

README

0. Vorbemerkungen
1. Projektbeschreibung
2. Installationsanleitung
3. Verwendete Bibliothekn
4. Kontakt

0. Vorbemerkungen

Zusätzlich zu dieser README steht eine detaillierte README mit einer Projektbeschreibung, einer Installationsanleitung, einer API-Dokumentation und allen relevanten (Download-)Links unter

<https://github.com/minau87/meansurveys> oder alternativ im meansurveys-Verzeichnis als

README.md

zur Verfügung.

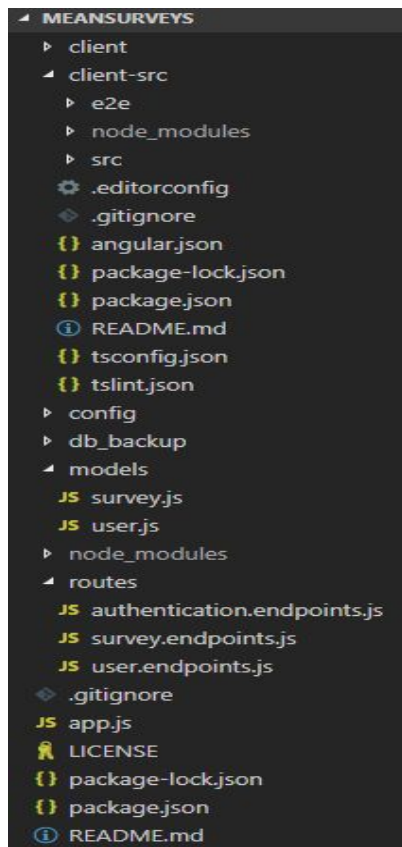
1. Projektbeschreibung

Ziel des Projektes war die Entwicklung eines Backends mit REST-Service und Datenbank sowie eines Client für eine Anwendung zur Durchführung von Umfragen.

Dabei waren folgende Anforderungen umzusetzen:

- Erstellung von Fragebögen: Ein Satz Fragen mit Antwortmöglichkeiten
- Erstellung von Umfragen: Basieren auf Fragebögen, die für einen vorgegebenen Zeitraum von Umfrageteilnehmern beantwortet werden können
- Eingaben je Umfrage und Teilnehmer
- Fragebögen (und damit Umfragen) sollen von Admin erstellt werden können
- Teilnehmer können Antworten für laufende Umfrage abgeben

Der Aufbau des Projektes gestaltet sich wie folgt:



(Abb. 1: Projektstruktur)

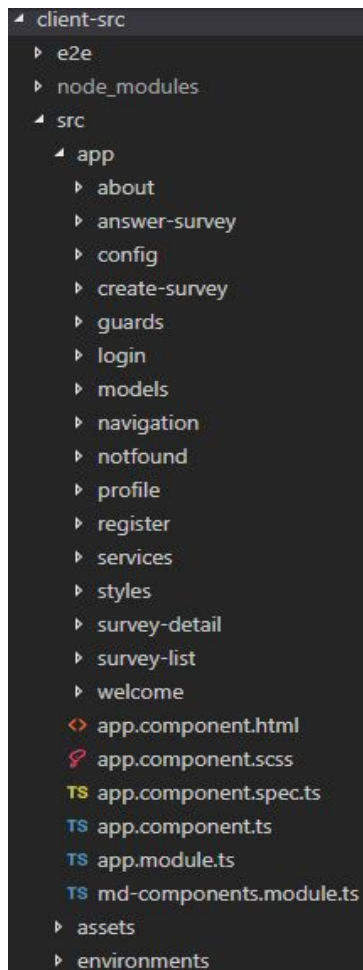
Die Anwendung besteht als Full-Stack-Applikation realisiert mit dem MEAN-Stack aus den folgenden Softwarekomponenten:

- *MongoDB*
- *Express*
- *Angular*
- *Node.js*

MongoDB dient der Datenhaltung und bei Node.js handelt es sich um eine Javascript-Runtime, die es ermöglicht, Javascript auch serverseitig ausführen zu können. Auf diese Grundlage stützt sich Express als Node.js-Framework, dass eine Reihe nützlicher Funktionalitäten zur Entwicklung von Web-Anwendungen liefert. Mit Express wurden im Rahmen dieses Projektes die API-Endpunkte entwickelt und bereitgestellt. Angular ist ein Framework zur Entwicklung von Single-Page-Applikationen, mit dem hier das Frontend der Gesamtanwendung entwickelt wurde.

