

# Grundlagen der Webtechnologie

WS 17/18

Dieter Konrad && Dieter Schuler

## Projekt Karte

Kundenkartenverwaltungssystem

### Motivation

Ein Unternehmer z.B. Herr Feldherr möchte die Kundenbindung an das eigene Unternehmen erzielen, indem er ein Kundenkartenverwaltungssystem einsetzt. Das System ist über ein Webportal aufrufbar. Ein Kunde z.B. Herr Kunz, soll eine Möglichkeit haben eine Kundenkarte zu beantragen, die Kartenpin zu ändern und die gespeicherten Kundendaten anzusehen. Herr Feldherr möchte die vorhandenen oder vorgefertigte Kundendaten in das System übernehmen können. Außerdem benötigt Herr Feldherr für die Verwaltung der Karten einen geschützten Bereich.

### Implementierung

Das System fungiert als Webseite mit einem Datenbanksystem. Die Weboberfläche ist responsive gestaltet worden. Daher ist die Benutzerfreundlichkeit für mobile Geräte gegeben. Die Bedienung der Funktionen geschieht über das obere Menü. Die erste Seite ist ein Webformular. Es gibt eine Seite für den Kunden, erreichbar über den Menüpunkt **Kunde**. Verwaltungsseiten sind über die Menüpunkte **Login**, **Abmelden**, **Adminin** und **Personen** erreichbar. Herr Feldherr kann alle Kunden einsehen, PIN für die Person neu generieren, einzelne Personen löschen, den gesamten Datenbestand löschen und die Datenbank mit Kunden aus einer Datei neu befüllen.

## Erste Schritte mit dem Projekt karte

### Benötigte Software (Windows und Ubuntu)

- YARN: <https://yarnpkg.com/lang/en/>
- NodeJS: <https://nodejs.org/en/>
- MongoDB: <https://www.mongodb.com/download-center#community>

## Projekt Quellcode herunterladen

- <https://github.com/dkotrada/karte/archive/master.zip>
- Nach dem entpacken zum ordner navigieren: `cd karte-master`

## Projekt initialisieren und Starten

1. in der konsole: `yarn` (Initialisieren)
2. in einer anderen Konsole: `mongod --dbpath mongodb` (Datenbankspeicherort angeben)
3. in der konsole: `yarn start` (Applikation starten)
4. Folgendes im Browser öffnen: `http://localhost:3131` alternativ `127.0.0.1:3113`

## Allgemeine Informationen zu benutzten Technologien

NodeJS Application mit Express, Bootstrap, MongoDB und Mongoose

### Node.js

Node.js ist ein in C++ geschriebene Laufzeitumgebung für Chrome V8 JavaScript Maschine. Es erlaubt den JavaScript-Programmierern serverseitige Programme zu schreiben. Durch ereignisgesteuertes nicht blockierendes Eingabe/Ausgabe Model ist Node.js effizient und ressourcenschonend. Das npm Ökosystem von Node.js ist das größte Ökosystem von Open Source Bibliotheken in der Welt.

<https://nodejs.org/de/>

### Node.js Module

Die Projekte auf der Basis des Node.js sind modular aufgebaut. So übernimmt jedes Modul eine bestimmte Aufgabe. Zum Beispiel Rendern der Templates oder verwalten von Cookies. Im weiteren werden die im Projekt eingesetzten Module kurz beschreiben.

### Express.js

Express ist ein schnelles, offenes, unkompliziertes Web-Framework für Node.js.

