

## **Realisierung**

---

# Realisierung Projektarbeit M158/M169

**Version 1.0**

Autor des Dokuments	Fabian Peter, David Bürge, Ben Davatz	Erstellt am	27.05.2024
Dateiname	Realisierung.pdf		
Seitenanzahl	19		

## Historie der Dokumentversionen

Version	Datum	Autor	Änderungsgrund / Bemerkungen
0.1	22.05.2024	David Bürge	Ersterstellung
0.2	23.05.2024	Fabian Peter	Anpassung der Formatierung
0.3	29.05.2024	David Bürge	Abfüllen der Arbeitsschritte
04	01.06.2024	David Bürge	Ergänzung weiterer Arbeitsschritte
0.4	02.06.2024	Fabian Peter	Verbesserungen

**Tabelle 1: Dokumentversionen**

## Inhaltsverzeichnis

Historie der Dokumentversionen.....	2
1 Einleitung .....	4
1.1 Auftrag .....	4
1.2 Anforderungen .....	4
1.2.1 Muss-Anforderungen .....	4
2 Realisierung.....	5
2.1 Bestehende Umgebung auf Port 8080 umstellen.....	5
2.1.1 Moodle auf Port 8080 Umstellen .....	5
2.1.2 Umstellung Apache2 zu Port 8080 .....	6
2.1.3 Sites-enabled auf Port 8080 umstellen .....	6
2.1.4 Kennzeichnung der veralteten Umgebung .....	7
2.1.5 Neustart des Services .....	8
2.2 Sichern der Daten aus alter Umgebung .....	9
2.2.1 Erstellung eines SQL-Dumps .....	9
2.2.2 Anpassung des SQL-Dumps .....	9
2.2.3 Ergänzung folgender Befehle .....	9
2.2.4 Sichern der Moodle Daten .....	10
2.3 Migration der Moodle Umgebung .....	11
2.3.1 Erklärung des Docker-Compose .....	11
2.3.2 Ausführen des Docker Compose .....	13
2.4 Importieren des SQL-Dumps .....	14
2.5 Updates durchführen .....	15
2.6 Moodle aufrufen .....	16
3 Glossar .....	17
4 Anhang / Ressourcen .....	18
4.1 Abbildungsverzeichnis.....	18
4.2 Tabellenverzeichnis .....	18
4.3 Quellenverzeichnis.....	19

# 1 Einleitung

## 1.1 Auftrag

In diesem Projekt sollte eine Moodle-Umgebung von einer Ubuntu Maschine in eine Docker-Umgebung migriert werden. Das folgende Dokument sollte die Realisierung beinhalten. Hier wird mithilfe von Bildern und Tabellen das Ganze Vorgehen beschrieben.

## 1.2 Anforderungen

Folgende Anforderungen wurden bei der Initialisierung des Projektes definiert, diese mussten nun bei der Umsetzung beachtet werden.

### 1.2.1 Muss-Anforderungen

Req_1	Neuste Moodle-Version (via localhost:80 abrufbar)
Req_2	Datenbank-Container, Moodle-Container
Req_3	Daten vollumfänglich migriert
Req_4	Altes System noch lauffähig unter Port 8080 (Klare Kennzeichnung, dass es sich um das alte System handelt)
Req_5	Moodle-Image selbst erstellen (nicht Standard Moodle-Image verwenden)
Req_6	Funktionstüchtiges Login
Req_7	Sicherheit
Req_8	Daten auf Hostsystem

**Tabelle 2: Muss-Anforderungen**

## 2 Realisierung

### 2.1 Bestehende Umgebung auf Port 8080 umstellen.

Zu Beginn musste zuerst die veraltete Umgebung auf Port 8080 umgestellt werden. Die Umgebung ist auf einer VM mit Ubuntu bereits installiert.

