

## Praktikum 2 (Semantisches HTML)

Im Rahmen des Praktikums entwickeln wir eine Web-Anwendung, welche wir Schritt für Schritt mit weiteren Anforderungen, Funktionen und Technologien erweitern.

**Hinweis zu Bonuspunkten:** Zur Vergabe der Bonuspunkte werden wir den von Ihnen produzierten Quellcode begutachten, sobald Sie bestimmte *Meilensteine* erreicht haben. Sobald ein Praktikum einen solchen Meilenstein markiert, finden Sie einen entsprechenden Hinweis auf dem jeweiligen Praktikumsblatt. Dieser Hinweis beinhaltet auch, wieviele Bonuspunkte für den jeweiligen Meilenstein erzielbar sind. Da die Praktika aufeinander aufbauen, enthält ein Meilenstein jeweils alle vorhergehenden Praktikumsaufgaben - zur Erlangung aller Bonuspunkte sind also alle Aufgaben zu bearbeiten. Um die Bonuspunkte zu erhalten, sprechen Sie uns einfach im Praktikum an, sobald Sie einen Meilenstein erreicht haben. Wir schauen uns dann gemeinsam Ihren Stand in einer kleinen Abnahme an. Dabei muss Ihre Gruppe, in welcher Sie die Aufgabe gelöst haben, vollständig anwesend sein. Insgesamt gibt es drei Meilensteine, und es sind maximal 10 Bonuspunkte über das Praktikum erreichbar. *Wichtig:* Abnahmen erfolgen nur bis maximal 20.01.23 (Ende der Vorlesungszeit)!

### Hinweis zum entstehenden Code:

- Zur Verwaltung und zum kollaborativen Bearbeiten Ihres Quellcodes empfehlen wir Ihnen die Nutzung von Git (z.B. in Verbindung mit dem [GitLab-Server des FB4](#), Login per FH-Account). Eine kurze Einführung in Git finden Sie [hier](#) sowie im [Bonusmaterial](#) der Lernmaterialien.
- Falls Sie Git nicht verwenden, so legen Sie Ihren Quellcode pro Praktikumsaufgabe in separaten Verzeichnissen (z.B. „praktikum2“, „praktikum3“) ab, damit die einzelnen Entwicklungsschritte bei den Abnahmen ersichtlich sind.

## Anwendungsszenario: Podcast-App

---

In diesem Semester entwickeln wir gemeinsam eine *Podcast-App*.

Ein [Podcast](#) ist eine Art Serie in Audio- oder (seltener) Videoform, welche aus einzelnen *Episoden* besteht. BenutzerInnen können für sie interessante Podcasts *abonnieren*, so dass sie stets informiert werden, sobald eine neue Episode erscheint. Zu diesem Zweck

gibt es spezielle Apps (auch *Podcatcher* genannt), die abonnierte Podcasts verwalten, regelmäßig auf neue Episoden überprüfen und diese bei Erscheinen schon einmal herunterladen, damit sie zum Anhören zur Verfügung stehen. Beispiele solcher Podcatcher-Apps sind AntennaPod, Pocket Casts oder Apple Podcasts.

Technisch wird das Abonnieren und Aktualisieren von Podcast-Episoden typischerweise mittels *RSS-Feeds* (RSS = Really Simple Syndication) realisiert. RSS bietet ein XML-basiertes Format zur Beschreibung von Änderungen/Neuigkeiten auf Web-Seiten (Details siehe [RSS 2.0 Specification](#)).

Es gibt mittlerweile ein großes Angebot an Podcasts zu verschiedensten Themen, hier ein paar Beispiele:

- Web- und Software-Entwicklung: [Working Draft](#), [programmier.bar](#)
- Politik und Nachrichten: [Lage der Nation](#), [Logbuch: Netzpolitik](#)
- Forschung und Wissenschaft: [Forschergeist](#), [Methodisch inkorrekt](#)

Im Rahmen des Praktikums werden wir schrittweise eine Web-Anwendung entwickeln, welche u.A. folgende Funktionen enthalten wird:

- Abonnieren beliebiger Podcasts
- Anzeige aller abonnierten Podcasts
- Abrufen und Anzeige aller Episoden zu einem Podcast
- Anzeigen der Informationen zu einer Episode (z.B. Dauer) und Abspielen der Episode

