

[gemini.google.com /app/c2bc00d01b638e6f](https://gemini.google.com/app/c2bc00d01b638e6f)

## Unknown Title

---



services:

mariadb:

  image: mariadb:latest

  container\_name: mariadb-moodle

  restart: always

  environment:

    MYSQL\_ROOT\_PASSWORD: rootpass

    MYSQL\_DATABASE: moodle

    MYSQL\_USER: cristobal

    MYSQL\_PASSWORD: 12345

  volumes:

    - mariadb\_data:/var/lib/mysql

networks:

- moodle\_network

moodle:

image: moodlehq/moodle-php-apache:latest

container\_name: moodle

restart: always

depends\_on:

- mariadb

ports:

- "8083:80" # HTTP

- "8443:443" # HTTPS

environment:

MOODLE\_DATABASE\_TYPE: mysqli

MOODLE\_DATABASE\_HOST: mariadb\_host

MOODLE\_DATABASE\_NAME: moodle

MOODLE\_DATABASE\_USER: cristobal

MOODLE\_DATABASE\_PASSWORD: 12345

MOODLE\_USERNAME: cristobal

MOODLE\_PASSWORD: 12345

MOODLE\_SITE\_NAME: "Moodle SUAREZ1"

MOODLE\_EMAIL: admin@suarez1.abad2

volumes:

- moodle\_data:/bitnami/moodle

- moodle\_mariadb\_data:/bitnami/moodledata

networks:

- moodle\_network

volumes:

```
mariadb_data:
```

```
moodle_data:
```

```
moodle_mariadb_data:
```

networks:

```
moodle_network: "[root@control01 Moodle]$docker compose up -d
```

```
[+] Running 1/1
```

```
✗ moodle Error manifest for moodlehq/moodle-php-apache:lat... 1.4s
```

Error response from daemon: manifest for moodlehq/moodle-php-apache:latest not found: manifest unknown: manifest unknown

Tu Nombre domingo 2 noviembre 2025 14:22

```
[root@control01 Moodle]$"
```

The error you're seeing, **Error response from daemon: manifest for moodlehq/moodle-php-apache:latest not found: manifest unknown: manifest unknown**, indicates that Docker couldn't find an image with the tag :latest for the repository **moodlehq/moodle-php-apache** on Docker Hub.

This typically happens when the image maintainer (Moodle HQ, in this case) *doesn't* provide a generic :latest tag for that specific image, or it has been temporarily removed or renamed. Looking at the official Docker Hub page for moodlehq/moodle-php-apache, it appears they primarily use tags that specify the **PHP version** and the **base operating system distribution** (e.g., 8.3-bookworm or just 8.3).

---

## The Fix: Specify an Image Tag

To resolve this, you need to change the `image` line for your `moodle` service in your `docker-compose.yml` file to use a **specific, existing tag** instead of :latest.

### 1. Find a Suitable Tag

You should choose a tag that corresponds to a **recent and supported PHP version**. For example, tags like 8.3, 8.2, or 8.1 are often available. I'll suggest using a recent, stable version.

### 2. Update the `docker-compose.yml`

Change the moodle service definition from this:

```
moodle:  
  image: moodlehq/moodle-php-apache:latest # <--- The problematic line
```

To something like this (using a recent stable tag, like **8.3**):

```
moodle:  
  image: moodlehq/moodle-php-apache:8.3 # <--- Change to a specific tag
```

*Note: You can check the Docker Hub page for moodlehq/moodle-php-apache to see all currently available tags.*

### 3. Re-run Docker Compose

Save the changes to your docker-compose.yml file and run the command again:

```
docker compose up -d
```

This should now successfully pull the specified Moodle image and start your services.

---

## Suggestion for Database Host

I also noticed a potential issue in your moodle service environment variables:

```
MOODLE_DATABASE_HOST: mariadb_host # <--- Potential Issue
```

In Docker Compose, services on the same network can communicate using their **service names** as the hostname. The service name for your MariaDB container is **mariadb**.

You should change the MOODLE\_DATABASE\_HOST to the **service name** of the database container, which is **mariadb**:

```
moodle:  
  # ...  
  environment:  
    # ...  
    MOODLE_DATABASE_HOST: mariadb # <--- Change this to the service name!  
    # ...
```

This ensures the Moodle container can connect to the MariaDB container correctly over the Docker network.

Would you like me to check any other configuration details in your docker-compose.yml?

services:

```
mariadb:  
  
image: mariadb:latest  
  
container_name: mariadb-moodle  
  
restart: always  
  
environment:  
  
  MYSQL_ROOT_PASSWORD: rootpass  
  
  MYSQL_DATABASE: moodle  
  
  MYSQL_USER: cristobal  
  
  MYSQL_PASSWORD: 12345  
  
volumes:  
  - mariadb_data:/var/lib/mysql  
  
networks:  
  - moodle_network  
  
  
moodle:  
  
image: moodle-php-apache:8.4  
  
container_name: moodle  
  
restart: always  
  
depends_on:  
  - mariadb  
  
ports:  
  - "8083:80" # HTTP  
  - "8443:443" # HTTPS  
  
environment:  
  
  MOODLE_DATABASE_TYPE: mysqli  
  
  MOODLE_DATABASE_HOST: mariadb_host  
  
  MOODLE_DATABASE_NAME: moodle
```

```
MOODLE_DATABASE_USER: cristobal  
MOODLE_DATABASE_PASSWORD: 12345  
MOODLE_USERNAME: cristobal  
MOODLE_PASSWORD: 12345  
MOODLE_SITE_NAME: "Moodle SUAREZ1"  
MOODLE_EMAIL: admin@suarez1.abad2
```

volumes:

- moodle\_data:/bitnami/moodle
- moodle\_mariadb\_data:/bitnami/moodledata

networks:

- moodle\_network

volumes:

mariadb\_data:

moodle\_data:

moodle\_mariadb\_data:

networks:

moodle\_network: "[root@control01 Moodle]\$docker compose up -d"

[+] Running 1/1

✗ moodle Error pull access denied for moodle-php-apache, r... 1.4s

Error response from daemon: pull access denied for moodle-php-apache, repository does not exist or may require 'docker login': denied: requested access to the resource is denied"

That error, **Error response from daemon: pull access denied for moodle-php-apache, repository does not exist or may require 'docker login': denied: requested access to the resource is denied**, means Docker cannot find the image you specified because the repository name is **incorrect or incomplete**.

