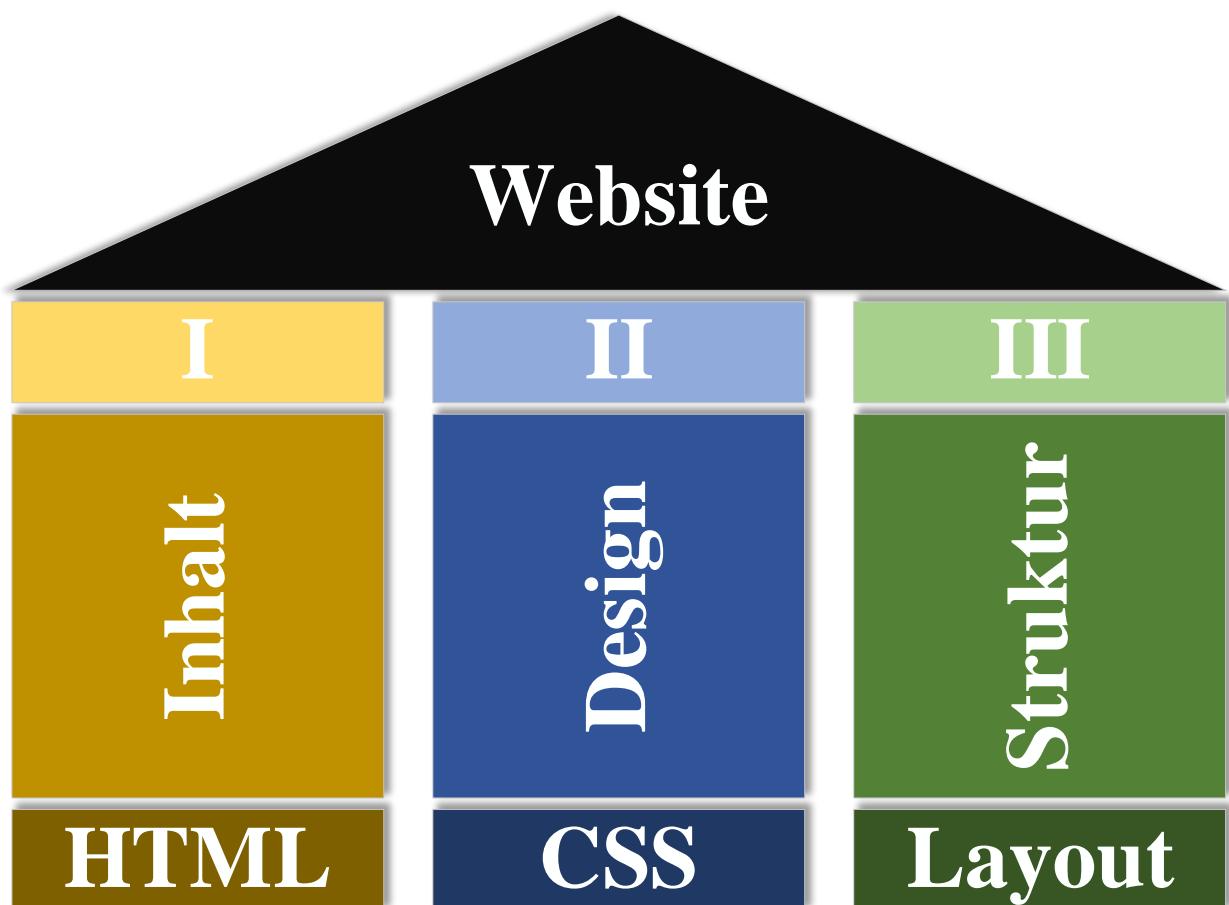


# **Webseite**



## **Modul WB**

## **Modulunterlagen**

Dieses Dokument darf ohne schriftliche Zustimmung des RAU weder kopiert noch anderweitig vervielfältigt werden.  
© RAU, 2021

## Inhaltsverzeichnis

<b>1</b>	<b>Handlungskompetenz.....</b>	<b>3</b>
<b>2</b>	<b>Über das Internet .....</b>	<b>4</b>
<b>3</b>	<b>Präsentationen .....</b>	<b>6</b>
3.1	Pflicht Themen .....	6
3.2	Optionale Themen.....	7
3.3	Vorgaben .....	7
<b>4</b>	<b>HTML Inhalt I .....</b>	<b>8</b>
4.1	Einleitung .....	8
4.2	Elemente, Tag und Attribut.....	8
4.3	Navigation .....	10
<b>5</b>	<b>CSS Design II.....</b>	<b>12</b>
5.1	Einleitung .....	12
5.2	Was ist CSS?.....	13
5.3	File Struktur anpassen .....	14
5.4	HTML Elemente gestalten .....	14
5.5	Navigationsleiste erstellen.....	15
<b>6</b>	<b>Layout Seitenstruktur III.....</b>	<b>16</b>
6.1	Einleitung .....	16
6.2	Layout in drei Varianten .....	16
6.3	Codereview .....	17
<b>7</b>	<b>Vertiefungsaufgaben .....</b>	<b>18</b>
7.1	Meta-Angaben.....	18
7.2	Zusatzaufgaben zur Vertiefung .....	19
<b>8</b>	<b>Webauftritt veröffentlichen .....</b>	<b>21</b>
<b>9</b>	<b>Projekt.....</b>	<b>22</b>
9.1	IPERKA.....	22
9.2	Dokumente der einzelnen Projektphasen.....	22
9.3	Prototypen.....	26
9.4	Umsetzung.....	27

**Identifikation und Änderungsgeschichte**

Dokumenttitel: Modulunterlagen  
Thema: Modul WB Webseite  
Autor: Michael Graf  
Firma: RAU, Regionales Ausbildungszentrum Au  
Dateiname: HandOut-WB\_Webseite\_v10.docx  
Ablageort: C:\Austausch\_synch\Informatik\Module\_ab\_2021\WS\_Webseite\Lernende\HandOut-WB\_Webseite\_v10.docx  
Druckdatum: 02.11.2021

Version	Datum	Bemerkungen
1.0	Januar 2021	Initialversion basierend auf M101 und M307 / MG

## 1 Handlungskompetenz

Quelle: ICT-Berufsbildung Schweiz

<b>Titel</b>	<b>Applikationen gemäss Entwurf implementieren und dabei die Sicherheitsanforderungen erfüllen</b>
Kompetenz	Auf Basis der definierten Anforderungen und ausgearbeiteten Entwürfe implementieren Informatikerinnen und Informatiker Applikationen. Es kann sich dabei sowohl um neue als auch um die Erweiterung von bestehenden Applikationen handeln:  In einem ersten Schritt richten sie eine geeignete Entwicklungs- und Laufzeitumgebung ein. Das erstellte Realisierungskonzept sowie Firmenvorgaben dienen ihnen als Grundlage.  Danach beginnen sie mit der Programmierung von Front-End gemäss den definierten Anforderungen. Sie setzen dazu vorgegebene Programmiersprachen sowie Entwicklungstools ein. Die Implementierung testen sie regelmässig auf Fehler und beheben diese. Sie achten laufend darauf, dass sie regulatorische und technische Richtlinien (z.B. von Frameworks, Designvorgaben) einhalten. Ausserdem berücksichtigen sie Sicherheitsanforderungen.  Bei der Implementation achten sie darauf, die allgemein bekannten Coderichtlinien einzuhalten, um die Nachvollziehbarkeit und Verständlichkeit zu sichern.
Bildungsplan	Handlungskompetenz g5 auf das Modul WS angepasst

## 2 Über das Internet

Anhand folgender Aufgaben machen Sie sich einleitend einige allgemeine Gedanken über das Internet und über das Erstellen von Webauftritten:

	Die Links der Webseiten für die Hilfestellung ändern immer wieder, da sie von den Betreibern aktualisiert werden. Fall Links in den Unterlagen nicht funktionieren, melden Sie sich bitte bei der Kursleitung.
---	--

Beantworten Sie nun folgende Fragen und fassen Sie in einem Word-File oder im OneNote zusammen. Ziel ist es, die Begriffe zu verstehen.

