**Artikel**

Besteht aus einem Globalen Head und Footer welche in extra dateien liegen zur schnellen wiederverwendung. Nach dem Head folgt ein Container welche den Artikel beinhaltet, der Artikel beginnt mit einem passendem Bild gefolgt mit einem Text zu dem jeweiligen Thema. Unterhalb des Artikels befindet sich noch eine weiterleitung zu weiteren Artikeln mit dem selben Thema.

**Article.css**

Besteht aus den styles für die Artikel.

Body-Style:

Setzt die allgemeine Body Farbe auf den Hex #f2f2f2 ein hellgrau.

Span1:

Die Klasse span1 stylt Unterüberschriften inerhalb eines Artikels, es wird die größe auf 1rem und der Zeilenabstand auf 1.7 gesetzt um es vom regulerem Text hervor zu heben.

Container-Style:

Der ".container"-Style enthält verschiedene Regeln, um den Container auf der Seite zu stylen, der den Inhalt der Website enthält. Es wird "flex" als "display" verwendet, um die Elemente im Container anzuordnen. Die Ausrichtung erfolgt in Spaltenrichtung und die Ausrichtung erfolgt an der linken Seite. Die Hintergrundfarbe wird auf Weiß gesetzt und es wird ein Abstand von 40 Pixeln um den Container hinzugefügt. Die maximale Breite des Containers beträgt 1200 Pixel, und der Container ist horizontal zentriert. Der Container wird auch eine z-index-Eigenschaft haben, um zu ermöglichen, dass andere Elemente auf der Seite unter oder über dem Container angezeigt werden können. Außerdem wird dem Container eine abgerundete Ecken-Eigenschaft mit einem Radius von 5 Pixeln zugewiesen.

Container-Image-Style:

Dieser Style wendet sich an das Bild innerhalb des Containers. Das Bild wird auf eine Breite von 100% eingestellt und der Höhe wird ein Wert von "auto" zugewiesen, damit es sich automatisch an die Breite des Containers anpasst. Es wird auch ein Abstand von 10 Pixeln am unteren Rand des Bildes hinzugefügt, um es vom Text abzugrenzen. Das Bild wird auch abgerundete Ecken mit einem Radius von 5 Pixeln haben.

Container-Text-Style:

Dieser Style wird auf alle Textelemente innerhalb des Containers angewendet, die nicht Überschriften oder Bilder sind. Der Text wird in einer Schriftgröße von 0,8rem angezeigt, und der Zeilenabstand beträgt 1,3.

Small-images-Style:

Dieser Style wird auf die Bilder am Ende des Artikels angewendet. Die Bilder werden in einer horizontalen Reihe mit einer zentralen Ausrichtung und einem Abstand von 10 Pixeln zwischen ihnen angezeigt. Sie haben eine Breite von 300 Pixeln und eine Höhe von "auto", um sich automatisch an die Breite des Containers anzupassen. Sie haben auch abgerundete Ecken mit einem Radius von 5 Pixeln und eine schwarze Umrandung. Wenn Sie auf eines der Bilder zeigen, wird es 10% größer skaliert. Die Bilder haben auch einen Cursor in Form einer Hand, um anzuzeigen, dass sie anklickbar sind.

Small-images-Links-Style:

Dieser Style wird auf die Links angewendet, die sich um jedes der kleinen Bilder befinden. Die Links werden in einer Spalte angeordnet und die Ausrichtung erfolgt an der linken Seite. Die Breite des Links beträgt 200 Pixel, mit einem abstand von 50px jeweils links und rechts.

Small-images-a-Img-Style:

Mit ‚object-fit: cover‘ wird das Bild so skaliert, dass es volständig in den Container passt, one Seitenverhältnisse zu verzerren. Die Höhe wird auf 100% gesetzt und ein Rechter Abstand von 10px zwischen den Bildcontainern wird gesetzt.

Small-images-a-text-Style:

Setzt den oberen Abstand zu den Bildern auf 20px und den untern und seitlichen auf 0.

Small-images-text-Style:

Schriftgröße wird auf 0.8rem festgelegt, innerhalb des Containers wird der text zentriert und der Abstand nach oben wird auf 10px gesetzt.

Medienabfrage:

Medienabfrage bei Bildschirmen mit maximaler Breite von 768px.

Die Flex-Ausrichtung wird in eine Spalte geändert, um auf kleinere Bildschirme zu passen, und die Ausrichtung wird linksbündig ausgerichtet. Der obere Abstand des Bildes wird auf 20px geändert und das Bild wird so skaliert, dass es vollständig in den Container passt. Der obere und untere Abstand wird auf 40px festgelegt. Der Abstand zwischen den Bildcontainern wird auf 20px festgelegt. Der Abstand für jedes dritte Bild (beginnend mit dem ersten Bild) wird auf 0px gesetzt. Der Abstand für jedes dritte Bild (beginnend mit dem letzten Bild) wird auf 0px gesetzt.

**Login**

Beinhaltet Globalen Head und Footer, ebenso hat es das Anmelde Formular im zentrum. Dieses benötigt den Benutzernamen und das Passwort mit welche dann als formular abgesendet werden zur weiterverarbeitung. Auch hat es hier einen Knopf welcher einen zum Registrierungsformular weiterleitet. Der Hintergrund ist ein Slider welchen zwischen drei Bildern rotiert.