Your configuration currently specifies the Moodle image as:

```
image: moodle-php-apache:8.4
```

This tells Docker to look for an image named `moodle-php-apache` in the `root` Docker Hub library, but the official Moodle image is located under the `moodlehq` organization's namespace.

---

## The Fix: Correcting the Image Name

You need to add the correct organization prefix, `moodlehq/`, to the image name.

### 1. Update the `docker-compose.yml`

Change the `moodle` service definition from this:

```
moodle:  
  image: moodle-php-apache:8.4 # <--- Missing the 'moodlehq/' prefix
```

To this:

```
moodle:  
  image: moodlehq/moodle-php-apache:8.4 # <--- Correct image name
```

## Other Important Corrections

While fixing the image name, there are **two critical configuration issues** that must be addressed to ensure Moodle runs correctly:

### 1. Database Hostname Correction (Crucial)

In Docker Compose, services communicate using their service names. Your MariaDB service is named `mariadb`, so the Moodle service must use that name to connect to the database container.

Change this:

```
MOODLE_DATABASE_HOST: mariadb_host # <--- Incorrect
```

To this:

```
MOODLE_DATABASE_HOST: mariadb # <--- Correct (using the service name)
```

### 2. Volume Path Correction (Moodle Image Type)

The official Moodle HQ images use standard paths, which are different from the paths used by Bitnami images (which your current paths seem to match).

- The Moodle application code should typically map to `/var/www/html` or similar, not `/bitnami/moodle`.

- The Moodle data directory (`moodledata`) should typically map to `/var/www/moodledata` or similar, not `/bitnami/moodledata`.

Since you are using the image `moodlehq/moodle-php-apache:8.4`, you should change the volume paths. Based on the configuration for that image, the standard Moodle code location is often mapped to where the web server expects it (`/var/www/html` is a common path for Apache-based images), and a dedicated `moodledata` volume is created internally.

For the `moodlehq/moodle-php-apache` image, you typically mount **your Moodle code** to `/var/www/html` and the **`moodledata`** to `/var/www/moodledata` (or let the container manage the data volume). However, if you are attempting a **one-container install**, you might consider using the officially recommended `moodle-docker` setup or an image specifically built for this, as the `moodle-php-apache` image is primarily for development environments and assumes you clone the Moodle code separately.

**Recommendation:** If you intend to use this specific `moodlehq` image, you should adjust the volumes:

Change your moodle volumes from:

```
volumes:
  - moodle_data:/bitnami/moodle
  - moodle_mariadb_data:/bitnami/moodledata
```

To the standard directory structure, assuming you have the Moodle code in a local directory mounted as `moodle_data` (though the exact paths depend on how the Moodle HQ image is configured and where your code is):

```
volumes:
  - moodle_data:/var/www/html # <--- Assuming Moodle application code
  - moodle_mariadb_data:/var/www/moodledata # <--- Assuming moodledata
    directory
```

## ✓ Final Corrected docker-compose.yml

Here is the fully corrected `docker-compose.yml`:

```
services:
  mariadb:
    image: mariadb:latest
    container_name: mariadb-moodle
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: rootpass
      MYSQL_DATABASE: moodle
      MYSQL_USER: cristobal
      MYSQL_PASSWORD: 12345
```

```

volumes:
  - mariadb_data:/var/lib/mysql
networks:
  - moodle_network

moodle:
  image: moodlehq/moodle-php-apache:8.4 # <-- FIX 1: Corrected image name
  container_name: moodle
  restart: always
  depends_on:
    - mariadb
  ports:
    - "8083:80"      # HTTP
    - "8443:443"     # HTTPS
  environment:
    MOODLE_DATABASE_TYPE: mysqli
    MOODLE_DATABASE_HOST: mariadb # <-- FIX 2: Corrected database
  hostname
    MOODLE_DATABASE_NAME: moodle
    MOODLE_DATABASE_USER: cristobal
    MOODLE_DATABASE_PASSWORD: 12345
    MOODLE_USERNAME: cristobal
    MOODLE_PASSWORD: 12345
    MOODLE_SITE_NAME: "Moodle SUAREZ1"
    MOODLE_EMAIL: admin@suarez1.abad2
  volumes:
    - moodle_data:/var/www/html # <-- FIX 3: Corrected app path
    - moodle_mariadb_data:/var/www/moodledata # <-- FIX 3: Corrected data
path
  networks:
    - moodle_network

volumes:
  mariadb_data:
  moodle_data:
  moodle_mariadb_data:

networks:
  moodle_network:

```

Would you like me to help you verify the necessary local folder structure to use with those new volume paths?

Forbidden

You don't have permission to access this resource.

Apache/2.4.65 (Debian) Server at 10.2.17.10 Port 8083

It appears your containers started successfully, but now you're encountering a "**Forbidden - You don't have permission to access this resource**" error when trying to access Moodle via your browser at <http://10.2.17.10:8083>. This is typically a file permission or directory issue related to how you've set up the **volumes** for the Moodle application code.

The moodlehq/moodle-php-apache image expects the Moodle application files to be available and readable by the Apache web server user inside the container.

## Fix: Volume and Permission Check

The most likely cause is that your local directory being mounted for `moodle_data` does not contain the Moodle files, or the permissions on those files are incorrect for the user inside the container.

### 1. Verify Moodle Code Location (Critical)

The moodlehq/moodle-php-apache image **does not include the Moodle code itself**; it only provides the PHP/Apache environment. You must **manually clone the Moodle code** into the local directory that you are mapping to the container.

You are mapping:

```
- moodle_data:/var/www/html
```

You must ensure that the local directory associated with the `moodle_data` volume (usually a directory created next to your `docker-compose.yml` file) contains the full Moodle application code (e.g., `index.php`, `config.php`, etc.).

#### Action:

1. **Stop** your containers: `docker compose down`

2. **Clone Moodle:** Navigate to the directory containing your `docker-compose.yml` file. If you haven't done so already, clone Moodle's code into a new directory (e.g., `moodle-code`):

```
git clone -b MOODLE_404_STABLE git://git.moodle.org/moodle.git  
./moodle-code
```

*(Note: Using a specific stable branch like `MOODLE_404_STABLE` is recommended instead of cloning the entire repository or using an unknown branch.)*

3. **Update `docker-compose.yml`:** Change your `moodle` service volumes to explicitly map this new directory. Let's create a specific volume definition instead of relying on the named volume placeholder for `moodle_data`:

```
# Update moodle service volumes:  
volumes:  
  - ./moodle-code:/var/www/html # <--- Map your local Moodle code here  
  - moodle_mariadb_data:/var/www/moodledata
```

*If you use a directory path (`./moodle-code`) instead of a named volume (`moodle_data`), you must remove `moodle_data:` from the `volumes:` section at the bottom of the file.*

## 2. Check File Permissions (If Cloning Doesn't Fix It)

If the Moodle code is present, the issue is likely due to file ownership/permissions. The Apache process inside the container runs as a non-root user (often `www-data`). If the files on your host system are owned by root, the container cannot read them.