A1	Was versteht man unter dem Begriff Client-Server-Technologie  Für die einzelnen Internetdienste wie WWW, E-Mail, FTP usw. muss auf einem Host Rechner eine entsprechende Server-Software laufen. Ein Host-Rechner kann einen Internet-Dienst nur anbieten, wenn eine entsprechende Software auf dem Rechner aktiv und wenn der Rechner "Online" ist.	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Client-Server-Technologie</a>
A2	Was versteht man unter dem Begriff TCP/IP?  Das Transmission Control Protocol (englisch für Übertragungssteuerungsprotokoll) ist ein Netzwerkprotokoll, das definiert auf welche Art und Weise Daten zwischen Computern ausgetauscht werden sollen.	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">TCP/IP-Protokoll</a>
A3	Was ist HTTP, HTTPS und welche Funktion erfüllen sie?  Das HTTP (HyperText Transfer Protocol) ist ein Protokoll, welches die Datenübertragung zwischen Anwendungen regelt. Es ist ein Frage-Antwort-Schema.  HTTPS (HyperText Transfer Protocol Secure) ist ein Protokoll zur abhörsicheren Übertragung von Dokumenten im WWW.	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">HTTP</a> <a href="#">HTTPS und TLS</a>
A4	Was ist eine URL?  Ein Uniform Resource Locator (URL) gibt den Ort (location) einer Ressource im Netz an. URLs waren ursprünglich die einzige Art von URIs, weshalb der Begriff URL oft gleichbedeutend mit URI verwendet wird.	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">URL (Uniform Resource Locator)</a>

## Modulunterlagen

A5	Was versteht man unter dem Begriff DNS?
	Das DNS (Domain Name System) ist ein Verzeichnisdienst, der für die Zuordnung von Domainnamen zu IP-Adressen verantwortlich ist.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">DNS (Domain Name System)</a>
A6	Was ist ein Browser?
	Ein Browser (englisch für Stöberer) ist eine Software, mit der im WWW Dokumente abgerufen und dargestellt werden können.  Die bekanntesten Browser sind: Google Chrome, Mozilla Firefox, MS Edge, MS Internet-Explorer, Opera Browser und Safari.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Browser</a>
A7	Was ist HTML?
	HTML bestimmt den strukturellen Aufbau einer Internetseite.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">HTML (Hypertext Markup Language)</a>
A8	Was ist CSS?
	CSS (Cascading Style Sheets, zu deutsch „Mehrstufige Formatvorlagen“) ist eine Formatierungssprache für HTML-, SVG- und XML-Dokumente.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">CSS (Cascading Style Sheets)</a>
A9	Was sind HTML-Editoren?
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Code-Editor</a>

## 3 Präsentationen

Die Pflichtthemen sind alle für die schriftliche Leistungsbeurteilung 2 relevant und einzelne Themengebiete werden daraus abgefragt. Als Vorgabe dienen die unten aufgeführten Links und die Präsentationen der Mitlernenden sind nur zur zusätzlichen Unterstützung des Lerninhaltes.

### 3.1 Pflicht Themen

#### 1) Rechtliche Aspekte bei Verwendung von fremden Inhalten auf eine Webseite

[https://de.wikipedia.org/wiki/Urheberrecht\\_\(Schweiz\)](https://de.wikipedia.org/wiki/Urheberrecht_(Schweiz))  
[https://de.wikipedia.org/wiki/Creative\\_Commons](https://de.wikipedia.org/wiki/Creative_Commons)  
 Public Domain: <https://de.wikipedia.org/wiki/Gemeinfreiheit>  
**Absolute und relative Verlinkung und Pfade**  
<https://de.wikipedia.org/wiki/Pfadname>

#### 2) Grafikformate allgemein. Gängige Medienformate für die Wiedergabe von Fotografien, Grafiken, Dokumenten und Filmen

Vektorgrafik: <https://de.wikipedia.org/wiki/Vektorgrafik>  
 Pixelgrafik: <https://de.wikipedia.org/wiki/Rastergrafik>  
 PNG: [https://de.wikipedia.org/wiki/Portable\\_Network\\_Graphics](https://de.wikipedia.org/wiki/Portable_Network_Graphics)  
 JPEG: <https://de.wikipedia.org/wiki/JPEG>  
 GIF: [https://de.wikipedia.org/wiki/Graphics\\_Interchange\\_Format](https://de.wikipedia.org/wiki/Graphics_Interchange_Format)  
 TIFF: [https://de.wikipedia.org/wiki/Tagged\\_Image\\_File\\_Format](https://de.wikipedia.org/wiki/Tagged_Image_File_Format)  
 BMP: [https://de.wikipedia.org/wiki/Windows\\_Bitmap](https://de.wikipedia.org/wiki/Windows_Bitmap)  
 PDF: [https://de.wikipedia.org/wiki/Portable\\_Document\\_Format](https://de.wikipedia.org/wiki/Portable_Document_Format)  
 MP4: <https://de.wikipedia.org/wiki/MP4>  
 MPEG: [https://de.wikipedia.org/wiki/Moving\\_Picture\\_Experts\\_Group](https://de.wikipedia.org/wiki/Moving_Picture_Experts_Group)

#### 3) Gestaltungsregeln für Schriften, Farben, Grafiken und Benutzeroberflächen

Benutzerfreundlichkeit selfhtml: <https://wiki.selfhtml.org/wiki/Benutzerfreundlichkeit>  
 Benutzerfreundlichkeit Wikipedia: <https://de.wikipedia.org/wiki/Benutzerfreundlichkeit>  
 Typography: <https://material.io/guidelines/style/typography.html>  
 Webfarben: <https://de.wikipedia.org/wiki/Webfarbe>  
 Colors: <https://material.io/guidelines/style/color.html>  
 Imagery: <https://material.io/guidelines/style/imagery.html>  
 UI Design Do's and Don'ts: <https://developer.apple.com/design/tips/>

#### 4) Systemtest eines Webauftritts mit funktionaler und nichtfunktionaler Anforderung

Ein Ziel: Unterschied zwischen **funktionaler** und **nichtfunktionaler** Anforderung verstehen.  
 Systemtest: <https://de.wikipedia.org/wiki/Softwaretest>  
 funktionale Anforderung, nichtfunktionale Anforderung:  
[https://de.wikipedia.org/wiki/Anforderung\\_\(Informatik\)](https://de.wikipedia.org/wiki/Anforderung_(Informatik))  
 Markup Validation: <https://validator.w3.org/>  
 Analyze Site Performance <https://developers.google.com/speed/pagespeed/>

## 5) Veröffentlichung eines Webauftritts

Ein Ziel: Erklären, wie eine Website veröffentlicht werden kann  
(Zusammenhänge aufzeigen)

ISP: <https://de.wikipedia.org/wiki/Internetdienstanbieter>

Top-Level-Domain: <https://de.wikipedia.org/wiki/Top-Level-Domain>

Registrierung Domain-Name .ch: <http://www.switch.ch/>

Backup: <https://de.wikipedia.org/wiki/Datensicherung>Zahlungsverfahren: <https://de.wikipedia.org/wiki/Zahlungsverfahren>

## 6) Metatags

<http://wiki.selfhtml.org/wiki/HTML/Kopfdaten/meta>,  
[https://www.w3schools.com/tags/tag\\_meta.asp](https://www.w3schools.com/tags/tag_meta.asp)

## 7) Cascading Style Sheets CSS

<http://wiki.selfhtml.org/wiki/CSS>, <https://www.w3schools.com/css/default.asp>

## 8) Responsive Webdesign

[https://wiki.selfhtml.org/wiki/HTML/Tutorials/responsive\\_Webdesign](https://wiki.selfhtml.org/wiki/HTML/Tutorials/responsive_Webdesign)  
[https://www.w3schools.com/html/html\\_responsive.asp](https://www.w3schools.com/html/html_responsive.asp)

