

Webseiten programmieren und gestalten

Das umfassende Handbuch

DAS INHALTS- VERZEICHNIS

» Hier geht's
direkt
zum Buch

Inhalt

1	 Websites, Webentwicklung, Full Stack – was Sie in diesem Buch erwartet	19
1.1	 Frontend vs. Backend	21
1.2	 Was müssen Sie können?	22
1.3	 Umfassendes oder gar vollständiges Wissen?	23
1.4	 Wie korrekt ist korrekt?	24
1.5	 In eigener Sache – Danksagung	24
2	 Good to know – etwas Vorwissen	25
2.1	 Die Planung einer Website	25
2.2	 Ziele einer Website	27
2.3	 Marktanalyse, um den Markt zu verstehen	29
2.4	 Der User, das (un)bekannte Wesen	30
2.4.1	Einfühlungsvermögen ist gefragt	30
2.4.2	Die Zielgruppenanalyse als Mittel, den User kennenzulernen	31
2.4.3	Personas	38
2.4.4	Die Anforderungsanalyse, um die Anforderungen der User zu erkennen und erfüllen zu können	39
2.4.5	Die Bedürfnisse des Users erfüllen: (Visual) Storytelling	42
2.5	 Konkurrenzanalyse: Wie sieht die Konkurrenz aus?	46
2.5.1	Die Vorgehensweise einer (einfachen) Konkurrenzanalyse	47
2.6	 Content is King – die Content Strategy	50
2.7	 Zielorientiertes Interaktionsdesign	51
2.8	 Ein wenig Wahrnehmungspsychologie	52
2.9	 Wie Farbe wirkt	53
2.9.1	Licht und Physik – wie Licht entsteht und was wir darunter verstehen	53
2.9.2	Licht und Biologie – wie wir Licht wahrnehmen	55
2.9.3	Wie Menschen Farben empfinden	57
2.9.4	Licht und Mathematik – die Farbmodelle als Ergebnis	67

2.9.5	Farbmodell versus menschliches Auge – das L*a*b-Modell	75
2.9.6	Farbzusammenstellung und Farbharmonie aus Liebe zum User	77
2.10	Die Unternehmensphilosophie wird in der Corporate Identity ausgedrückt	80
3	Das liebe Internet – ein wenig Basiswissen	83
3.1	Dienste im Internet, Server und Client	84
3.2	Protokolle und Ports – eine Frage der Kommunikation	85
3.2.1	HTTP/1.1, HTTP/2 oder HTTP/3 und die Statuscodes	88
3.2.2	Ports – die Türchen der Protokolle	90
3.3	IP-Adressen – unsere Anschrift im Internet	91
3.3.1	IPv4 – sind 4,3 Milliarden Adressen zu wenig?	92
3.3.2	IPv6 – mehr als ausreichend viele Adressen	94
3.3.3	Internet Service Provider sind das Tor zur Internetwelt	95
3.4	Domains	97
3.4.1	Top-Level Domain (TLD) – der Ausgangspunkt	99
3.4.2	Second-Level Domain (SLD) – die Qual der Wahl	101
3.4.3	Um's Eck zum Kiosk: Vergabestellen von Domains	102
3.4.4	Nameserver und Domain Name Service (DNS) – einmal IP-Adresse und wieder zurück	103
3.5	Mime Types und Content Types – Sein und Schein	104
3.6	Ein paar notwendige Begriffsdefinitionen	105
3.7	Wie gelangen unsere Daten auf den Server? FTP macht's vor	106
3.8	Wie ein Webserver funktioniert	107
3.8.1	Die Qual der Wahl: Apache vs. NGINX	108
3.8.2	HTTP vs. HTTPS – unsicher oder doch sicher?	115
3.8.3	Ein paar Worte zum Cloud-Hosting	116
3.9	Ach ja, und wie funktioniert nun eine Website?	116
4	Die lokale Entwicklungsumgebung	121
4.1	Server-Software – wir spielen Provider	121
4.2	Editoren	123
4.3	Der Browser als Interpreter und Testumgebung	125

