

Web-Technologien, Sommersemester 2025

Teilleistung 2

Abgabe der Lösungen

- *Abgabetermin.* **03. Juni 2025, 23:55 Uhr**
- *Abgabemodus.* Ordner als ZIP-Datei im Virtuellen Campus im entsprechenden VC-Kurs des Moduls. Bei einer Bearbeitung im Team, genügt die Abgabe durch ein einzelnes Teammitglied. Die zuletzt hochgeladene Abgabe wird bewertet.
- *Dateiformat der Abgabe.* Die Abgabe erfolgt durch den Upload des ZIP-Archivs, das alle notwendigen Dateien beinhalten soll.
- *Update der Lösung.* Bis zur oben angegebenen Deadline könnt Ihr Eure Lösung beliebig oft durch eine neue (korrigierte) Fassung ersetzen. Zuvor hochgeladene Lösung werden dabei überschrieben. Wir haben keine Möglichkeit, alte Fassungen wiederherzustellen!

Hinweise

- Die vollständige und korrekte Bearbeitung einer Aufgabe ergibt die volle Punktzahl dieser Aufgabe.
- Für diese Teilleistung erwarten wir einen Bearbeitungsaufwand von insgesamt etwa 45 Stunden, da wir von vier Personen pro Team ausgehen. Dies ist als Richtwert zu verstehen und keinesfalls als Unter- oder Obergrenze.
- In Verdachtsfällen behalten wir uns vor, Plagiate von der Bewertung auszunehmen. Wir werden in einem solchen Fall Kontakt mit den Betroffenen aufnehmen.
- Programmieraufgaben sollen in nachvollziehbar kommentierter Form im Quelltext abgegeben werden. Dabei dürfen keine anderen als die angegebenen Technologien verwendet werden. Nicht lauffähige Programme können nicht bewertet werden. Der Abgabe soll eine README-Datei hinzugefügt werden, die beschreibt, wie man die Lösung ausführen kann.
- Bei der Auswahl der Medien sind keine Urheberrechte zu berücksichtigen, da eine Veröffentlichung der Teilleistung nicht vorgesehen ist.

Aufgabe: Erstellung einer Website zur Festival-Organisation mittels HTML, CSS und JavaScript (30 Punkte)

Die Werbung für das Rock am See Festival ist viral gegangen. Die Anzahl der verkauften Tickets übersteigt alle Erwartungen. Deshalb werden nun viele Helfer:innen benötigt, um das Festival stemmen zu können. Dafür soll eine eigene Website entwickelt werden, sodass die Helfer:innen optimal den einzelnen Events auf dem Festival zugeordnet werden können.

Die neue Webanwendung soll eine **Single-Page-Application** sein, d. h. nur eine einzige HTML-Datei umfassen, auf der alle Informationen zu den Helfer:innen zu finden sind.

Auf der Website soll es eine Liste mit allen Helfer:innen geben. Außerdem ist eine Übersicht nötig, welche Daten grundsätzlich zur Verfügung stehen. Zudem soll hier gezeigt werden, welche Events auf dem Festival bestimmte Anforderungen bzgl. der Eigenschaften der Helfer:innen voraussetzen.

Abbildung 1 zeigt die Struktur der Single-Page-Application. Die weiteren Anforderungen an Inhalt, Design, Layout und Funktionalität werden im Folgenden aufgelistet.

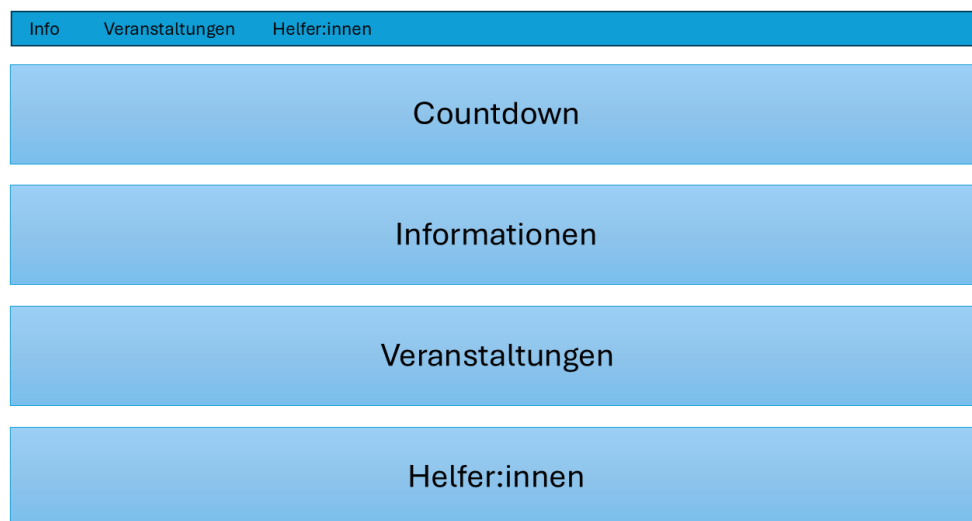


Abbildung 1: Struktur der Single-Page-Application

Zugriff auf die API

Auf die API kann mithilfe von GET-Requests zugegriffen werden. Die Informationen zu den Helfer:innen sind unter <https://randomuser.me/api/?results=30&seed=a> verfügbar.

