

RaspberryPi-LAMP-Wordpress-Setup

Dieses Dokument zeigt eine Mitschrift zur Ausarbeitung eines Arbeitsauftrage/Laborauftrages im Unterrichtsfach **Betriebssysteme & Labor**.

Inhalt

1. Zielsetzung	2
2. Geräteübersicht	2
2.1 Netzwerkeinstellungen (Hardware)	2
2.2 User & Zugangsdaten (Software)	2
2.3 Quellen	2
3. Installation OS	3
3.1 Erstinbetriebnahme & Konfiguration OS	4
4. Apache einrichten	4
4.1 PHP installieren & einrichten	4
5. MariaDB	5
6. phpMyAdmin einrichten	6
6.1 WebGUI	6
7. Wordpress Installation	7
8. Root-Rechte vergeben	8

1. Zielsetzung

Aufsetzen und Konfiguration einer LAMP-Umgebung auf einem RASPBERRY-PI

2. Geräteübersicht

	Device (# & Name)	Rolle	Hypervisor (VM)	Betriebssystem (OS)	IP-Adresse (LAN)
1	LENOVO E16	Client	nein	WIN11	192.168.1.222
2	Raspi 400	Server	nein	Raspberry Pi OS light 64bit)	192.168.1.15
3					

2.1 Netzwerkeinstellungen (Hardware)

Devices	Device #	# 1	# 2	# 3	# 4	# 5
	IP Adr.	192.168.1.222	192.168.1.15			
	DHCP/static	DHCP	DHCP			
	SUB	255.255.255.0				
	Gateway	192.168.1.1	192.168.1.1			
	DNS					
	Mac					

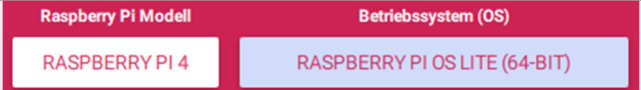
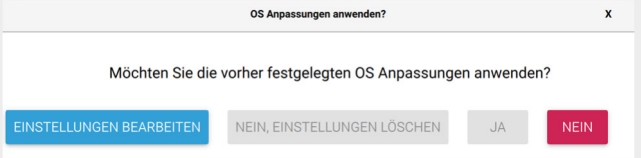
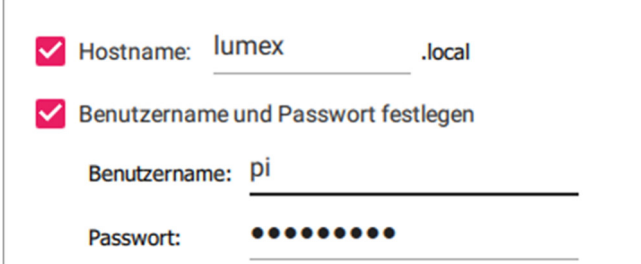
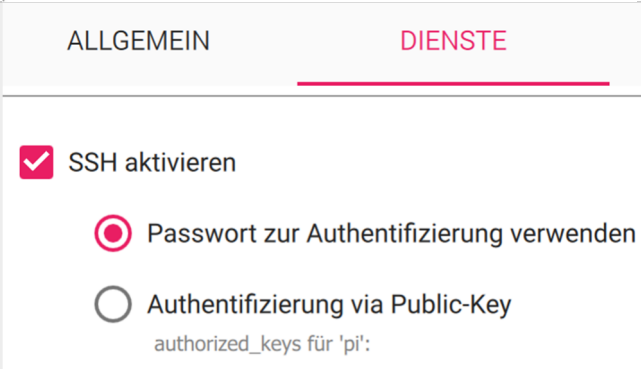
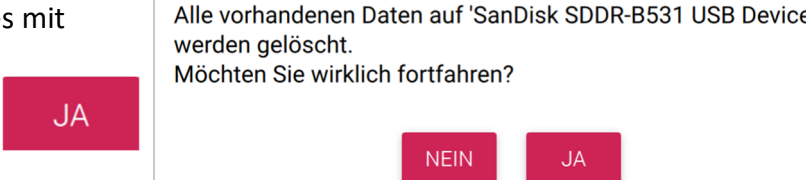
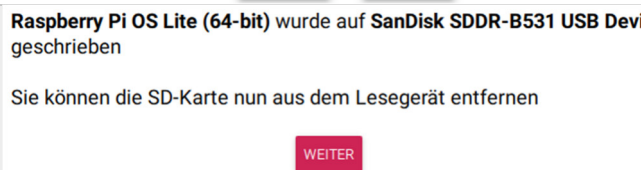
2.2 User & Zugangsdaten (Software)