Am Ende des Semesters werden Sie eine funktionierende Basis-Anwendung entwickelt haben, welche durchaus geeignet ist, über die Veranstaltung hinaus zu einem vollwertigen Podcatcher weiterentwickelt zu werden. 😊

## Aufgabe 1: Statischer Prototyp der Podcast-App

Wir beginnen mit einem ersten statischen Prototyp der Podcast-App, der die Inhalte und Strukturierung der Web-Anwendung demonstrieren soll. Wir nutzen zu diesem Zweck im ersten Schritt ausschließlich HTML - auf optisches Styling verzichten wir zunächst (fast) völlig. **Verwenden Sie HTML-Elemente aus der Vorlesung zur Erstellung des Prototypen - insbesondere die semantischen Sektionselemente von HTML5!**

Erstellen Sie *drei statische* HTML-Seiten mit folgenden Inhalten:

1. `index.html` : Liste der abonnierten Podcasts

- Diese Seite stellt alle abonnierten Podcasts in Form einer *ungeordneten Aufzählung* dar.

- Zu jedem Podcast in der Liste werden folgende Daten angezeigt:
  - Titel des Podcasts (z.B. "Working Draft")
  - Anzahl der verfügbaren Episoden
- Der Titel jedes Podcasts in der Liste ist dabei ein *Hyperlink* (→ [Lernmodul C-06](#)), welcher auf die *Detailseite zu einem Podcast* (siehe Punkt 2) verlinkt.

## 2. `podcast.html` : Detailseite zu einem Podcast

- Diese Seite zeigt alle Informationen zu einem Podcast an.
- Die Seite besteht aus *zwei Sektionen* (verwenden Sie entsprechende *semantische HTML-Elemente*):

### 1. Informationen zum Podcast:

- Titel
- Beschreibung (kurzer Text als Inhaltsbeschreibung)
- Autor\*in (ein oder mehrere Personen, die bei dem Podcast mitwirken),
- Besitzer\*in (Person, die für den Podcast hauptverantwortlich ist)
- Kategorien (kommaseparierte Liste von Kategorien, welchen der Podcast zugeordnet ist)
- letztes Update (Datum des letzten Updates, d.h. Erscheinen der neuesten Episode)

### 2. Episoden zum Podcast:

- Eine *geordnete Aufzählung* aller Episoden zum Podcast.
- Stellen Sie dabei für jede Episode folgende Daten dar:
  - Titel der Episode
  - Datum (des Erscheinens der Episode)
- Der Titel jeder Episode ist ein Hyperlink, welcher auf die *Detailseite zu einer Episode* (siehe Punkt 3) verweist.

## 3. `episode.html` : Detailseite zu einer Episode:

- Diese Seite zeigt alle Informationen zu einer Episode an.
- Die Seite besteht aus *einer Sektion "Informationen zur Episode"* (verwenden Sie entsprechende *semantische HTML-Elemente*) mit folgenden Inhalten:
  - Titel
  - Beschreibung (kurzer Text als Inhaltsbeschreibung)
  - Datum (Erscheinungsdatum der Episode)

