



Auf einen Blick

Über den Autor	7
Einleitung	19
Teil I: Die ersten Schritte	23
Kapitel 1: Die Werkzeuge	25
Kapitel 2: Die erste Angular-Anwendung	49
Kapitel 3: Modularer Aufbau	71
Teil II: Mit dem Nutzer interagieren	93
Kapitel 4: Der Nutzer und was er sieht	95
Kapitel 5: Daten vom Nutzer bekommen	117
Kapitel 6: Nutzereingaben validieren	141
Teil III: Mit dem Server kommunizieren	161
Kapitel 7: Grundlagen der Server-Kommunikation	163
Kapitel 8: Datenaustausch mit einem REST-Server	187
Teil IV: Routing	209
Kapitel 9: Routing: Grundlagen	211
Kapitel 10: Weitere Router-Features	231
Teil V: Die Anwendung testen	251
Kapitel 11: Unit-Tests	253
Kapitel 12: End-to-End-Tests	279
Teil VI: Der Top-Ten-Teil	295
Kapitel 13: Zehn Angular-Bibliotheken	297
Kapitel 14: (Nicht ganz) Zehn Angular-Informationsquellen	303
Anhang A: Einführung in TypeScript	307
Stichwortverzeichnis	321







Inhaltsverzeichnis

Über den Autor	7
Einleitung	19
Konventionen in diesem Buch	19
Was Sie nicht lesen müssen	19
Törichte Annahmen über die Leser	20
Wie dieses Buch aufgebaut ist	20
Teil I: Die ersten Schritte	20
Teil II: Mit dem Nutzer interagieren	20
Teil III: Mit dem Server kommunizieren	20
Teil IV: Routing	20
Teil V: Die Anwendung testen	21
Teil VI: Der Top-Ten-Teil	21
Symbole, die in diesem Buch verwendet werden	21
Wie es weitergeht	21
TEIL I	
DIE ERSTEN SCHRITTE	23
Kapitel 1	
Die Werkzeuge	25
Die Entwicklungsumgebung	25
Node.js und npm	26
Texteditoren	28
Der TypeScript-Compiler	29
Kompilierfehler	30
Compiler-Optionen	31
TSLint, codelyzer und Prettier	33
Das Hauptwerkzeug: Angular CLI	34
Features von Angular CLI	34
Projekt initialisieren	36
Die Verzeichnisstruktur von Angular CLI	36
Die Konfigurationsdatei von Angular CLI	39
Die Anwendung bauen	41
Die Beispielanwendung tritt auf	44
Beispielanwendung: Umgebung einrichten	46
Kapitel 2	
Die erste Angular-Anwendung	49
Die Bausteine einer Angular-Anwendung	49
Angular-Module	50
Angular-Module definieren	51
Der NgModule-Decorator	52





12 Inhaltsverzeichnis

Arten von Angular-Modulen	53
Hauptmodul	53
Feature-Module	54
Routing-Module	54
Service-Module	55
Widget-Module	55
Komponenten	55
Komponenten definieren	55
Der Component-Decorator	57
Das Template der Komponente	57
Die Styles der Komponente	59
Komponenten deklarieren	60
Services – Ein Ort für die Logik	61
Einen Service definieren	61
Einen Service registrieren	62
Einen Service nutzen	63
Dependency Injection und Providers	64
Hierarchische Injektoren	64
Die Hauptdatei der Anwendung	66
Beispielanwendung: Erste Schritte	68
Service implementieren und registrieren	68
Service nutzen	69

Kapitel 3

Modularer Aufbau	71
Der Komponentenbaum	71
Komponenten und ihre Kinder	72
Komponenten in ein Template einfügen	73
Interaktion zwischen Komponenten	74
Der Lebenszyklus einer Komponente	82
Lifecycle-Hooks definieren	83
Konstruktor	83
ngOnChanges	83
ngOnInit	85
ngDoCheck	85
ngAfterContentInit	85
ngAfterContentChecked	85
ngAfterViewInit	86
ngAfterViewChecked	86
ngOnDestroy	86
Dumme und kluge Komponenten	86
Dumme Komponenten	86
Kluge Komponenten	87
Vorteile dieser Aufteilung	87
Beispielanwendung: Neue Komponenten	87