Client	Geräte #	Anwendung	Benutzername	Passwort	Gerätename	domain (AD)
Server	Geräte #	Anwendung	Benutzername	Passwort	Hostname	
	2	Raspi OS	pi	Aa_123456	lumex	
	2	MariaDB	lumex	123456	pi	
	2	apache2	pi	Aa_123456		
	2	phpmyadmin	phpmyadmin	Aa_123456		

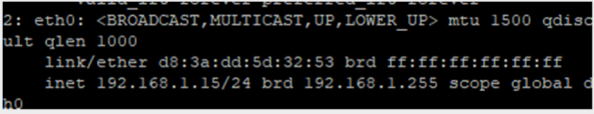
2.3 Quellen

https://projects.raspberrypi.org/de-DE/projects/lamp-web-server-with-wordpress/2
https://techgeeks.de/raspberry-pi-als-webserver-mit-wordpress/
https://www.ionos.at/digitalguide/server/konfiguration/einen-raspberry-pi-webserver-mit-lamp-einrichten/

3. Installation OS

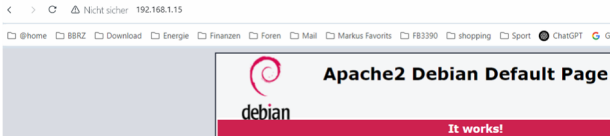
Erklärung	Beispiel
Auswahl eines geeigneten Betriebssystems	
Auswahl - PI-Modell - Betriebssystem - Datenträger	
Einstellungen bearbeiten	
Unter „Allgemein“ Eingabe des Hostnames Festlegen von Benutzernamen & Passwort	
Unter „Dienste“ Aktivierung von SSH	
Bestätigung des Schreibvorganges mit	
Wurde der Schreibvorgang beendet, kann die SD-Karte in den Raspi gesteckt und das System gestartet werden.	

3.1 Erstinbetriebnahme & Konfiguration OS

Erklärung	Beispiel
Nach dem die SD-Karte beschrieben wurde, kann der Raspi gebootet werden. Um ihn per PuTTY zu erreichen, wird mit Hilfe eines Netzwerkscanners diese ausfindig gemacht.	
sudo apt-get update, sudo apt-get upgrade	update und upgrade durchgeführt

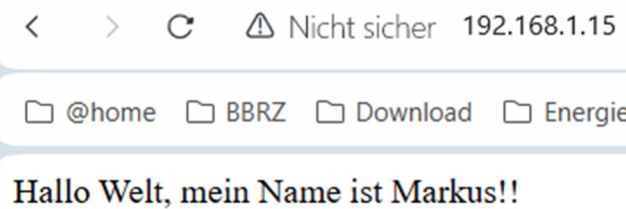
4. Apache einrichten

Die Installation des Apache HTTP Servers mit einem Befehl

Syntax, Eingabe	
sudo apt-get install apache2 -y	Download & Installation von Apache2
Installation erfolgreich!	
sudo apt-get install php -y	

4.1 PHP installieren & einrichten

PHP ist ein Präprozessor: Es ist Code der ausgeführt wird, wenn der Server eine Anforderung für eine Webseite über einen Webbrowser empfängt. Es ermittelt, was auf der Seite angezeigt werden soll, und sendet diese Seite dann an den Browser.