**Action:** After cloning, you might need to adjust the local permissions before running `docker compose up -d`.

```
# Change ownership of the Moodle files on the host machine  
sudo chown -R 33:33 ./moodle-code
```

*(The user ID 33 is the common UID for `www-data` on Debian/Ubuntu, which is the base for the Moodle image. This ensures the file permissions match the user inside the container.)*

**After applying these changes, run:**

```
docker compose up -d
```

Then try accessing Moodle again at <http://10.2.17.10:8083>.

---

I've provided the full solution for cloning the code and correcting permissions. Would you like me to walk you through the exact steps for cloning the Moodle code into a local directory and updating your `docker-compose.yml` to reflect that change?

Me funciona? "

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/install.php on line 183

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/lib/setuplib.php on line 38

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/lib/setuplib.php on line 40

Deprecated: moodle\_url::\_\_construct(): Implicitly marking

parameter \$params as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/lib/weblib.php on line 332

Deprecated: moodle\_url::params(): Implicitly marking parameter

\$params as nullable is deprecated, the explicit nullable type must be

used instead in /var/www/html/lib/weblib.php on line 405

Deprecated: moodle\_url::merge\_overrideparams(): Implicitly

marking parameter \$overrideparams as nullable is deprecated, the

explicit nullable type must be used instead in /var/www/html/lib/weblib.php on line 485

Deprecated: moodle\_url::get\_query\_string(): Implicitly marking

parameter \$overrideparams as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/weblib.php on line 513

Deprecated: moodle\_url::out(): Implicitly marking parameter

\$overrideparams as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/lib/weblib.php on line 580

Deprecated: moodle\_url::raw\_out(): Implicitly marking parameter

\$overrideparams as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/lib/weblib.php on line 612

Deprecated: moodle\_url::out\_as\_local\_url(): Implicitly marking

parameter \$overrideparams as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/weblib.php on line 995

Deprecated: user\_picture::fields(): Implicitly marking parameter  
\$extrafields as nullable is deprecated, the explicit nullable type must  
be used instead in /var/www/html/lib/outputcomponents.php on line 256

Deprecated: user\_picture::unalias(): Implicitly marking  
parameter \$extrafields as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 284

Deprecated: user\_picture::get\_url(): Implicitly marking  
parameter \$renderer as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 326

Deprecated: pix\_icon::\_\_construct(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 659

Deprecated: action\_link::\_\_construct(): Implicitly marking  
parameter \$action as nullable is deprecated, the explicit nullable type  
must be used instead in /var/www/html/lib/outputcomponents.php on line 1628

Deprecated: action\_link::\_\_construct(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 1628

Deprecated: action\_link::\_\_construct(): Implicitly marking  
parameter \$icon as nullable is deprecated, the explicit nullable type  
must be used instead in /var/www/html/lib/outputcomponents.php on line 1628

Deprecated: html\_writer::tag(): Implicitly marking parameter  
\$attributes as nullable is deprecated, the explicit nullable type must  
be used instead in /var/www/html/lib/outputcomponents.php on line 1744

Deprecated: html\_writer::start\_tag(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 1755

Deprecated: html\_writer::empty\_tag(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 1776

Deprecated: html\_writer::nonempty\_tag(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 1788

Deprecated: html\_writer::attributes(): Implicitly marking  
parameter \$attributes as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/outputcomponents.php on line 1823

Deprecated: html\_writer::img(): Implicitly marking parameter  
\$attributes as nullable is deprecated, the explicit nullable type must  
be used instead in /var/www/html/lib/outputcomponents.php on line 1840

Deprecated: html\_writer::link(): Implicitly marking parameter  
\$attributes as nullable is deprecated, the explicit nullable type must

be used instead in /var/www/html/lib/outputcomponents.php on line 1884

Deprecated: html\_writer::checkbox(): Implicitly marking

parameter \$attributes as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/outputcomponents.php on line 1901

Deprecated: html\_writer::checkbox(): Implicitly marking

parameter \$labelattributes as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 1901

Deprecated: html\_writer::select\_yes\_no(): Implicitly marking

parameter \$attributes as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/outputcomponents.php on line 1935

Deprecated: html\_writer::select(): Implicitly marking parameter

\$attributes as nullable is deprecated, the explicit nullable type must

be used instead in /var/www/html/lib/outputcomponents.php on line 1956

Deprecated: html\_writer::select\_time(): Implicitly marking

parameter \$attributes as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/outputcomponents.php on line 2064

Deprecated: html\_writer::alist(): Implicitly marking parameter

\$attributes as nullable is deprecated, the explicit nullable type must

be used instead in /var/www/html/lib/outputcomponents.php on line 2144

Deprecated: html\_writer::input\_hidden\_params(): Implicitly

marking parameter \$exclude as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2160

Deprecated: html\_writer::add\_class(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2499

Deprecated: html\_writer::div(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2523

Deprecated: html\_writer::start\_div(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2534

Deprecated: html\_writer::span(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2555

Deprecated: html\_writer::start\_span(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2566

Deprecated: js\_writer::function\_call(): Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2599

Deprecated: js\_writer::function\_call\_with\_Y(): Implicitly marking parameter \$extraarguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2622

Deprecated: js\_writer::object\_init(): Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2642

Deprecated: js\_writer::object\_init(): Implicitly marking parameter \$requirements as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2642

Deprecated: js\_writer::event\_handler(): Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2702

Deprecated: html\_table\_row::\_\_construct(): Implicitly marking parameter \$cells as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 2939

Deprecated: block\_contents::\_\_construct(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 3506

Deprecated: custom\_menu\_item::\_\_construct(): Implicitly marking parameter \$url as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 3633

Deprecated: custom\_menu\_item::\_\_construct(): Implicitly marking parameter \$parent as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 3633

Deprecated: custom\_menu\_item::add(): Implicitly marking parameter \$url as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 3656

Deprecated: action\_menu::get\_primary\_actions(): Implicitly marking parameter \$output as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputcomponents.php on line 4599

