

Study Case

Elektronisches Tagebuch

Markus Gehrig | Webtechnologie 1 | 10.11.18

1 Konzept 2

1.1 System Architecture 2

1.2 Software Architecture 3

1.2.1 Fremd-Module 3

1.3 Design 3

1.4 DatenBank 3

1.5 UML 3

2 Entwicklungsumgebung 4

2.1 Server Voraussetzungen 4

2.2 Zusätzliche WerkezEUge 4

2.3 Zusätzliche Software Komponenten 5

3 Testfälle (Platzhalter) 5

# Konzept

## System Architecture

## Software Architecture

**Systemgrenze**

DB  
MySQL

DBAL

File Service

Datei-  
System

Business - Logik

Twig

Frontend  
Bootstrap  
oder  
Foundation

### Fremd-Module

**Doctrine DBAL (Database Abstraction Layer) -** Mächtiger Datenbank Abstraktion Layer, welcher sowohl mit MySQL als auch mit MariaDB, Microsoft SQL Server, Oracle, PostgreeSQL, SAP Sybase SQL Anywhere und SQLite umgehen.

**Symfony Finder (PHP Datei System Interface) -** Einfacher Zugriff auf Dateien des Datei-Systems.Doku: <https://symfony.com/doc/current/components/finder.html>

**Twig -** Einfache PHP Frontend Anbindung, welche Marker in PHP/HTML Files austauscht.

## Design

## DatenBank

## UML

Vielleicht gefällt Ihnen das Foto auf dem Deckblatt genau so gut wie uns – aber wenn es nicht ideal für Ihren Bericht ist, können Sie es auch ganz leicht durch ein eigenes ersetzen.

Löschen Sie einfach das Platzhalterbild. Klicken Sie dann auf der Registerkarte "Einfügen" auf "Bild", um eins aus Ihren Dateien auszuwählen.

# System-Komponenten

## Server Voraussetzungen

|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Version | Beschreibung |
|  | Apache | Benötigt: Apache/1.4  verwendet: Apache/2.4.7 (Ubuntu) | Der Apache 2 ist der führende Open Source Webserver auf dem Markt. |
|  | PHP | Benötigt: PHP 7.0 oder höher  verwendet: PHP 7.2 | PHP ist eine Skriptsprache, die für die Erstellung von dynamischen Webseiten und Webanwendungen verwendet wird. |
|  | MySQL | Benötigt: MySQL 8.0.0  verwendet: MySQL 8.0.12 | MySQL ist eine Datenbanklösung der Firma Oracle, welche oft zusammen mit PHP eingesetzt wird. |

## WerkezEUge

|  | Name | Version | Beschreibung |
| --- | --- | --- | --- |
| Bildergebnis für Docker | Docker for Windows |  | Docker stellt eine Virtualisierungsumgebung zur Verfügung. Der Vorteil von Docker zu anderen Virtualisierungsumgebungen ist, dass vorgefertigte Container wie PHP und MySQL geladen werden können. |
|  | Composer | Benötigt: Composer 1.0.0  Verwendet: Composer 1.7.3 | Composer ist ein Paketmanager für PHP, mit welchem man einfach und schnell PHP Bibliotheken herunterladen, installieren und autoloaden kann. |
| Datei:Visual Studio Code 1.18 icon.svg | Microsoft Visual Studio Code |  | Freier Quelltext-Editor von Microsoft.  Visual Studio Code stellt praktische Module zur Verfügung, welche die Entwicklung mit GitHub und Docker unterstützen. |
| Bildergebnis für github | GitHub |  | GitHub ist die bekannteste Plattform zum Austausch von Quellcode. |

## Zusätzliche Software Komponenten

|  | Name | Version | Beschreibung |
| --- | --- | --- | --- |
|  | Symfony | Benötigt: Symfony 3.0  verwendet: Symfony 3.0 | Symfony ist eine PHP Bibliothek, welche zum einen ein PHP Framework ist und zum anderen eine Komponentensammlung, die eine Vielzahl von entkoppelten Komponenten zur Webentwicklung bereitstellt. |
| Datei:JQuery-Logo.svg | jQuery | Benötigt: jQuery 3.3.1  Verwendet: jQuery 3.3.1 |  |
| Boostrap logo.svg | Bootstrap |  |  |
| Bildergebnis für twig | Twig |  |  |
| Bildergebnis für unit test php | PHPUnit |  |  |

# Entwicklungsumgebung

## Docker

Docker ist eine moderne Virtualisierungsumgebung, welche darauf ausgerichtet ist Dienste und ganze Systeme als „Container“ bereitzustellen. Der Vorteil von Docker ist, dass Systeme einfach und schnell ausgerollt werden können. In meinem Fall braucht es sogar keine spezielle Einrichtung, da man fertige Container für Apache, PHP und MySQL laden kann.

### Docker Netzwerk

Da es keinen fertigen Container gibt, welcher Apache, PHP und MySQL hat. Müssen zwei Container verwendet werden. Einen für Apache und PHP und einen für MySQL. Aus Sicherheitsgründen ist es aber verschiedenen Containern nicht gestattet untereinander zu kommunizieren, solange man kein dedicated Netzwerk für beide Container anlegt. Dies geht einfach über folgenden Befehl.

docker network create dairy --driver bridge

### Docker Container

Die beiden Docker Container für Apache, PHP und MySQL kommen direkt von den jeweiligen Entwicklern, welche ihre Systeme auf Docker Hub zur Verfügung stellen.

#### PHP & Apache Container

Der PHP Container wird direkt von der Docker Community bereitgestellt und besteht aus einem Linux Kernel (Debian) auf welchem nur Apache und PHP laufen. Dadurch ist das System sehr klein und hat dadurch eine extreme Performance.

Dadurch, dass es sich um einen vollwertigen Server handelt, sollte es im Verlauf des Projekts (vor allem bei Rollout) zu fast keinen unerwarteten Problemen kommen.

Die vollständige Anleitung zum PHP Docker findet man hier: <https://hub.docker.com/_/php/>.

Für das Tagebuch Projekt reicht folgender Befehl.

docker run --rm --network dairy -dit -p 80:80 -v C:\Projects\Dairy\src:/var/www/html php:7.2-apache

#### MySQL Container

https://hub.docker.com/\_/mysql/

docker run --rm --network dairy --name db -p 3306:3306 -v C:\Projects\Dairy\mysql:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=geheim -dit mysql:8

# Testfälle (Platzhalter)

1. Tagebucheinträge müssen erfasst werden können.
2. Jeder Tagebucheintrag hat ein Datum. Das Datum kann durch den Benutzer angegeben werden.
3. Jeder Tagebucheintrag muss einer Kategorie (z.B. Ferien, Geburtstag, Familienfest…) zugeordnet werden können.
4. Pro Tagebucheintrag kann ein Freitext inkl. Sonderzeichen bis zu einer maximalen Länge von 1000 Zeichen erfasst werden.
5. Zu jedem Tagebucheintrag kann ein Bild erfasst werden.
6. Jeder Tagebucheintrag wird in der Datenbank gespeichert.