5	HTML – Die Grundlage einer Website	133
5.1	Ein bisschen in der Geschichte von HTML stöbern	135
5.2	Das Grundgerüst einer HTML-Seite	139
5.3	Der Dokumentbaum	142
5.4	Die Elemente einer HTML-Seite	145
5.4.1	Textueller Inhalt	145
5.4.2	Überschriften	146
5.4.3	Listen	149
5.4.4	Tabellen für die Darstellung tabellarischer Daten (und sonst nichts)	154
5.4.5	Bilder als optisches Glück und technische Herausforderung	165
5.4.6	Links sind die Grundidee des WWW	183
5.5	Easy: Block vs. Inline – die grundlegendste Unterscheidung von Elementen	187
5.6	Not so easy: Content Categories ab HTML5	191
5.6.1	Main Content Categories	192
5.6.2	Form-related Content Categories	194
5.7	Bedeutungslose Elemente – nichts für echte Webdesigner*innen	194
5.8	Elemente mit (besonderer) Bedeutung	195
5.8.1	Strukturgebende Elemente sind die Basis jeder Webseite	195
5.8.2	Ein wichtiges Spezialthema: die Navigation	209
5.8.3	Übersicht und Detail – angeteaserte Inhalte	210
5.9	Formulare als Schnittstelle zwischen Client und Server	212
5.9.1	Arten der Datenübertragung	213
5.9.2	Die Anfrage an den Server	214
5.9.3	Der User ist gefordert: Formularelemente	215
5.9.4	Mehr Übersicht, mehr Restriktion: Formularfeld-Attribute	226
5.9.5	Für den User und die Suchmaschinen: Beschriftung von Eingabeelementen	228
5.9.6	Ein wenig Übersicht tut gut: Wie Sie Formulare in mehrere Abschnitte gliedern	229
5.9.7	Formulare aufgebohrt I: Datei-Upload	230
5.9.8	Formulare aufgebohrt II: Dialoge	232
5.9.9	Formulare aufgebohrt III: Suchbereiche	234
5.9.10	Übung: Ein exemplarisches Formularbeispiel	235
5.10	Multimedia mit Audio, Video & Co.	236
5.10.1	Das bewegte Bild: Video-Einbettung	236

5.10.2	Das gesprochene Wort: Audio-Einbettung	237
5.11	Meta-Informationen sind der Mehrwert von Websites	238
5.11.1	Wofür, weshalb und von wem? Beschreibende Meta-Tags	239
5.11.2	Ein paar Angaben zur Site-Darstellung	240
5.11.3	Ein paar Anweisungen für Suchmaschinen	241
5.11.4	Ein paar Angaben zur Content Security Policy und zum Referer	242
6	CSS – Formatierung rulez	249
6.1	Die drei Säulen von CSS	250
6.2	Die Stylesheets – so geht der Browser vor	250
6.3	CSS und HTML – ein schönes Paar	256
6.3.1	So geht's: Das style-Tag	256
6.3.2	So geht's auch: Den CSS-Code in eine externe Datei auslagern	257
6.3.3	So wollen wir's nicht: Einbinden des CSS-Codes in ein style-Attribut	258
6.4	Und die Schreibweise von CSS-Regeln?	258
6.5	Ein Muss: Selektionen und Spezifitäten	260
6.5.1	Die Berechnung der Spezifität	265
6.5.2	Selektion auf Basis von Attributen	267
6.5.3	Eltern, Kinder und Geschwister – verschachtelte Elemente	268
6.5.4	Pseudoklassen und Pseudoelemente für weitere Auswahlkriterien	273
6.6	Die Vererbung – mehr Pro als Kontra	277
6.7	Einheiten in CSS – nicht alles ist relativ	280
6.8	Sind ja alles nur Boxen – das Box-Modell	282
6.8.1	Das Box-Modell für Block-Elemente	283
6.8.2	Das Box-Modell für Inline-Elemente	288
6.8.3	Wir haben es in der Hand: Die Darstellung von Boxen steuern	289
6.9	Glanz und Gloria – Farben	294
6.10	Das geschriebene Wort – Schriften	295
6.10.1	Corporate Design Fonts auf Websites – Pro und Contra	299
6.10.2	Ein paar Worte zur Typografie	299
6.10.3	Wie lesbar sind Schriften?	301
6.10.4	Statische vs. variable Fonts – die Zukunft der Schriftformate?	304
6.10.5	Schriftnutzung auf Websites – Webfonts, Google Fonts und Lizenzschriften	305