Deprecated: core\_renderer::action\_link(): Implicitly marking parameter \$action as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2084

Deprecated: core\_renderer::action\_link(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2084

Deprecated: core\_renderer::action\_icon(): Implicitly marking parameter \$action as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2124

Deprecated: core\_renderer::action\_icon(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/outputrenderers.php on line 2124

Deprecated: core\_renderer::single\_button(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2233

Deprecated: core\_renderer::image\_icon(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2410

Deprecated: core\_renderer::pix\_icon(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2438

Deprecated: core\_renderer::spacer(): Implicitly marking parameter \$attributes as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2670

Deprecated: core\_renderer::user\_picture(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 2724

Deprecated: core\_renderer\_maintenance::user\_picture(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrenderers.php on line 5936

Deprecated: page\_requirements\_manager::js\_function\_call():

Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1017

Deprecated: page\_requirements\_manager::yui\_module(): Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1103

Deprecated: page\_requirements\_manager::js\_init\_call():  
Implicitly marking parameter \$extraarguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1140

Deprecated: page\_requirements\_manager::js\_init\_call():  
Implicitly marking parameter \$module as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1140

Deprecated: page\_requirements\_manager::js\_init\_code():  
Implicitly marking parameter \$module as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1162

Deprecated: page\_requirements\_manager::event\_handler():  
Implicitly marking parameter \$arguments as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/outputrequirementslib.php on line 1313

Deprecated: dml\_read\_exception::\_\_construct(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dmllib.php on line 135

Deprecated: dml\_multiple\_records\_exception::\_\_construct():

Implicitly marking parameter \$params as nullable is deprecated, the

explicit nullable type must be used instead in /var/www/html/lib/dmllib.php on line 164

Deprecated: dml\_missing\_record\_exception::\_\_construct():

Implicitly marking parameter \$params as nullable is deprecated, the

explicit nullable type must be used instead in /var/www/html/lib/dmllib.php on line 193

Deprecated: dml\_write\_exception::\_\_construct(): Implicitly

marking parameter \$params as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/dmllib.php on line 246

Deprecated: moodle\_database::connect(): Implicitly marking

parameter \$dboptions as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 304

Deprecated: moodle\_database::store\_settings(): Implicitly

marking parameter \$dboptions as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 316

Deprecated: moodle\_database::create\_database(): Implicitly

marking parameter \$dboptions as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 376

Deprecated: moodle\_database::print\_debug(): Implicitly marking

parameter \$params as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 600

Deprecated: moodle\_database::where\_clause(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 661

Deprecated: moodle\_database::fix\_sql\_params(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 894

Deprecated: moodle\_database::execute(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1317

Deprecated: moodle\_database::get\_recordset(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1353

Deprecated: moodle\_database::get\_recordset\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1399

Deprecated: moodle\_database::get\_recordset\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1426

Deprecated: moodle\_database::get\_records(): Implicitly marking

parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1462

Deprecated: moodle\_database::get\_records\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1506

Deprecated: moodle\_database::get\_records\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1530

Deprecated: moodle\_database::get\_records\_menu(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1551

Deprecated: moodle\_database::get\_records\_select\_menu(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1580

Deprecated: moodle\_database::get\_records\_sql\_menu(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1606

Deprecated: moodle\_database::get\_record\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1651

Deprecated: moodle\_database::get\_record\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1677

Deprecated: moodle\_database::get\_field\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1733

Deprecated: moodle\_database::get\_field\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1756

Deprecated: moodle\_database::get\_fieldset\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1789

Deprecated: moodle\_database::get\_fieldset\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1804

Deprecated: moodle\_database::set\_field(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1919

Deprecated: moodle\_database::set\_field\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1935

Deprecated: moodle\_database::count\_records(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1946

Deprecated: moodle\_database::count\_records\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1961

Deprecated: moodle\_database::count\_records\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 1981

Deprecated: moodle\_database::record\_exists\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 2011

Deprecated: moodle\_database::record\_exists\_sql(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 2029

Deprecated: moodle\_database::delete\_records(): Implicitly marking parameter \$conditions as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 2045

Deprecated: moodle\_database::delete\_records\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/dml/moodle\_database.php on line 2101

Deprecated: remove\_course\_contents(): Implicitly marking

parameter \$options as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/lib/moodlelib.php on line 4727

Deprecated: moodle\_page::set\_url(): Implicitly marking parameter

\$params as nullable is deprecated, the explicit nullable type must be

used instead in /var/www/html/lib/pagelib.php on line 1496

Deprecated: admin\_externalpage\_setup(): Implicitly marking

parameter \$extraurlparams as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/adminlib.php on line 8711

Deprecated: admin\_setting\_configcolourpicker::\_\_construct():

Implicitly marking parameter \$previewconfig as nullable is deprecated,

the explicit nullable type must be used instead in /var/www/html/lib/adminlib.php on line 10444

Deprecated: admin\_setting\_configstoredfile::\_\_construct():

Implicitly marking parameter \$options as nullable is deprecated, the

explicit nullable type must be used instead in /var/www/html/lib/adminlib.php on line 10611

Deprecated: xmldb\_object::validateDefinition(): Implicitly

marking parameter \$xmldb\_table as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/xmldb/xmldb\_object.php on line 419

Deprecated: xmldb\_table::validateDefinition(): Implicitly

marking parameter \$xmlDb\_table as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/xmlDb/xmlDb\_table.php on line 716

Deprecated: xmlDb\_field::validateDefinition(): Implicitly marking parameter \$xmlDb\_table as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/xmlDb/xmlDb\_field.php on line 763

Deprecated: xmlDb\_index::validateDefinition(): Implicitly marking parameter \$xmlDb\_table as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/xmlDb/xmlDb\_index.php on line 339

Deprecated: database\_manager::check\_database\_schema():  
Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/ddl/database\_manager.php on line 958

Deprecated: file\_prepare\_draft\_area(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filelib.php on line 408

Deprecated: file\_rewrite\_pluginfile\_urls(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filelib.php on line 492

Deprecated: file\_save\_draft\_area\_files(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filelib.php on line 1107

Deprecated: file\_merge\_files\_from\_draft\_area\_into\_filearea():

Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filelib.php on line 2986

Deprecated: file\_storage::delete\_area\_files\_select(): Implicitly marking parameter \$params as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/file\_storage.php on line 917

Deprecated: file\_storage::create\_file\_from\_url(): Implicitly marking parameter \$options as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/file\_storage.php on line 1224