<http://expressjs.com/de/>

## MongoDB

Eine der Führenden NoSQL Datenbanken. Ist eine Allzweckdatenbank mit offenem Quellcode. Sie hat u.a. folgende Merkmale: - Dokumentdatenmodell mit dynamischen Schemata - Umfassende, flexible Indexunterstützung und mächtige Abfragen - Aggregation-Framework und MapReduce - Auto-Sharding für horizontale Skalierbarkeit - Eingebaute Replikation für Hochverfügbarkeit - Textsuche - Erweiterte Sicherheit (z.B. Kerberos Unterstützung) - Speicherung großer Dateien mit GridFS

<https://www.mongodb.com/de>

### MongoDB Installation

Um MongoDB installieren zu können genügt die Einsicht der Anleitungen auf den folgenden folgenden Webseiten: - <https://docs.mongodb.com/manual/administration/install-community/> - <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/>

### Mongo Consolen Bedienung

- <https://docs.mongodb.com/manual/reference/mongo-shell/>
- Shell starten: `mongo --shell`
- Zeige Datenbanken: `show dbs`
- Auswählen der Datenbank: `use databasename`
- Kollektionen anzeigen: `show collections` oder `show tables`
- Inhalt aller Daten in der Kollektion anzeigen: `db.collectionname.find()`
- Gesamten Inhalt aus der Kollektion entfernen: `db.collectionname.remove({})`  
<https://docs.mongodb.com/getting-started/shell/remove/>

### MongoDB befüllen der Datenbank mit Beispieldaten

- Dokument: `karte_db`
- Kollektion: `people`
- `mongoimport --db karte_db --collection people --drop --file DATABASE_SEED.json --jsonArray`
- <https://docs.mongodb.com/getting-started/shell/import-data/>

### Verzeichnis für MongoDB angeben

- Ein weiteres Terminalfenster öffnen und folgendes Kommando ausführen:  
`mongod --dbpath mongodb/`

## Mongoose

Mongoose ist ein ODM Objekt-Datenbank-Management-System für Node.js. Erlaubt uns das Arbeiten mit der MongoDB.

<http://mongoosejs.com/>

## Hilfsmodule

- body-parser.js (Anfragebody verarbeiten)
- cookie-parser.js (Cookies verarbeiten)
- cookie-session.js (Sessionen verarbeiten)
- path.js (Pfade im Projekt setzen)
- shelljs.js (Ausführen der Schellbefehle)

## Design mit Bootstrap Framework

Bootstrap ist ein Open-Source-Toolkit zur Entwicklung responsiver Designs mithilfe von HTML, CSS und JS.

<http://getbootstrap.com/>

## Projekterstellung und Entwicklung

Um das Projekt zu Kompilieren und den Entwickler zu unterstützen werden zusätzliche Module eingesetzt. Diese sind in der Datei `packages.json` im Bereich `devDependencies` aufgeführt.

## Babeljs Kompiler

Ermöglicht das Benutzen von Ecma Script 2015 im Projekt.

## Yarn

Manager der Node.js Module, initialisieren des Projekts und starten des Servers.

## Nodemon

Überwachung der Änderungen des Quellcodes und automatisches Recompilieren der Quelldateien.

## Eigen erstellte Module

- dbcontroller.js
- personcontroller.js
- routescontroller.js
- config.js

## Dokumentation Quellcode des Projekts

Die Dokumentation zum Quellcode ist in den Kommentaren in den jeweiligen Dateien nachzulesen.

## Datenbank Eintrag Beispiel

```
{
  "title": "Dr.",
  "name": "Jung",
  "vorname": "Julia",
  "strasse": "Verwalter",
  "hausnummer": "33",
  "stadt": "Superstadt",
  "postleitzahl": "37877",
  "telefon": "5572136578",
  "email": "julia@karte.de",
  "card": {
    "pin": 0,
    "uuid": 123456
  }
}
```

## Projektstruktur nach der Installation

- **mongodb** (Ordner für die Dokumentendatenbank)
- **node\_modules** (NodeJS notwendige Module “Bibliotheken”)
- **server** (Applikationsquellcode, Logik, Design)
  - **controller** (MVC Applikationslogik)
  - **model** (MVC Datenbank)
  - **public** (Statische Dateien)
  - **views** (MVC Darstellung)
    - \* **pages** (Templates für die einzelne Seiten)
    - \* **partials** (Template Blöcke für Wiederverwendung)
  - **config.js** ( Port und Host Einstellungen )

