



Web-Programmierung WWI21SEB

Aaron Schweig, SAP

Non-Business

Wer bin ich?

Aaron Schweig, 25 Jahre

- Seit 2016 freiberufliche Tätigkeiten im Bereich Webentwicklung
- 2018-2021 Bachelor Wirtschaftsinformatik
- Seit 2021 Fullstack Development @SAP

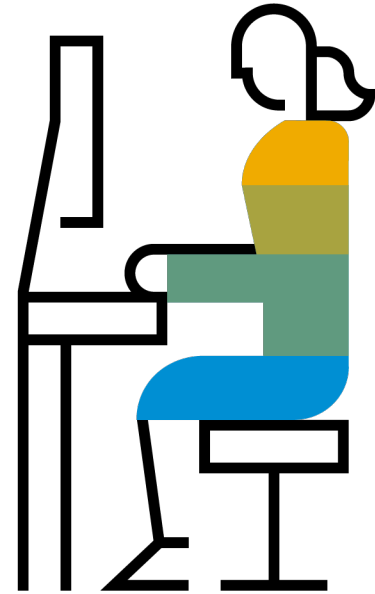
Hobbies:

- Musik (Klavier, Gitarre, etc.)
- Videospiele



Wer seid ihr?

- Name, Alter, Firma
- Hobbies
- Erfahrung mit Web-Development (privat, in der Firma, etc.)
- **Was erhoffe ich mir von der Vorlesung?**



Prüfungsleistung: Portfolio (70 P.)

1. Projekt: Entwicklung einer Webanwendung

- **Abgabe 14.07.2023:** Code als Git-Repository

2. Dokumentation

- Projektidee
- Anforderungen
- Wireframes / Mockups
- Architektur (Technologien, Gründe für die Auswahl von Technologien, Diagramme etc.)
- **Abgabe 14.07.2023:** PDF

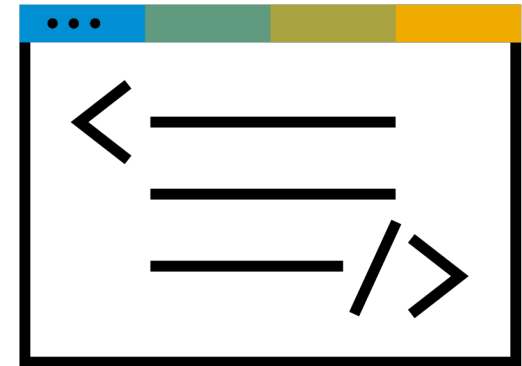
3. Präsentation

- In der letzten Vorlesung (14.07.2023)
- Individuelle Leistungen müssen erkennbar sein
- Gruppengröße: 5 – 6 Studierende



Voraussetzungen

- [GitHub](#) Account
- Entwicklungsumgebung
 - [Visual Studio Code](#) (recommended)
 - WebStorm
- Arbeiten mit [GIT](#) als Version Control System (VCS)
 - <https://github.com/git-guides>
 - <https://docs.github.com/en/get-started/quickstart/hello-world>



Das Vorlesungsrepository klonen

1

```
git clone https://github.com/aaronschweig/wwi21seb.git
```

2

Repository in eurer Entwicklungsumgebung öffnen

Agenda

1

Einführung und Grundlagen

Grundlegende Einführung in die Webentwicklung und das Ökosystem

2

Architektur- und Kommunikationspattern

Muster und Best Practices bei der Entwicklung von Webanwendungen

3

Fortgeschrittene Frontend-Entwicklung

Betrachtung aktueller Frameworks am Beispiel von Svelte

4

Authentifizierungsmechanismen

Mechanismen zur Authentifizierung von Nutzern, Standards wie OAuth2, OIDC, etc.