Deprecated: stored\_file::extract\_to\_pathname(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/stored\_file.php on line 542

Deprecated: stored\_file::extract\_to\_storage(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/stored\_file.php on line 560

Deprecated: file\_system::extract\_to\_pathname(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/file\_system.php on line 287

Deprecated: file\_system::extract\_to\_storage(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/file\_system.php on line 306

Deprecated: zip\_packer::archive\_to\_storage(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 56

Deprecated: zip\_packer::archive\_to\_pathname(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 98

Deprecated: zip\_packer::archive\_stored(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 176

Deprecated: zip\_packer::archive\_pathname(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 218

Deprecated: zip\_packer::extract\_to\_pathname(): Implicitly marking parameter \$onlyfiles as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 265

Deprecated: zip\_packer::extract\_to\_pathname(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 265

Deprecated: zip\_packer::extract\_to\_storage(): Implicitly marking parameter \$progress as nullable is deprecated, the explicit nullable

type must be used instead in /var/www/html/lib/filestorage/zip\_packer.php on line 425

Deprecated: file\_packer::archive\_to\_storage(): Implicitly

marking parameter \$progress as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/filestorage/file\_packer.php on line 62

Deprecated: file\_packer::archive\_to.pathname(): Implicitly

marking parameter \$progress as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/filestorage/file\_packer.php on line 87

Deprecated: file\_packer::extract\_to.pathname(): Implicitly

marking parameter \$onlyfiles as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/filestorage/file\_packer.php on line 101

Deprecated: file\_packer::extract\_to.pathname(): Implicitly

marking parameter \$progress as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/filestorage/file\_packer.php on line 101

Deprecated: file\_packer::extract\_to\_storage(): Implicitly

marking parameter \$progress as nullable is deprecated, the explicit

nullable type must be used instead in /var/www/html/lib/filestorage/file\_packer.php on line 117

Deprecated: cache\_helper::purge\_store(): Implicitly marking

parameter \$config as nullable is deprecated, the explicit nullable type

must be used instead in /var/www/html/cache/classes/helper.php on line 554

Deprecated: cache\_helper::warnings(): Implicitly marking

parameter \$stores as nullable is deprecated, the explicit nullable type  
must be used instead in /var/www/html/cache/classes/helper.php on line 859

Deprecated: GuzzleHttp\Promise\queue(): Implicitly marking  
parameter \$assign as nullable is deprecated, the explicit nullable type  
must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 24

Deprecated: GuzzleHttp\Promise\each(): Implicitly marking  
parameter \$onFulfilled as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 260

Deprecated: GuzzleHttp\Promise\each(): Implicitly marking  
parameter \$onRejected as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 260

Deprecated: GuzzleHttp\Promise\each\_limit(): Implicitly marking  
parameter \$onFulfilled as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 285

Deprecated: GuzzleHttp\Promise\each\_limit(): Implicitly marking  
parameter \$onRejected as nullable is deprecated, the explicit nullable  
type must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 285

Deprecated: GuzzleHttp\Promise\each\_limit\_all(): Implicitly  
marking parameter \$onFulfilled as nullable is deprecated, the explicit  
nullable type must be used instead in /var/www/html/lib/guzzlehttp/promises/src/functions.php on line 307

Deprecated: DI\create(): Implicitly marking parameter \$className as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/php-di/php-di/src/functions.php on line 33

Deprecated: DI\autowire(): Implicitly marking parameter \$className as nullable is deprecated, the explicit nullable type must be used instead in /var/www/html/lib/php-di/php-di/src/functions.php on line 46

## Installation

### Language

Choose a language

Please

choose a language for the installation. This language will also be used as the default language for the site, though it may be changed later.

### Language

"

¡Sí, te está funcionando! 🎉

Aunque veas una gran cantidad de mensajes de **Deprecated**, lo importante es que la instalación de Moodle ha avanzado y te está mostrando la pantalla para "Elegir un idioma".

---

## ⚠ ¿Qué significan los mensajes Deprecated?

Estos mensajes de advertencia (**Deprecated**) no son errores fatales que detengan la aplicación. Simplemente te están informando que:

1. Estás utilizando una versión de **PHP (probablemente PHP 8.x)** que es **más nueva** de lo que el código de Moodle espera.
2. El código de Moodle que está intentando ejecutar utiliza **funciones, constantes o sintaxis** que han sido **marcadas como obsoletas** (deprecated) en esa versión reciente de PHP y que serán eliminadas en futuras versiones de PHP.

## La solución:

Moodle HQ lanza versiones de Moodle que son compatibles con las versiones estables y recientes de PHP.

- Si estás usando **Moodle 4.x**, deberías asegurarte de que tu imagen de Moodle (como `moodlehq/moodle - php - apache :8.4` que mencionaste) sea la **última versión estable de Moodle**. Los desarrolladores de Moodle generalmente corrigen estas advertencias con cada nueva versión.
- En tu caso, al usar una imagen con **PHP 8.4** y una versión de Moodle que quizás no está 100% actualizada para ese PHP, aparecen muchas de estas advertencias.

**No tienes que detenerte.** Puedes y debes continuar con el proceso de instalación de Moodle haciendo clic en el botón para seleccionar el idioma.

## → Siguiente Paso

Ahora, debes **seleccionar el idioma** deseado para la instalación y luego continuar con la configuración de la base de datos y los ajustes del sitio Moodle.

**Continúa la instalación desde la pantalla que se muestra.** ¿Quieres que te guíe con los siguientes pasos de la instalación?

El directorio de los datos (`/var/www/moodledata`) no puede ser creado por el instalador.

¡Ese es un error común y crítico! Significa que el contenedor de Moodle no tiene los permisos necesarios para crear o escribir en el directorio de datos persistente que especificaste en tu `docker-compose.yml`.

Tu configuración de volúmenes para el servicio `moodle` es:

```
volumes:  
- ./moodle-code:/var/www/html  
- moodle_mariadb_data:/var/www/moodledata # <-- Este es el problema
```

El volumen `moodle_mariadb_data` se está creando como un **volumen con nombre (named volume)** que Docker gestiona. Para que el instalador de Moodle pueda escribir en él, debes asegurarte de que el usuario de Apache dentro del contenedor (`www-data`, UID 33) tenga los permisos adecuados.

# Solución: Configuración de Permisos para moodledata

Tienes dos formas principales de solucionar esto: la forma recomendada con la imagen de Moodle HQ o la forma manual de configurar los permisos en el volumen.