#### 2.1.1 Moodle auf Port 8080 Umstellen

Damit man das Ganze umstellen kann, muss der definierte Port in der Config.php unter `/var/www/html` angepasst werden. Dazu muss die `wwwroot` zu `'http://localhost:8080'` geändert werden. Im Bild ist zu sehen wie es dann aussehen sollte.

```
GNU nano 6.2
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype      = 'mysqli';
$CFG->dblibrary   = 'native';
$CFG->dbhost      = 'localhost';
$CFG->dbname      = 'moodle';
$CFG->dbuser      = 'debian-sys-maint';
$CFG->dbpass      = 'vaIdfgRPSXzKbPPd';
$CFG->prefix      = 'mdl_';
$CFG->dboptions   = array (
    'dbpersist' => 0,
    'dbport'    => '',
    'dbsocket'  => '',
    'dbcollation' => 'utf8mb4_unicode_ci',
);

$CFG->wwwroot     = 'http://localhost:8080';
$CFG->dataroot    = '/var/www/moodledata';
$CFG->admin       = 'admin';

$CFG->directorypermissions = 0777;

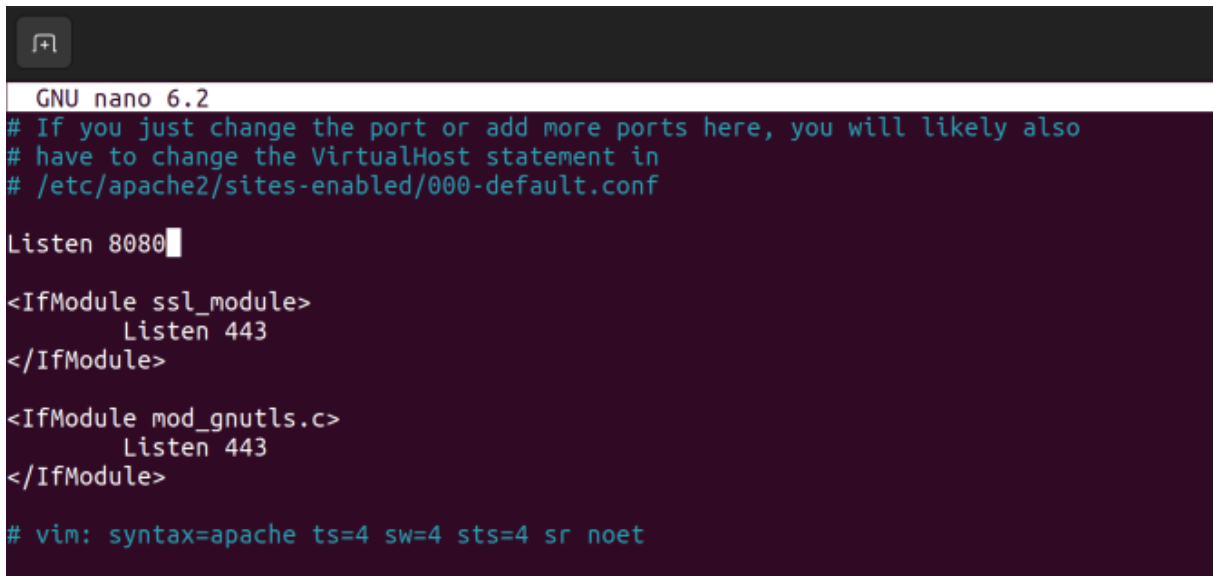
require_once(__DIR__ . '/lib/setup.php');

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!
```

Abbildung 1: Port 8080 umstellen

### 2.1.2 Umstellung Apache2 zu Port 8080

Als nächstes muss der Apache Listen Port auf 8080 angepasst werden. Das kann man unter `/etc/apache2/ports.conf`. Dafür bearbeitet man die Datei am einfachsten mit «nano».



```
GNU nano 6.2
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 8080

<IfModule ssl_module>
    Listen 443
</IfModule>

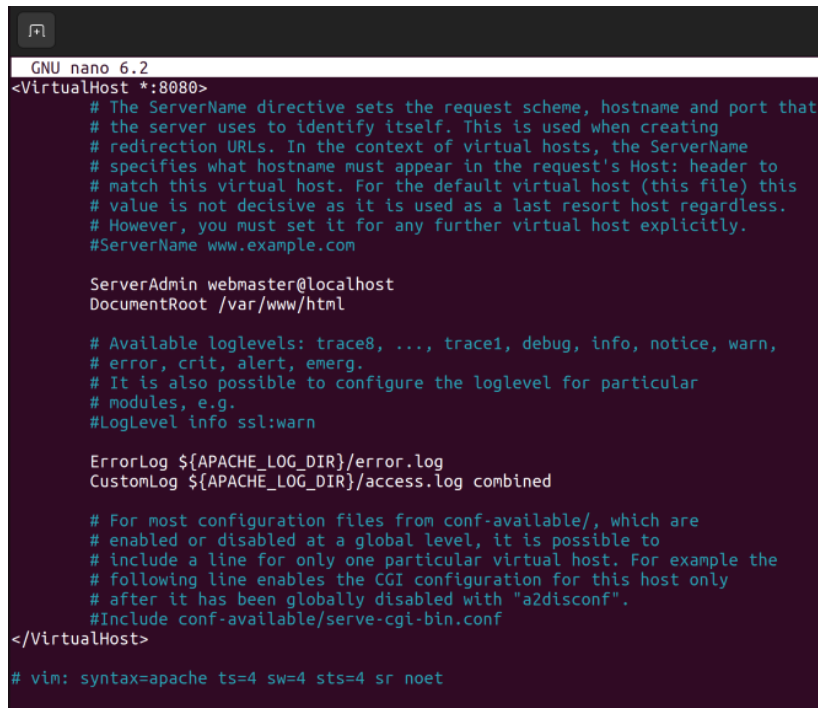
<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

**Abbildung 2: Umstellung Apache2 zu Port 8080**

### 2.1.3 Sites-enabled auf Port 8080 umstellen

Zum Schluss muss die `000-default.conf` unter `/etc/apache2/sites-enabled` auch auf Port 8080 Umgestellt werden. Dafür muss man oben einfach von 80 auf 8080 ändern. Das ganze sollte auch wieder mit «nano» gemacht werden.