Syntax, Input																	
sudo apt-get install php php-mysql -y	Download & Installation von php																
sudo nano /var/www/html/phpinfo.php	neue Datei und benennen diese „phpinfo.php“																
Der Dateiinhalte sieht folgendermaßen aus:	<pre><?php phpinfo(); ?></pre> PHP Version 8.2.26																
Mit Strg + x, y und Enter abspeichern																	
sudo nano index.php	Erstellt eine Datei mit dem Namen „index.php“ und öffnet sie im nano-Editor!																
Im editor machen wir folgenden Eintrag und speichern sie! (Strg „O“ & „X“	<pre><?php echo "Hallo Welt"; ?></pre>																
sudo rm index.html	Löscht index.html, weil es Vorrang vor index.php hat!																
<table border="1"> <tr><td>System</td><td>Linux lumex 6.6.51+rpt-rpi-v8 #</td></tr> <tr><td>Build Date</td><td>Nov 25 2024 17:21:51</td></tr> <tr><td>Build System</td><td>Linux</td></tr> <tr><td>Server API</td><td>Apache 2.0 Handler</td></tr> <tr><td>Virtual Directory Support</td><td>disabled</td></tr> <tr><td>Configuration File (php.ini) Path</td><td>/etc/php/8.2/apache2</td></tr> <tr><td>Loaded Configuration File</td><td>/etc/php/8.2/apache2/php.ini</td></tr> <tr><td>Scan this dir for additional .ini files</td><td>/etc/php/8.2/apache2/conf.d</td></tr> </table>	System	Linux lumex 6.6.51+rpt-rpi-v8 #	Build Date	Nov 25 2024 17:21:51	Build System	Linux	Server API	Apache 2.0 Handler	Virtual Directory Support	disabled	Configuration File (php.ini) Path	/etc/php/8.2/apache2	Loaded Configuration File	/etc/php/8.2/apache2/php.ini	Scan this dir for additional .ini files	/etc/php/8.2/apache2/conf.d	
System	Linux lumex 6.6.51+rpt-rpi-v8 #																
Build Date	Nov 25 2024 17:21:51																
Build System	Linux																
Server API	Apache 2.0 Handler																
Virtual Directory Support	disabled																
Configuration File (php.ini) Path	/etc/php/8.2/apache2																
Loaded Configuration File	/etc/php/8.2/apache2/php.ini																
Scan this dir for additional .ini files	/etc/php/8.2/apache2/conf.d																

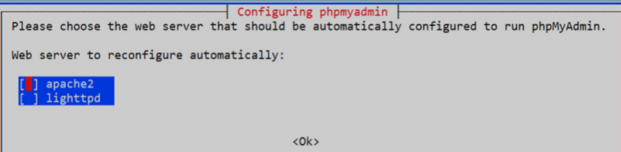
5. MariaDB

Mit MySQL legen Sie die Datenbank des Webservers an.

Syntax, Input	
<code>sudo apt install mariadb-server</code>	Installiert den MariaDB-Datenbankserver.
<code>sudo apt install php-mysql</code>	Installiert das PHP-MySQL-Modul, das die Verbindung zwischen PHP und MariaDB ermöglicht.
<code>sudo mysql -u root -p</code>	Meldet sich als root-Benutzer am MariaDB-Server an. Es wird ein Passwort abgefragt. <pre>pi@lumex:~ \$ sudo mysql -u root -p Enter password:</pre>
Nach der Eingabe des selbst gewählten Passwortes befindet man sich in der DB	<pre>Enter password: Welcome to the MariaDB monitor. Commands end with ; or \g. Your MariaDB connection id is 31 Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12 Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement. MariaDB [(none)]></pre>
Strg + D	Datenbank verlassen =Exit
<code>MariaDB [(none)]> CREATE USER 'lumex'@'localhost' IDENTIFIED BY '123456';</code>	Neuen Benutzer „lumex“ am localhost anlegen
<code>MariaDB [(none)]> SET PASSWORD [FOR lumex] = 123456</code>	Passwort für User „lumex“ gesetzt = 123456
<code>MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'lumex'@'localhost';</code>	ALL PRIVILEGES: Gewährt dem Benutzer alle verfügbaren Rechte. *.*: Bedeutet alle Datenbanken (*) und alle Tabellen (*). 'lumex'@'%': Der Benutzername 'lumex' und das Host-Muster '%', das alle Hosts erlaubt. <pre>MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'lumex'@'localhost'; Query OK, 0 rows affected (0.003 sec)</pre>
<code>FLUSH PRIVILEGES;</code>	Nach dem Erteilen der Privilegien ist es wichtig, die Änderungen mit dem Befehl zu übernehmen! WICHTIG!