## 3.2 Optionale Themen

### 9) Image-Map

[https://wiki.selfhtml.org/wiki/HTML/Multimedia\\_und\\_Grafiken/verweissensitive\\_Grafiken](https://wiki.selfhtml.org/wiki/HTML/Multimedia_und_Grafiken/verweissensitive_Grafiken)  
[https://www.w3schools.com/tags/tag\\_area.asp](https://www.w3schools.com/tags/tag_area.asp)

### 10) Seitenstrukturierung mittels CSS-Bildschirmunterteilung (CSS-basierte Layouts)

[https://wiki.selfhtml.org/wiki/CSS/Anwendung\\_und\\_Praxis/mehrspaltige\\_Layouts](https://wiki.selfhtml.org/wiki/CSS/Anwendung_und_Praxis/mehrspaltige_Layouts)  
[https://www.w3schools.com/css/css\\_website\\_layout.asp](https://www.w3schools.com/css/css_website_layout.asp)

### 11) Einfache Animationen mit CSS und Animated Gif

<https://wiki.selfhtml.org/wiki/CSS/Eigenschaften/Animation/Transition>  
[https://www.w3schools.com/css/css3\\_animations.asp](https://www.w3schools.com/css/css3_animations.asp)

### 12) Videos ohne Plugins oder Add-Ons wiedergeben

[https://wiki.selfhtml.org/wiki/HTML/Multimedia\\_und\\_Grafiken/video](https://wiki.selfhtml.org/wiki/HTML/Multimedia_und_Grafiken/video)  
[https://www.w3schools.com/tags/av\\_met\\_play.asp](https://www.w3schools.com/tags/av_met_play.asp)

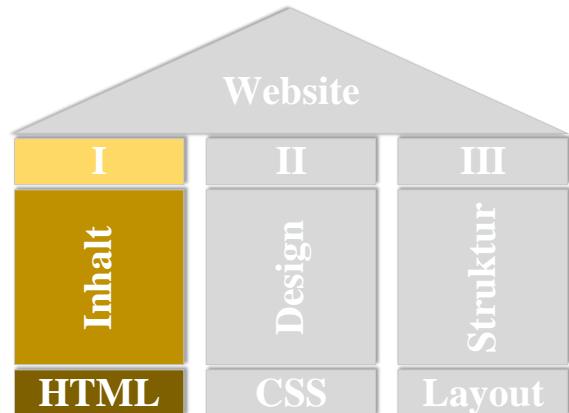
## 3.3 Vorgaben

A1	Es wird ein Thema aus dem Bereich der Webentwicklung erarbeitet und das Ergebnis wird im Plenum präsentiert. Die Themenauswahl wird gemeinsam im Plenum besprochen. Die Dauer der Präsentation ist 4 - 8 Minuten pro Person. Die Präsentation hat eine Einführung, einen Mittelteil und einen Abschluss mit Fazit.
----	--

## 4 HTML Inhalt I

### 4.1 Einleitung

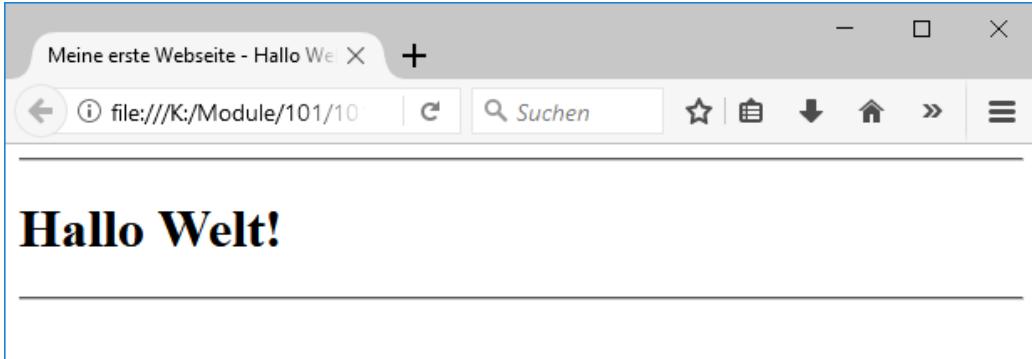
In der Webentwicklung werden Inhalt und Design strikt getrennt. Der Inhalt trägt alle Informationen eines Webauftritts und ist nach wie vor das wichtigste Element, da er die Botschaft vermittelt. Die Botschaft kann noch so brisant sein, ohne eine attraktive Gestaltung besucht trotzdem niemand Ihre Seite. Vorerst konzentrieren Sie sich aber auf Elemente, die Inhalte aufnehmen und auf verschiedene Art und Weise strukturieren können. Seien Sie also nicht enttäuscht, wenn Ihre ersten Webseiten noch eher fad aussehen, dies wird sich schon bald ändern.



Der Fokus liegt in diesem Kapitel ganz klar auf der **Hypertext Markup Language HTML**, der Auszeichnungssprache für das Web schlechthin.

### 4.2 Elemente, Tag und Attribut

#### 4.2.1 HTML-Grundgerüst und Ihre erste Webseite

A1	Erstellen Sie nun an Ihrem Arbeitsplatz ein Verzeichnis für das Modul WS. In diesem Verzeichnis erstellen Sie wiederum ein Verzeichnis mit dem Namen <code>1_grundgeruest</code> . Auf der Datei <code>index.html</code> erstellen Sie nun eine Webseite ohne Inhalt, die Sie künftig für alle weiteren Aufgaben als Vorlage benutzen können.	
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a>	<a href="#">Grundgerüst</a>
A2	In einem weiteren Verzeichnis <code>2_hallowelt</code> realisieren Sie wiederum auf der Datei <code>index.html</code> Ihre erste Webseite gemäss untenstehender Vorlage.  Beachten Sie auch die Meta-Daten im <code>&lt;head&gt;</code> .	
	  <a href="#">Lernvideo</a> <a href="#">Hello World</a>	
A3	Erstellen Sie künftig für alle Aufgaben ein eigenes Verzeichnis mit einer Nummer und dem Titel der Aufgabe ( <code>nummer_titel</code> ). Die Hauptseite heisst immer <code>index.html</code> . Die Übersicht über die Aufgaben wird so erleichtert.	

#### 4.2.2 Struktur und Kommentar

F1	Beachten Sie die HTML-Codestruktur. Wie sieht ein guter Stil aus?	
		wiki.selfhtml.org <a href="#">Guter HTML-Stil</a>
F2	Wie bringen Sie in HTML Kommentare an? Haben Sie Ihren Code angemessen kommentiert?	
		wiki.selfhtml.org <a href="#">Kommentar</a>

#### 4.2.3 Elemente zur Textstrukturierung

A2	Sie erhalten die Vorlage (textstrukturierung.pdf) einer Webseite in Form einer pdf-Datei. Erstellen Sie nun diese Webseite so originalgetreu wie möglich im Ordner 3_textstrukturierung nach.	
		wiki.selfhtml.org <a href="#">Hierarchien beachten</a> <a href="#">Textabsatz &lt;p&gt;</a> <a href="#">Zeilenumbruch &lt;br&gt;</a> <a href="#">Listen</a>
		<b>Vorgaben</b> 2_Textstrukturierung
A3	Achten Sie auf eine gute Struktur Ihres Codes und auf verständliche Kommentare.	

