

# STATE OF THE ART



## REFUNDABLE

effiziente Reise- und Exkursions-  
verwaltung für Schulen

Dehner Linus, Foster Ryan, Beier Michael

**tgm**  
Die Schule der Technik

**JUST DO IT**  
HÖHERE ABTEILUNG FÜR  
INFORMATIONSTECHNOLOGIE

Version	Autor	QS	Datum	Status	Kommentare
0.1	Idehner	mbeier	2020-09-24	Draft	Create
1	mbeier	Idehner	2020-10-24	Draft	Backend - Überblick
2	mbeier	Idehner	2020-11-08	Draft	Layout finalisiert
3	mbeier	Idehner	2020-11-08	Draft	Backend - Docker

# Inhaltsverzeichnis

<b>1</b>	<b>Projektleitung &amp; Frontend - responsives Webdesign</b>	<b>3</b>
<b>2</b>	<b>Frontend - Webapplikation als REST-Client</b>	<b>4</b>
2.1	Überblick . . . . .	4
2.2	Design-Patterns . . . . .	4
2.2.1	MVVM . . . . .	4
2.2.2	MVC . . . . .	4
2.3	Datenformate . . . . .	4
2.4	Umsetzungsmöglichkeiten . . . . .	4
2.4.1	Vue . . . . .	4
2.4.2	React . . . . .	4
2.4.3	Angular . . . . .	4
2.4.4	Ohne Framework . . . . .	4
2.5	Aufbereitung der Daten . . . . .	4
<b>3</b>	<b>Backend - REST-Schnittstelle und Infrastruktur</b>	<b>5</b>
3.1	Überblick . . . . .	5
3.2	Docker . . . . .	6
3.2.1	Datenbank . . . . .	6
3.2.2	Backend-Container . . . . .	6
3.2.3	Webserver . . . . .	6
3.3	Deployment . . . . .	6
3.4	REST-Schnittstelle . . . . .	6
3.4.1	Framework . . . . .	6
3.4.2	Endpoints . . . . .	6
3.5	Funktionalität . . . . .	6
3.5.1	TGM-LDAP Schnittstelle . . . . .	6
3.5.2	Datenbank Schnittstelle . . . . .	6
3.5.3	Google Maps . . . . .	6
3.5.4	WebUntis . . . . .	6
3.5.5	E-Mails . . . . .	6
3.5.6	PDF-Dateien . . . . .	6
3.6	Kommunikation und Datenformate . . . . .	6
<b>4</b>	<b>Fazit</b>	<b>7</b>
	<b>Glossar</b>	<b>8</b>
	<b>Literaturverzeichnis</b>	<b>8</b>
	<b>Abbildungsverzeichnis</b>	<b>8</b>

# **1 Projektleitung & Frontend - responsives Webdesign**

## **2 Frontend - Webapplikation als REST-Client**

### **2.1 Überblick**

### **2.2 Design-Patterns**

#### **2.2.1 MVVM**

#### **2.2.2 MVC**

### **2.3 Datenformate**

### **2.4 Umsetzungsmöglichkeiten**

#### **2.4.1 Vue**

#### **2.4.2 React**

#### **2.4.3 Angular**

#### **2.4.4 Ohne Framework**

### **2.5 Aufbereitung der Daten**

### 3 Backend - REST-Schnittstelle und Infrastruktur

#### 3.1 Überblick

Das Backend besteht aus mehreren Komponenten. Einerseits muss eine gewisse Software-Infrastruktur aufgebaut werden, um [Webinterface](#) und die REST-Schnittstelle bereitzustellen. Andererseits muss die Anwendung selbst auch entwickelt werden. Diese besteht wiederum auch aus mehreren Teilen. Darunter fällt die REST-Schnittstelle, inklusive der implementierten Endpoints, selbst, Schnittstellen zu diversen Diensten, wie dem TGM-LDAP Server, zur Datenbank, zu Google Maps und zu WebUntis, aber auch die weitere Funktionalität der Anwendung, unter anderem das Versenden von E-Mails oder Erstellen von PDF-Dateien.