6.10.6	Die Einbettung von Google Fonts – ein ehrlich gemeinter Dank an Google	312
6.11	Textumfluss (float) – umflossene Elemente wie in Druckmedien	320
6.12	Der Hintergrund als Gestaltungselement	325
6.12.1	Hintergrundbilder sind keine Bilder	328
6.12.2	Das Farbenspiel der Hintergrundverläufe	330
6.13	Positionierungsarten – Normal Flow vs. individuelle Positionierung	332
6.13.1	position:relative	334
6.13.2	position:absolute	335
6.13.3	position:fixed	339
6.13.4	position:sticky	340
6.13.5	Der z-index	342
6.14	Tabellenartiges Design mit CSS	343
6.15	Listenartiges Design mit CSS	350
6.16	CSS Counters – Individuelle Nummerierung mit CSS gefällig?	352
6.17	Fortgeschrittenes Layout – Jetzt wird's fein	356
6.17.1	Flexboxen – eindimensionales Denken	356
6.17.2	Grids – zweidimensionales Denken	363
6.18	Alles ist in Bewegung – Übergänge und Animationen	379
6.18.1	Transitions als Vorstufe zu Animationen – die einfache Variante	380
6.18.2	Keyframe-Animationen – Animation in Vollendung	383
6.18.3	Das Spiel mit der Perspektive: Der Parallax-Effekt	387
6.19	Ein erster Schritt in Richtung Programmierung mit CSS-Variablen	391
6.19.1	Mathematische Berechnungen in CSS	393
6.20	Media Queries und @-Regeln aus Rücksichtnahme auf die User-Anforderungen	395
6.20.1	width und height	397
6.20.2	orientation	398
6.20.3	pointer	399
6.20.4	resolution	401
6.20.5	Dark Mode vs. Light Mode	402
6.20.6	Was wird vom Browser unterstützt? Ein Blick auf die @supports-Regel	407
6.21	Advanced Stuff: CSSOM	408
6.21.1	Hindernisse	411
6.22	Sass, LESS & Co – der Einsatz von CSS-Präprozessoren	411
6.23	Übungsbeispiel	412