**Login-Stylesheet.css**

@keyframes slide:

Definiert die Schritte in der sich die slides von links nach rechts bewegen, dies wird mit der eigenschaft translateX angewandt. Mit den ‚animation-timing-function‘ wird der übergang der Bewegung bestimmt, für einen Flüßigen übergang der Bilder.

Slider:

Hält den Inhalt innerhalb des sichtbaren Bereichs mit der eigenschaft hidden für den overflow. Der Inhalt wird auf die volle breite des Bildschirms gestreckt und die höhe wird auf 90% begrenzt.

Slider-slide:

Definiert die position der einzelnen slides damit diese überhalb des elternelements zentriert bleiben und sich über den ganzen Bildschirm erstrecken. Schließlich wird die "slide"-Animation mit der Eigenschaft "animation: slide 16.5s infinite" angewendet, um das Element in einer Dauerschleife zu animieren.

Slider-slide:nth-child:

Definiert die einzelnen Bilder der jeweiligen slides und teilt die Zeit auf die drei slides auf damit diese gleich lange sichtbar sind.

Login-container:

Positioniert den Contaoiner für den Login Horizontal zentriert, mit einer breite von 400px und einer höhe von 480px. Es hat einen box Schatten damit dieser einen leichten 3D effekt aufweist, mit leicht abgerundetem Rand. Die Farbe des Containers wird auf ein leicht transparentes Weiß gesetzt.

Labels:

Die einzelnen labels werden untereinander angeordnet mit deren jeweiligen inputs direkt unter sich.

Button:

Der submit und register button sind im Container ganz unten angeordnet und sind auch transparent mit einer hover eigenschaft welches sie nichtmehr transparent lässt.

Show-password:

Positioniert die checkbox neben das dazu gehörige Label.

Medienabfrage:

Medienabfrage bei Bildschirmen mit maximaler Breite von 768px.

Lässt den Slider automatisch kleiner machen bei kleineren Bildschirmen.

**Login-js.js:**

function togglePassword():

Ermöglicht es das Passwort als sichtbaren text anzuzeigen.

**Footer**

Beinhaltet den Kontak zu den drei ProjektTeilnehmern und ist auf allen Seiten global als Footer eingestellt. Befindet sich in einer externen Datei für schnellere und leichteres einfügen.

**partialStyle.css**

Footer-Style:

Hat die eine linear gradiente farbe, und streckt sich auf die ganze breite des bildschirm

Footer-Kontakt-Style:

Ordnet die Kontakte in einen Grid an um die Position der Namen und Email Adressen zu vereinfachen.

Konatkt:

Text wird jeweils zentriert innerhalb der grid area und hat eine weiße Farbe mit einer größe von 12px.

**Startseite**

Beinhaltet kleinen Wilkommenstext zu der Website, unterhalb des textes befinden sich drei weiterleitungen zu drei aktuell beliebten Artikeln.

**Startseite.css**

Body

Farbe wird auf ein hellgrau gesetzt mit dem hex #f2f2f2.

Span

Farbe wird auf rot gesetzt.

Button

Button anordnung wird gesetzt mit transparentem Hintergrund und einem pointer als cursor.

Buttoncontainer

Die größe des Containers für die Buttons wird auf eine Breite von 200px gesetzt und der z-index auf 2 damit es immer oberhalb liegt.

Insta/Twitter/Facebook Button

Positioniert die einzelnen Buttons. Sind untereinander angeordnet und haben die hover eingenschaft versteckt, ebenso haben sie eine hover animation welche sie 5px nach rechts verschiebt.

Container

Container wird zentriert und schmaler als der Body definiert damit links und rechts Balken mit farbe des Bodys entsehen. Der z-index wird auf 1 gesetzt damit dieser immer oberhalb des Bodys liegt. Die ecken werden mir einem border-radius von 5px abgerundet.

Artikel

Artikel wird innerhalb des Containers zentriert und erhält einen Abstand nach links und rechts. Die Schriftgröße der Überschrift h1 wird auf groß gesetzt mit einem Zeilenabstand von 2, damit diese sich von Rest abheben kann.

Images

Diese werden unterhalb des textes innerhalb des Containers positioniert. Sie sind in einer Reihe angeordnet zu einander angeordnet, zu dem text unter sich sind sie jeweils in einer Spalte angeordnet. Sie haben auch eine hover animation welche sie mit zu 1.1 skaliert.

Medienabfrage

Medienabfrage bei Bildschirmen mit maximaler Breite von 768px.

Sorgt für eine automatische anpassung bei kleineren Bildschirmen. Mit padding und margin wird für eine passende anpassung gesorgt.

**Routes**

Hier wird zuerst das express framework definiert und auch das darauf bassierend die Route-Definition, ebenso wird definiert das man das data Verzeichnis benötigt um auf inhalte wie nutzer und kommentare zugreifen zu können. Im nächsten Teil werden die ganzen Routen definiert, wird eine HTML-GET-Request am die jeweilige URL gesendet wird diese gerendert. In den oberen drei Routen wird noch die methode getComments aufgerufen welche aus dem data Verzeichnis die zu dem jeweiligem Artikel gehörenden Kommentare zurück gibt. Beim rendern der Seite werden hier dann auch noch die Kommentare mit übergeben, welche dann auf der Seite auch zu sehen werden. Die restlichen Routen werden einfach ohne die Kommentare gerendert. Zum schluss wird das Modul noch exportiert, damit auch die index.js zugriff auf die einzelnen Routes bekommt.