Inhaltsverzeichnis 13

TEIL II MIT DEM NUTZER INTERAGIEREN 93

Kapitel 4

Der Nutzer und was er sieht 95

Daten anzeigen	95
Listen anzeigen	95
Daten vor dem Anzeigen transformieren	100
Auf Ereignisse reagieren	104
Parameter übergeben	105
Pseudo-Ereignisse	107
Das Aussehen verändern	107
Die NgStyle-Direktive	107
Die NgClass-Direktive	108
Teile der Ansicht bedingt anzeigen	110
Die NgIf-Direktive	110
Die NgSwitch-Direktive	111
Beispielanwendung: Daten anzeigen	112
Kontaktlisten anzeigen	112
Kontakte anzeigen	113
Die Ansicht aufräumen	115

Kapitel 5

Daten vom Nutzer bekommen 117

Formulare in Angular	117
Formulararten	118
Voraussetzungen für die Nutzung von Formularen	119
Ein einfaches Formular implementieren	119
Template-gesteuerte Formulare implementieren	120
Textfelder in Template-gesteuerten Formularen	121
Zahlenfelder in Template-gesteuerten Formularen	123
Kontrollkästchen in Template-gesteuerten Formularen	124
Optionsschaltflächen in Template-gesteuerten Formularen	125
Auswahlfelder in Template-gesteuerte Formularen	126
Komplexere Formulare implementieren	127
Modell-gesteuerte Formulare implementieren	128
Eingabefelder in Modell-gesteuerten Formularen	129
Auf Datenänderungen im Formular reagieren	131
Dynamische Formulare	134
Beispielanwendung: Daten bekommen	136
Formular für die Kontaktlisten	137
Das Formular nutzen	138
Das Formular anzeigen	139





14 Inhaltsverzeichnis

Kapitel 6	
Nutzereingaben validieren	141
Grundlagen	141
Validierungsfunktionen	142
Validierungsdirektiven	143
Zustände von Eingabefeldern	144
Zustände von Gruppen	146
Validierung in Template-gesteuerten Formularen	147
Validierungsdirektiven nutzen	147
Den Zustand von Template-gesteuerten Formularen abfragen	148
Validierung in Modell-gesteuerten Formularen	149
Validierungsfunktionen nutzen	149
Den Zustand von Modell-gesteuerten Formularen abfragen	151
Validierungsfehler anzeigen	153
Eigene Validierungsfunktionen	155
Synchrone Validierungsfunktionen	155
Asynchrone Validierungsfunktionen	156
Gruppenvalidierung	158
Beispielanwendung: Eingabe validieren	159

TEIL III

MIT DEM SERVER KOMMUNIZIEREN

Kapitel 7	
Grundlagen der Server-Kommunikation	163
Webserver	163
Reaktive Programmierung mit RxJS	165
Observables	165
Signale von Observables	166
Observables erzeugen	167
Methoden von Observables	168
Operatoren für Observables	171
Fehlerbehandlung	174
Das HTTP-Modul	177
Angular-Module im HTTP-Modul	177
Hilfsklassen im HTTP-Modul	178
Der HttpClient-Service	179
Methoden des HttpClient-Services	179
Optionen des HttpClient-Services	179
Weitere Features des HttpClient-Services	180
Beispielanwendung: API-Server	180
Mit dem API-Server arbeiten	181
API-Beschreibung	182

