



Softwareprojekt WS 2014/2015

Studentenclub-Verwaltung

Installationsanleitung

Inhaltsverzeichnis

0.	Beispiel der Installationsanleitung.....	3
1.	Voraussetzungen.....	3
2.	Composer installieren	3
3.	Laravel installieren.....	4
4.	Laravel app- und public-Ordner installieren.....	4
5.	Datenbank erstellen/auswählen	5
6.	Datenbank Konfiguration unter Laravel hinterlegen	5
7.	Datenbankschema anlegen.	6
8.	Apache Konfiguration.....	6
9.	Weitere Laravel Konfiguration	7
10.	LDAP Konfigurieren.....	7

0. Beispiel der Installationsanleitung

In dieser Installationsanleitung werden folgende Variablen beispielhaft verwendet, bei der Installation können diese Variablen den eigenen Wünschen entsprechend gesetzt / abgeändert werden, bzw. müssen der Konfiguration des Systems entsprechend gesetzt werden.

Installationsverzeichnis:	<code>/var/www/laravel</code>
Name der Datenbank:	<code>laravelDB</code>
Beispiele Domain für Laravel:	<code>laravel.bc-club.de</code>
Benutzergruppe des Apache Webserver:	<code>www-data</code>

1. Voraussetzungen

Für die Installation werden folgende vorinstallierte Komponenten vorausgesetzt

- MySQL Datenbank Server
- Apache Server
- PHP Version 5.6
 - MCRYPT PHP Erweiterung
 - JSON PHP Erweiterung
 - LDAP PHP Erweiterung
 - MySQL PHP Erweiterung

Je nach verwendetem Ziel-Betriebssystem ist die Installation dieser Komponenten unterschiedlich. Unter Linux Betriebssystemen kann dazu ein Paketmanager wie APT oder YUM verwendet werden. Allerdings unterscheiden sich die Paketnamen je nach Version und verwendetem Betriebssystem, deswegen wird empfohlen das Handbuch zum Betriebssystem zur Installation der Komponenten zurate zu nehmen.

Beispiel für Debian basierte Betriebssysteme (am Beispiel Debian 8)

```
apt-get install apache2 libapache2-mod-php5 mysql-server php5 php5-json  
php5-ldap php5-mcrypt php5-mysql
```

Beispiel für RedHat basierte Betriebssysteme (am Beispiel CentOS 7)

```
yum install httpd mariadb-server php php-ldap php-mysql
```

2. Composer installieren

Für Laravel wird eine Installation über den Abhängigkeitsmanager Composer empfohlen.

Dieser Punkt 2 und die Punkte 3 und 4 können übersprungen werden wenn als Grundlage für die Installation die Version aus der übergebenen Zip-Datei verwendet wird.

Composer kann über folgende Befehle ins Verzeichnis /usr/local/bin/ installiert werden um den Befehl global im System verfügbar zu machen.

Ins Verzeichnis wechseln:

```
cd /usr/local/bin/
```

Composer herunterladen.

```
php -r "readfile('https://getcomposer.org/installer');" | php
```

Symbolische Verknüpfung erstellen damit Composer neben ,composer.phar‘ auch über ,composer‘ aufgerufen werden kann:

```
ln -s /usr/local/bin/composer.phar /usr/local/bin/composer
```

Für alternative Installationsweisen empfiehlt sich ein Blick auf die Webseite von Composer: <https://getcomposer.org/download/>

3. Laravel installieren

Nach der Installation von Composer kann eine beliebige Laravel Installation wie folgt heruntergeladen werden

Ins Verzeichnis wechseln

```
cd /var/www/
```

Laravel herunterladen

```
composer create-project laravel/laravel --prefer-dist laravel
```

4. Laravel app- und public-Ordner installieren

Jetzt sollten app- und public- Verzeichnisse aus der übergebenden Zip-Datei ersetzt werden.

Erst wechseln wir in das Verzeichnis von Laravel:

```
cd /var/www/laravel/
```

Dann löschen wir die über Laravel automatisch erzeugten Verzeichnisse app und public:

```
rm -r app/ public/
```

Jetzt entpacken wir die beiden Ordner aus der übergebenen Zip-Datei in das Verzeichnis:

```
unzip Studentenclubverwaltung.zip app/*
```

```
unzip Studentenclubverwaltung.zip public/*
```

5. Datenbank erstellen/auswählen

Es wird eine eigene Datenbank benötigt. (Optional: Wenn mehrere Anwendungen auf dem Server laufen, sollte für jede Anwendung ein eigener Datenbank-Nutzer definiert werden). Wenn eine vorhandene Datenbank verwendet werden soll, kann dieser Punkt übersprungen werden, wenn sich die Anwendung die Datenbank mit anderen Anwendungen teilt, sollte ein Prefix für die Tabellen verwendet werden (siehe dazu 6).