7	JavaScript – die logische Abteilung auf der Clientseite	415
7.1	Eine erste Abgrenzung zu anderen Sprachen und Frameworks	417
7.2	Scriptsprachen und HTML – her mit der Interaktion	418
7.3	Bevor es los geht, ein wenig Vorbereitung	419
7.3.1	Wie schreibt man Programmcode?	420
7.3.2	Kommentare in JavaScript	421
7.3.3	Hilfe bei der (notwendigen) Fehlersuche: Diverse Ein- und Ausgabemöglichkeiten	421
7.3.4	Die Entwicklerkonsole ist ein Segen fürs Webdesign	422
7.3.5	Kein Inhalt sichtbar?	423
7.3.6	async und defer – zwei Lösungsansätze für blockierende Scripts	427
7.4	Die essenziellen Grundlagen der Programmierung	435
7.4.1	Kleine Wertespeicher – Variablen	435
7.4.2	Wir treffen Entscheidungen: Bedingungen	454
7.4.3	Schleifen lassen Code im Kreis laufen	462
7.4.4	Nicht eine, sondern Hunderte Variablen benötigt? Arrays!	471
7.4.5	Benannte Array-Einträge? Objekte (assoziative Arrays)!	489
7.4.6	Vorbereitet, aber (noch) nicht serviert: Funktionen	494
7.4.7	Bekannt oder anonym?	506
7.4.8	Pfeilfunktionen (Arrow Functions)	509
7.5	Das Zusammenspiel zwischen HTML und JavaScript – das Dokumentobjektmodell (DOM)	510
7.5.1	Die Geschichte des DOM	511
7.5.2	Unsere erste Wahl: document	512
7.6	Ereignisse bestimmen den Ablauf	527
7.6.1	Event Listener vs. on-Events vs. Inline Event Handler	528
7.6.2	Event Bubbling: Worauf wurde (alles) geklickt?	539
7.6.3	Schritt für Schritt: Ein Bildwechsler mit JavaScript	546
7.7	Wir haben ja Zeit: Zeitlich gesteuerte Befehle	550
7.8	Mehr als nur document: Das window-Objekt	552
7.8.1	Unser Browser, der »Navigator«	553
7.8.2	Wo wir uns befinden – die Location	554
7.8.3	Wo wir waren – die History	555
7.9	Das mühselige Arbeiten mit Datum und Uhrzeit	556
7.10	Synchron oder besser asynchron? AJAX, Promises und die fetch-API	557
7.10.1	Datenübergabe an den Server	566

7.10.2	Promises als das bessere Ereignismodell?	569
7.10.3	Die fetch-API als bessere Alternative zu AJAX	576
7.11	Web-APIs	579
7.12	Auslagern von Code und Modulen	581
7.13	Robuste Programmierung – der Umgang mit Fehlern	587
7.13.1	Progressive Enhancement vs. Graceful Degredation	588
7.13.2	Fehlertoleranz	589
7.13.3	Was geht, was geht nicht?	589
7.13.4	Exceptions und try-catch	592
7.13.5	Eigene Exceptions	593
7.13.6	Globale und lokale Variablen	595
7.13.7	Weitere Hilfsmittel zur Zielerreichung	596
7.14	Wo das W3-Konsortium nicht mehr weiter wusste:	
	HTML-Elemente mit JS-API	597
7.14.1	Die Video- und Audio-API zum Steuern von Medieninhalten	597
7.15	Lokale Datenspeicherung – wohin mit den Daten?	602
7.16	Vorlagen (Template Literals)	605
7.17	Das Finale? Prototypen, Klassen, Architektur und Co.	605
7.17.1	Website-Architektur	606
7.17.2	Lust auf mehr!	608
7.18	Übungsbeispiel	608

8 Responsive Webdesign – Verantwortung dem User gegenüber 611

8.1	Die Grundlagen des Responsive Design	611
8.1.1	Die fünf Säulen des Responsive Design	612
8.1.2	Eine Abgrenzung zum adaptiven und liquiden Design	613
8.1.3	Frontend oder Backend?	613
8.1.4	Wie mobil ist mobil?	614
8.2	Wie geht man's an? Der Workflow	614
8.2.1	Zwei Ansätze: Mobile First und Desktop First	615
8.2.2	Der Anteil der mobilen User überwiegt: richtig und falsch	616
8.3	Content- und Layoutstrategien für responsive Websites	616
8.3.1	Grundbausteine responsiven Designs im Schnelldurchlauf	618
8.3.2	Überwiegend flüssiges Design	618

