

# Praktikum 7 (Kachelansicht, Responsive Layout, CTF)

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

## Meilenstein 2: "CSS"

Sobald Sie Praktikum 4, 5, 6 und 7 vollständig bearbeitet haben, haben Sie Meilenstein 2 erreicht! Sprechen Sie uns (Sven Jörges, Andreas Harrer oder Julian Feder) im Praktikum an. Wir schauen uns dann gemeinsam Ihren Stand in einer kleinen Abnahme an. Dabei muss Ihre Gruppe, in welcher Sie die Praktika bearbeitet haben, vollständig anwesend sein. Für diesen Meilenstein sind insgesamt 3 Bonuspunkte erreichbar. Es folgt noch ein weiterer Meilenstein.

## 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](#).
- 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.

## Aufgabe 1: Kachelansicht

---

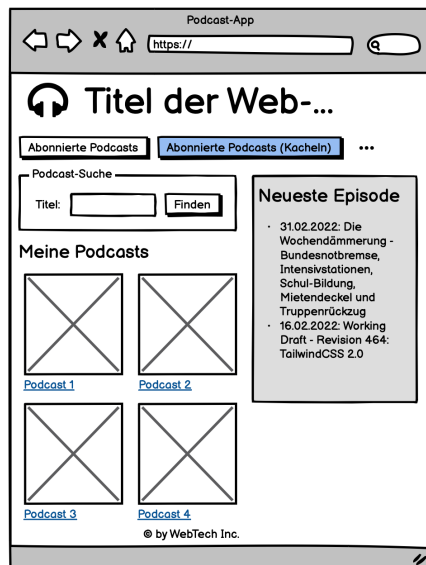
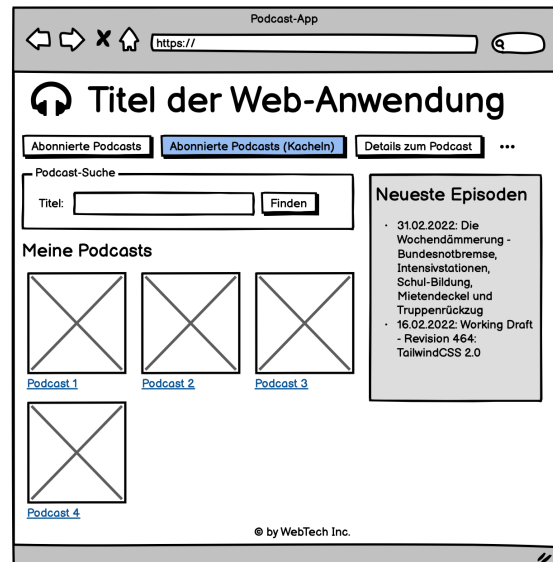
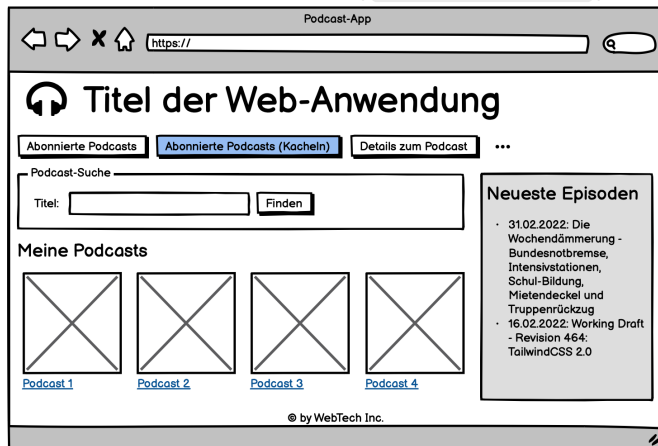
Auf der Seite `index.html` zeigen wir die Liste der abonnierten Podcasts aktuell in Form einer ungeordneten Aufzählung an.

Wir erstellen nun eine alternative Version dieser Seite mit einer etwas moderneren Ansicht in Form von *Kacheln*. Zunächst erstellen wir diese alternative Ansicht in einer zusätzlichen Seite `index2.html`. In einem späteren Praktikum werden wir die beiden Ansichten wieder zusammenführen und es Benutzer\*innen ermöglichen, per Knopfdruck zwischen den Ansichten zu wechseln.

