

UserTable-Project

jan.schulz@digitalcareerinstitute.org

Aufgabe 1: Layout

120px



UserTable

Show All Users | Search

ID	Name	Email
1	Stefan	stefan@abx.net
2	Lara	lara@gmx.net
3	Julia	julia@gmail.com
4	Karsten	karsten@foo.net
5	Peter	peter@xing.com
6	Steffi	steffi@xing.net

Aufgabe 2: /users

1. Implementiere eine Route /users die alle Users aus der users.json zurückgibt. Allerdings sollen nur die id, name und email herausgegeben werden. Öffne dafür die users.json mit einem **asynchronen** fs.readFile – Befehl und parse die erhaltenen Daten.
2. Wenn die Webseite lädt, lade nun alle Userdaten über das Backend /users.

Aufgabe 3: Layout Userdetail-Seite

UserTable	
Show All Users Search	
	<p><u>Steffi</u></p> <p>Id: 6</p> <p>Email: steffi@gmx.net</p> <p>Lorem30</p>

Aufgabe 4: /userdetail/:id


1. Implementiere eine Route /userdetail die alle Informationen über den Nutzer mit der Id = :id wiedergibt.
2. Auf der Startseite soll ein onClick für jede Row in der Tabelle erstellt werden. Mit einem Klick auf den jeweiligen Nutzer soll die Detailseite angezeigt werden.

Aufgabe 4

UserTable

Show All Users | Search

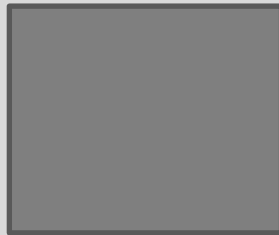
ID	Name	Email
<u>1</u>	<u>Stefan</u>	<u>stefan@abx.net</u>
<u>2</u>	<u>Lara</u>	<u>lara@gmx.net</u>
<u>3</u>	<u>Julia</u>	<u>julia@gmail.com</u>
<u>4</u>	<u>Karsten</u>	<u>karsten@foo.net</u>
<u>5</u>	<u>Peter</u>	<u>peter@xing.com</u>
<u>6</u>	<u>Steffi</u>	<u>steffi@xing.net</u>



Aufgabe 4

UserTable

Show All Users | Search



Peter

Id: 5

Email: peter@xing.com

Lorem30

Aufgabe 5: Navigations-Feedback

1. Füge ein visuelles Feedback hinzu, auf welcher Seite sich der Nutzer gerade befindet. Wenn der Nutzer z.B.
 1. die Seite lädt, soll „Show All Users“ fett und unterstrichen sein.
 2. Analog für den zweiten Menüpunkt „Search“

Aufgabe 6: /users?q=peter

1. Für deine Resource /users auf dem Backend soll eine Suchfunktion implementiert werden. Implementiere dazu einen Check, ob req.query.q existiert und falls ja, gib alle Nutzer zurück deren Name, Email oder Beschreibung ein Oberstring von von q ist.
2. Teste Deine Implementierung indem Du den Link <http://localhost:3000/users?q=peter> direkt auf Deinem Browser ausführst.

Aufgabe 7: Suchfunktion

120px

UserTable

Show All Users | Search

Suchbegriff eingeben ...

Suchen

Aufgabe 7: Suchfunktion

UserTable

Show All Users | Search

stef

Suchen

Aufgabe 7: Suchfunktion

UserTable

Show All Users | Search

ID	Name	Email
<u>1</u>	<u>Stefan</u>	<u>stefan@abx.net</u>
<u>6</u>	<u>Steffi</u>	<u>steffi@xing.net</u>

Aufgabe 8: Nutzer anlegen

1. Erstelle eine neue POST-Resource /users, die einen JSON-Body mit den Keys erwartet.
 - name
 - email
 - description
 - profilePic
2. Generiere in Deinem Server eine ID als randomstring von 10 Zeichen und gib sie dem Nutzer-Objekt als id.
3. Das neue Nutzer-Objkt soll in die Datei users.json hinzugefügt werden. Dafür solltest Du die users.json
 1. Öffnen
 2. Parsen
 3. Das Nutzer-Objekt dem neuen Nutzer hinzufügen
 4. Stringifien
 5. Speichern
4. Das Nutzer-Objekt soll nun zurückgeschickt werden. Bitte denke daran, die generierte Id mitzuschicken. Falls es irgendeinen Fehler gab, schicke zurück:


```
{ error: "Fehler" }
```
5. Teste Deine Implementierung mit Postman.

Aufgabe 8: Nutzer anlegen

UserTable

Show All Users | Search | Nutzer anlegen

Name

Email

Description

Dateiname Foto

Erstellen

Aufgabe 8: Nutzer anlegen Frontend

1. Erstelle einen neuen Navigationspunkt “Nutzer anlegen” im Header-Bereich.
2. Erstelle eine neues Formular “Nutzer anlegen” wie auf dem nächsten Slide.
3. Wenn der Nutzer auf den Button “Erstellen” klickt, soll
 1. Validiert werden, dass keines der Felder leer ist.
 2. Validiert werden, dass die Email-Adresse auch eine richtig Email-Adresse ist.
 - Einfach Version: schauen, dass genau ein @-Zeichen enthalten ist
 - Schwere Version: Google/Stackoverflow
 3. Ein POST-Request an /users geschickt werden, mit einem Body so ähnlich wie:

```
{  
  "name" : "Lars",  
  "email" : "lars@abc.com",  
  "description" : "Lars geht gerne Spazieren mit seinem Hund. Am liebsten tun sie das im Sommer wenn die Sonne scheint.",  
  "profilePic": "lars.jpg"  
}
```

Aufgabe 8: Nutzer anlegen Frontend

4. Wenn es keinen Fehler vom Backend gab (kein error) , soll die Start-Ansicht gezeigt werden inklusive dem neu hinzugefügten Nutzer.

Optional Aufgabe 9 (schwer): Bild hochladen

1. Implementier in Deinem Backend eine neue POST-Resource `/imageupload`, die eine Datei empfängt. Schaue Dir dafür selbstständig die Express-Library `express-fileupload` an.
2. Implementiere den Frontendpart mit dem `<input type="file"/>` Tag.

Aufgabe 9: Nutzer anlegen

UserTable

Show All Users | Search | Nutzer anlegen

Name

Email

Description

Dateiname Foto

Dateiname Foto

Erstellen