

# Übung: Entwicklung einer Todo-App

## Layout und Design mit Bootstrap

Im dritten Teil der Übungsaufgabe entwickeln wir auf Basis des CSS-Frameworks Bootstrap ein ansprechendes Layout und Design für unsere Todo-App.

### Vorbereitung

Für diese Aufgabe müsst ihr die vorherigen beiden Aufgaben *“Die GUI mit HTML, CSS und JavaScript”* sowie *“Speicherung der Daten im JSON-Format”* abgeschlossen haben.

### Aufgabe 1: Ein Layout für unsere Todo-App

Jeder Webanwendung liegt eine Struktur zugrunde, die bestimmt, an welcher Stelle Inhalte erscheinen sollen. Im ersten Schritt überlegen wir uns eine solche Struktur für unsere Todo-App.

- Informiert euch über das Layoutsystem basierend auf der Einteilung einer Webseite in ein Grid bestehend aus Zeilen und jeweils 12-Spalten, [wie es auch von Bootstrap verwendet wird](#).
- Die Struktur unserer Todo-App soll zukünftig auf einem solchen Grid basieren. Dabei soll es auf vertikaler Ebene (Zeilen) drei Bereiche geben:
  1. Einen Header, der in zwei Spalten geteilt ist. In der linken Spalte, die fast die gesamte Breite in Anspruch nehmen soll, wird der Titel unserer App dargestellt. Ein kleiner Bereich auf der rechten Seite (**A**) zeigt zudem das Profilfoto des angemeldeten Benutzers.
  2. Einen Hauptbereich, in dem auf der linken Spalte (**B**) die verschiedenen Todo-Listen eines Benutzers aufgelistet werden sollen. In der Mitte des Bereichs (**C**) wird die aktuell ausgewählte Todo-Liste mit ihren Einträgen angezeigt. Dieser Bereich soll den meisten Platz einnehmen. Im dritten Bereich auf der rechten Seite (**D**) werden Details zum aktuell angeklickten Todo dargestellt. Hier kann ein Todo später auch bearbeitet werden.

3. Einen Footer (**E**), in dem neben Informationen zu unserem Unternehmen ein Menü mit Einträgen zum Impressum etc. erscheinen soll.
- Überlegt euch für jede der drei Bereiche und deren Spalten passende Größenverhältnisse. Erstellt eine Skizze eures Layouts.
  - Setzt das Layout mithilfe der entsprechenden Bootstrap-Klassen wie `.container`, `.row` und `.col` und seinen Spezialfällen um.

## Aufgabe 2: Bootstrap-Elemente in unserer Todo-App

Bootstrap beinhaltet nicht nur wichtige Klassen für die Erstellung professioneller Layouts. Es definiert auch zahlreiche Inhaltskomponenten für Webseiten, darunter Formularelemente, Navigationsleisten, ausklappbare Inhalte und vieles mehr.

- Ersetzt die einfache HTML-Liste durch eine [mit Bootstrap gestylte Liste](#), die eher dem Stil einer Todo-App entspricht.
- Verwendet für [sämtliche Benutzereingaben](#) sowie [Buttons](#) die von Bootstrap bereitgestellten Elemente.
- Setzt in den Bereich **A** ein [mit Bootstrap gestyltes Profilfoto](#) eines beliebigen Benutzers.
- Verwendet für den Bereich **D** und die Darstellung der Details zu einem Todo eine sogenannte [Card von Bootstrap](#).

## Aufgabe 3: Responsiveness

Mit *Responsiveness* meinen wir, dass eine Seite sich auf unterschiedlichen Bildschirmgrößen so verhält, dass alle wichtigen Inhalte erfasst werden können. Zum Beispiel wird auf größeren Bildschirmen oft Inhalt nebeneinander angezeigt, der auf Smartphones dann untereinander platziert wird. Führt die folgenden Anpassungen am Layout durch, sodass unsere Todo-App auf verschiedenen Bildschirmgrößen nutzbar wird.

1. Versteckt den Bereich mit den Details zum aktuellen Todo-Item (**D**), wenn die Bildschirmbreite kleiner als 992 Pixel (**large**) ist. Dafür soll der Abschnitt **C** den Rest des Platzes, an dem ursprünglich **D** war, einnehmen.

**HINWEIS:** Nutzt hierfür die [Display-Properties](#) von Bootstrap. Weist dem Abschnitt im Default die Klasse `.d-none` zu. Nutzt die richtige Klasse, um es bei der entsprechenden Bildschirmgröße wieder einzublenden.

2. Zeigt die Bereiche **B** und **C** untereinander an, wenn der Bildschirm kleiner als 768 Pixel (**medium**) groß ist. Der Bereich **D** sollte bei dieser Größe versteckt sein, wenn die erste Maßnahme richtig umgesetzt wurde.

**HINWEIS:** Nutzt hierfür die [Responsive Classes](#) von Bootstrap. Damit könnt ihr unterschiedliche Spaltenbreiten für unterschiedliche Bildschirmgrößen definieren.

3. Ändert die Reihenfolge für die 2. Anpassung, sodass **C** oben erscheint und **B** darunter.

**HINWEIS:** Verwendet hierfür Bootstraps [Reordering Properties](#).

4. Bei **kleinen Bildschirmen (< 576 Pixel)** soll das Thumbnail in der rechten oberen Ecke (**A**) versteckt werden.

**HINWEIS:** Nutzt hierfür wieder die [Display Properties](#) von Bootstrap. Weist dem Abschnitt im Default die Klasse `.d-none` zu. Nutzt die richtige Klasse, um es bei der entsprechenden Bildschirmgröße wieder einzublenden.

Damit sind wir mit dem dritten Teil der Aufgabe fertig. Unsere Todo-App kann sich mittlerweile sehen lassen. Im vierten Teil sollen die Daten über eine zentrale Datenbank in der Cloud über verschiedene Geräte synchronisiert werden.