Um sich einen Überblick über die Informationen der API zu verschaffen, kann es hilfreich sein, die API-Antwort zunächst in Firefox oder auch mit Tools wie Postman¹ oder Insomnia² anzusehen. Diese Tools ermöglichen es, die Antwort strukturiert darzustellen und erleichtern Euch so das Verschaffen eines Überblicks.

Weitere Anforderungen

- Oben auf der Seite soll ein Countdown zu sehen sein, der die Tage runterzählt bis zum Start des Festivals. Darunter folgen drei Abschnitte: Informationen, Events und eine Liste der Helfer:innen.
- Es soll eine Navigationsleiste geben, über die man mithilfe von HTML-anchors jeweils zu den drei Abschnitten springen kann. Diese Leiste soll immer am oberen Bildschirmrand angezeigt werden, egal wo man sich auf der Webseite aufhält.
- Der erste Abschnitt soll Informationen dazu enthalten, welche Daten von den Helfer:innen zur Verfügung stehen.
- Der nächste Abschnitt besteht aus einer Tabelle, in der links die Events auf dem Festival stehen. In der Mitte folgt eine kurze inhaltliche Beschreibung und rechts findet man, welche Merkmale die Helfer:innen erfüllen sollen (wie bspw. Mindestalter). Euch sind hier bei der Veranstaltungsart keine Grenzen gesetzt, einzig die Merkmale sollen auch in der API vorhanden sein. Es sollen zunächst ca. drei Events aufgelistet werden, aber so, dass die Tabelle theoretisch jederzeit erweitert werden könnte.
- Der letzte Abschnitt soll alle Helfer:innen in einer Liste darstellen. Es sollen die von der API zur Verfügung gestellten Daten dazu mithilfe der Fetch API abgerufen und im Webaufttritt angezeigt werden. Aber Achtung: Nur folgende Eigenschaften sollen angezeigt werden: *Name, Geschlecht, Email-Adresse, Geburtsdatum, Registrierungsdatum, Telefon, Handy, ID und Bild*. Letzteres soll in einer angemessenen Größe dargestellt werden. Auch hier wurde als Orientierung ein grobes Mockup erstellt 2. Das Mockup gilt nur als grobe Richtlinie, ihr könnt die Liste natürlich auch anders gestalten.
- Zudem soll es die Möglichkeit geben, die Liste nach Name und dem Registrierungsdatum zu sortieren. Dabei soll man zwischen aufsteigender und absteigender Sortierung wählen können.
- Die `Helfer:innen` Liste ist im *ISO-8601*-Format³ angeben und kann mittels des `Date()`⁴ Konstruktors von JavaScript verarbeitet werden. Dazu hier ein Beispiel:

¹<https://www.postman.com>

²<https://insomnia.rest>

³https://de.wikipedia.org/wiki/ISO_8601

⁴https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date/Date

```
const date = new Date("2025-07-05T12:00");
const formattedDate = date.toLocaleDateString("de-DE", {
  minute: "numeric",
  hour: "numeric",
  day: "numeric",
  month: "long",
  year: "numeric",
});
console.log(formattedDate) // 5. Juli 2025 um 12:00
```

- Für den Fall, dass die API nicht erreichbar ist, soll eine sinnvolle und verständliche Fehlermeldung an passender Stelle angezeigt werden.
- Während Daten von der API abgefragt werden, soll eine Ladeanzeige dargestellt werden. Die Ladeanzeige soll ausschließlich mit CSS implementiert werden. Ein paar Beispiele sind unter <https://codesandbox.io/s/luoos?file=/loaders/circular-trails/circular-trails.css> zu finden.
- Der Webauftritt soll ein klares, übersichtliches und einheitliches Design besitzen. Buttons sollen also bspw. leicht als solche erkennbar sein.

Helfer:innen

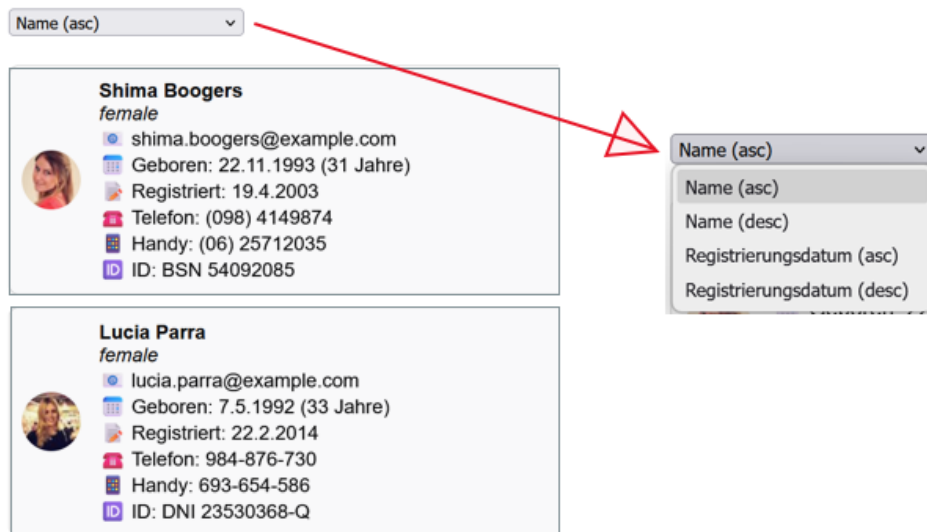


Abbildung 2: Mockup der Helfer:innen-Liste.

**Wir wünschen Euch viel Erfolg
bei der Bearbeitung der Teilleistung!**