8.3.3	Spalten verschieben	619
8.3.4	Layout umschalten	620
8.3.5	Kleine Veränderungen	622
8.3.6	Off-Canvas	622
8.4	Umbruchpunkte definieren, wo das Layout ein anderes wird	623
8.5	Die Grundbausteine einer responsiven Site	625
8.5.1	Media Queries als das verbindende Element	625
8.5.2	Der Gestaltungsraster als Basis	626
8.5.3	Flexible Medien als Kür	638
8.5.4	Navigation	680
8.6	Mobile Usability – Usability weitergedacht	693
8.6.1	Bedienfähigkeit	694
8.7	Alles performt – optimiert geht’s schneller	697
8.7.1	Weniger Bilder laden	704
8.7.2	CSS- und JavaScript-Dateien zusammenfassen	704
8.7.3	Weitere Tipps	706
8.8	Über den Tellerrand geblickt: Fortgeschrittene Themen	707
8.8.1	Device Detection vs. Feature Detection	708
8.8.2	RESS – Responsive Webdesign with Server-Side Components	715
8.9	Ausblick	716
9	Veröffentlichung und Versionierung	717
9.1	Der Veröffentlichungsprozess	717
9.2	Die Versionierung	720
10	Usability, User Experience und Barrierefreiheit	723
10.1	Usability	724
10.1.1	Klappt das alles, wie wir es uns gedacht haben? Usability-Evaluierung	726
10.1.2	Sind Tests überhaupt notwendig?	734
10.1.3	Einen Test planen und durchführen	735
10.1.4	Den Unwahrheiten auf der Spur: Usability-Mythen	739
10.1.5	Usability-Tipps für gute Usability	740
10.1.6	Typische Usability-Fehler	743

10.2	User Experience	744
10.2.1	User Centered Design	746
10.3	Barrierefreiheit	747
10.3.1	Relevante Einschränkungen	748
10.3.2	Was hilft, was unterstützt? Assistierende Technologien	749
10.3.3	Barrierefreiheit im Internet – Web Accessibility	750
10.3.4	Was bleibt für uns zu tun?	755
10.3.5	Weitere Quellen und hilfreiche Tools	760
11	Die Serverseite bzw. das Backend – Programmieren mit PHP	761
11.1	PHP gegen den Rest der Welt?	762
11.2	Aller Anfang ist leicht	763
11.3	Die Entwicklungsumgebung	766
11.4	Imperativer oder objektorientierter Ansatz?	768
11.5	Jeder macht Fehler – das Fehlermodell von PHP	769
11.6	Die Unterschiede in der Schreibweise von JavaScript und PHP	771
11.6.1	Variablen und Konstanten	772
11.6.2	Bedingungen	772
11.6.3	Schleifen	773
11.6.4	Arrays und Objekte	773
11.6.5	Funktionen	777
11.6.6	Konstanten und globale Variablen	781
11.7	Trial and Error: try-catch	782
11.8	Hilfreich: Code in externe Dateien auslagern	784
11.9	Debugging für PHP-Entwickler?	786
11.9.1	Vorbereitungen treffen	789
11.9.2	Variableninhalte und Co. auf dem Bildschirm ausgeben	789
11.9.3	Error-Logging	793
11.10	Der Anknüpfungspunkt an HTML – Formulare als wesentliches Kommunikationsmittel	793
11.11	Den HTTP-Header manipulieren – serverseitige Weiterleitungen	804
11.12	»Kennst Du mich noch?« – Sessions (Sitzungen)	807
11.12.1	Ablegen von serverseitigen Informationen – das \$_SESSION-Array	815

11.12.2 Weg damit: Session und Session-Variablen löschen	818
11.13 Formulare aufgebohrt: Datei-Upload	819
11.14 Die Dateistruktur auf der Serverseite – unsere Ablage	828
11.14.1 Den Inhalt von Verzeichnissen auslesen	829
11.14.2 Neue Verzeichnisse anlegen	839
11.14.3 Bestehende Verzeichnisse löschen	845
11.14.4 Bestehende Verzeichnisse und Dateien umbenennen bzw. verschieben	847
11.14.5 Nicht für alle bestimmt – Dateien und Verzeichnisse schützen	852
11.15 Bilder skalieren	855
11.15.1 Verbesserungspotenzial detektiert	857
11.15.2 Alternative Funktionen zu imagescale	859
11.16 Die Kommunikation mit der Außenwelt – der Versand von E-Mails	864
11.16.1 Zu einfach und schnell im Spam: Die mail-Funktion	865
11.16.2 Die sinnvolle Variante: Die PHPMailer-Library	869
11.17 Ausblick	872