```
GNU nano 6.2
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

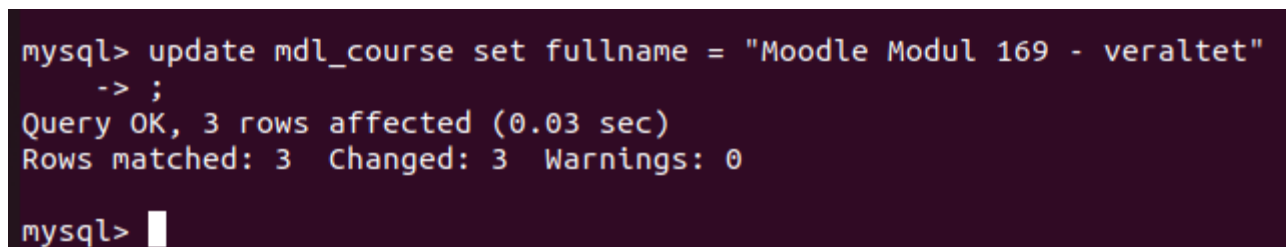
# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

**Abbildung 3: 000-default.conf**

### 2.1.4 Kennzeichnung der veralteten Umgebung

Dass es zukünftig keine Verwechslungen gibt, haben wir den Namen des veralteten Kurses zu „Moodle Modul 169 – veraltet“ angepasst. Um dies umzusetzen, muss auf der Datenbank «moodle» folgender Befehl ausgeführt werden:



```
mysql> update mdl_course set fullname = "Moodle Modul 169 - veraltet"
-> ;
Query OK, 3 rows affected (0.03 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> █
```

**Abbildung 4: Kennzeichnung der veralteten Umgebung**

### 2.1.5 Neustart des Services

Damit alles auch aktualisiert wird, muss noch der Service Apache2 neu gestartet werden. Dies erfolgt durch den Befehl im Bild.

```
root@lp-22-04:/var/www/html# systemctl restart apache2
root@lp-22-04:/var/www/html#
```

#### Abbildung 5: Neustart von Apache2

Anschliessend ist die Moodle Seite über <http://localhost:8080> mit der entsprechenden Notiz, dass diese veraltet ist, erreichbar.

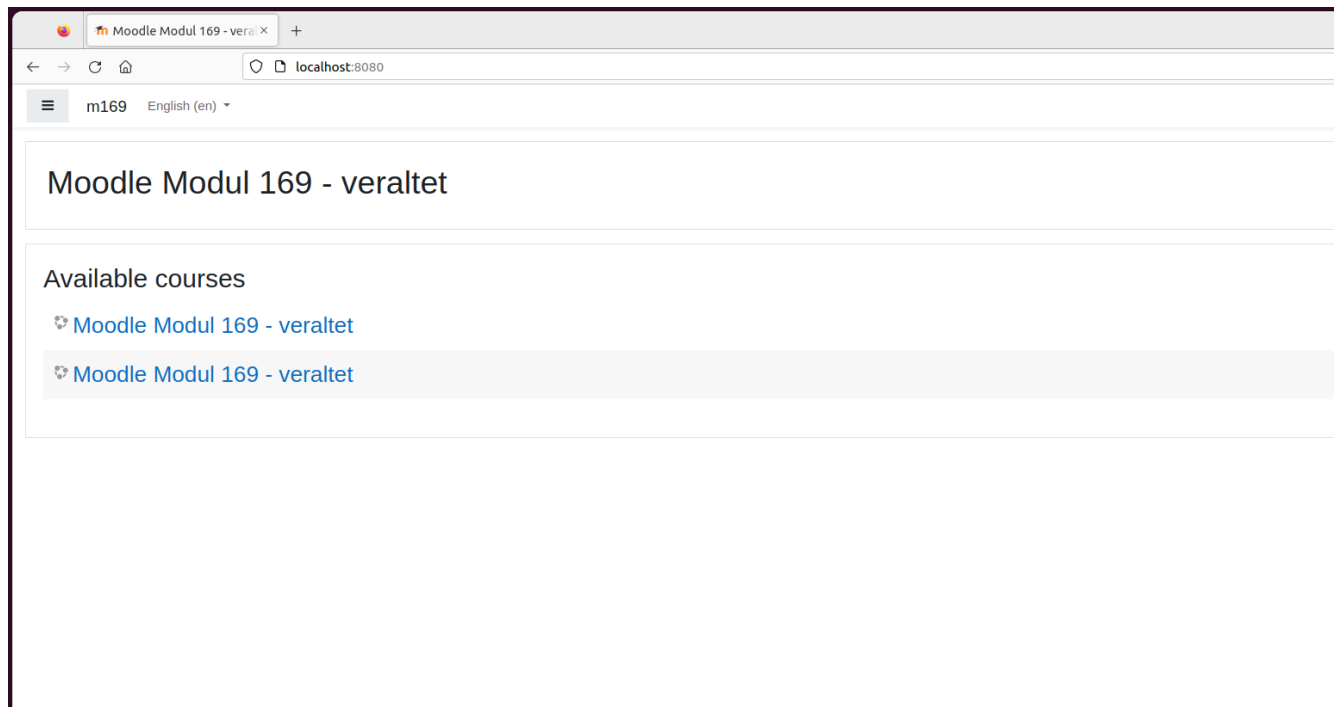


Abbildung 6: Altes Moodle auf Port 8080



## Sichern der Daten aus alter Umgebung

### 2.1.6 Erstellung eines SQL-Dumps

Um die Daten aus der alten Umgebung zu migrieren, musste zuerst ein SQL-Dump erstellt werden. Dazu wurde der folgende Befehl verwendet.

```
root@lp-22-04:/home/vmadmin# mysqldump moodle > dump.sql
root@lp-22-04:/home/vmadmin# ls
app Desktop DockerProjekt Documents Downloads dump.sql Music Pictures Public snap Templates tmp Videos
root@lp-22-04:/home/vmadmin#
```

**Abbildung 7: Erstellung SQL-Dump**

### 2.1.7 Anpassung des SQL-Dumps

Die in die Container geladenen Moodle-Daten und die dazugehörige Datenbank stammen aus der Version 3.10.11, während das Container-Image einer neueren Version basiert. Bei der Erstellung der Container richtet Moodle eine neue Datenbankstruktur ein, die zusätzliche Tabellen enthält, die in der älteren Version nicht vorhanden waren.