#### 4.2.4 Elemente zur Darstellung von Inhalten

A1	Sie erhalten wiederum eine Vorlage (elementedarstellung.pdf) inklusive Bilddateien. Gehen Sie wie bei Aufgabe 4.2.3 vor. Der Ordner heisst 4_elementedarstellung.	
		wiki.selfhtml.org <a href="#">Bilder im Internet</a> <a href="#">Tabellen</a> <a href="#">Links</a> <a href="#">Formulare</a>
		<b>Vorgaben</b> 3_Elementedarstellung

#### 4.2.5 Eigenschaften und Eigenheiten

Nach diesen zwei ersten Übungen lernen Sie HTML langsam kennen. Es wird Zeit, über einige Eigenschaften und Eigenheiten von HTML zu diskutieren. Halten Sie die Ergebnisse fest:

F1	Machen Sie sich über den Grundaufbau von HTML-Tags Gedanken. Was sind Attribute? Was sind Werte? Wie können Attribute Werte annehmen?	
		wiki.selfhtml.org <a href="#">Attribute in Tags</a>

F2	Umlaute und Sonderzeichen: Wie können Sie einen HTML-Tag im Browser darstellen? (Beispiel: Sie wollen den Tag <table> im Browser darstellen, natürlich mit den Klammern.)	
		<a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Zeichenreferenz</a>
F5	<p>Achten Sie auf die beiden HTML-Tags &lt;img&gt; und &lt;a&gt;. Versehen den Tag &lt;img&gt; ab sofort immer auch mit dem Attribut alt und den entsprechenden Werten.</p> <p>Versehen Sie den Tag &lt;a&gt; ab sofort immer auch mit dem Attribut target und den entsprechenden Werten.</p> <p>Was bewirken diese Attribute (alt und target) auf den Tags &lt;img&gt; und &lt;a&gt;?</p>	
		<a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#"><u>&lt;img&gt;</u></a> <a href="#"><u>&lt;a&gt;</u></a>

## 4.3 Navigation

### 4.3.1 Eine erste Navigation

Um sich auf Webauftritten komfortabel bewegen zu können, sind Navigationselemente unerlässlich. In diesem Kapitel realisieren Sie Ihre ersten Navigationsmenüs, beginnend mit einer Seitennavigation. Solche werden mit Vorteil mit einer <ul> in Kombination mit einer Verlinkung realisiert.

A1	<p>Vorbereitungen:</p> <ul style="list-style-type: none"> <li>• Erstellen Sie wiederum ein Verzeichnis für diese Aufgabe: 5_navigation</li> <li>• Ziehen Sie eine Kopie der Datei index.html aus dem Verzeichnis 3_textstrukturierung in das neue Verzeichnis und benennen Sie sie auf textstruktur.html um.</li> <li>• Ziehen Sie eine Kopie der Datei index.html aus dem Verzeichnis 4_elementedarstellung in das neue Verzeichnis und benennen Sie sie auf inhaltelemente.html um.</li> <li>• Kopieren Sie die index.html aus dem Verzeichnis 1_grundgeruest ebenfalls in das neue Verzeichnis. Lassen Sie diese vorderhand noch leer.</li> </ul>	
A2	<p>Erstellen Sie nun auf textstruktur.html und auf inhaltelemente.html eine Seiteninterne Navigation nachfolgendem Vorbild.</p>	
		<a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#"><u>Einfache Menüs</u></a> <a href="#"><u>Seiteninterne Verweise</u></a>
		<a href="#">Lernvideo</a> <a href="#"><u>Erste Navigation</u></a>

		<b>Elemente zur Textstrukturierung</b>	<b>Weitere Elemente zur Darstellung von Inhalten</b>
		<b>Menu</b> <ul style="list-style-type: none"> <li>• <a href="#">Trennlinien</a></li> <li>• <a href="#">Überschriften</a></li> <li>• <a href="#">Textabsätze</a></li> <li>• <a href="#">Zeilenumbruch</a></li> <li>• <a href="#">Listen</a></li> </ul>	<b>Menu</b> <ul style="list-style-type: none"> <li>• <a href="#">Grafiken</a></li> <li>• <a href="#">Tabellen</a></li> <li>• <a href="#">Verweise</a></li> <li>• <a href="#">Formulare</a></li> </ul>
A3		Bleibt noch die Datei index.html. Auf dieser realisieren Sie nun eine seitenübergreifende Navigation, damit textstruktur.html und inhaltelemente.html einzeln aufgerufen werden können. In den beiden Menüs braucht es nun noch einen weiteren Eintrag, um zu dieser Startseite zurückzugelangen.	<b>Webauftritt erstellen I, der Inhalte</b>
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a>	<a href="#"><code>&lt;a&gt;</code></a>	<b>Hauptmenü</b> <ul style="list-style-type: none"> <li>• <a href="#">Elemente zur Textstrukturierung</a></li> <li>• <a href="#">Weitere Elemente zur Darstellung von Inhalten</a></li> </ul>

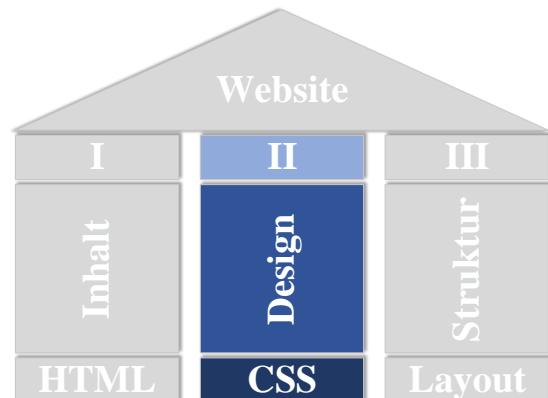
### 4.3.2 Absolut- und Relativpfad

F1	Diskutieren Sie bereits jetzt den Unterschied zwischen einer absoluten und einer relativen Verlinkung. Welche Konsequenzen hat es, wenn Sie einen Webauftritt absolut verlinken und diesen einem Mitlernenden zum Testen ausliefern? Falls Ihre Verlinkung mit absoluten Pfaden aufgebaut sind, ändern Sie diese in relative Pfade.	
	<a href="#">Wikipedia</a>	<a href="#"><u>Pfadname</u></a>

## 5 CSS Design II

### 5.1 Einleitung

Einen Designpreis können Sie mit Ihren bisher erstellten Webauftritten sicher nicht gewinnen. Immerhin haben Sie sich jedoch um den Inhalt gekümmert und Elemente kennengelernt, um diesen zu strukturieren.



- |    |  |
|----|--|
| A1 | Lassen Sie sich den Webauftritt der Neuen Zürcher Zeitung <a href="http://www.nzz.ch">www.nzz.ch</a> ohne jegliche Gestaltungselemente demonstrieren (Unter Firefox: Ansicht > Webseiten-Stiel > Kein Stiel wählen). Was stellen Sie fest? |
|----|--|

Es wird nun höchste Zeit, dass Sie Ihren Webauftritt gestalten können. Im Vergleich zum Printbereich werden im Screenbereich eher häufiger und differenzierter Formen und Farben eingesetzt. Die Kunst besteht darin, die unbeschränkten Gestaltungsmöglichkeiten gezielt und passend anzuwenden. Oft haben Sie sich als Webentwicklerin auch an Vorgaben des Auftraggebers oder Ihres Lehrbetriebes zu halten. Man spricht in diesem Zusammenhang von:

	<b>Corporate Identity (CI)</b>	Die Identität (lat. idem «derselbe») eines Unternehmens. Die Unternehmensidentität ist die Gesamtheit der kennzeichnenden und als Organisation von anderen Unternehmen unterscheidenden Merkmale.
	<b>Styleguide</b>	<p>Ein Styleguide (oder deutsch Gestaltungsrichtlinie) beschreibt, wie bestimmte Elemente eines Druckerzeugnisses oder einer Webseite zu gestalten sind. Diese Richtlinien sollen ein einheitliches Erscheinungsbild verschiedener Kommunikationsmittel einer Institution oder Firma, z. B. Werbe- und Informationsmaterial, Geschäftsbriefe oder Internetauftritte, gewährleisten.</p> <p>Bekannte Beispiele sind</p> <p>Microsoft Windows: <a href="#">Entwerfen und Codieren von Windows-Apps</a></p> <p>Apple iOS: <a href="#">Human Interface Guidelines</a></p> <p>Google Android: <a href="#">Design for Android</a></p>
Quelle: Wikipedia, die freie Enzyklopädie		

Wie lassen sich Webauftritte gestalten? HTML selbst bietet über Attribute und Werte einige Möglichkeiten:

A2	Codieren Sie folgendes HTML und betrachten Sie das Resultat im Browser:  .. <body bgcolor="maroon"> .. .. </body> .. Welche gravierenden Nachteile bestehen, wenn Sie Webauftritte so gestalten?
	Inhalt und Gestaltung werden strikt getrennt. HTML-Attribute, die die Gestaltung beeinflussen, sind nicht erlaubt. Webauftritte, die so gestaltet sind, müssen überarbeitet werden!