Abb.1 zeigt den Aufbau des Projektes. Sowohl *app.js*, das *routes*-, das *config*- und das *models*-Verzeichnis bündeln den Code für das Backend. Hier wird u.a. definiert, welche API-Endpunkte durch die Anwendung bereitgestellt werden und wie die Anwendung reagiert (z.B. auf das Speichern einer neuer Umfrage, das Anlegen eines Benutzers, ...), wenn die einzelnen API-Endpunkte beansprucht werden.

Das *client-src*-Verzeichnis (vgl. Abb. 2) bündelt den Code des Frontends.



(Abb. 2: Projektstruktur des Client)

Im Unterverzeichnis *app* befinden sich sämtliche Komponenten der Client-Applikation, sowie die Services, die mit dem Backend kommunizieren und Models, durch die die vom Backend empfangenen Daten clientseitig repräsentiert werden.

2. Installationsanleitung

Für die Installation der Anwendung gibt es zwei Herangehensweisen:

- 1) Installation aus der .zip-Datei
- 2) Installation via GitHub (Link unter 0. Vorbemerkungen)

2.1 Vorbedingungen

Die folgenden Schritte müssen für eine korrekte Installation der Anwendung vorab erledigt sein:

- Installation von MongoDB (<https://www.mongodb.com/download-center#community>)
- Installation von Node.js (<https://nodejs.org/en/>)

Sollte die Konfiguration von MongoDB jener entsprechen, wie sie auch in der Veranstaltung genutzt wurde, sollte dies ausreichen. Dann ist lediglich Node.js zu installieren.

2.2 Installation aus der .zip-Datei

1. Entpacken Sie den in der .zip-Datei enthaltenen *meansurveys*-Ordner an einen Ort Ihrer Wahl.
2. Öffnen Sie hier ein Kommandozeilenfenster und führen folgenden Befehl aus:
`npm install`
3. Öffnen Sie hier ein Kommandozeilenfenster und führen folgenden Befehl aus:
`npm install -g nodemon`
4. Navigieren Sie in den *client-src*-Ordner und öffnen ein neues Kommandozeilenfenster. Führen Sie den folgenden Befehl aus:
`npm install`
5. Anschließend ist das Angular Command Line Interface (CLI) herunterzuladen:
<https://cli.angular.io/>

Beide `npm install`-Befehle sorgen dafür, dass die benötigten Dependencies der Anwendungen heruntergeladen werden.

2.3 Installation über GitHub

1. Installieren Sie Git (<https://git-scm.com/downloads>)
2. Navigieren Sie an einen Ort Ihrer Wahl und öffnen ein Kommandozeilenfenster. Geben Sie anschließend folgenden Befehl ein:
`git clone https://github.com/minau87/meansurveys.git`
3. Die Anwendung wird nun heruntergeladen und das *meansurveys*-Verzeichnis erstellt
4. Fahren Sie fort mit den Schritten 2 - 5 aus Abschnitt 2.2

2.4 Starten der Anwendung im Entwicklungsmodus

- Navigieren Sie in den *meansurveys*-Ordner. Öffnen Sie hier ein Kommandozeilenfenster und geben folgenden Befehl ein:
`nodemon`
(Sollte es bei diesem Schritt zu Fehlern kommen, versuchen Sie vorab eine MongoDB-Instanz via *mongod* zu starten)
- Navigieren Sie in den *client-src*-Ordner, öffnen ein Kommandozeilenfenster und geben folgenden Befehl ein:
`ng serve`
- Ist die Anwendung gestartet, öffnen Sie den Browser und geben in die Adresszeile <http://localhost:4200> ein.