Nach dem Import des Dumps erkennt die neue Moodle-Version die veralteten Daten und versucht ein Update durchzuführen. Dieses Update scheitert jedoch, weil die neuen Tabellen bereits vorhanden sind. Um das Update ohne Fehler durchzuführen, müssen diese Tabellen vor dem Import des Dumps gelöscht werden.

### 2.1.8 Ergänzung folgender Befehle

Damit der Import Problemlos funktioniert müssen die folgenden Befehle vor den Commands im SQL-Dump ergänzt werden.

```
-- Tabellenamen für mdl_tool_brickfield-Tabellen sammeln

SELECT GROUP_CONCAT(table_schema, ',', table_name) INTO @tables

FROM information_schema.tables

WHERE table_schema = 'moodle' AND table_name LIKE 'mdl_tool_brickfield%';

-- DROP TABLE-Befehl für mdl_tool_brickfield-Tabellen vorbereiten und ausführen

SET @tables = CONCAT('DROP TABLE IF EXISTS ', @tables);

PREPARE stmt FROM @tables;

EXECUTE stmt;

DEALLOCATE PREPARE stmt;

-- Tabellenamen für mdl_bigbluebutton-Tabellen sammeln

SELECT GROUP_CONCAT(table_schema, ',', table_name) INTO @tables

FROM information_schema.tables
```

```
WHERE table_schema = 'moodle' AND table_name LIKE 'mdl_bigbluebutton%';

-- DROP TABLE-Befehl für mdl_bigbluebutton-Tabellen vorbereiten und ausführen

SET @tables = CONCAT('DROP TABLE IF EXISTS ', @tables);

PREPARE stmt FROM @tables;

EXECUTE stmt;

DEALLOCATE PREPARE stmt;

-- Die Tabelle mdl_tiny_autosave löschen

DROP TABLE IF EXISTS `mdl_tiny_autosave`;
```

**Tabelle 3: Ergänzungen SQL Dump****2.1.9 Sichern der Moodle Daten**

Die Daten der Moodle Seite, werden über ein Volume bereitgestellt. Damit dies möglich ist, müssen diese zuerst aus der alten Umgebung exportiert werden. Dazu müssen die gesamten Daten aus /var/www/moodledata in einen weiteren Ordner «moodledata» kopiert werden.

## 2.2 Migration der Moodle Umgebung

### 2.2.1 Erklärung des Docker-Compose

```
version: '3.9' # Version des Docker Compose Formats

services:

  mariadb: # Definition des MariaDB-Datenbankdienstes

    image: mariadb # Das MariaDB-Docker-Image wird verwendet

    restart: always # Der Container wird immer neu gestartet, wenn er stoppt

    volumes:

      - DB-Volume:/var/lib/mysql # Datenbankdaten werden in einem benannten Volumen gespeichert

    networks:

      - moodle-network # Der Dienst wird dem moodle-network-Netzwerk hinzugefügt

  environment:

    - MYSQL_RANDOM_ROOT_PASSWORD=true # Das Root-Passwort wird zufällig generiert

    - MYSQL_USER=system-manager # Benutzername für die Datenbank

    - MYSQL_PASSWORD=akdsfksdfksdf # Passwort für den Benutzer

    - MYSQL_DATABASE=moodle # Name der zu erstellenden Datenbank

  phpmyadmin: # Definition des phpMyAdmin-Dienstes

    image: phpmyadmin/phpmyadmin # Das phpMyAdmin-Docker-Image wird verwendet

    restart: always # Der Container wird immer neu gestartet, wenn er stoppt

    ports:

      - 88:80 # Der Dienst ist über Port 88 auf dem Host erreichbar und wird auf Port 80 im Container weitergeleitet

    environment:

      - PMA_HOST=mariadb # Der Hostname der MariaDB, zu der sich phpMyAdmin verbindet

    networks:

      - moodle-network # Der Dienst wird dem moodle-network-Netzwerk hinzugefügt
```

moodle: # Definition des Moodle-Dienstes

image: bitnami/moodle:4.1 # Das Bitnami Moodle-Docker-Image wird verwendet, Version 4.1

restart: always # Der Container wird immer neu gestartet, wenn er stoppt

ports:

- 80:8080 # Der Dienst ist über Port 80 auf dem Host erreichbar und wird auf Port 8080 im Container weitergeleitet

volumes:

- Moodle-Volume:/bitnami/moodle # Moodle-Daten werden in einem benannten Volumen gespeichert

- ./moodledata:/bitnami/moodledata # Moodle-Daten werden im lokalen Verzeichnis `./moodledata` gespeichert und im Container nach `/bitnami/moodledata` gemountet

networks:

- moodle-network # Der Dienst wird dem moodle-network-Netzwerk hinzugefügt

environment:

- MOODLE\_DATABASE\_HOST=mariadb # Hostname der Datenbank

- MOODLE\_DATABASE\_USER=system-manager # Benutzername für die Datenbankverbindung