Gehen Sie wie folgt vor:

1. Kopieren Sie die Inhalte von `index.html` in eine weitere Datei `index2.html`.
2. Lagern Sie zur Wahrung der Übersicht alle benötigten CSS-Regeln in ein separates externes Stylesheet `index.css` aus.
3. Verlinken Sie `index2.html` *auf allen Seiten* in der Navigation (zunächst einmal als zusätzlicher Link).
4. Ersetzen Sie die bisherige Darstellung der abonnierten Podcasts durch eine Kachelansicht. Realisieren Sie den entsprechenden Teil der Seite mit Hilfe von **Flexbox ohne Zuhilfenahme von Media Queries** - das umgebende Flexbox-Layout der Seite soll beibehalten werden! Das Layout der Kachelansicht soll dem in der unten stehenden Abbildung dargestellten Schema entsprechen:
  1. Jede Kachel stellt einen abonnierten Podcast dar. Es sollen insgesamt *mindestens vier Kacheln* dargestellt werden (d.h. vier beispielhafte Podcasts).
  2. Eine Kachel besteht aus zwei Teilen:
    1. Dem Bild zum Podcast (z.B. Logo) - analog zum Bild auf der *Detailseite zu einem Podcast* (`podcast.html`). In der unten stehenden Abbildung ist das Bild jeweils durch eine Box mit einem X symbolisiert.
    2. Dem Titel des Podcasts (in der unten stehenden Abbildung symbolisch "Podcast 1", "Podcast 2" etc. - ersetzen Sie die Titel durch entsprechende Beispiele). Die Anzahl der Episoden zeigen wir in dieser Ansicht nicht mehr an.
  3. Alle Kacheln sollen *gleich groß* sein. Zwischen den Kacheln soll horizontal und vertikal ein *gleichmäßiger Abstand* herrschen.
  4. Jede Kachel soll ein *klickbarer Hyperlink* sein, welcher auf die *Detailseite zu einem Podcast* (`podcast.html`) verweist. Dabei soll jeweils die gesamte Kachel (also Bild *und* Titel) klickbar sein.
  5. Bei Verkleinerung der Breite des Browserfensters sollen sich die Kacheln (wie auf der Abbildung unten dargestellt) rearrangieren. Ist für die rechteste Kachel kein Platz mehr, so rutscht sie in die Zeile darunter ganz nach links.

Aufbau der Kachelansicht ( `index2.html` ):