Agenda

5

Testen von Webanwendungen

Webentwicklungsprojekte automatisiert testen

6

Deployment / Hosting von Webanwendungen

Überblick über Möglichkeiten zur Bereitstellung von Webanwendungen

7

Projektvorstellung 🎉

Vorstellung der Prüfungsleistung in der Vorlesung



Web-Programmierung Einführung und Grundlagen

Non-Business

Agenda

1.1

**HTML, CSS &
JavaScript**

+ Übung

1.2

NodeJS, (p)npm

1.3

TypeScript

1.4

**Nächste
Vorlesung &
Fragen**

1.1

HTML, CSS & JavaScript

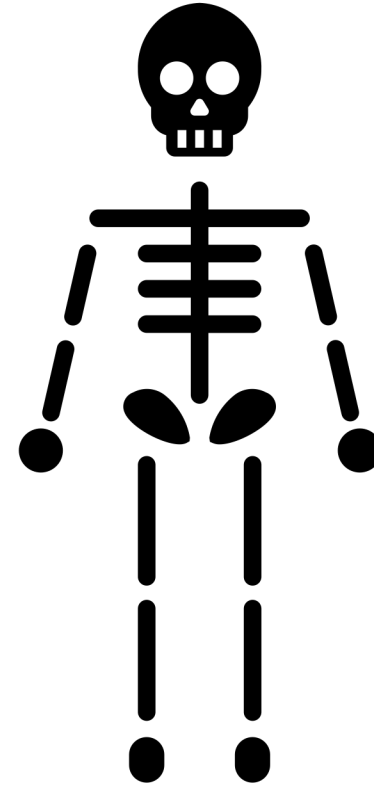
Grundlegende Technologien zur Entwicklung von Webanwendungen

Non-Business

HTML

HyperText Markup Language

- Das *Skelett* einer Website / Webanwendung
- Dient zur **Strukturierung** der Inhalte einer Website
 - Durch die Nutzung von **Tags**
- *Beispieltags:*
 - `<div> </div>`
 - `<h1> </h1>`
 - ...



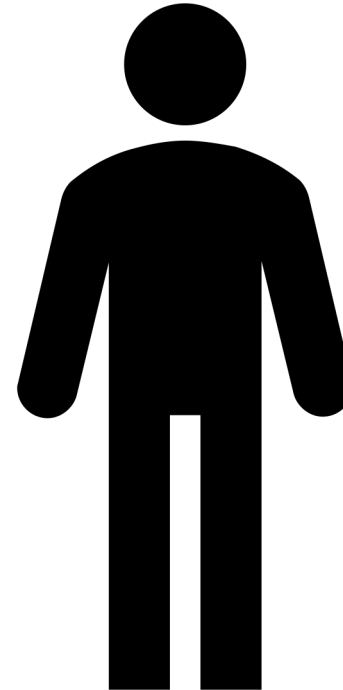
HTML Beispiel

```
<!DOCTYPE html>
<html>
  <head>
    <title>Das ist ein Titel</title>
  </head>
  <body>
    <div>
      <p>Hallo WW21SEB!</p>
    </div>
  </body>
</html>
```

CSS

Cascading Style Sheets

- Die Gestaltung einer Website / Webanwendung
- Dient zur Anpassung der Darstellung des HTML-Skeletts
 - Layout
 - Farben
 - Typographie
- *Beispielklassen:*
 - `display: flex`
 - `background-color: green`
 - ...



CSS Beispiel

```
<!DOCTYPE html>
<html>
  <head>
    <title>Das ist ein Titel</title>
  </head>
  <body>
    <div>
      <p style="color: red">Hallo WW21SEB!</p>
    </div>
  </body>
</html>
```

Inline Styles

```
p {
  color: red;
}
```

Separate CSS Klassen

JavaScript

- Die Programmiersprache für Webentwicklung
 - Wird durch alle Browser unterstützt
- Definiert das Verhalten einer Webanwendung
- Standardisiert als ECMAScript



JavaScript Beispiele

Variablen (nicht typisiert 😬)

```
let y;  
y = 10;  
const z = "this value cannot be reassigned!";
```

Funktionen

```
function foo(n) {  
  return n + 1;  
}
```

Mit einem DOM interagieren

```
const myElem = document.createElement('span');  
  
myElem.classList.add('foo');  
myElem.id = 'bar';  
myElem.setAttribute('data-attr', 'baz');  
  
document.body.appendChild(myElem);  
  
document.querySelector('.class');
```

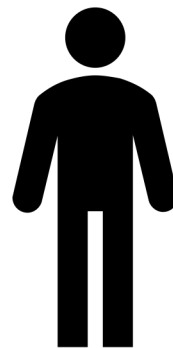
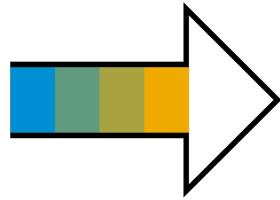
Klassen, Prototyping ...