Gesucht ist also eine Technologie, mit der die Inhalte zentral gestaltet werden können. Hier hat sich ein mächtiges und trotzdem einfach zu erlernendes Werkzeug durchgesetzt. Ab sofort steht Ihnen ausschliesslich dieses Werkzeug für sämtliche Gestaltungsaufgaben von Webauftritten zur Verfügung:

## 5.2 Was ist CSS?

Die Gestaltungsmöglichkeiten mit CSS sind schier unbegrenzt. Ein h1 kann beispielsweise folgendermassen formatiert werden:

### Ich bin eine Überschrift 1, aber mit CSS gestaltet!

A1	Was bedeutet 'cascading'?	
		wikipedia.org
A2	Machen Sie Ihre ersten Erfahrungen mit CSS, versuchen Sie die Erklärungen und die Beispiele aus <a href="#">wiki.selfhtml.org</a> nachzuvollziehen. Studieren Sie in diesem Zusammenhang auch das <b>Box-Modell</b> ganz genau und probieren Sie gleich aus.	
		<a href="#">wiki.selfhtml.org</a>
		<a href="#">Syntax</a>
		<a href="#">Einbinden einer externen Datei</a>
		<a href="#">Box-Modell</a>
A3	Schauen Sie sich die Beispiele 1_schlechtes_Beispiel-HTML_und_CSS_gemischt_und 2_gutes_Beispiel-HTML_und_CSS_getrennt an. Was ist der Unterschied?	

Das schlechte hat mehrer Style formatierung.

		Vorgaben	4_CSS_Gute_und_schlechte_Beispiele
--	---	----------	------------------------------------

## 5.3 File Struktur anpassen

A3	Nebst HTML-Seiten und Bilddateien kommt nun ein weiteres Element – nämlich eine oder mehrere CSS-Dateien – zu Ihrem Webauftritt dazu. Es wird Zeit, über Ordnung und Struktur innerhalb eines Webauftritts nachzudenken. Entwerfen Sie eine allgemeingültige Verzeichnisstruktur für Webauftritte.
	<b>Tipp</b> <code>index.html</code> muss innerhalb eines Webauftritts immer auf Root stehen. Warum?

## 5.4 HTML Elemente gestalten

A1	Professionelle Webdesigner beschäftigen sich selbstverständlich auch mit der Farbenlehre. Nehmen Sie einen kurzen Einblick in folgende Websites und Sie erhalten wertvolle Tipps im Umgang mit Farben.		
		<b>99designs.de</b>	<a href="#">Die Grundlagen der Farbenlehre</a>
		<b>material.io</b>	<a href="#">COLOR TOOL</a>
A2	Denken Sie an die Optimierung Ihres CSS-Codes im Sinne von: «Kann ich dasselbe Resultat auch mit <b>weniger Codezeilen</b> erreichen?»		
		<ul style="list-style-type: none"> <li><b>Beispiel 1:</b> Die Elemente <code>h1</code>, <code>h2</code>, <code>h3</code>, <code>p</code>, <code>table</code> und <code>ul</code> haben gemäss Styleguide alle dieselbe Hintergrund- und Schriftfarbe. Wie sieht der optimale CSS-Code dafür aus?</li> <li><b>Beispiel 2:</b> <code>h1</code>, <code>h2</code>, <code>h3</code> und <code>h4</code> sollen einheitlich formatiert sein. Lediglich die Schriftgrösse soll gemäss Header-Hierarchie immer etwas kleiner werden. Auch hier lässt sich einiges an CSS-Code einsparen. Wie?</li> </ul>	
		<b>wiki.selfhtml.org</b>	<a href="#">Typselektor</a>
		<b>wiki.selfhtml.org</b>	<a href="#">ids und Klassen</a>
A3	Ziehen Sie nun eine Kopie Ihrer Lösung von Aufgabe 5_navigation. Gestalten Sie diesen Webauftritt mit Cascading Style Sheets.		
		<b>Tipps</b>	<ul style="list-style-type: none"> <li>Achten Sie auf ein einheitliches Auftreten der verschiedenen Elemente.</li> <li>Gehen Sie eher sparsam mit Formen und Farben um.</li> <li>Die Formatierung der Menüliste folgt im Kapitel 5.5</li> <li>Arbeiten Sie noch nicht mit Klassierungen. Formatieren Sie direkt die HTML-Tags, Beispiel: <code>h1</code></li> </ul>

```
{  
    background-color:#336699;  
    color:white;  
    ..  
}
```

## 5.5 Navigationsleiste erstellen

Es fehlt noch die Gestaltung der Navigationsleisten. Diese können ebenfalls sehr effektvoll mit CSS gestaltet werden. Dabei wird das HTML-Element `ul`, das die Menüeinträge beinhaltet, entsprechend ausgestaltet. Besonders interessant ist die Möglichkeit, beim Überfahren mit dem Mauszeiger über einen Listeneintrag gewisse Effekte auslösen zu können. Damit können mehrstufige Navigationsleisten realisiert werden.

- |    |  |
|----|--|
| A1 | Die Navigationsleiste wird mit CSS erweitert. Es soll ein optisch ansprechendes und schlichtes Menü entstehen. |
|----|--|



[wiki.selfhtml.org](http://wiki.selfhtml.org)

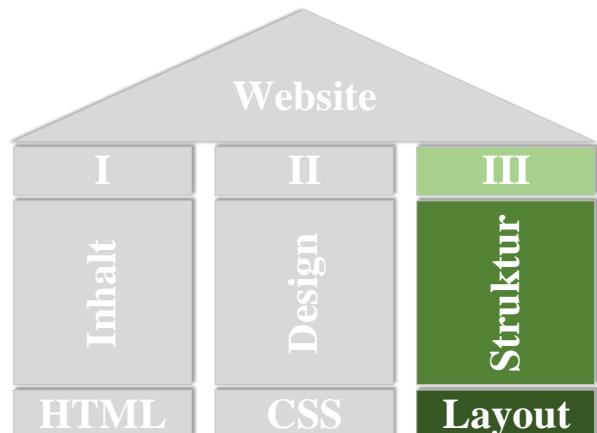
[Einfache Menüs](#)

## 6 Layout Seitenstruktur III

## 6.1 Einleitung

Sie haben nun Ihren Webauftritt gestaltet und sind damit einen grossen Schritt weitergekommen. Nach wie vor werden jedoch alle Elemente schön der Reihe nach dargestellt, gerade so, wie wir sie im HTML-Code aufgezeichnet haben. Was nun, wenn wir die Navigationsleiste linker Hand des Webauftritts platzieren wollen und uns dazu noch Bereiche für Inhalte, Bilder, Kopf- und Fusszeile zulegen wollen?