- MOODLE\_DATABASE\_PASSWORD=akdsfkdsfkdsf # Passwort für die Datenbankverbindung

- MOODLE\_DATABASE\_NAME=moodle # Name der Moodle-Datenbank

volumes:

DB-Volume: # Definition des benannten Volumens für die MariaDB-Daten

Moodle-Volume: # Definition des benannten Volumens für die Moodle-Daten

networks:

moodle-network: # Definition des benutzerdefinierten Netzwerks für die Dienste

**Tabelle 4: Erklärung Docker-Compose**

## 2.2.2 Ausführen des Docker Compose

Um die Umgebung schlussendlich zu starten, wird der untenstehende Befehl verwendet. Danach wird die Datenbank, der Moodle Container sowie der phpmyadmin Container gestartet, welche später für den Import der Datenbank verwendet wird.

Beim Ausführen des Docker-compose, muss darauf geachtet werden, dass sich der Benutzer im entsprechenden Ordner befindet, wo auch die zuvor exportierten moodledata gespeichert sind.

Command: *docker compose up -d*

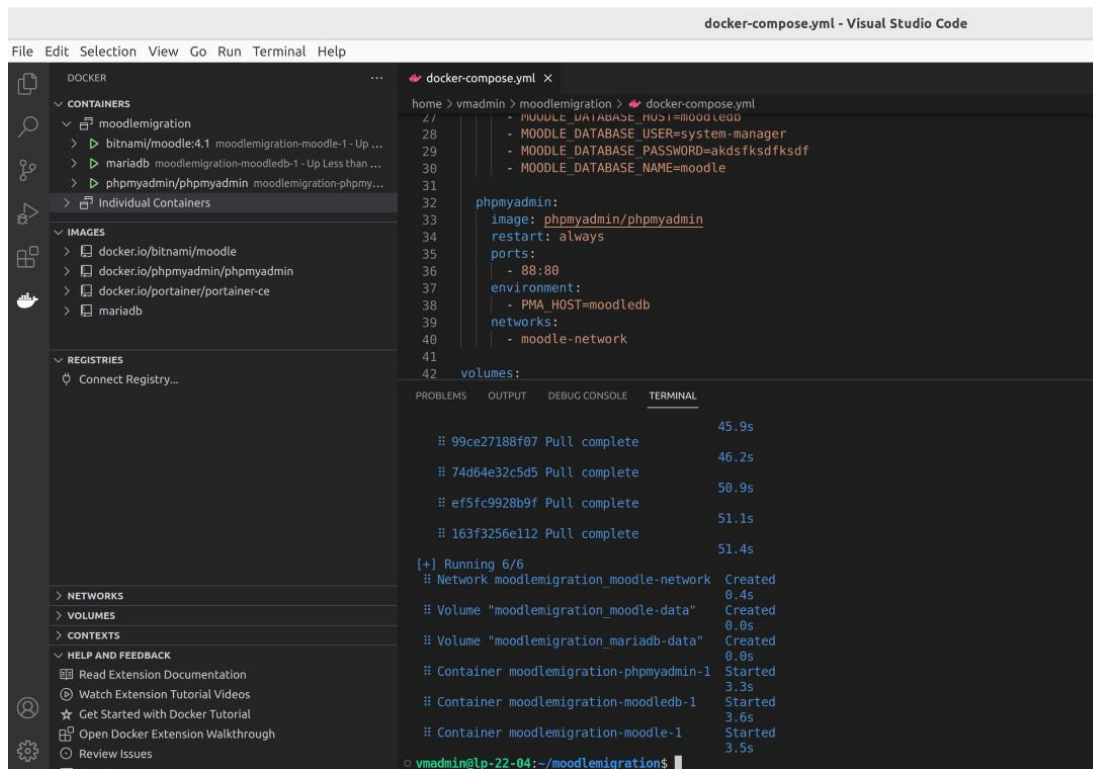


Abbildung 8: Docker Compose wird ausgeführt

## Importieren des SQL-Dumps

Um den SQL dump ohne Datenverluste zu importieren und später auf einfache Art und Weise zu verwalten wird ein weiterer Container mit phpmyadmin benutzt. Dieser ist nach ausführen des Docker-Compose unter <http://localhost:88> verfügbar. Auf phpmyadmin wird der zuvor definierte Benutzer system-manager mit dem im Docker-Compose gegebenen Passwort verwendet. Aus Sicherheitsaspekten sollte dieses Passwort nach der Erstellung der Umgebung angepasst werden.

Um nun die Datenbank zu importieren, muss zuerst die DB Moodle auf der linken Seite angewählt werden. Anschliessend kann der angepasste Dump mit der Import Funktion importiert werden.