Das Starten der Anwendung auf diese Art bewirkt, dass einerseits Änderungen am Client-Code dazu führen, dass die Anwendung automatisch neu lädt. Gleiches gilt, dank *nodemon*, auch für den serverseitigen Code.

2.5 Starten der Anwendung im Produktionsmodus

- Unabhängig davon, ob Sie die Anwendung über die .zip-Datei oder GitHub bezogen haben, liegt ein fertiges Build der Anwendung im *client*-Verzeichnis
- Beabsichtigen Sie nicht, testweise Änderungen am Code vorzunehmen, wird dazu geraten, die Anwendung folgendermaßen zu starten:
 - Navigieren Sie in das *meansurveys*-Verzeichnis, öffnen ein Kommandozeilenfenster und geben folgenden Befehl ein:

npm start

- Die Anwendung startet. Ist die Anwendung gestartet, öffnen Sie Ihren Browser und geben in die Adresszeile <http://localhost:3000> ein.

2.6 Admin user und das Laden vordefinierter Testdaten

Im Normalfall führt die Registrierung eines neuen Nutzers nicht dazu, dass dieser Administratorberechtigungen besitzt. Zwischen Administratoren und normalen Nutzern wird innerhalb dieses Projektes nur dadurch unterschieden, dass das user-Dokument innerhalb der MongoDB-Datenbank für einen Administrator lediglich ein weiteres Schlüssel-Wert-Paar umfasst.

An dieser Stelle stehen Ihnen zwei Optionen zur Verfügung:

- Neuanlegen eines Benutzers über den Browser und anschließende manuelle Ergänzung des Schlüssel-Wert-Paares *"role"* : *"admin"*, z.B. über die MongoDB-Shell
- Laden des mitgelieferten Backups der MongoDB-Datenbank

Letzteres Vorgehen wird empfohlen, da so unmittelbar zwei Nutzer (Benutzername: admin, Passwort: 123456; Benutzername: user, Passwort: 123456) und eine Umfrage in ihrer lokalen MongoDB angelegt werden.

Haben Sie den /bin-Pfad ihrer MongoDB-Installation zu ihren Umgebungsvariablen hinzugefügt, können Sie über den Befehl *mongorestore --db meansurveys db_backup/meansurveys* (über ein Kommandozeilenfenster im *meansurveys*-Verzeichnis) das Backup laden. Alternativ können Sie in den /bin-Pfad ihrer MongoDB-Installation wechseln um *mongorestore* ausführen zu können, dann ist jedoch der Pfad zum Backup entsprechend anzupassen.

3. Verwendete Bibliotheken

Die Installation von Dependencies, d.h. Abhängigkeiten wie z.B. Bibliotheken, wird in Node.js-Umgebungen oft durch die Verfügbarkeit des Node Package Managers automatisiert. In diesem Zusammenhang sind die beiden *package.json*-Dateien (einmal im *meansurveys*- und einmal im *client-src*-Verzeichnis) von Bedeutung. Sie sammeln alle notwendigen Dependencies. Wird bspw. eine Anwendung, wie im vorliegenden Fall, neuinstalliert, reicht dann der Bezug der Dependencies mittels des Befehls *npm install*, der ein Herunterladen aller Abhängigkeiten auslöst. Zu diesen Dependencies zählen für dieses Projekt z.B. Angular und Express, aber auch Highlight.js, dass vereinzelt für ein Code-Highlighting von JSON-Objekten genutzt wurde. Dies ist - auch, wenn es keinen direkten Projektbezug gibt - u.a. dann interessant, wenn man beobachten möchte, wie sich derartige Objekte bei Ihrer Manipulation (vgl. bei der Erstellung einer neuer Umfrage über die Anwendung) verändert. Durch eine Bibliothek wie Highlight.js wird dies deutlich nachvollziehbarer.

Eine vollständige Auflistung aller Abhängigkeiten dieses Projektes kann daher den beiden o.g. *package.json*-Dateien entnommen werden.

4. Kontakt

Bei Fragen zum Projekt oder eventuell auftauchenden Problemen, wenden Sie sich bitte jederzeit per Mail an:

lukas.franken@stud.hn.de
michel.nauen@stud.hn.de

oder