

Inhaltsverzeichnis

1	Einleitung	11
	Kurz oder lang?.....	12
2	Plan & Concept	15
	Ein »Quick & Dirty«-Projekt planen und konzipieren	15
2.1	Planung mit dem Kunden	16
2.1.1	Bevor Sie anfangen zu programmieren	16
2.1.2	Während der Entwicklung	20
2.1.3	Nach der Entwicklung.....	25
2.2	Planung für sich selbst.....	28
2.2.1	Bevor Sie anfangen zu programmieren	28
2.2.2	Während der Entwicklung	31
3	Small & light.....	33
	Die Werkzeuge	33
3.1	PHP-Entwicklungsumgebungen.....	34
3.1.1	Die Schweren.....	34
3.1.2	Die Leichten	36
3.1.3	Die Ultraleichten	41
3.2	MySQL-Benutzeroberflächen.....	42
3.3	Sonstige Werkzeuge.....	45
4	Present & Future	51
	Für zukünftige Projekte vorbereitet sein.....	51
4.1	Grundlegende Ordnerstruktur	51
4.2	Helferfunktionen und -klassen	54
4.2.1	Debug-Funktionen.....	55
4.2.2	String-Funktionen	59
4.2.3	Eigene Klassen.....	63
4.2.4	Externe Funktionen	75
4.3	Einzubindende Dateien	77
4.4	Frameworks	85

5	Build & Write	89
5.1	Quick & dirty programmieren	89
5.1.1	Namensgebung	90
5.1.2	Die Datenbank	92
5.1.3	Die Programmierung – die Themen-Administration	95
5.1.3.1	Prinzipielle Struktur der PHP-Datei	97
5.1.3.2	Der HTML-Bereich – ein erstes Formular	99
5.1.3.3	Der PHP-Programmbereich – erste Aktionen	103
5.1.3.4	Einrücken, Auskommentieren und Co	111
5.1.3.5	Optik – das Auge arbeitet mit	116
6	Copy & Paste und Search & Replace	121
6.1	Warum alles neu machen, wenn man doch kopieren kann?	121
6.1.1	Kopieren, einfügen, suchen und ersetzen	122
6.1.2	Arrays und ternäre Operatoren	136
7	Show & Sort	151
7.1	Den Blog zusammensetzen	151
7.1.1	Die Struktur	151
7.1.2	Arrays	160
7.1.3	Arrays sortieren	170
8	Safe & Secure	183
8.1	Sicherheit	183
8.1.1	PHP-Konfiguration	184
8.1.1.1	register_globals = off	186
8.1.1.2	allow_url_fopen = off	189
8.1.1.3	open_basedir	189
8.1.2	XSS (Cross-Site-Scripting)	191
8.1.3	SQL-Injection	194
8.1.4	Captcha	197
8.1.5	Die Macht des md5-Algorithmus	199
8.1.5.1	Passwörter in der Datenbank	199
8.1.5.2	Parameterübergabe	202
8.1.6	Sicherheitskopien	203
8.1.6.1	PHP-Dateien sichern	203
8.1.6.2	Datenbankdaten sichern	204

9 Deploy & Deliver	207
Ausliefern und Aktualisieren.....	207
9.1 Dateien	208
9.2 Datenbank	212
9.2.1 Möglichkeit 1: das große SQL-Skript	214
9.2.2 Möglichkeit 2: das kleine Update-Skript.....	217
10 Seek & Destroy	229
Fehler finden und beheben.....	229
10.1 Prinzipielles Vorgehen bei der Fehlersuche	230
10.2 Werkzeuge zur Fehlersuche.....	231
10.2.1 echo und echo_r, die und die_r.....	233
10.2.2 Das Debug-Panel	239
10.3 Besondere Situationen bei der Fehlersuche	252
10.3.1 Kein Debugging auf der Live-Seite	252
10.3.2 Keine Fehlerausgaben auf dem Livesystem	257
10.3.3 Fehler, die nur beim Nutzer auftreten	260
10.4 Typische PHP-Stolpersteine	262
10.4.1 Der Klassiker: == ist nicht =.....	262
10.4.2 Einige Vergleiche sind gleicher als gleich: == ist nicht gleich ==	264
10.4.3 Ein Leerzeichen ist nicht leer.....	265
10.4.4 Formfehler im Formular	267
11 Expand & Optimize	271
Ein paar Ideen zur Erweiterung und Optimierung Ihres Projekts	271
11.1 Mehrsprachigkeit.....	271
11.1.1 Administration und Datenbank	272
11.1.2 Webseiten-Elemente aus der Datenbank.....	274
11.1.3 Webseite – statische Elemente	278
11.2 Pimp My Website – den Kunden beeindrucken.....	282
11.2.1 TinyMCE	283
11.2.2 phpMailer	286
12 Tipps & Tricks	295
Tipps und Tricks fürs »Quick & dirty«-Arbeiten	295
12.1 PHP-Tipps	295
12.1.1 Doppelteinträge verhindern – header hilft!.....	295

12.1.2	Brich den String	297
12.1.3	Tabellen mit Modulo	302
12.1.4	Einfach oder doppelt, was hält besser?	309
12.2	MySQL-Tipps.....	311
12.2.1	1 = 2 und 1 = 1	311
12.2.2	CREATE TABLE SELECT – die schnellste Sicherheitskopie der Welt	313
12.2.3	Tu's mit Excel: Viele SQL-Strings auf einmal erstellen.....	314
13	Epilog & die goldenen Regeln	319
	Wenn Sie noch einmal wissen wollen, worum es eigentlich ging.....	319
	Stichwortverzeichnis	323