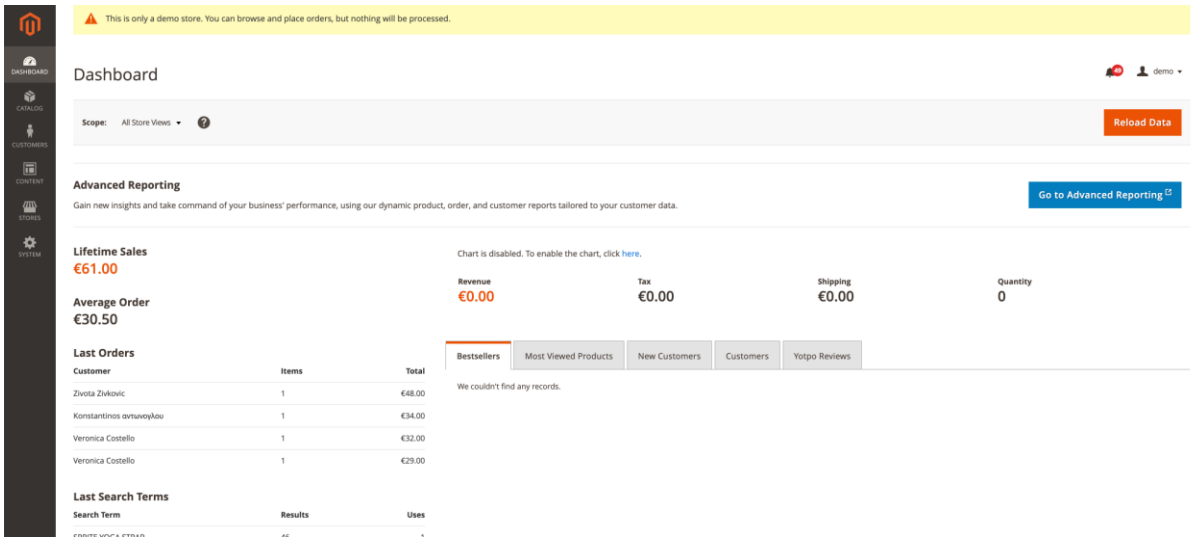
When the installation is completed, we get two links: a link for visiting the front-end of the site (a storefront) and the back-end (Magento Admin), as in the screenshot below. At the end of the installation you'll see the following:

```
Post installation file permissions check...
For security, remove write permissions from these directories: '/var/www/app/etc'
[Progress: 841 / 842]
Write installation date...
[Progress: 842 / 842]
[SUCCESS]: Magento installation complete.
[SUCCESS]: Magento Admin URI: /admin_w47lci
Nothing to import.
```

That's it! There is a back-end:

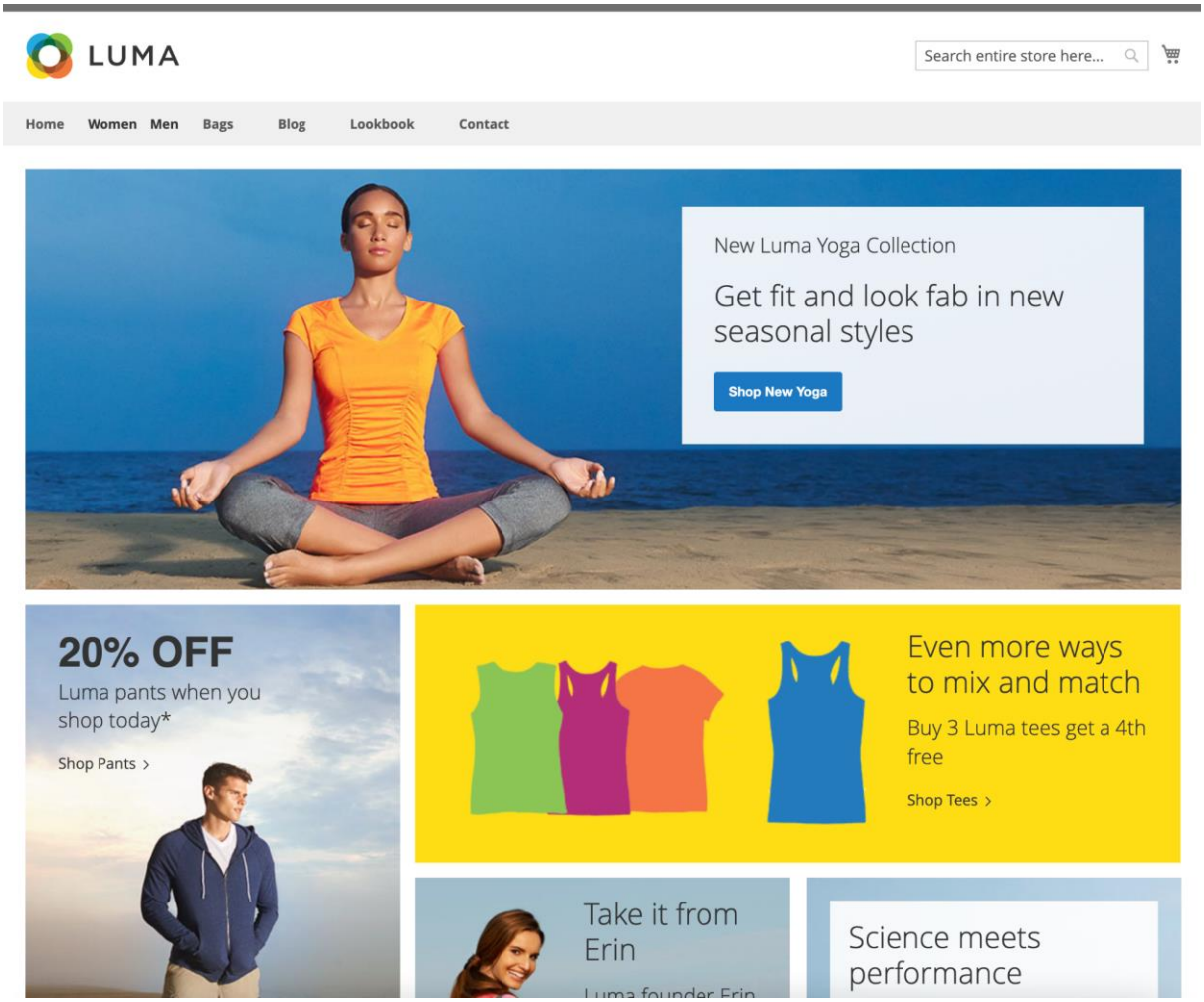
Published at https://arkit.co.in

There is a dashboard of your store:

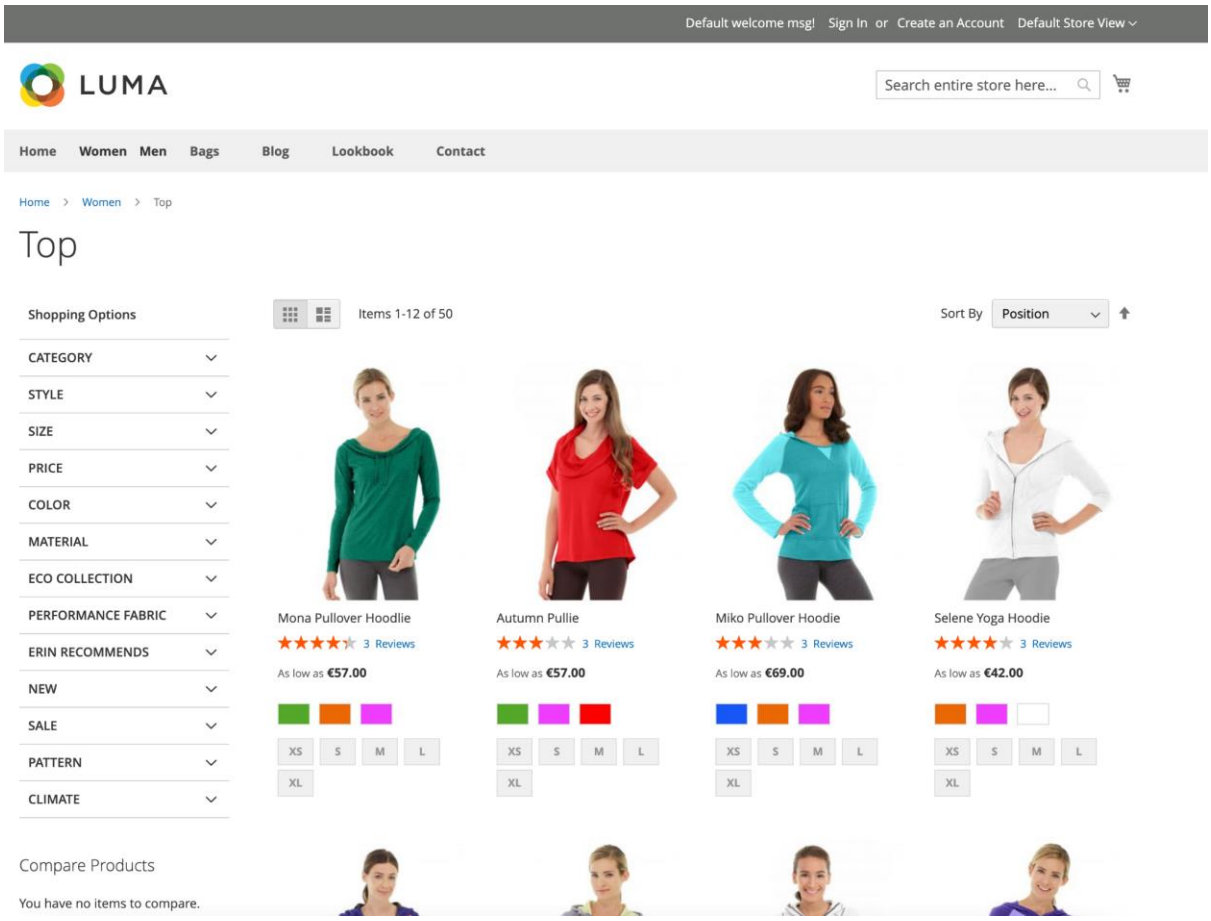



There are screenshots from the front-end:

Published at https://arkit.co.in/how-to-install-magento-2-on-linux/


LUMA
Search entire store here...
Home
Women
Men
Bags
Blog
Lookbook
Contact
New Luma Yoga Collection
Get fit and look fab in new seasonal styles
Shop New Yoga
20% OFF
Luma pants when you shop today*
Shop Pants ›
Even more ways to mix and match
Buy 3 Luma tees get a 4th free
Shop Tees ›
Take it from Erin
Science meets performance

Published at https://arkit.co.in




## To Conclude

The installation of Magento does not require advanced technical knowledge and skills: even a junior or trainee can handle it. In a normal situation, the process can take from half an hour to a couple of hours. You only need to be very scrupulous about the system requirements.

## About the Author

Alex Husar

Published at https://arkit.co.in/how-to-install-magento-2-on-linux/

Published at https://arkit.co.in

Alex Husar, Chief Technology Officer at Onilab with 8+ years of experience in developing PWAs, Magento migration, and Salesforce development. He graduated from the Czech Technical University and obtained a bachelor's degree in Computer Software Engineering. Alex's expertise includes both full-stack dev skills and a strong ability to provide project-critical guidance to the whole team.