## Opción 1: (Recomendada) Usar el Script de Entrada del Contenedor

El error indica que el instalador de Moodle (el script `install.php`) no puede crear el directorio, pero la imagen `moodlehq/moodle-php-apache` está diseñada para manejar esto al iniciar el contenedor.

**El problema suele ser que el directorio `moodledata` ya existe, pero con permisos incorrectos.**

### 1. Elimina el Volumen Existente con Permisos Erróneos:

Asegúrate de que los contenedores estén detenidos y luego elimina el volumen `moodle_mariadb_data` para que se cree de nuevo de forma limpia:

```
docker compose down  
docker volume rm moodle_mariadb_data
```

### 2. Verifica la Configuración del Volumen:

Asegúrate de que tu `docker-compose.yml` use el **nombre del servicio** para el host de la base de datos (ya lo corrigiste, pero es bueno confirmarlo) y que los volúmenes estén bien definidos:

```
moodle:  
  # ...  
  environment:  
    # Asegúrate que el host es 'mariadb' (el nombre del servicio)  
    MOODLE_DATABASE_HOST: mariadb  
  volumes:  
    - ./moodle-code:/var/www/html  
    - moodle_mariadb_data:/var/www/moodledata # Usando named volume  
  # ...
```

### 3. Vuelve a Ejecutar:

```
docker compose up -d
```

Al recrear el volumen con el comando anterior, el **script de entrada** del contenedor de Moodle debería inicializarlo con los permisos correctos (propiedad del usuario `www-data`).

---

## Opción 2: (Si el problema persiste) Ajustar permisos manualmente

Si el problema persiste al usar el volumen con nombre, es posible que estés en un entorno (como SELinux en CentOS/RHEL) que interfiere con los volúmenes de Docker. La solución es ejecutar un comando *dentro* del contenedor para cambiar la propiedad del directorio.

## 1. Inicia los contenedores:

```
docker compose up -d
```

## 2. Ejecuta el comando para corregir permisos:

Usa docker exec para ingresar al contenedor y cambiar la propiedad del directorio moodledata al usuario www-data (UID 33):

```
docker exec moodle chown -R www-data:www-data /var/www/moodledata
```

(moodle es el *container\_name* de tu servicio Moodle).

Después de ejecutar este comando, **actualiza la página del navegador** donde se mostraba el error de instalación. El instalador ahora debería tener permiso para escribir en el directorio y avanzar al siguiente paso.

¿Pudiste ejecutar el comando docker volume rm moodle\_mariadb\_data y luego intentaste de nuevo con docker compose up -d? Si no funciona, por favor, prueba la Opción 2.

He probado las dos opciones y no funciona. Te paso el .yaml "services":

mariadb:

```
image: mariadb:latest
```

```
container_name: mariadb-moodle
```

```
restart: always
```

```
environment:
```

```
MYSQL_ROOT_PASSWORD: rootpass
```

```
MYSQL_DATABASE: moodle
```

```
MYSQL_USER: cristobal
```

```
MYSQL_PASSWORD: 12345
```

volumes:

```
- mariadb_data:/var/lib/mysql
```

networks:

```
- moodle_network
```

moodle:

```
image: moodlehq/moodle-php-apache:8.4
```

```
container_name: moodle
```

```
restart: always
```

```
depends_on:
```

```
  - mariadb
```

```
ports:
```

```
  - "8083:80" # HTTP
```

```
  - "8444:443" # HTTPS
```

```
environment:
```

```
MOODLE_DATABASE_TYPE: mysqli
```

```
MOODLE_DATABASE_HOST: mariadb
```

```
MOODLE_DATABASE_NAME: moodle
```

```
MOODLE_DATABASE_USER: cristobal
```

```
MOODLE_DATABASE_PASSWORD: 12345
```

```
MOODLE_USERNAME: cristobal
```

```
MOODLE_PASSWORD: 12345
```

```
MOODLE_SITE_NAME: "Moodle SUAREZ1"
```

```
MOODLE_EMAIL: admin@suarez1.abad2
```

```
volumes:
```

```
  - ./moodle-code:/var/www/html
```

```
  - moodle_mariadb_data:/var/www/moodledata
```

```
networks:
```

```
  - moodle_network
```

```
volumes:
```

```
mariadb_data:
```

```
moodle_data:
```

moodle\_mariadb\_data:

networks:

moodle\_network:

Gracias por compartir tu docker-compose.yml actualizado. Se ve que has incorporado todas las correcciones anteriores (imagen correcta, host de DB correcto y mapeo de volumen local).

El error "**El directorio de los datos (/var/www/moodledata) no puede ser creado por el instalador.**" persiste porque, aunque eliminaste el volumen, el problema se debe a que la **carpeta de la aplicación Moodle** no es legible. Además, el volumen de datos en Moodle HQ necesita una configuración adicional.

Aquí tienes un plan de ataque de dos pasos para resolver el problema de permisos de manera definitiva:

---

## 1. Corregir la Persistencia de Permisos en el Volumen moodledata

El problema principal es que el usuario dentro del contenedor (www-data, UID 33) no puede escribir en el volumen gestionado por Docker (moodle\_mariadb\_data).

Aunque el comando chown temporalmente funciona, la forma más robusta y *oficialmente recomendada* para el contenedor moodlehq es usar una imagen auxiliar que configure los permisos del volumen de datos *antes* de que Moodle lo intente usar.

### Solución 1: Usar la Imagen Oficial de moodle-docker

Te recomiendo migrar a la configuración oficial que usa la imagen **moodle-docker** (un wrapper para desarrollo) o la imagen de Moodle que ya incluye todo preconfigurado, como la de Bitnami, ya que la imagen moodle-php-apache está diseñada para escenarios más avanzados.

**Opción alternativa (y más simple) que usa la configuración que tienes:**

Vamos a usar una solución de permisos de Docker Compose conocida: montar la carpeta *mientras existe*.

#### 1. Detén los contenedores:

```
docker compose down
```

#### 2. Asegúrate de que tu carpeta de código local tenga permisos correctos:

```
# Asume que estás en el mismo directorio que docker-compose.yml
sudo chown -R 33:33 ./moodle-code
```

*Esto le da al usuario www-data (UID 33) del contenedor los permisos para leer tu código local.*

### 3. Ejecuta un servicio temporal para inicializar el directorio moodledata:

Agrega el siguiente servicio *temporal* a tu docker-compose.yml para crear la carpeta moodledata con los permisos correctos en el volumen con nombre.

```
# AÑADE ESTE SERVICIO TEMPORAL ARRIBA DEL SERVICIO 'moodle'  
fix_permissions:  
    image: moodlehq/moodle-php-apache:8.4  
    container_name: fix_permissions  
    volumes:  
        - moodle_mariadb_data:/var/www/moodledata  
    command: chown -R www-data:www-data /var/www/moodledata  
    depends_on:  
        - mariadb
```

- **Propósito:** El servicio `fix_permissions` simplemente arranca, cambia la propiedad del volumen `moodle_mariadb_data` al usuario `www-data` (UID 33) y luego se detiene inmediatamente.