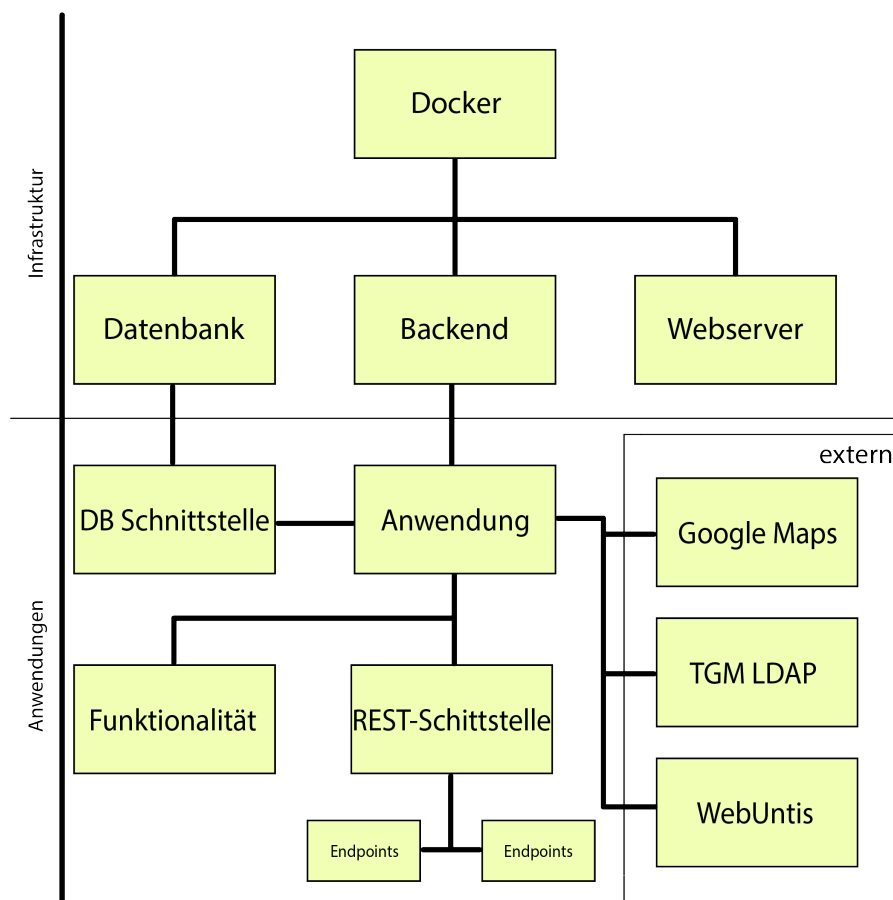


Abbildung 1: Übersicht über die verschiedenen Komponenten der Infrastruktur und der Anwendung

## **3.2 Docker**

Um die Infrastruktur des Projektes einfach aufbauen zu können, wird [Docker](#) genutzt. Da es sich hier um eine komplex strukturierte Infrastruktur handelt wird zusätzlich das Werkzeug [Docker Compose](#) genutzt. Mit Docker Compose kann eine Infrastruktur aufgebaut die der [1. Abbildung \(Übersicht\)](#) entspricht. Für diese sind folgende Container vorgesehen, die in den nächsten Kapiteln noch ins Detail beschrieben werden.

### **3.2.1 Datenbank**

### **3.2.2 Backend-Container**

### **3.2.3 Webserver**

## **3.3 Deployment**

## **3.4 REST-Schnittstelle**

### **3.4.1 Framework**

### **3.4.2 Endpoints**

## **3.5 Funktionalität**

### **3.5.1 TGM-LDAP Schnittstelle**

### **3.5.2 Datenbank Schnittstelle**

### **3.5.3 Google Maps**

### **3.5.4 WebUntis**

### **3.5.5 E-Mails**

### **3.5.6 PDF-Dateien**

## **3.6 Kommunikation und Datenformate**

## **4 Fazit**



## Glossar

**Docker** „Docker ist eine Software, welche es ermöglicht Programme in einer abgeschnittenen Umgebung (genannt Container) laufen zu lassen. Das Erstellen dieser Umgebung und das Installieren und Laufen des Programms darin, gestaltet sich hierbei sehr einfach.[1]“. 6

**Docker Compose** „Docker Compose ist eine Erweiterung von Docker. Mit ihr kann man multiple Container gleichzeitig aufbauen, womit es ermöglicht wird komplexe Infrastruktur - wie in Refundable benötigt - einfach aufzubauen, zu reproduzieren und letztlich auf die Computer, auf denen während der Produktion die Infrastruktur laufen wird, zu liefern.[2]“. 6

**Webinterface** „Ein Web Interface ist ein System, durch welches Anwender mit dem Netz interagieren. Der Begriff Web Interface steht zumeist für grafische Oberflächen.“. 5

## Literaturverzeichnis

- [1] *Docker Engine overview | Docker Documentation*. URL: <https://docs.docker.com/engine/> (besucht am 08. 11. 2020).
- [2] *Overview of Docker Compose | Docker Documentation*. URL: <https://docs.docker.com/compose/> (besucht am 08. 11. 2020).

## Abbildungsverzeichnis

1	Übersicht über die Komponenten	5
---	--------------------------------	---