12 Wohin mit all den Daten? Datenbanken liefern die Antwort 875

12.1 Alles Datenbank, oder was? Ein paar Begriffe	876
12.2 Wir sind Administrator – die Verwaltung der Datenbank	876
12.3 Tabellen und Co. – der Aufbau einer Datenbank	878
12.3.1 Wertetypen	880
12.3.2 Tabellen innerhalb einer Datenbank anlegen	884
12.3.3 Werte in eine Tabelle einfügen	887
12.4 Sicher ist sicher – Sichern einer Datenbank	889
12.5 Die zwei Freunde Webserver und Datenbankserver – Datenbankanbindung mittels PHP	891
12.5.1 MySQLi	892
12.5.2 PDO (PHP Data Objects)	894
12.5.3 Räumen wir auf: MySQLi und PDO in einem Wrapper	895
12.6 Noch eine Sprache – SQL	898
12.6.1 PDO-Special: Varianten zum Auslesen von Datensätzen	901
12.6.2 Das Abfragen von Datensätzen: SELECT	902
12.6.3 Das Einfügen neuer Datensätze: INSERT	920

12.6.4	Das Aktualisieren bestehender Datensätze: UPDATE	923
12.6.5	Das Löschen bestehender Datensätze: DELETE	924
12.6.6	Ein Anwendungsbeispiel in Form eines Redaktionssystems für eine Tabelle	925
12.7	Alles eine Sache der Vorbereitung: Prepared Statements	933
12.7.1	Verwendung eines Parameters	934
12.7.2	Verwendung mehrerer Parameter	935
12.8	Datenbankzugriff verbessert: PDO aufgebohrt	936
12.8.1	Persistente Verbindungen	936
12.8.2	Transaktionen	936
12.9	Praxis vs. Theorie – Normalformen, Beziehungen, Joins	937
12.9.1	Das perfekte Datenbankdesign in der Theorie: Normalformen	938
12.9.2	Constraints und die Praxis	946
12.9.3	Arten von Beziehungen zwischen Tabellen	954
12.9.4	Zurück zur Praxis: Beziehungen in phpMyAdmin anlegen	960
12.9.5	Gemeinsame Daten abfragen: Joins	965
12.10	Auch die Datenbank kann's – rekursive Datenspeicherung	972
12.11	Ende der Fahnenstange? Weitere SQL-Befehle	978
13	1984 und Big Brother – Sicherheitsaspekte	985
13.1	Angriffsszenario 1: Cross-Site Scripting (XSS)	985
13.1.1	Key-Tracker	989
13.1.2	CSS-Keylogger	991
13.2	Angriffsszenario 2: SQL-Injections	992
13.2.1	First-Order SQL-Injections	993
13.2.2	Second-Order SQL-Injections	997
13.3	Angriffsszenario 3: Formulardaten entfernter Sites	998
13.4	Angriffsszenario 4: Upload von Dateien	1000
13.5	Sichere Verschlüsselung von Passwörtern – asymmetrische Verschlüsselungsverfahren	1001
13.5.1	Alt, gut, unsicher: Die MD5-Verschlüsselung	1002
13.5.2	Neu, besser, sicher: die Blowfish-Algorithmen (bcrypt)	1004
13.6	Ver- und Entschlüsselung von Daten – symmetrische Verschlüsselungsverfahren	1005
13.6.1	Der Standard: AES-128-CBC	1006

13.6.2	Der bessere Standard: AES-128-GCM	1007
13.7	Damit haben wir nicht gerechnet – Variablentypen und Konvertierungsfunktionen	1008
13.8	Weitere Tipps zur Sicherheit einer Site	1009
Index		1011