Übung 1

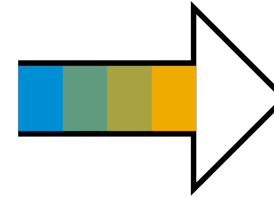
- **Erstellt eine Website über euch selbst.**
 - Name, Alter, Bild (freiwillig)
 - Liste mit Hobbies / Erfahrung mit Webentwicklung
 - Neue Hobbies und Erfahrungen können über Textfeld mit Button hinzugefügt werden



HTML



+ CSS



+ JavaScript

- Website in einem Git-Repository speichern / commiten

1.2

NodeJS, NPM und Bundling

Bereitstellung von Webanwendungen, Einbindung von 3rd Party Code

Non-Business

NodeJS, (P)NPM and Bundling

- plattformübergreifende Open-Source-JavaScript-Laufzeitumgebung
- JavaScript kann außerhalb des Webbrowsers ausgeführt werden
- Basiert auf Googles V8 JavaScript engine
- <https://nodejs.org/en>



(P)NPM

- **N**ode **P**ackage **M**anager
- Paketmanager für die JavaScript-Laufzeitumgebung Node.js
- Erlaubt Veröffentlichung und Konsumierung von 3rd Party JavaScript Paketen im eigenen Code
- <https://www.npmjs.com/>
- <https://pnpm.io/>



Bundling

- Verkleinern und Optimieren von Assets
- Fingerprinting von Assets für besseres Caching
- Erzeugen client-spezifischer Assets
- Unterstützen neuer JS-Features
- Einbinden von Polyfills
- Transpilieren von Code
- Beispiel: <https://vitejs.dev/>



Was ist TypeScript?

- **Strikt typisierte** Programmiersprache, die auf JavaScript basiert
- Besseres Tooling (IDE Autocompletion, Compile Time Errors, besseres Refactoring, etc)
- Superset von JavaScript → Jeder valide JavaScript Code ist auch valider TypeScript Code
- Danke modernen Tools wie Bundlern und Deno sehr weit verbreitet



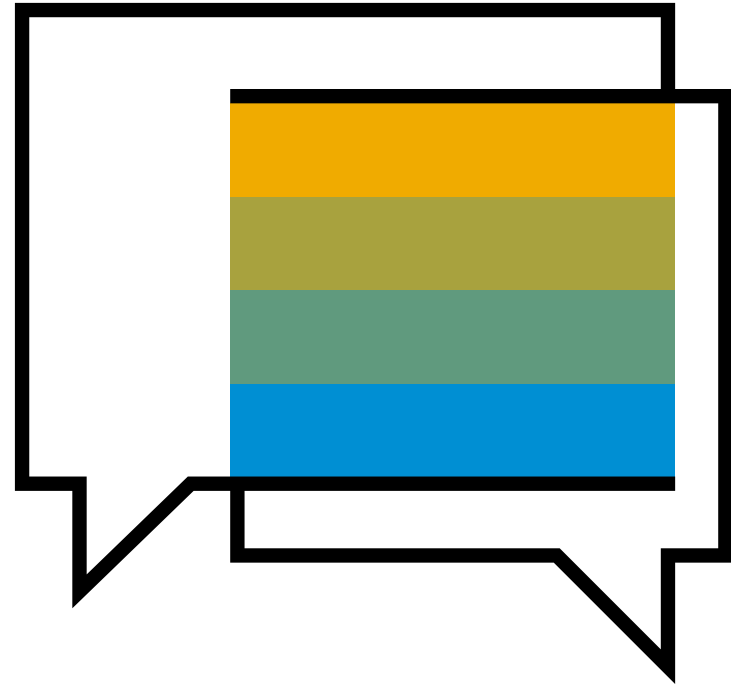
Nützliche Dokumentation/Links zu Web-Standards

- MDN <https://developer.mozilla.org/en-US/>
- W3Schools <https://www.w3schools.com/tags/default.asp>
- IETF standards <https://datatracker.ietf.org/> (useful in the later parts of the lecture)

Aufgaben bis zum 26.05.2023

- Gruppen selbstständig festlegen
- Projektidee festhalten → Als Issue im Vorlesungsrepo

Q & A



Thank you.

Contact information:

Aaron Schweig
aaron.schweig@sap.com