Jetzt wollen wir die Webseite in verschiedene Bereiche unterteilen und diesen einzelnen Teile den entsprechenden Elementen zuteilen. Dies kann mit verschiedenen Methoden realisiert werden. Bevor wir uns auf eine festlegen, befassen Sie sich mit folgender einleitender Aufgabe:



## 6.2 Layout in drei Varianten

A1	Sie erhalten 3 verschiedene Layouts. Schauen Sie sich die Layouts an, wie sie aufgebaut sind und stellen Sie jeweils pro Layout die Vor- und Nachteile auf. Überlegen Sie sich, welches Layout für eine Website am geeignetsten ist.		
Layout mit <div> Element	Layout mit HTML5	Responsive Design	
Vor- und Nachteile	Vor- und Nachteile	Vor- und Nachteile	
 <b>Vorgaben</b>	<a href="#">5_Layout_in_drei_Varianten</a>		
A2	Damit Ihr Layout auch auf verschiedenen Endgeräten optimal dargestellt werden kann, sind zwei Beispiele ( <a href="#">Responsive_Design-bootstrap.html</a> , <a href="#">Responsive_Design-float.html</a> ) vorgegeben. Studieren Sie beide Beispiele und erweitern Sie es mit den Städten Zürich und Bern.		
	<a href="#">wiki.selfhtml.org</a>		<a href="#">float</a>

Z1	Für ein dynamisch angepasstes Layout können Sie sich mit den folgenden zwei Themen auseinandersetzen.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Flexbox</a>
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">Grid Container</a>
A3	Strukturieren Sie nun Ihren Webauftritt. Ziehen Sie wiederum eine Kopie aus der jetzigen Webseite, damit Ihnen die bisherigen Resultate erhalten bleiben und erweitern Sie Ihre Webseite mit einem responsiven Design. Hier sind viele Experimente und Analysen nötig, um die Materie zu beherrschen.
	 <a href="http://wiki.selfhtml.org">wiki.selfhtml.org</a> <a href="#">mehrspaltige Layouts</a>

### 6.3 Codereview

F1	Demonstrieren Sie Ihre Webseite im Plenum. Sie haben nun einen komplett strukturierten, gestalteten und mit Inhalten versehenen Webauftritt.
----	--

## 7 Vertiefungsaufgaben

### 7.1 Meta-Angaben

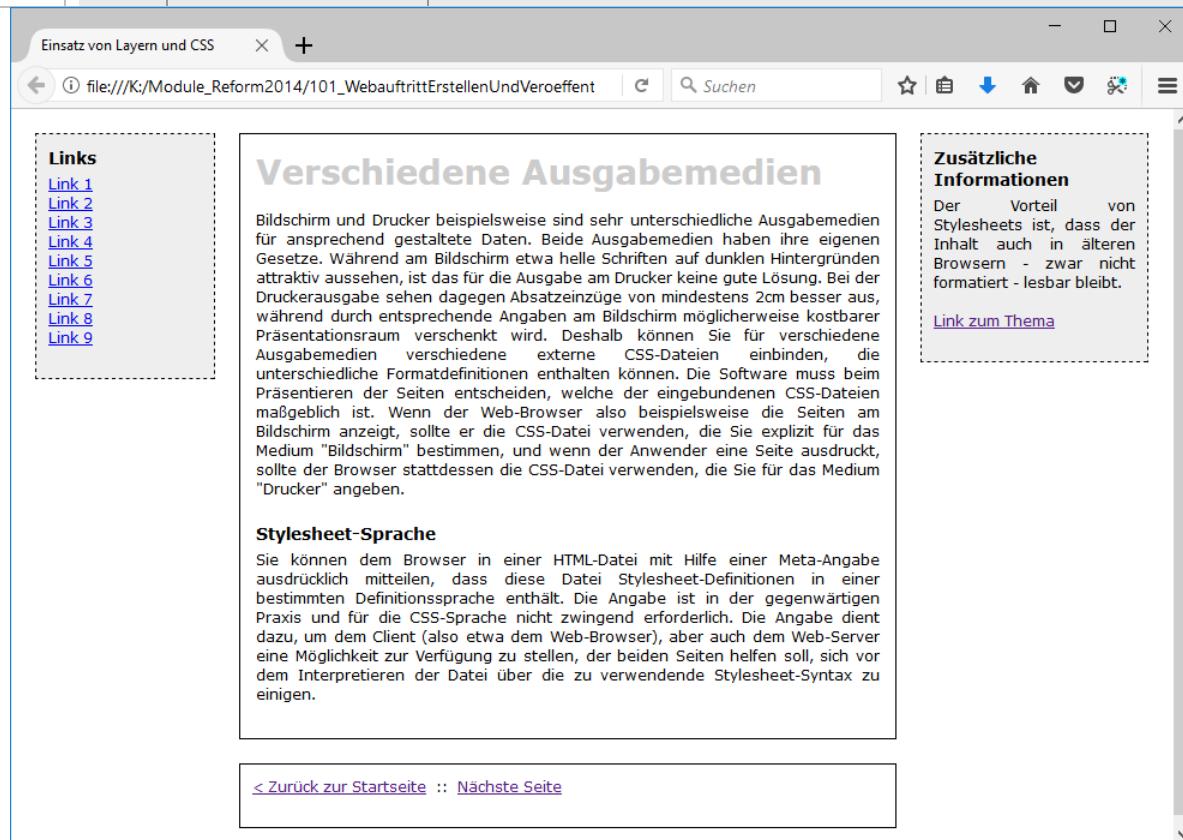
A1	<p>Erkundigen Sie sich über die Meta-Angaben. Sie werden feststellen, dass Sie in Meta-Angaben verschiedene nützliche Anweisungen notieren können. Erstellen Sie eine neue Seite, in deren Head folgende Angaben mit Meta-Tags definiert werden müssen.</p> <ul style="list-style-type: none"><li>• Zeichencodierung der Seite</li><li>• Schlüsselwörter mit den Worten: <i>Meta-Angaben, Aufgabe, Übungen, HTML</i></li><li>• Beschreibungstext mit dem Satz: <i>Dies ist eine Übungsseite mit Meta-Angaben. Schaut sie euch an und lernt, wie man Meta-Angaben macht und welche Aufgabe sie erfüllen.</i></li><li>• Sprache der Seite</li><li>• Datum der Publikation (das heutige Datum)</li></ul>
A3	<p>Die einzelnen Meta-Angaben sollen zusätzlich im Body der Seite angezeigt werden. Die Seite soll ein Tutorial für Metadaten sein. Beachten Sie, dass Sonderzeichen wie &lt; oder &gt; durch entsprechende Zeichenfolgen (z. B. &amp;lt; ;) ersetzt werden müssen, damit sie richtig angezeigt werden.</p>

## 7.2 Zusatzaufgaben zur Vertiefung

### 7.2.1 CSS für Print und Screen

- A1 Erstellen Sie folgendes Dokument in HTML und CSS. Der Printscreen veranschaulicht, wie das Endprodukt aussehen soll.

	Vorgaben	7_Projekt_Fischerfreunde
	wiki.selfhtml.org	<a href="#">Media Queries</a>



The screenshot shows a browser window with the title "Einsatz von Layern und CSS". The main content area displays a page with the heading "Verschiedene Ausgabemedien". On the left, there is a sidebar with a list of links labeled "Links" containing items like "Link 1" through "Link 9". On the right, there is a box labeled "Zusätzliche Informationen" containing text about the benefits of CSS for print and screen. At the bottom of the page is a navigation bar with links to "Zurück zur Startseite" and "Nächste Seite".

-  Die drei Elemente «Links», «Zusätzliche Informationen» und die untere Navigation sind freischwebende Layer und werden in der Druckanzeige nicht angezeigt.
- A2 Die Seite soll also ausgedruckt anders strukturiert sein als auf der Bildschirmansicht. Dies macht insofern Sinn, als dass gerade Navigationsleisten auf dem Papierausdruck nutzlos sind und blass Platz verschwenden. Der nebenstehende Printscreen zeigt das Layout im Modell. Den Styleguide und das Originallayout erhalten Sie ebenfalls als pdf-Vorgabe.



### 7.2.2 Imagemap

- A1 Erstellen Sie ein einziges Bild mit verschiedenen Farben. Sie können beispielsweise die 16 Grundfarben auf einem Bild darstellen. Verlinken Sie die 16 Farbflächen mittels Image-map so, das sich die dazugehörige Farbe in einem neuen Fenster öffnet.



[wiki.selfhtml.org](http://wiki.selfhtml.org)

[verweisensitive Grafiken](#)

### 7.2.3 Canvas

- A1 Erstellen Sie in einem Canvas eine Animation nach einer eigenen Idee.



