Moodle

Moodle is a free and open-source learning management system used for blended learning, distance education, flipped classroom and other online learning projects.

- Moodle Installation

Moodle Installation

*In this article, we will install **Moodle** directly on Raspberry Pi OS using the **LAMP** (**L**inux, **A**pache, **M**ariaDB, **P**HP) stack.*

> Why not using Docker?
>
> The bare metal setup gives us more **fined-grained control** over our Moodle installation and let Moodle to use more RPi resources.

? REQUIREMENTS

Requirement	Recommendation
Raspberry Pi Model	Pi 4 with 2GB+ RAM (4GB+ ideal)
OS	Raspberry Pi OS 64-bit Lite
Storage	SSD recommended (faster & safer)
Internet Access	Required during install
Domain (optional)	For HTTPS

?? INSTALLATION

? 1. Update your System

```
sudo apt update && sudo apt upgrade -y
```

? 2. Install Apache

```
sudo apt install apache2 -y
```

Enable and start Apache:

```
sudo systemctl enable apache2
sudo systemctl start apache2
```

Check: Visit `http://<raspberry-pi-ip>` — you should see the Apache welcome page.

? 3. Install MariaDB (MySQL-compatible)

```
sudo apt install mariadb-server -y
```

Secure it:

```
sudo mysql_secure_installation
```

You'll be asked to:

- Set root password
- Remove anonymous users
- Disallow remote root login
- Remove test DB
- Reload privileges

 Answer "yes" to all.

? 4. Create Moodle Database

```
sudo mariadb
```

Inside the MariaDB shell:

```
CREATE DATABASE moodle DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
CREATE USER 'moodleuser'@'localhost' IDENTIFIED BY 'moodlepass';
GRANT ALL PRIVILEGES ON moodle.* TO 'moodleuser'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

? 5. Install PHP & Extensions

```
sudo apt install php php-cli php-fpm php-mysql php-zip php-gd php-xml php-curl php-mbstring
php-soap php-intl php-xmlrpc php-bcmath php-ldap php-readline -y
```

Set timezone in `php.ini`:

```
sudo nano /etc/php/*/apache2/php.ini
```

Search for `date.timezone` and set your timezone (e.g., `Europe/Paris`):

```
date.timezone = Europe/Paris
```

? 6. Download Moodle

```
cd /var/www/html
sudo rm index.html  # remove Apache default page
sudo apt install git -y
sudo git clone -b MOODLE_402_STABLE https://github.com/moodle/moodle.git
sudo chown -R www-data:www-data moodle
sudo chmod -R 755 moodle
```

Create a Moodle data directory (not web-accessible):

```
sudo mkdir /var/moodledata
sudo chown -R www-data:www-data /var/moodledata
sudo chmod -R 755 /var/moodledata
```

? 7. Configure Apache for Moodle

Create a new config:

```
sudo nano /etc/apache2/sites-available/moodle.conf
```

Paste this:

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    DocumentRoot /var/www/html/moodle
    ServerName moodle.local

    <Directory /var/www/html/moodle>
        Options +FollowSymlinks
        AllowOverride All
        Require all granted
    </Directory>
```

```
    ErrorLog ${APACHE_LOG_DIR}/moodle_error.log
    CustomLog ${APACHE_LOG_DIR}/moodle_access.log combined
</VirtualHost>
```

Enable it:

```
sudo a2ensite moodle.conf
sudo a2enmod rewrite
sudo systemctl reload apache2
```

Edit your hosts file (optional for local use):

```
echo "127.0.0.1 moodle.local" | sudo tee -a /etc/hosts
```

? 8. Run the Moodle Web Installer

Visit:

```
http://<raspberrypi-ip>/moodle
```

Or if using domain:

```
http://moodle.local
```

Follow the wizard:

- Select language
- Confirm paths
- Choose MariaDB (MySQL)
- Use: DB: `moodle` User: `moodleuser` Password: `moodlepass`
- Set admin account

Let it finish setting up the database (takes a few mins).

? 9. Enable HTTPS (Optional but Recommended)

Install Caddy or Let's Encrypt for HTTPS.

Example with Caddy (much easier than Certbot):

```
sudo apt install -y debian-keyring debian-archive-keyring curl
curl -1sLf 'https://dl.cloudsmith.io/public/caddy/stable/gpg.key' | sudo gpg --dearmor -o
/usr/share/keyrings/caddy-stable-archive-keyring.gpg
curl -1sLf 'https://dl.cloudsmith.io/public/caddy/stable/debian.deb.txt' | sudo tee
/etc/apt/sources.list.d/caddy-stable.list
sudo apt update
sudo apt install caddy
```

Create a `Caddyfile`:

```
sudo nano /etc/caddy/Caddyfile
```

```
moodle.yourdomain.com {
    root * /var/www/html/moodle
    php_fastcgi unix//run/php/php8.2-fpm.sock
    file_server
    encode gzip
}
```

Then reload:

```
sudo systemctl restart caddy
```

Done — you're now running Moodle over HTTPS with a free Let's Encrypt cert!

? EXTRAS

Feature	How
Backups	Use `rsync` or `cron` to back up `/var/www/html/moodle`, `/var/moodledata`, and database
Email (SMTP)	Configure `Site admin > Server > Email`
Themes & Plugins	Use built-in plugin installer
Performance	Enable caching (Redis or Memcached), configure cron

---

? Summary

Step	Done

Apache & PHP installed	☐
MariaDB & DB created	☐
Moodle downloaded	☐
Apache config created	☐
Moodle web installer run	☐
HTTPS enabled	☐