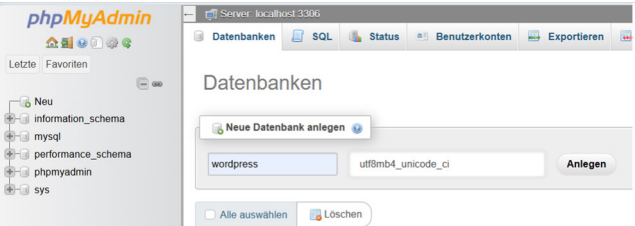
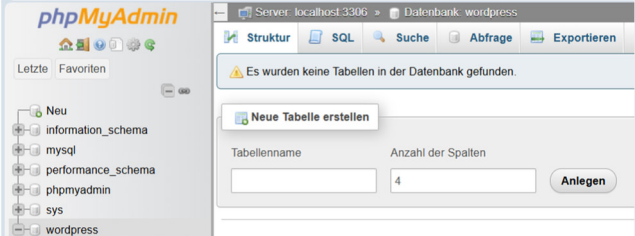
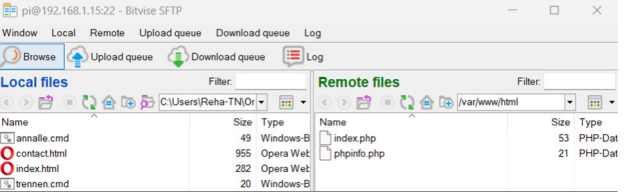
6. phpMyAdmin einrichten

Für die Administration von MySQL bietet sich die Nutzung von phpMyAdmin an. Mithilfe dieser freien Software verwalten Sie die Datenbank ganz einfach über eine grafische Benutzeroberfläche im Browser.

sudo apt-get install phpmyadmin Auswahl „apache2“	
Im Folgenden Fenster werden Sie gefragt, ob die phpMyAdmin-Verwaltungsdatenbank installiert werden soll – was Sie bejahen. Daraufhin bestimmen Sie noch ein Passwort für phpMyAdmin → fertiggestellt.	
sudo nano /etc/apache2/apache2.conf	phpMyAdmin mit dem Apache Webserver verbinden
ans Ende der Konfigurationsdatei und eine neue Zeile eingeben: Include /etc/phpmyadmin/apache.conf	<pre># Include the virtual host configurations: IncludeOptional sites-enabled/*.conf Include /etc/phpmyadmin/apache.conf</pre>
„Strg“ + „O“ → speichern der Datei „Strg“ + „X“ → schließen der Datei	
/etc/init.d/apache2 restart	Neustart des Dienstes <pre>==== AUTHENTICATION COMPLETE ==== pi@lumex:~ \$</pre>

6.1 WebGUI

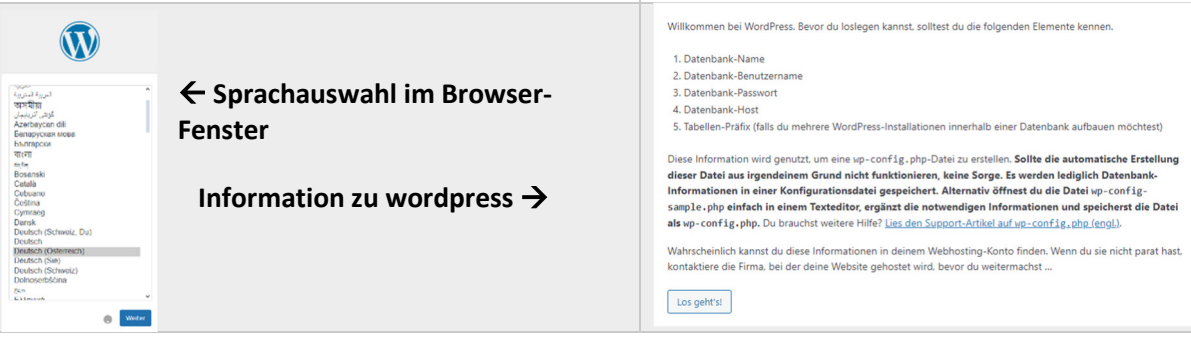
Aktion	
Eingabe in einen Web-Browser: http://192.168.1.15/phpmyadmin/ Sprach-Setup & Erstanmeldung User: lumex Passwort: 123456	