Für versierte Nutzer kann dies über die Kommandozeile erledigt werden.

Am MySQL Server anmelden:

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

Datenbank erstellen:

```
create database laravelDB;
```

Zur handlicheren Verwaltung der MySQL Datenbank empfehlen wir phpMyAdmin, damit lässt sich relativ schnell und einfach mit Datenbanken arbeiten und diese verwalten.

6. Datenbank Konfiguration unter Laravel hinterlegen

Damit Laravel auf die Datenbank zugreifen kann muss der Zugriff auf die Datenbank in der Konfigurationsdatei „app/config/database.php“ hinterlegt werden.

Hier ist eine Beispielkonfiguration der Datei app/config/database.php:

```
'mysql' => array(
    'driver'   => 'mysql',
    'host'     => 'localhost',
    'database' => 'laravelDB',
    'username' => 'root',
    'password' => '',
    'charset'  => 'utf8',
    'collation' => 'utf8_unicode_ci',
    'prefix'   => '',
),
```

Host: Wenn die Datenbank auf einem externen Server läuft muss hier die Adresse angegeben werden, bei einem lokalen Datenbankserver kann 'localhost' verwendet werden

Database: Hier wird der Name der zu verwendeten Datenbank angegeben

Username: Der Datenbank-Nutzername

Password: Das Passwort des Datenbank-Nutzers

Charset: Zeichenkodierung mit der die Daten in die Datenbank geschrieben werden

Prefix: Präfix für die Tabellennamen, sinnvoll wenn mehrere Anwendungen dieselbe Datenbank verwenden, damit sich Tabellennamen nicht überschneiden.

7. Datenbankschema anlegen.

Mithilfe der Laravel Migrations wird das Datenbankschema angelegt, dazu muss folgender Befehl ausgeführt werden. Mit ‚cd‘ in das Verzeichnis wechseln und php artisan migrate ausführen.

```
cd /var/www/laravel/
```

```
php artisan migrate
```

8. Apache Konfiguration

Bei der Konfiguration von Apache für Laravel sollte beachtet werden dass das Modul **mod_rewrite** aktiviert werden muss. Außerdem sollte das Zielverzeichnis von **DocumentRoot** des VirtualHost auf den **public** Ordner von Laravel verweisen.

Eine Beispielkonfiguration für Laravel für Apache:

```
<VirtualHost *:80>

    ServerName laravel.bc-club.de

    DocumentRoot "/var/www/laravel/public"

    <Directory "/var/www/laravel/public">

        AllowOverride all

    </Directory>

</VirtualHost>
```

Nutzer und Nutzergruppe anpassen. Damit Apache die Dateien lesen kann und teilweise auch in die entsprechenden Verzeichnisse schreiben kann (z.B. für Cache und Fehlerlogs) ist es notwendig dass die Nutzer und Nutzergruppen der Dateien und Verzeichnisse von Laravel angepasst werden. Dies kann mit folgendem Befehlen bewerkstelligt werden.

```
chown -R www-data:www-data /var/www/laravel
```

9. Weitere Laravel Konfiguration

/app/config/app.php

Hier sollten folgende Variablen den eigenen Wünschen gesetzt werden:

Damit kann das Debugging für die Fehlersuche ein- oder ausgeschaltet werden:

```
'debug' => true,      ODER      'debug' => false,
```

Die Domain / IP unter der Laravel verfügbar ist:

```
'url' => 'http://laravel.bc-club.de',
```

Schlüssel zur Verschlüsselung sollte auf einen zufälligen 32 Zeichen String gesetzt werden:

```
'key' => '',
```

/public/.htaccess

Hier sollte die Rewrite-Rule angepasst werden, wenn Laravel zum Beispiel in einem Unterverzeichnis ‚test123‘ installiert wurde und über ‚http://laravel.bc-club.de/test123‘ erreichbar ist:

```
RewriteRule ^(.*)/$ /test123/$1 [L,R=301]
```

```
RewriteBase /test123/
```

Wenn es im Wurzelverzeichnis installiert ist:

```
RewriteRule ^(.*)/$ /$1 [L,R=301]
```

```
RewriteBase /
```

10. LDAP Konfigurieren

In der Datei /app/config/bcLDAP.php muss die Serveradresse und die Zugangsdaten entsprechend konfiguriert werden. Außerdem muss beachtet werden dass die Datei speziell für den bc-Club Server angelegt wurde und für andere LDAP Server ggf. umfangreiche Anpassungen nötig sind.

Die Adresse des LDAP Servers:

```
'server' => '',
```

Die Zugangsdaten für den LDAP Server:

```
'adminUsername' => '',
```

```
'adminPassword' => '',
```