



# Internet-Technologien in der Praxis

## Webframeworks

Seminar im WiSe 2025 / 2026

Prof. Dr. Jan-Peer Rudolph

# Motivation

- Webframeworks sind der Grundpfeiler moderner Anwendungsentwicklung, insbesondere in der Entwicklung von Unternehmensanwendungen
- Heutzutage wird geradezu erwartet, dass eine Applikation über den Browser oder ein mobiles Endgerät benutzt werden kann
- Ziel des Seminars ist die Entwicklung einer beispielhaften Web-Applikation in Einzelarbeit oder im Team von 2 bis 3 Personen

# Technologien

- Die Entwicklung soll im **Frontend** erfolgen mittels des Frontend-Frameworks [React](#) und im **Backend** entweder mit dem Python-Framework [Flask](#) oder dem Java-Framework [Spring Boot](#)
- Technologien:
  - Frontend:
    - [React](#)
    - [Bootstrap](#) (CSS)
    - [HTML](#)
  - Backend:
    - [Flask](#) (Python) oder alternativ [Spring Boot](#) (Java)
    - Datenbank (z. B. SQLite oder MariaDB)
- Ihre Applikation soll in Form von [Docker](#)-Containern installierbar sein

# Use Cases / Themengebiete der zu entwickelnden Anwendung (1/2)

Fachlich kann aus folgenden Themen ausgewählt werden:

- Organisations-Management:
  - Web-Applikation zum (rollenbasierten) **Team-Management** in einem Unternehmen
  - Web-Applikation zum **Aufgaben-Management** (z. B. in Form eines Kanban-Boards)
  - Web-Applikation zur Erstellung von **Organigrammen** für Unternehmen
  - Web-Applikation zur Buchung von **Büro-Arbeitsplätzen** (Verwaltung Office-Plätze, Home-Office, Urlaub)

## Use Cases / Themengebiete der zu entwickelnden Anwendung (2/2)

- Personalführung:
  - Web-Applikation zur Durchführung von **Mitarbeitergesprächen** in einem Unternehmen (Führungskraft bewertet Mitarbeiter / gibt Feedback)
  - Web-Applikation zur Durchführung von **Mitarbeiterbefragungen** in einem Unternehmen (Mitarbeiter bewertet Unternehmen)
- IT-Management:
  - Web-Applikation zum Management von **IT-Assets** (Hardware, Software, Lizenzen, Verträge, Schnittstellen, sonstige Informationen)
  - Web-Applikation zur Verwaltung von **IT-Tickets** (Erstellung, Bearbeitung, Verwaltung von IT-Tickets)

# Ablauf

- Einzelbearbeitung oder im Team von 2 bis 3 Personen
- Es soll eine beispielhafte Web-Applikation mit den genannten Technologien entwickelt werden
- Die Applikation soll einen der aufgelisteten Use Cases / Themen abbilden

# Leistungserbringung und Bewertung

- Die Bewertung erfolgt individuell und setzt sich aus drei Faktoren zusammen:
  - **40 % Vortrag**
  - **40 % Schriftl. Ausarbeitung**
  - **20 % Vollständige Abgabe** (insb. vollständige Code-Abgabe mit installierbarer und lauffähiger Applikation)

# Leistungserbringung und Bewertung

- **Vortrag:**

- Jeder Teilnehmer hat einen Vortragsanteil von 8 min (Einzelarbeit → 8 min, 2er Teams → insgesamt 16 min, 3er Teams → insgesamt 24 min)
- Inhalt des Vortrags: Demo der Applikation, Architektur, Problemstellungen in der Entwicklung und deren Lösung, Auszüge aus der Implementierung, Besonderheiten, ...

# Leistungserbringung und Bewertung

- **Schriftl. Ausarbeitung:**

- Jeder Teilnehmer verfasst eine schriftliche Ausarbeitung im Umfang von mindestens 5 und maximal 10 Seiten (Einzelarbeit → 5 - 10 Seiten, 2er Teams → 10 - 20 Seiten, 3er Teams → 15 - 30 Seiten)
- Teams geben eine gemeinsame schriftliche Ausarbeitung ab. Dabei muss klar ersichtlich sein, wer welchen Teil geschrieben hat. Pro Kapitel gibt es nur einen Autor, das muss eindeutig gekennzeichnet sein.
- Inhalt der Ausarbeitung: Einführung, Stand der Technik (bestehende Applikationen im Themenkontext), Beschreibung der Architektur und Implementierung, Fazit und Ausblick

# Leistungserbringung und Bewertung

- **Abgabe** soll eine **ZIP-Datei** sein, die enthält:
  - PDF der **schriftl. Ausarbeitung**
  - PDF der **Präsentationsfolien** (wenn Folien im Vortrag gezeigt wurden)
  - **Code-Abgabe** der entwickelten Applikation
    - Ordner `Backend` (enthält den Source-Code des Backends und ein `Dockerfile` zum Bauen)
    - Ordner `Frontend` (enthält den Source-Code des Frontends und ein `Dockerfile` zum Bauen)
    - `docker-compose.yml` (zur gemeinsamen Installation / Deployment aller Services)
  - **Installationsanleitung** der Applikation
    - Ihre Applikation soll in Form von Docker-Containern installierbar und lauffähig sein → am Ende muss sich Ihre Applikation durch Ausführung des Kommandozeilenbefehls `docker-compose up --build` starten lassen



Vielen Dank für Ihre Aufmerksamkeit!