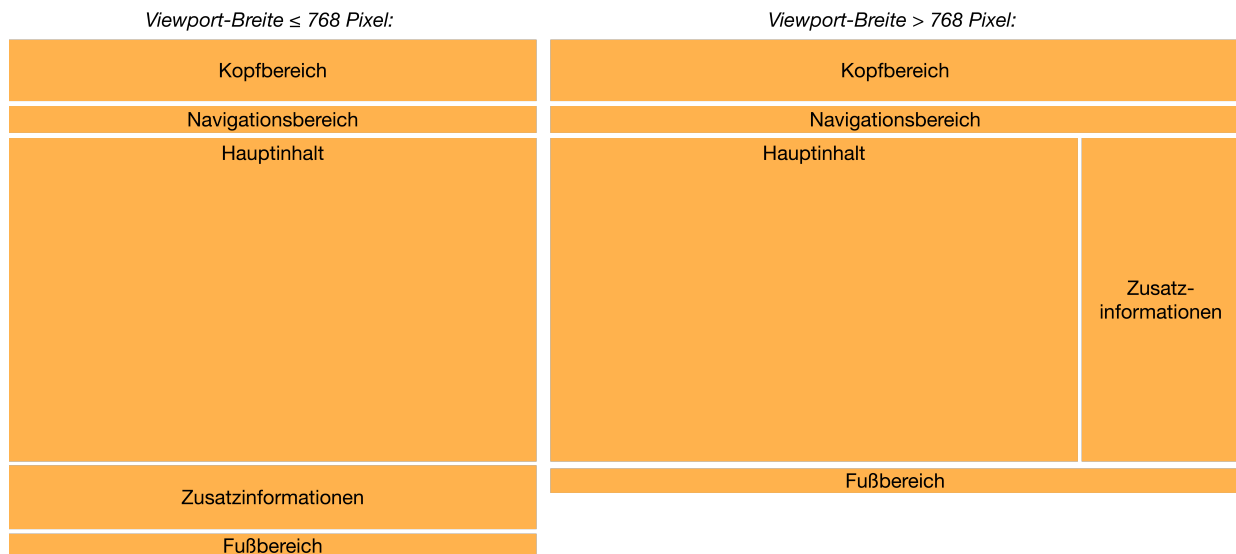


## Aufgabe 2: Responsive Flexbox-Layouts

Erweitern Sie die Flexbox-Layouts für die *Liste der abonnierten Podcasts* ( `index.html` , `index2.html` ) und die *Detailseite zu einer Episode* ( `episode.html` ) so, dass sie responsiv sind. Folgendes soll gelten:

1. Bei Änderung der Viewport-Breite soll sich das Layout fluide verhalten und den verfügbaren Platz ausfüllen.
2. Bei einer Viewport-Breite  $\leq 768$  Pixel soll der Bereich mit den Zusatzinformationen unter dem Hauptinhalt angezeigt werden (siehe folgende Abbildung).

### Aufbau der responsiven Flexbox-Layouts:



## Aufgabe 3: Responsive Grid-Layouts

Erweitern Sie die Grid-Layouts für die *Detailseite zu einem Podcast* ( `podcast.html` ) und die *Seite zum Anlegen eines neuen Podcasts* ( `newPodcast.html` ) so, dass sie responsiv sind. Folgendes soll gelten:

#### 1. Beide Seiten:

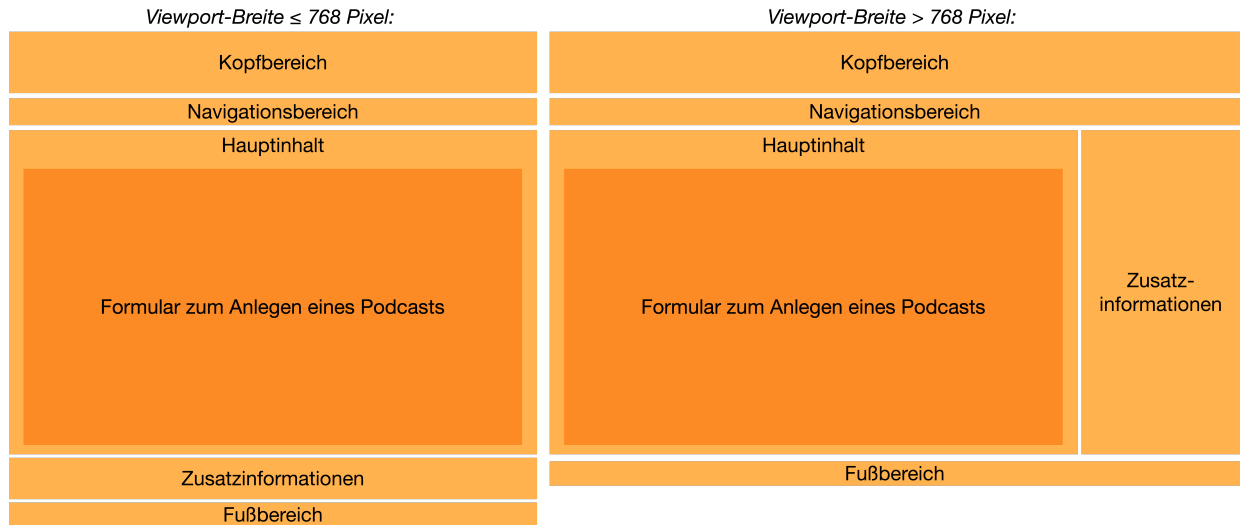
1. Bei Änderung der Viewport-Breite soll sich das Layout fluide verhalten und den verfügbaren Platz ausfüllen.
2. Bei einer Viewport-Breite  $\leq 768$  Pixel soll der Bereich mit den Zusatzinformationen unter dem Hauptinhalt angezeigt werden (siehe folgende Abbildungen).