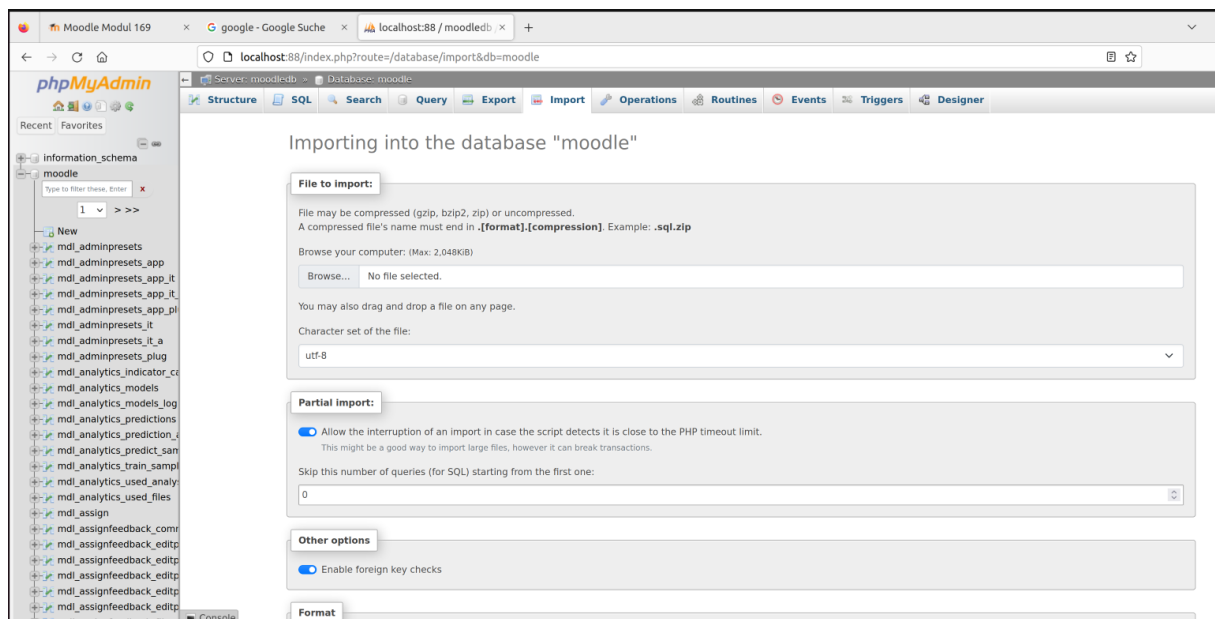
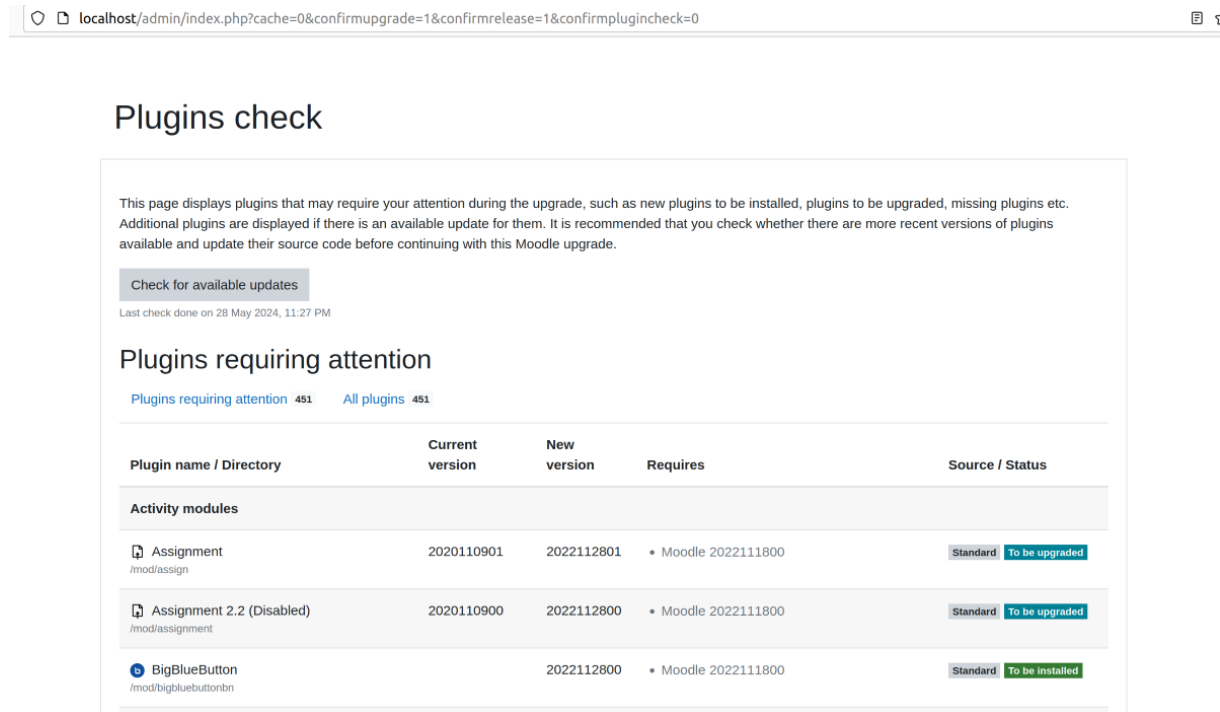


Abbildung 9: Import des Dumps in phpmyadmin

## 2.3 Updates durchführen

Nachdem der Dump importiert wurde, müssen auf der neuen Moodle Seite einige Updates durchgeführt werden. Dies muss mit «Check for available Updates» und danach mit Continue bestätigt werden.



This page displays plugins that may require your attention during the upgrade, such as new plugins to be installed, plugins to be upgraded, missing plugins etc. Additional plugins are displayed if there is an available update for them. It is recommended that you check whether there are more recent versions of plugins available and update their source code before continuing with this Moodle upgrade.