[wiki.selfhtml.org](http://wiki.selfhtml.org)

[Canvas](#)



[w3schools.com](http://w3schools.com)

[Canvas](#)

## 8 Webauftritt veröffentlichen

A1 Eigene Website lokal betreiben

1. Installieren Sie Windows in einer virtuellen Maschine unter Hyper-V. Das ist nun Ihr Webserver.
2. Installieren Sie auf dem neuen Rechner [XAMPP](#) und starten Sie den Webdienst.
3. Geben Sie das Laufwerk C:\XAMPP\htdocs frei (*Ordner > Eigenschaften > Freigabe > Erweiterte Freigabe...*)
  - a. Erweiterte Freigabe: Haken für «Diesen Ordner freigeben» setzen.
  - b. Berechtigungen > Vollzugriff zulassen > OK
  - c. Den Netzpfad \\Computername\htdocs merken.
4. Verbinden Sie sich nun vom physischen Rechner mit dem freigegebenen Netzlaufwerk des virtuellen Computers, indem Sie im Explorer \\Computername\htdocs eingeben.
5. Nun können Sie Ihre Website vom physischen auf den virtuellen Computer kopieren.
6. Nun ist die Website online. Testen Sie diese mittels <http://computername> auf dem physischen Rechner.



[apachefriends.org](http://apachefriends.org)

[XAMPP Apache + MariaDB + PHP + Perl](http://xampp-project.com)

## 9 Projekt

### 9.1 IPERKA

Als Projektmanagementmethode wird IPERKA verwendet und die Webseite wird über die fünf Phasen entwickelt.

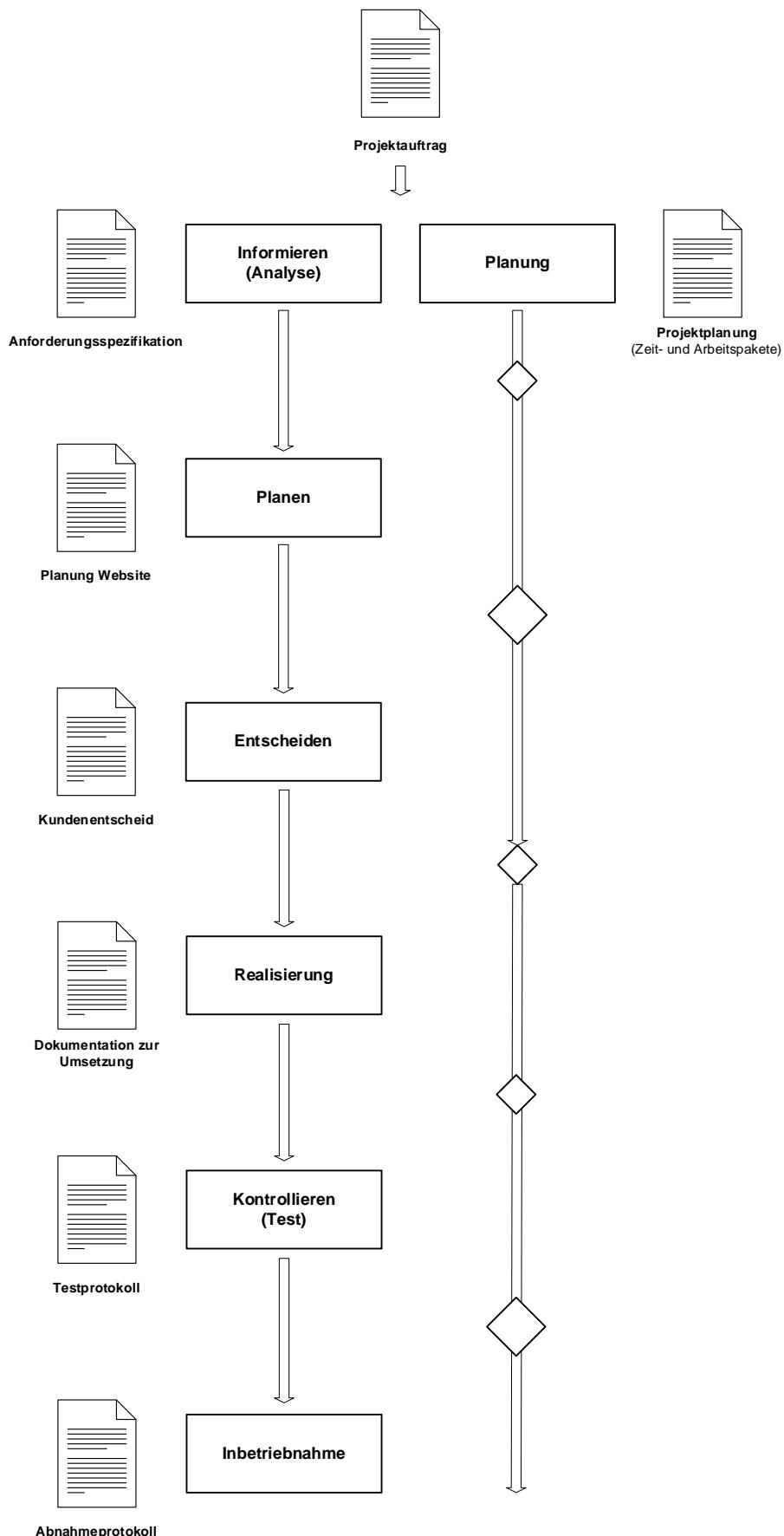
	wikipedia.org	<a href="#">IPERKA</a>
---	---------------	------------------------

### 9.2 Dokumente der einzelnen Projektphasen

Neben der effektiven Software, welche innerhalb eines Projektes entwickelt wird, resultiert aus jeder Phase eines Projektes im Minimum noch ein Dokument. Auch bei der Bezeichnung dieser Dokumente werden in der Praxis zum Teil unterschiedliche Begriffe verwendet, was oft zu Missverständnissen führt. Undurchsichtig wird es immer dann, wenn alle Dokumente ohne Differenzierung einfach als Konzept bezeichnet werden.

Modulunterlagen

---



### **9.2.1 Anforderungsspezifikation**

Eine Anforderungsspezifikation ist das Resultat der Analyse und beschreibt hauptsächlich die Funktionalität und die nichtfunktionalen Anforderungen an eine Software. Die Funktionalität beschreibt dabei, was die Software effektiv kann (z.B. Kontaktmöglichkeit mit E-Mail). Die nichtfunktionalen Anforderungen (auch Produktanforderungen genannt) beschreiben zusätzlich Anforderungen an die Erscheinung, an die Sicherheit und an die Wartbarkeit der Software.

Am Beispiel eines Webauftritts wird mit einer sauberen Anforderungsspezifikation sichergestellt,

- dass die Grundfunktionalität des Webauftritts definiert ist,
- dass alle Anforderungen an den Webauftritt ausreichend durchdacht und definiert sind,
- dass jeder in Ihrem Projektteam die gleiche Sprache spricht,
- dass jeder im Projektteam das gleiche Verständnis über die einzelnen Elemente des Webauftritts hat,
- dass die Programmierer und Designer die Ideen des Kunden genau verstanden haben.

Ferner bildet die Anforderungsspezifikation mit der Funktionsbeschreibung auch die wichtigste Grundlage für die Abnahme des fertigen Produktes am Ende des Projektes. Zu diesem Zeitpunkt wird die Anforderungsspezifikation herangezogen, um objektiv nachprüfen zu können, inwiefern die am Projekt beteiligten Parteien, ihre vertraglichen Verpflichtungen erfüllt haben.

### **9.2.2 Projektdokumentation**

Nachdem die Analyse mit der Anforderungsspezifikation abgeschlossen ist, folgt die Phase des Entwurfs oder auch die Phase der Planung der Webseite. Zum einen gibt es eine Zeitplanung, wann welches Arbeitspaket erledigt wird. Zum anderen gibt es eine Planung des grafischen Designs. Das bedeutet, dass man bereits mit dem Kunden zusammensitzen kann und den Entwurf der Webseite bespricht.