### 4. Actualiza el servicio moodle:

Añade la dependencia del nuevo servicio:

```
# ... en el servicio 'moodle'  
restart: always  
depends_on:  
    - mariadb  
    - fix_permissions # <-- AÑADE ESTA DEPENDENCIA  
ports:  
# ...
```

### 5. Ejecuta Docker Compose:

```
docker compose up -d
```

El nuevo servicio `fix_permissions` se ejecutará primero, asegurando que el directorio `/var/www/moodledata` en el volumen gestionado sea propiedad del usuario correcto antes de que el servicio `moodle` intente acceder a él.

---

## 2. ! Advertencia de Volúmenes Redundantes

He notado que en la sección `volumes` : al final de tu archivo tienes un volumen llamado `moodle_data` que no se está utilizando en el servicio `moodle`.

```
volumes:  
    mariadb_data:
```

```
moodle_data:           # <-- No se usa
moodle_mariadb_data:
```

**Acción:** Es seguro **eliminar** moodle\_data: de la sección volumes: ya que no tiene ninguna función. Tu volumen de aplicación es ahora el mapeo local ./moodle-code.

---

### Resumen de la Acción a Realizar:

1. Añade el servicio **fix\_permissions** temporal a tu .yml.
2. Añade la dependencia **fix\_permissions** al servicio moodle.
3. Ejecuta **sudo chown -R 33:33 ./moodle-code** en tu host.
4. Ejecuta **docker compose up -d**.

Este proceso debería solucionar todos los problemas de permisos del directorio de datos. Una vez que Moodle se instale, puedes eliminar el servicio fix\_permissions de tu .yml.

¿Quieres que te proporcione el archivo docker-compose.yml final y limpio con el servicio fix\_permissions añadido?

Si, por favor

¡Por supuesto! Aquí tienes el archivo docker-compose.yml corregido. He añadido el servicio temporal **fix\_permissions** para garantizar que el directorio /var/www/moodledata tenga los permisos correctos antes de que Moodle intente acceder a él.

También he eliminado el volumen no utilizado moodle\_data: de la sección inferior.

```
services:
  mariadb:
    image: mariadb:latest
    container_name: mariadb-moodle
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: rootpass
      MYSQL_DATABASE: moodle
      MYSQL_USER: cristobal
      MYSQL_PASSWORD: 12345
    volumes:
      - mariadb_data:/var/lib/mysql
  networks:
    - moodle_network

fix_permissions: # NUEVO SERVICIO TEMPORAL para corregir permisos
  image: moodlehq/moodle-php-apache:8.4
```

```

container_name: fix_permissions
volumes:
  - moodle_mariadb_data:/var/www/moodledata
command: chown -R www-data:www-data /var/www/moodledata
depends_on:
  - mariadb
networks:
  - moodle_network

moodle:
  image: moodlehq/moodle-php-apache:8.4
  container_name: moodle
  restart: always
  depends_on:
    - mariadb
    - fix_permissions # Depende de que los permisos se hayan corregido
  ports:
    - "8083:80"      # HTTP
    - "8444:443"     # HTTPS
  environment:
    MOODLE_DATABASE_TYPE: mysqli
    MOODLE_DATABASE_HOST: mariadb
    MOODLE_DATABASE_NAME: moodle
    MOODLE_DATABASE_USER: cristobal
    MOODLE_DATABASE_PASSWORD: 12345
    MOODLE_USERNAME: cristobal
    MOODLE_PASSWORD: 12345
    MOODLE_SITE_NAME: "Moodle SUAREZ1"
    MOODLE_EMAIL: admin@suarez1.abad2
  volumes:
    - ./moodle-code:/var/www/html
    - moodle_mariadb_data:/var/www/moodledata
  networks:
    - moodle_network

volumes:
  mariadb_data:
  moodle_mariadb_data: # Eliminado el volumen 'moodle_data' no utilizado

networks:
  moodle_network:

```

## Pasos Finales de Ejecución

Antes de ejecutar el nuevo archivo, sigue estos pasos finales:

1. **Guarda** el archivo con el nombre docker-compose.yml.
2. **Detén y elimina** los contenedores y volúmenes anteriores para asegurar una limpieza:

```
docker compose down -v
```

3. **Asegúrate de que el código de Moodle tenga permisos de lectura en el host:**

```
sudo chown -R 33:33 ./moodle-code
```

(*Esto evita el error "Forbidden" que tuviste antes.*)

4. **Ejecuta la nueva configuración:**

```
docker compose up -d
```

Ahora, cuando el servicio moodle se inicie, el volumen moodle\_mariadb\_data ya habrá sido inicializado y configurado con los permisos correctos por el servicio fix\_permissions, resolviendo el error del directorio de datos. Reinicia el navegador y la instalación debería continuar.

Me sigue dando el error "El directorio de los datos (/var/www/moodledata) no puede ser creado por el instalador."

El hecho de que el error persista incluso después de intentar las soluciones estándar (eliminar el volumen y usar el servicio fix\_permissions) sugiere un problema de **permisos más profundo** relacionado con cómo tu sistema operativo host (probablemente Linux) maneja la seguridad y los volúmenes de Docker, especialmente con **SELinux o AppArmor**.

Aquí tienes el plan de ataque final, que incluye la solución para **SELinux** y una forma alternativa de montar el volumen para saltarse los problemas de permisos.

---

## ⓧ Plan de Solución Final (SELinux y Montaje Explícito)

### 1. Corrección de Permisos de SELinux (Linux Host)

Si estás ejecutando Docker en un sistema como **CentOS, RHEL o Fedora**, **SELinux** está probablemente impidiendo que el contenedor modifique el volumen. Tienes dos opciones para resolver esto:

#### Opción A: Desactivar las restricciones de SELinux (Solo para entornos de prueba)

Añade la etiqueta :z o :Z al mapeo del volumen para indicarle a Docker que ajuste automáticamente el contexto de SELinux. Usaremos :z en el volumen de datos para que el directorio sea compartido entre contenedores.