[Check for available updates](#)

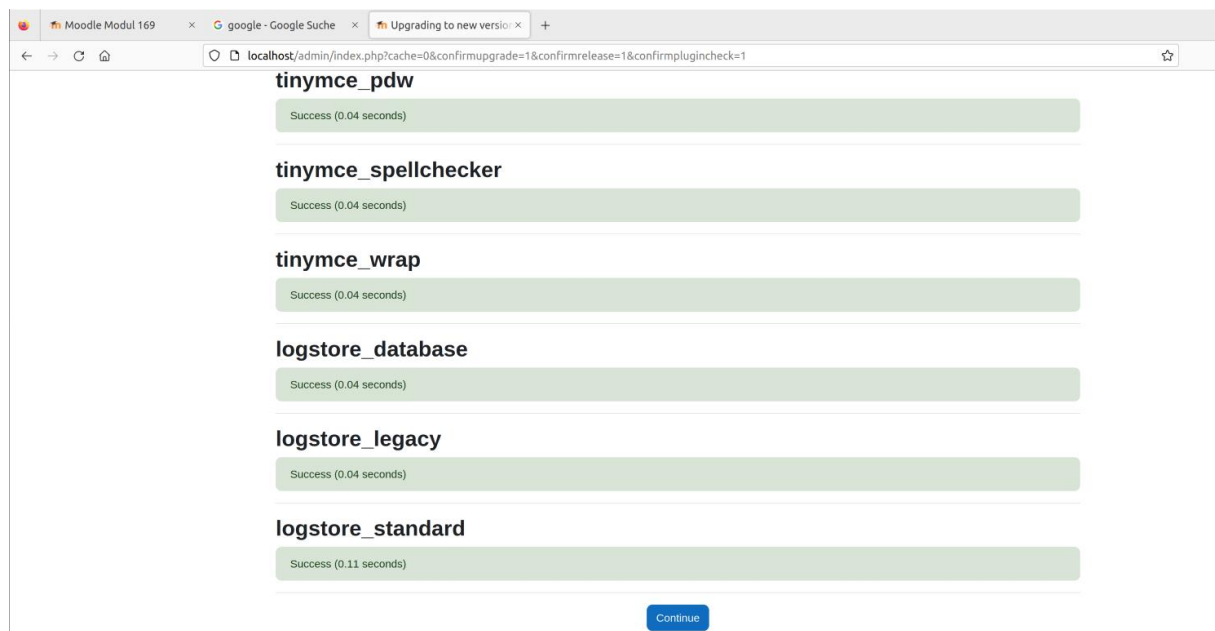
Last check done on 28 May 2024, 11:27 PM

### Plugins requiring attention

[Plugins requiring attention](#) 451 [All plugins](#) 451

Plugin name / Directory	Current version	New version	Requires	Source / Status
<b>Activity modules</b>				
Assignment /mod/assign	2020110901	2022112801	• Moodle 2022111800	Standard <a href="#">To be upgraded</a>
Assignment 2.2 (Disabled) /mod/assignment	2020110900	2022112800	• Moodle 2022111800	Standard <a href="#">To be upgraded</a>
BigBlueButton /mod/bigbluebuttonbn		2022112800	• Moodle 2022111800	Standard <a href="#">To be installed</a>

Abbildung 10: Update Plugins



tinymce\_pdw  
Success (0.04 seconds)

tinymce\_spellchecker  
Success (0.04 seconds)

tinymce\_wrap  
Success (0.04 seconds)

logstore\_database  
Success (0.04 seconds)

logstore\_legacy  
Success (0.04 seconds)

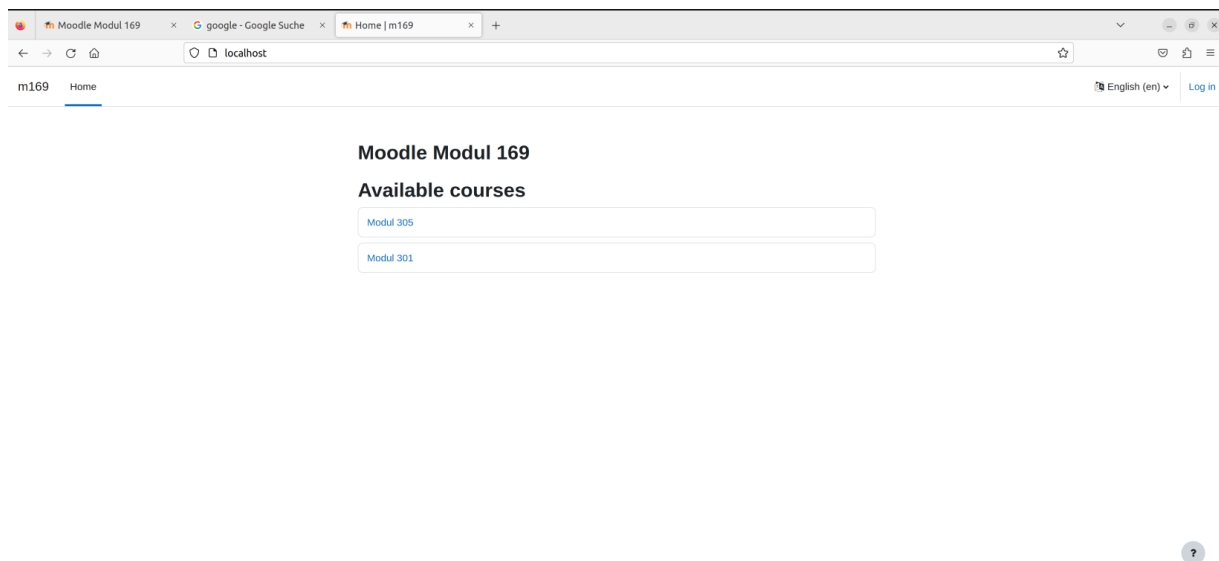
logstore\_standard  
Success (0.11 seconds)

[Continue](#)

Abbildung 11: Migriertes Moodle offene Updates

## 2.4 Moodle aufrufen

Zum Schluss kann auf die migrierte Moodle Umgebung über <http://localhost:80> zugegriffen werden.