- **index.js** ( Applikationseinstiegspunkt )
- **.babelrc** (Konfiguration für den Babeljs Kompiler “es2015”)
- **.gitignore** (Versionsverwaltungsdatei für das GIT Repositorium)
- **DATABASE\_SEED.json** (Beispieldaten für den Datenbank)
- **package.json** (Projektkonfigurationsdatei)
- **README.md** (Information zum Projekt)
- **yarn.lock** (Persistieren der Modulabhängigkeiten für das Projekt “karte”)

## Lizenzen

### Quelltexte und Dateien

Copyright 2018 Dieter Konrad && Dieter Shuler

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

**Bild von Ed Gregory**

<https://creativecommons.org/publicdomain/zero/1.0/>

## Screenshots der Anwendung

[Home](#)
[Kunde](#)
[Personen](#)
[Admin](#)
[Login](#)
[Abmelden](#)

## Karte beantragen

Benutzen Sie das unten stehende Formular um Ihre Karte zu beantragen.

Titel:
Name:
Vorname:
Strasse:
Hausnummer:
Stadt:
Postleitzahl:
Email:

Karte beantragen

© Copyright 2017 The Awesome People Dieter & Dieter

Figure 1: Startseite

Pin ändern	Eigene Daten ansehen
<div>Email eingeben</div> <div>Alte Pin eingeben</div> <div>Neue Pin eingeben</div> <div>PIN Übernehmen</div>	<div>Email eingeben</div> <div>Ihre Pin eingeben</div> <div>Ihre Daten ansehen</div>

Figure 2: Kundenbereich



Figure 3: Adminbereich

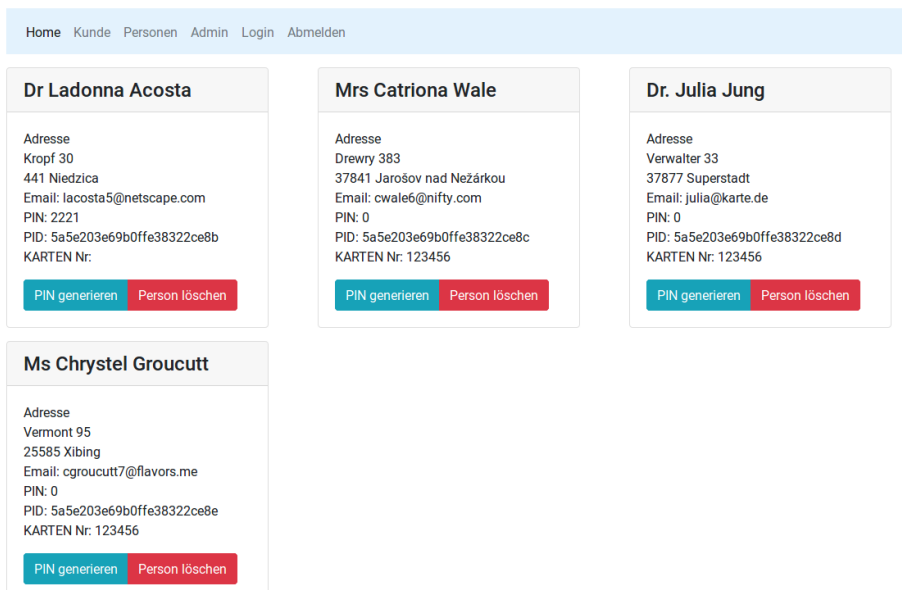


Figure 4: Personenliste

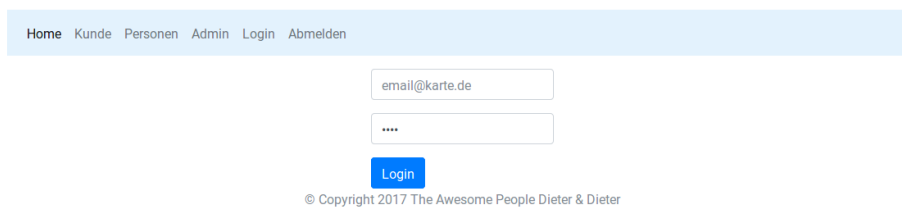


Figure 5: Loginseite