Am Beispiel eines Webauftritts müssen in der Planung folgende Punkte beschrieben werden:

- Projektplanung (ist für dieses Modul optional)
  - Zeitplan
  - Arbeitspakete
- Grafische User Interface
  - Beschreibung der Funktionen
  - Struktur resp. Gliederung
  - Navigationskonzept, wie wird welche Seite erreicht?
  - Darstellung mit Prototypen
  - Darstellung der Grafik
  - grundsätzlicher Seitenaufbau
  - Schriftarten, welche Schriftarten werden verwendet?
  - Schriftgrößen, welche Schriftgrößen werden verwendet? Wird in einer Tabelle definiert.
  - Farben, welche Farben werden verwenden (mit Farbcode)? Wird in einer Tabelle definiert.
  - Logos, was gibt es für Logos und wo werden sie platziert?
  - Bilder resp. Fotos
  - Größen von Tabellen, Menüs, Bilder usw.
  - Animation
- Gestaltungskonzept (Corporate Design)
- Redaktionelle Richtlinien
- Technologie mit Versionsangabe z. B. HTML5, CSS3 etc.

**Modulunterlagen**

---

- Frameworks z. B. Bootstrap in CSS
- Veröffentlichung, wie soll die Webseite veröffentlicht werden? z. B. mit XAMPP

In der Umsetzung wird vor allem schwierig verständliche Code kommentiert, damit das Lesen des Codes von anderen Personen einfacher ist. Damit ist die Webseite auch wartbar und kann einfach angepasst und erweitert werden.

- Kommentar im Code

Für das Testen braucht es zum Schluss ein Testprotokoll

- Testprotokoll
  - Funktionale Anforderungen überprüfen
  - Nichtfunktionale Anforderungen überprüfen

Am Schluss wird die Webseite dem Kunden übergeben und mit einem Abnahmeprotokoll bestätigt der Kunde, dass er zufrieden ist. Es kann aber auch sein, dass es Mängel gibt, welche behoben werden müssen.

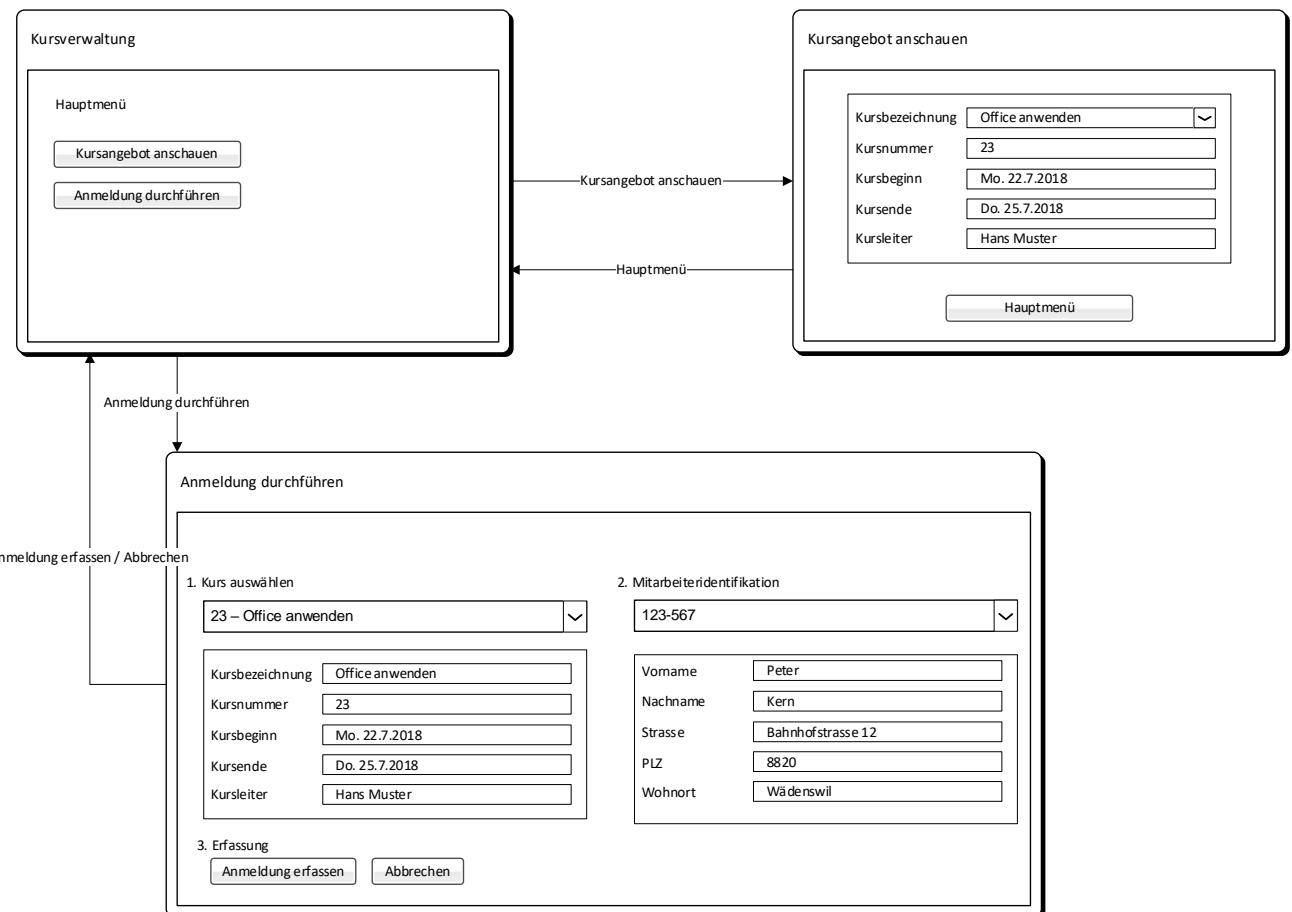
- Abnahmeprotokoll mit dem Kunden zusammen.

## 9.3 Prototypen

Als Basis für den Entscheid werden in der Praxis oft auch sogenannte Papier-Prototypen erstellt. Dabei handelt es sich um nicht funktionierende Darstellungen der zukünftigen grafischen Benutzeroberfläche. Prototypen werden einerseits erstellt, damit sich der Kunde besser vorstellen kann, wie sich der Webauftritt später präsentieren wird und als weiterer wichtiger Punkt dienen Prototypen der Kontrolle, ob die Anforderungen (Vorstellungen) des Kunden richtig interpretiert wurden.

Prototypen können bei einem Webauftritt als rein statische HTML-Seiten ausgeführt werden oder können auch einfach mit anderen Hilfsmitteln (z.B. PowerPoint, Visio) dargestellt werden, damit nicht bereits aufwendige Buttons und Bilder hergestellt werden müssen.

### Beispiel schwarz/weiss Papier Prototyp



## 9.4 Umsetzung

Das Projekt wird gestartet und die komplette Projektdokumentation inklusive der Realisierung wird in den einzelnen Projektphasen umgesetzt.

	<b>Vorgaben</b>	7_Projekt_Fischerfreunde
---	-----------------	--------------------------

A1	<b>Informieren</b> Lesen Sie die Anforderungsspezifikation und informieren Sie sich über den Projektauftrag.
A2	<b>Planen</b> Erstellen Sie die vollständige Projektdokumentation für die Planung.
A3	<b>Entscheidung</b> Sitzen Sie mit dem Kunden, in diesem Fall der Kursleiter, zusammen und besprechen Sie Ihre Planung mit ihm. Für die Umsetzung der Planung braucht es die Zustimmung des Kunden. Es soll eine gemeinsame Entscheidung sein. Der Kunde ist mit dem Plan zufrieden und Sie können diesen Plan in der vorgegebenen Zeit erfolgreich umsetzen.
A4	<b>Realisieren</b> Entwickeln Sie die Webseite anhand Ihres Planes.
A5	<b>Testen</b> Testen Sie den Webauftritt
A6	<b>Abnahme</b> Veröffentlichen Sie die Webseite und lassen Sie den Kunden die Seite anschauen. Ist der Kunde zufrieden, haben Sie einen hervorragenden Job gemacht.