

## Beschreibung für einen neuen SITA-Kurs

Dozent/in: Stefan Böhringer / Nils Hellwig  
Semester: SoSe 2023

### Kurstitel: Einführung in die moderne (Full-Stack) Webentwicklung mit JavaScript, Node.js und React.js

**Bemerkung:** [= Angabe des Kursformats, z.B.: „Der Kurs findet in Präsenz statt“]

Das Kursformat ist hybrid. Die Kursleiter sind in Präsenz vor Ort; die Teilnehmer:innen können sich online per Zoom zuschalten. Präferiert wird aber Präsenz. Das Seminar behandelt sowohl die serverseitige und clientseitige Programmierung mit insgesamt 4 SWS. Die Aufteilung erfolgt im Seminar-Übungs-Duktus.

**Inhalte:** [stichpunktartige Auflistung der Inhalte]

- Auffrischung *JavaScript* (inkl. *HTML* und *CSS*) [ca. 6 Std.]
- Einrichtung Entwicklungsumgebung (git, IDE) [ca. 2 Std.]
- Einführung in *Node.js* (Backend-Entwicklung mit JavaScript) [ca. 8 Std.]
- Arbeiten mit dem *Express.js* Framework für Node [ca. 12 Std.]
- Anbindung an SQL und NoSQL Datenbanken (inkl. Einführung in *MongoDB*) [ca. 8 Std.]
- Einfaches Templating mit *EJS* [ca. 4 Std.]
- Einführung in die Frontend-Entwicklung mit *React.js* [ca. 16 Std.]

**Kommentar:** [Beschreibung des Kurses und des Kursformates in Textform]

Die Teilnehmenden sollen einen Einblick in die heute gängige Praxis des (Full-Stack) Webdevelopments mit der Programmiersprache JavaScript bekommen. JavaScript lässt sich heute zum Lösen von Problemen sowohl auf der Frontend-, als auch auf der Backend-Seite einsetzen. Ein wichtiger Meilenstein dabei war die Entwicklung von Node.js, das als Fundament für den Kurs dient. Aufbauend darauf, sollen die Teilnehmer:innen durch die Hilfe eines, den Kurs begleitenden Projektes (eine ToDo-App inkl.

Authentifizierung) typische **JavaScript**-Werkzeuge kennenlernen:

- Express.js als Node-Framework zur Erstellung und Verwaltung der Applikation auf dem Server,
- MongoDB als Datenbank, die mit Hilfe von JSON arbeitet,
- EJS für einfaches Templating,
- React.js für die Umsetzung komplexerer Graphical User Interfaces.

Bei alledem arbeiten die Teilnehmenden ihrer eigenen Entwicklungsumgebung und verwalten ihren Code mit der Hilfe von git.

**Lernziele:** [Beschreibung der Ziele in Text- oder Auflistungsform]

- Auffrischung der eigenen JavaScript Kenntnisse und Vertiefung des Verständnisses asynchroner Programmierung mit Promises und „async, await“.
- Kennenlernen der wichtigsten Neuerungen seit ECMAScript 6.
- Auffrischung der eigenen HTML und CSS Kenntnisse, mit Schwerpunkt *responsives Design*.
- Kennenlernen der basalen Mechanismen des Internets (z.B. den Request-Response-Cycle, http-Methoden, etc.)
- Installation, Hintergrund und Handhabung von Node.js inkl. des Paketmanagers npm
- Verständnis über den Aufbau einer modernen responsiven Web-App

- Umsetzung des Aufbaus mit einem Node-Framework (Express.js) und Verständnis für die Arbeit mit Middlewares.
- Basales Verständnis von Datenbanken insbesondere der NoSQL-Datenbank MongoDB (Verständnis für die Dokumentenstruktur mit JSON).
- Verständnis und Umsetzung sicherer User-Authentifizierung mit Cookies und CSRF-Schutz.
- Kennenlernen einfacher HTML-Templates mit der Hilfe von EJS.
- Kennenlernen und Umsetzung komplexerer GUI-Web-Anwendungen mit React.js (Components, JSX, States, Props und Eventhandling).
- Im Ganzen: Kennenlernen des Design-Prozesses und des Aufbaus einer modernen Web-Applikation.

**Leistungsnachweis:** [Nennung der zu erbringenden Prüfungsleistung, z.B. Projektarbeit]

Erstellung eines eigenen kleinen Projektes mit Hilfe der im Seminar kennengelernten Werkzeuge und Techniken. Abgabe des Projektes bis zu Beginn des folgenden Semesters.

**Zielgruppe:** [Beschreibung, für wen der Kurs geeignet ist]

Studierende aller Fächer, insbesondere Teilnehmer:innen an der studienbegleitenden IT-Ausbildung mit JavaScript Vorkenntnissen.

**Zwingende Voraussetzung:** [Nennung der technischen Voraussetzungen + eventuelle Vorkenntnisse]

Grundkenntnisse JavaScript, HTML, CSS

Von Vorteil: Verwendung eines eigenen Laptop/PCs

**Modulzuordnung:** [wird unsererseits ergänzt]