Um eine neue Datenbank anzulegen, klickt Ihr im linken Bereich auf „Neu“. Danach muss ein Datenbankname vergeben werden, z.B. „wordpress“,	
Jetzt sollte die Datenbank im linken Bereich zu sehen sein.	
löschen der Datei „index.html“ per Bitvise SFTP unter /var/www/html oder per Terminal <pre>sudo rm /var/www/html/index.html</pre> durch bearbeiten der config-Datei von apache2 wird phpMyAdmin mit dem Apache Webserver verbunden. <pre>sudo nano /etc/apache2/apache2.conf</pre> <pre>/etc/init.d/apache2 restart</pre>	 <pre># Include the virtual host configurations: IncludeOptional sites-enabled/*.conf Include /etc/phpmyadmin/apache.conf</pre> Eintrag in der letzten Zeile: Include /etc/phpmyadmin/apache.conf Neustart des Systems durchführen

7. Wordpress Installation

Wenn alle Dateien kopiert wurden, wechselt Ihr in den Browser und loggt Euch in phpMyAdmin ein, um eine neue Datenbank anzulegen.

Syntax	Funktion
Installation	
<pre>sudo wget https://wordpress.org/latest.zip -O /var/www/html/wordpress.zip</pre>	Lädt die neueste Version von WordPress als ZIP-Datei herunter und speichert sie im Webserver-Verzeichnis.
<pre>sudo wget https://wordpress.org/latest.tar.gz</pre>	Alternativ kann WordPress als TAR.GZ-Datei heruntergeladen werden.
<pre>sudo tar xzf latest.tar.gz</pre>	Entpackt die heruntergeladene TAR.GZ-Datei.
<pre>sudo unzip wordpress.zip</pre>	Entpackt die heruntergeladene ZIP-Datei.
<pre>cd /var/www/html</pre>	Wechseln in das Webserver-Verzeichnis, um die WordPress-Dateien zu verwalten.
<pre>sudo mv wordpress/* .</pre>	Verschiebt den Inhalt des WordPress-Ordners ins Webserver-Root-Verzeichnis.
<pre>sudo chown -R www-data: .</pre>	Ändert die Eigentümerschaft der Dateien und Ordner auf den Webserver-Benutzer www-data, damit der Webserver korrekt auf die Dateien zugreifen kann.

sudo service apache2 restart	Startet den Apache-Webserver neu, um sicherzustellen, dass alle Änderungen angewendet werden.															
Abschließen der WordPress-Installation																
Die WordPress-Installationsseite wird im Browser aufgerufen, z.B. http://192.168.0.45/index.php , um die Installation abzuschließen.																
sudo rm index.html	Entfernt die Standard-index.html-Datei, die möglicherweise im Weg ist.															
																
Unten solltest du die Details deiner Datenbank-Verbindung eintragen. Wenn du dir nicht sicher bist, wende dich an dein Webhosting-Unternehmen. <table border="1"> <tr> <td>Datenbank-Name</td> <td> wordpress </td> <td>Datenbankname:</td> </tr> <tr> <td>Benutzername</td> <td> Benutzername </td> <td>Benutzername: lumex</td> </tr> <tr> <td>Passwort</td> <td> Passwort Anzeigen</td> <td>Passwort: 123456</td> </tr> <tr> <td>Datenbank-Host</td> <td> localhost </td> <td>Datenbank-Host: localhost</td> </tr> <tr> <td>Tabellen-Präfix</td> <td> wp_ </td> <td>Tabellen-Präfix: wp_</td> </tr> </table> Senden		Datenbank-Name	wordpress	Datenbankname:	Benutzername	Benutzername	Benutzername: lumex	Passwort	Passwort Anzeigen	Passwort: 123456	Datenbank-Host	localhost	Datenbank-Host: localhost	Tabellen-Präfix	wp_	Tabellen-Präfix: wp_
Datenbank-Name	wordpress	Datenbankname:														
Benutzername	Benutzername	Benutzername: lumex														
Passwort	Passwort Anzeigen	Passwort: 123456														
Datenbank-Host	localhost	Datenbank-Host: localhost														
Tabellen-Präfix	wp_	Tabellen-Präfix: wp_														
Nach der Installation kann der Administrator sich unter http://192.168.0.45/wp-admin/ anmelden, um die Website zu verwalten.																

8. Root-Rechte vergeben

Syntax

```
sudo groupadd www
sudo adduser pi www
sudo chgrp www /var/www/html
sudo chmod g+w /var/www/html
sudo find /var/www/html -type d -exec chmod g+s {} \;
```