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

1. Insbesondere soll das Bild zum Podcast flexibel sein (d.h. fluide mitwachsen bzw. mitschrumpfen).
2. Bei einer Viewport-Breite  $\leq 480$  Pixel soll das Bild zum Podcast unter der Liste der Episoden angezeigt werden (siehe folgende Abbildungen).

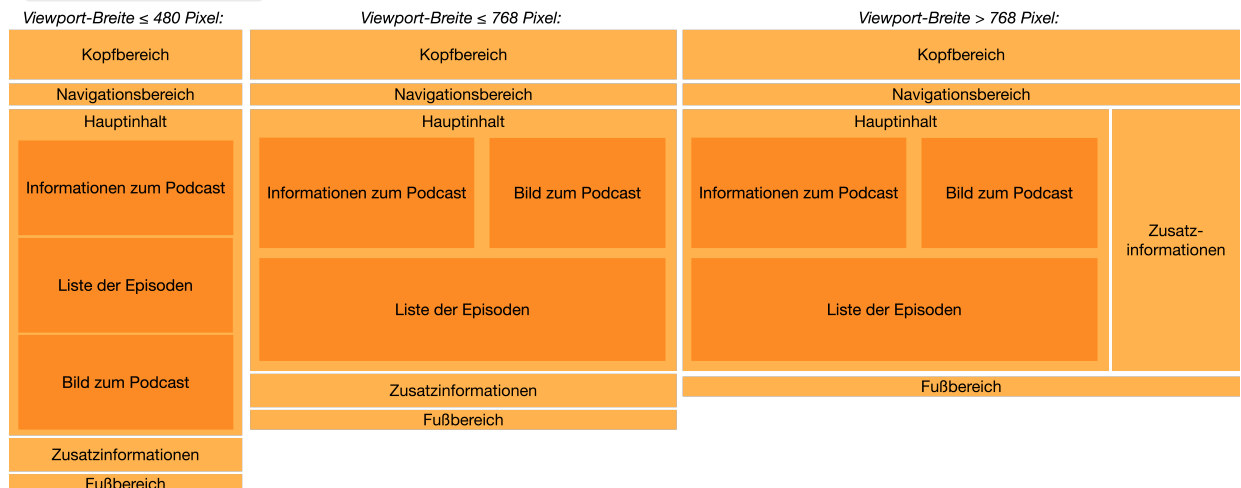
## Aufbau des responsiven Grid-Layouts der Seite zum Anlegen eines neuen Podcasts

( `newPodcast.html` ):



## Aufbau des responsiven Grid-Layouts der Detailseite zu einem Podcast

( `podcast.html` ):



## Allgemeine Hinweise:

- Validieren Sie Ihren HTML-Code mit dem [W3C Markup Validator](#). Der entstehende HTML-Code soll beim Check keine Fehler und Warnungen produzieren.
- Ausnahme: Warnungen bzgl. `date` - und `time` -Eingabefeldern können Sie ignorieren.
- Validieren Sie Ihren CSS-Code mit dem [W3C CSS Validation Service](#). Der entstehende CSS-Code soll beim Check keine Fehler und Warnungen produzieren.

## Zusatzaufgabe 2: Capture the Flag (CTF)

Die folgende Aufgabe ist freiwillig – es gibt für diese Aufgabe keine Bonuspunkte (nur Ruhm, Ehre, Spaß, etc. pp.)!

In dieser freiwilligen Zusatzaufgabe gibt es eine neue Challenge vom Typ *Capture the Flag* (CTF, siehe Praktikum 3 für die grundlegende Beschreibung des Prinzips). Auch dieses Mal ist das Flag wieder eine Zeichenkette, die mit `ctf_` beginnt (z.B.

`ctf_This_i5_the_Flag_forma7` ).

Finden Sie das Flag, das hier versteckt ist:

### Challenge 2: "Hexy Colors" (Schwierigkeitsgrad: leicht)

<https://labs.inf.fh-dortmund.de/ctfd-challenge-1/hexy-colors-everywhere.html>

Wenn Sie das Flag gefunden haben, schicken Sie es per E-Mail an [sven.joerges@fh-dortmund.de](mailto:sven.joerges@fh-dortmund.de) (gerne inklusive Lösungsweg, den Sie gewählt haben). Und nichts verraten!