Cambia la línea del volumen de datos en el servicio moodle a:

```
volumes:  
- ./moodle-code:/var/www/html  
- moodle_mariadb_data:/var/www/moodledata:z # <-- ¡Añade :z aquí!
```

### Opción B: Mapear una carpeta local en lugar de un volumen con nombre

Si la Opción A falla, es más fiable usar una **carpeta local** para el moodledata en lugar de un volumen con nombre gestionado por Docker, y luego aplicar manualmente los permisos al host.

**Pasos:**

#### 1. Detén y elimina todo lo anterior:

```
docker compose down -v
```

#### 2. Crea un directorio local para los datos:

```
mkdir moodledata_local
```

#### 3. Ajusta los permisos de la carpeta local:

```
sudo chown -R 33:33 moodledata_local
```

*Esto le da al usuario www-data (UID 33) del contenedor propiedad sobre la carpeta.*

#### 4. Actualiza el `docker-compose.yml` (¡Mira los cambios abajo!).

---

## 2. docker-compose.yml Corregido con Mapeo Local

Este es el archivo modificado para usar el mapeo de una carpeta local (`./moodledata_local`) y eliminar el volumen gestionado por Docker que estaba causando problemas.

```
services:  
mariadb:  
# ... (Sin cambios)  
image: mariadb:latest  
container_name: mariadb-moodle  
restart: always  
environment:  
    MYSQL_ROOT_PASSWORD: rootpass  
    MYSQL_DATABASE: moodle  
    MYSQL_USER: cristobal  
    MYSQL_PASSWORD: 12345  
volumes:
```

```

    - mariadb_data:/var/lib/mysql
networks:
    - moodle_network

moodle:
    image: moodlehq/moodle-php-apache:8.4
    container_name: moodle
    restart: always
    depends_on:
        - mariadb
    ports:
        - "8083:80"      # HTTP
        - "8444:443"     # HTTPS
    environment:
        MOODLE_DATABASE_TYPE: mysqli
        MOODLE_DATABASE_HOST: mariadb
        MOODLE_DATABASE_NAME: moodle
        MOODLE_DATABASE_USER: cristobal
        MOODLE_DATABASE_PASSWORD: 12345
        MOODLE_USERNAME: cristobal
        MOODLE_PASSWORD: 12345
        MOODLE_SITE_NAME: "Moodle SUAREZ1"
        MOODLE_EMAIL: admin@suarez1.abad2
    volumes:
        - ./moodle-code:/var/www/html
        - ./moodledata_local:/var/www/moodledata  # <-- Mapeo a carpeta local
networks:
    - moodle_network

volumes:
    mariadb_data:
        # Eliminados moodle_data y moodle_mariadb_data

networks:
    moodle_network:

```

### 3. Ejecución Final

1. Sigue los pasos 1, 2 y 3 de la sección anterior (Crear y dar permisos a moodledata\_local).
2. **Guarda** el nuevo docker-compose.yml.
3. Ejecuta:

```
docker compose up -d
```

El mapeo directo a una carpeta local con permisos correctos en el host debería eludir cualquier restricción que impida al instalador de Moodle escribir en el directorio de datos.

Si después de estos pasos, sigues viendo el error de permisos, necesitarás revisar la configuración de seguridad de tu sistema operativo (como SELinux en modo estricto). Si es así, ¿puedes indicarme qué sistema operativo utilizas?

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/install.php on line 183

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/lib/setuplib.php on line 38

Deprecated: Constant E\_STRICT is deprecated in /var/www/html/lib/setuplib.php on line 40

Warning: mkdir(): No space left on device in /var/www/html/lib/installlib.php on line 111

¡Hemos pasado del problema de permisos a un problema de **espacio en disco!**

El mensaje de advertencia más importante aquí, que anula todos los problemas anteriores de permisos, es:

**Warning: mkdir(): No space left on device in  
/var/www/html/lib/installlib.php on line 111**

Esto significa que, aunque el instalador de Moodle tiene permiso para crear el directorio de datos (/var/www/moodledata), el sistema operativo (tu máquina host o el sistema de archivos de Docker) está informando que **no queda espacio disponible** en el dispositivo de almacenamiento.

## Solución: Liberar Espacio en Disco

El contenedor de Moodle está intentando escribir archivos de datos o de caché, pero el sistema de archivos subyacente está lleno. Debes revisar y liberar espacio en tu sistema host o en Docker.

### 1. Revisar el Espacio en Disco del Host

Verifica cuánto espacio queda en la unidad de tu máquina que aloja los volúmenes de Docker:

```
df -h
```

*Si este comando muestra que el uso de disco está cerca del 100% para la partición donde Docker almacena sus datos, debes liberar espacio en tu host.*

## 2. Limpiar Docker (Si el Host No Está Lleno)

A menudo, Docker acumula imágenes, contenedores y volúmenes antiguos que pueden ocupar gigabytes. Puedes usar un comando de limpieza general:

```
docker system prune -a
```

Este comando:

- Detiene y elimina todos los contenedores que no se están ejecutando.
- Elimina todas las redes no utilizadas.
- Elimina todas las imágenes que no están asociadas a ningún contenedor existente.
- Elimina el caché de compilación (build cache).
- **Advertencia:** Eliminará **todos** los datos no utilizados, incluidos los volúmenes que no estén asociados a un contenedor. Asegúrate de que no tienes datos importantes en contenedores o volúmenes detenidos que necesites conservar.

## 3. Eliminar Volúmenes Antiguos de Moodle/MariaDB

Asegúrate de que no quedan residuos de intentos de instalación anteriores.

```
# Detener los contenedores actuales
docker compose down
# Eliminar los volúmenes sin usar
docker volume prune
```

## Reintentar la Instalación

Una vez que hayas liberado espacio suficiente en tu máquina host y/o limpiado Docker:

1. **Asegúrate de que estás usando la configuración más reciente** (la que utiliza la carpeta local `./moodledata_local` para evitar problemas de permisos):

```
volumes:
  - ./moodle-code:/var/www/html
  - ./moodledata_local:/var/www/moodledata
```

2. **Vuelve a ejecutar Docker Compose:**

```
docker compose up -d
```

3. **Actualiza la página del navegador.**

El instalador de Moodle ahora debería tener la capacidad de escribir los archivos de datos y completar la configuración.

¿Pudiste ejecutar el comando df -h para verificar si el espacio en disco es realmente el problema?

C