

# Sitzungsprotokoll: Technologie- & Framework-Meeting, 31.03.2021

## TOP

1. Nächstes Treffen
  - Mittwoch, 14.04.
  - <https://fau.zoom.us/j/92803652912?pwd=VEp6NFAzbXhrRkpCMmNxbXFqK1J3dz09>
2. Umgang mit Technologien
3. Eigenen Demodatensatz
  - Jeder überlegt sich Text (bspw. Sätze, Schlagworte, Emojis, oder andere Formate) mit denen man sich selbst beschreiben würde
  - Jeder überlegt sich Fragestellungen, mit denen man nach solchen Daten fragen könnte.
  - Jeder überlegt sich die Schlagwörter, die man sich selbst geben würde
  - Jeder überlegt sich die Schlagwörter, nach denen man suchen würde
  - -> Jeder für sich in einer Textdatei o.Ä., dann nächstes Mal besprechen.
  - Wenn wir die Fragen und das Format haben, dann auch weitere Personen fragen.
4. Webdesign
  - Kris, Matthias, Patrick: In 2 Wochen ein CSS-Framework und JS-Framework auswählen

## JS-Framework

- Angular
  - Für Bewerbungen evtl. interessant zu können, weil viele Stellenausschreibungen
  - Stability <https://angular.io/guide/releases>
    - Mehrmals im Jahr Breaking Changes durch Major Updates
  - Nur für Web-Applikationen da, keine Anleitungen dazu, wie man einzelne Angular-Komponenten in Nicht-Angular-Websites einbauen kann?
  - Sehr Typescript-lastig. Ist das ein Vorteil? Ist das ein Hindernis?
- React
  - Ressourcen

- Tutorial: <https://reactjs.org/tutorial/tutorial.html>
- Architekturdiagramm: Keins?
- Requirements:
  - Build mit npm auf Entwicklerrechner
  - Building for Relative Paths / By default, Create React App produces a build assuming your app is hosted at the server root.
  - React-Komponenten in Statischer MPA einbauen: <https://reactjs.org/docs/add-react-to-a-website.html>
- Beispiel mit Python: <https://www.digitalocean.com/community/tutorials/build-a-to-do-application-using-django-and-react>
- Stability <https://reactjs.org/docs/design-principles.html#stability>
  - Größere Breaking Changes zuletzt 2017
  - Letztes Major Release 2020; It Facebook aber fast nicht relevant (“In practice, we’ve only had to change fewer than twenty components out of 100,000+”)
- Vue
  - <https://vuejs.org/v2/guide/>
  - Stability:
    - 2.0 veröffentlicht in 2016
    - 3.0 veröffentlicht in 2020
  - Requirements
    - Funktioniert auch ohne CLI-Tool, und ohne npm, wenn man sich den node-Stack nicht antun möchte – so auch die offizielle Empfehlung für Anfänger:
      - <https://vuejs.org/v2/guide/#Getting-Started>
      - <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
  - Ein Vue-Objekt wird an einen Container gebunden, dadurch vermutlich auch hier wie bei React mehrere unabhg. Komponenten auf der gleichen Website unproblematisch
    - <https://codesandbox.io/s/stupefied-tdd-sivrd?file=/index.html>

## Vergleiche

- <https://towardsdatascience.com/vue-vs-react-vs-angular-114718eac3c5>
  - Vue einfacher
    - Würde Phillip nach kurzem Ausprobieren bestätigen – minimaler Overhead, um es zum laufen zu bekommen:
      - <https://codesandbox.io/s/upbeat-maxwell-7dmp7?file=/index.html:593-643>
    - Geht mit anderen Sprachen auch einigermaßen, wenn man das nicht auf der offiziellen Homepage im Tutorial sucht:
      - <https://dev.to/luispa/lets-try-react-without-nodejs-3a7>

- <https://github.com/antsmartian/hello-world-angular-2/blob/master/index.html>
- Angular umfangreich, aber auch viel Schreibaarbeit und anspruchsvoll. Fokus auf ganzen Anwendungen
- <https://vuejs.org/v2/guide/comparison.html>
  - React erschwert Anpassungen an HTML, da alle Templates in JS; aber bessere Toolunterstützung dafür vorhanden
  - Vue nutzt auch HTML-Templates, dadurch Arbeit für Designer einfacher.
  - Angular hat Restriktionen zur Strukturierung von Anwendungen

Abseits:

- Multi-Page-Applikation mit statischem Rendering der eigentlichen Inhalte (bspw. Benutzerprofile), um evtl. Probleme mit Suchindizierbarkeit und Zugänglichkeit zu vermeiden?
- JS für einzelne Komponenten der Seiten nutzen?
  - Suchbox
  - Suchergebnisanzeige mit Filterfunktionen oder Suchbegriffs-Eingabefeld
  - Profileditor

# Design

- <https://getbootstrap.com/>
- <https://materializecss.com/>
- <https://bulma.io/>
- <https://semantic-ui.com/>
- <https://purecss.io/grids/>

---

Revision #2

Created 2021-04-06 16:41:11 UTC by Christian Borss

Updated 2021-11-12 12:55:05 UTC by Christian Borss