**Abbildung 12: Funktionierendes Moodle**



## Glossar

Begriff	Erläuterung
Sudo	Ein Befehl, der es einem Benutzer erlaubt, Programme mit den Sicherheitsprivilegien eines anderen Benutzers, standardmässig des Superusers, auszuführen.
Mysqldump	Ein Dienstprogramm zum Sichern von MySQL-Datenbanken. Es erstellt einen SQL-Dump, der die SQL-Befehle enthält, um die Datenbank wiederherzustellen.
Moodle	Eine Open-Source-Lernplattform, die als Learning Management System (LMS) verwendet wird.
Dump.sql	Eine Datei, die die exportierten Datenbankinformationen enthält, die mit mysqldump erstellt wurden.
Container	Eine standardisierte Einheit der Software, die den Code und alle seine Abhängigkeiten umfasst, sodass die Anwendung schnell und zuverlässig von einer Computerumgebung zur anderen übertragen werden kann.
Container-Image	Eine unveränderliche, leichtgewichtige Datei, die alle Abhängigkeiten, Konfigurationsdateien und Einstellungen enthält, die für die Ausführung einer Anwendung in einem Container erforderlich sind.
Update	Ein Prozess, bei dem eine Software auf eine neuere Version aktualisiert wird, um Fehler zu beheben, neue Funktionen hinzuzufügen oder Sicherheitslücken zu schliessen.
www-data	Der Standardbenutzer, unter dem der Apache-Webserver auf vielen Linux-Distributionen läuft. Wird oft verwendet, um sicherzustellen, dass Webanwendungen wie Moodle die richtigen Berechtigungen haben.
Php	Eine weit verbreitete Skriptsprache, die besonders geeignet ist für die Webentwicklung und die Einbettung in HTML. Moodle ist in PHP geschrieben und benötigt PHP, um zu funktionieren.

**Tabelle 5: Glossar**

## 3 Anhang / Ressourcen

### 3.1 Abbildungsverzeichnis

Abbildung 1: Port 8080 umstellen.....	5
Abbildung 2: Umstellung Apache2 zu Port 8080 .....	6
Abbildung 3: 000-default.conf .....	7
Abbildung 4: Kennzeichnung der veralteten Umgebung .....	7
Abbildung 5: Neustart von Apache2.....	8
Abbildung 6: Altes Moodle auf Port 8080 .....	8
Abbildung 7: Erstellung SQL-Dump.....	9
Abbildung 8: Docker Compose wird ausgeführt .....	13
Abbildung 9: Import des Dumps in phpmyadmin .....	14
Abbildung 10: Update Plugins .....	15
Abbildung 11: Migriertes Moodle offene Updates .....	15
Abbildung 12: Funktionierendes Moodle.....	16

### 3.2 Tabellenverzeichnis

Tabelle 1: Dokumentversionen .....	2
Tabelle 2: Muss-Anforderungen.....	4
Tabelle 3: Ergänzungen SQL Dump .....	10
Tabelle 4: Erklärung Docker-Compose .....	12
Tabelle 5: Glossar.....	17

## Quellenverzeichnis

- Offizielle Moodle-Dokumentation - Docker Guide
  - Webseite: Moodle Entwicklerdokumentation - <https://docs.moodle.org/dev/Docker>
  - Zugriff: 28.05.2024
- Docker Hub - Moodle Image (Bitnami)
  - Webseite: Bitnami Moodle auf Docker Hub - <https://hub.docker.com/r/bitnami/moodle/>
  - Zugriff: 31.05.2024
- GitHub - Moodle Docker Repository
  - Webseite: Moodle Docker Repository auf GitHub - <https://github.com/moodlehq/moodle-docker>
  - Zugriff: 25.05.2024
- Docker Dokumentation - Installation
  - Webseite: Docker Installationsdokumentation - <https://docs.docker.com/get-docker/>
  - Zugriff: 02.06.2024
- Docker Compose Dokumentation - Installation
  - Webseite: Docker Compose Installationsdokumentation - <https://docs.docker.com/compose/install/>
  - Zugriff: 27.05.2024
- MySQL Dokumentation - mysqldump
  - Webseite: MySQL mysqldump-Dokumentation - <https://dev.mysql.com/doc/refman/8.0/en/mysqldump.html>
  - Zugriff: 29.05.2024
- Beispiel Docker-Compose-Konfigurationsdatei
  - Zugriff: 01.06.2024
- Docker-Dokumentation - Volumes
  - Webseite: Docker Volumes-Dokumentation - <https://docs.docker.com/storage/volumes/>
  - Zugriff: 22.05.2024