Inhaltsverzeichnis 15

Kapitel 8	
Datenaustausch mit einem REST-Server	187
Voraussetzungen	187
Daten vom Server anfordern	188
JSON-Daten anfordern	188
Daten in einem anderen Format anfordern	190
Daten an den Server schicken	191
Daten hinzufügen	192
Daten aktualisieren	193
Daten löschen	195
Anfrageparameter	195
Auf die HttpResponseMessage-Instanz zugreifen	196
Eigene Headers definieren	198
Fehlerbehandlung	199
Fehlerarten	200
Fehler in einem Service behandeln	201
Fehler in einer Komponente behandeln	203
Beispielanwendung: Datenaustausch	203
Kontaktlisten anfordern	204
Kontaktlisten hinzufügen	205
Kontaktlisten aktualisieren	206
Kontaktlisten löschen	207
Fehlerbehandlung	208
TEIL IV	
ROUTING	209
Kapitel 9	
Routing: Grundlagen	211
Was ist Routing?	211
Der Angular-Router	212
Das Angular-Router-Paket	212
Routerkonfiguration	214
Routingbaum und Zustand	216
Routing-Module	217
Routing-Modul für ein Feature-Modul	218
Haupt-Routing-Modul	219
Komponenten anzeigen	220
Navigieren mit dem Angular-Router	221
Absolute und relative Pfade	221
Navigation im Template	223
Navigation in der Klasse	224
Parametrisierte Pfade	225
Routing in der Beispielanwendung	227
In Angular-Module aufteilen	227
Routing implementieren	229

16 Inhaltsverzeichnis

Kapitel 10

Weitere Router-Features	231
Verschachtelte Routes	231
Verschachtelte Routes und RouterOutlets	232
Vorteile von verschachtelten Routes	234
Angular-Module nachladen	235
Module zum Nachladen definieren	235
Preloading	236
Bedingte Navigation	237
Guards definieren	238
Guards nutzen	238
canLoad-Guard	239
canActivate-Guard	240
canActivateChild-Guard	240
canDeactivate-Guard	240
Daten mit dem Router übergeben	241
Statische Daten übergeben	241
Dynamische Daten übergeben	243
Navigationsereignisse	245
Überblick über die Navigationsereignisse	245
Auf Navigationsereignisse reagieren	246
Umleitungen	247
Umleitungen definieren	248
Absolute und relative Umleitungen	248
Beispielanwendung: Login	249

TEIL V

DIE ANWENDUNG TESTEN

251

Kapitel 11

Unit-Tests	253
Was sind Unit-Tests?	253
jasmine	254
Testaufbau	254
Erwartungen definieren	256
Test-Spione	257
Asynchrone Tests	259
Unit-Tests ausführen	260
Angular-Test-Module definieren	262
Services testen	264
Bausteine mit Abhängigkeiten testen	266
Tests mit dem HttpClient	268
Komponenten testen	270
Das DOM einer Komponente testen	271
Verschachtelte Komponenten	274
Die Beispielanwendung Unit-testen	275

Inhaltsverzeichnis 17

Kapitel 12	
End-to-End-Tests	279
Was sind End-to-End-Tests?	279
Protractor	281
Die Funktionsweise von Protractor	282
Browserunterstützung	283
Tests mit Protractor schreiben	283
Das browser-Objekt	284
Das by-Objekt	285
Das element-Objekt	286
Beispiel für einen End-to-End-Test	287
Seitenobjekte	289
End-to-End-Tests ausführen	290
Timeouts	292
Die Beispielanwendung testen	293
TEIL VI	
DER TOP-TEN-TEIL	295
Kapitel 13	
Zehn Angular-Bibliotheken	297
Angular Material	297
ng-bootstrap	297
ngx-bootstrap	298
PrimeNG	298
NgRx	298
ag-Grid	299
NGX-Translate	300
Augury	300
Angular Universal	300
Ionic	301
Kapitel 14	
(Nicht ganz) Zehn Angular-Informationsquellen	303
Angular-Dokumentations-Webseite	303
Angular-Blog	304
Angular GitHub-Repository	304
Gitter	304
Stack Overflow	305
Angular in Depth	305
AngularJS.DE	305
Angular University	306
Awesome Angular	306



18 Inhaltsverzeichnis

Anhang A

Einführung in TypeScript.....307

Basistypen.....	308
Boolean.....	308
Number.....	309
String.....	309
Arrays.....	310
null und undefined	310
Any.....	310
Funktionen	311
Optionale und Default-Parameter.....	312
Pfeilfunktionen.....	313
Interfaces.....	313
Interfaces erweitern.....	314
Optionale Eigenschaften/Methoden.....	315
Klassen.....	315
Klassen und Interfaces.....	316
Sichtbarkeit von Eigenschaften und Methoden.....	317
Statische Methoden	318
Generische Typdefinitionen	319

Stichwortverzeichnis.....321