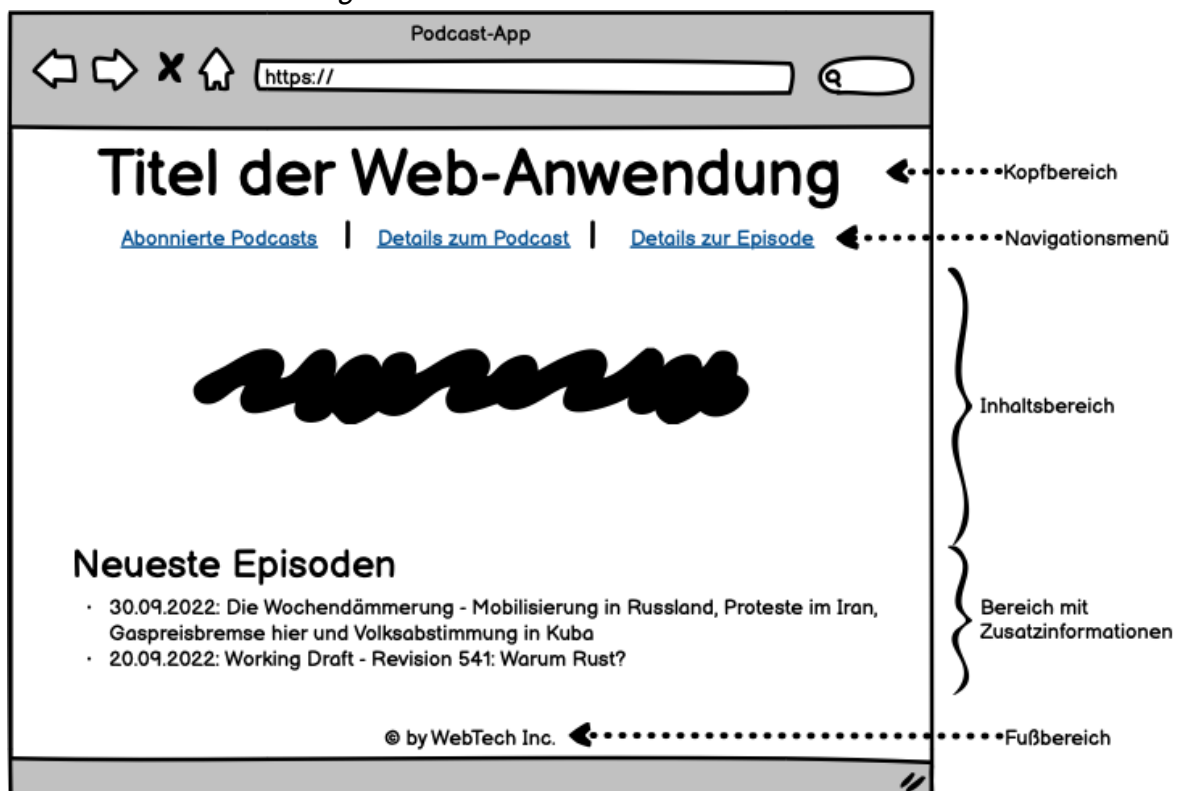
**Hinweis:** Da es sich erst einmal um einen Prototyp handelt, reicht für die *Detailseite zu einem Podcast* (Punkt 2) und die *Detailseite zu einer Episode* (Punkt 3) jeweils eine Beispielseite mit Daten aus. Sie müssen die Seiten nicht mehrfach mit unterschiedlichen Beispieldaten anlegen. Die Hyperlinks in der *Liste der abonnierten Podcasts* (Punkt 1) sowie in der Episodentabelle (Punkt 2.2) können einfach immer auf die gleichen Beispielseiten verlinken.

## Aufgabe 2: Gemeinsame Bereiche

Erweitern Sie die HTML-Seiten aus Aufgabe 1, so dass *alle drei Seiten* folgende gemeinsamen Bereiche enthalten:

- einen Kopfbereich mit dem Titel der Web-Anwendung,
- ein Navigationsmenü mit Einträgen für die drei Seiten (als entsprechende Hyperlinks),
- einen Bereich mit *Zusatzinformationen*, welcher eine *ungeordnete Liste* mit den neusten Episoden aller abonnierten Podcasts enthält (jeweils mit Erscheinungsdatum, Titel des Podcasts und Titel der Episode), und
- einen Fußbereich mit einem Copyright-Hinweis.

Schematische Darstellung des Seitenaufbaus:



Auch hier gilt: Verwenden Sie HTML-Elemente aus der Vorlesung zur Erweiterung der drei Seiten - insbesondere die semantischen Sektionselemente von HTML5!

# Allgemeine Hinweise:

---

- Verwenden Sie HTML-Elemente aus der Vorlesung zur Erstellung der drei Seiten - insbesondere die *semantischen Sektionselemente*!
- Zur Lösung der Aufgabe müssen Sie *exakt drei HTML-Dateien* (nicht mehr!) erstellen. Verwenden Sie statische Beispieldaten in den Seiten.
- Validieren Sie Ihren HTML-Code mit dem [W3C Markup Validator](#). Der entstehende HTML-Code soll beim Check keine Fehler und Warnungen produzieren.