Contents

Book Revision 1
Prerelease 1
Bug Reports 1
Chat With The Community! 1
Be notified of updates via Twitter 1

Writing your First Angular 2 Web Application 1
Simple Reddit Clone 1
Getting started 4
TypeScript 4
Example Project 4
Angular's Dependencies 6
All Dependencies Loaded 7
Adding CSS 8
Our First TypeScript 8
Making a `Component` 9
Adding a `template` 11
Booting Our Application 12
Loading our Application 12
Running The App 13
Compiling the TypeScript Code to `.js` 13
Using `npm` 14
Serving The App 15
Compiling on every change 17
Adding Data to the Component 17
Working with arrays 19
Expanding our Application 22
The Application Component 24
Adding Interaction 26
Adding the Article Component 29
Rendering Multiple Rows 38
Creating an Article class 38
Storing multiple `Articles` 42
Configuring the `ArticleComponent` with `inputs` 43

Rendering a List of Articles 46
Adding New `Articles` 48
Finishing Touches 49
Displaying the Article Domain 49
Re-sorting Based on Score 50
Full Code Listing 50
Wrapping Up 54
Getting Help 54

TypeScript 55
Angular 2 is built in TypeScript 55
What do we get with TypeScript? 56
Types 57
Trying it out with a REPL 58
Built-in types 59
Classes 61
Properties 61
Methods 62
Constructors 64
Inheritance 65
Utilities 67
Fat Arrow Functions 67
Template Strings 69
Wrapping up 70

How Angular Works 71
Application 71
The Navigation Component 72
The Breadcrumbs Component 72
The Product List Component 73
Product Model 75
Components 76
Component Decorator 78
Component `selector` 78
Component `template` 78
Adding A `Product` 79
Viewing the `Product` with Template Binding 80
Adding More Products 81
Selecting a `Product` 83
Listing products using `<products-list>` 83
The `ProductsList` Component 86
Configuring the `ProductsList` `@Component` Options 87
Component `inputs` 87

- Component `outputs` . . . 90
- Emitting Custom Events . . . 92
- Writing the `ProductsList` Controller Class . . . 94
- Writing the `ProductsList` View Template . . . 95
- The Full `ProductsList` Component . . . 96
- The `ProductRow` Component . . . 98
- `ProductRow` Component Configuration . . . 99
- `ProductRow` Component Definition Class . . . 100
- `ProductRow` `template` . . . 100
- `ProductRow` Full Listing . . . 101
- The `ProductImage` Component . . . 101
- The `PriceDisplay` Component . . . 102
- The `ProductDepartment` Component . . . 103
- The Completed Project . . . 104
- A Word on Data Architecture . . . 105

Built-in Components . . . 107
- Introduction . . . 107
- `NgIf` . . . 107
- `NgSwitch` . . . 107
- `NgStyle` . . . 110
- `NgClass` . . . 112
- `NgFor` . . . 115
- Getting an index . . . 120
- `NgNonBindable` . . . 121
- Conclusion . . . 122

Forms in Angular 2 . . . 123
- Forms are Crucial, Forms are Complex . . . 123
- Controls and Control Groups . . . 123
- `Control` . . . 123
- `ControlGroup` . . . 124
- Our First Form . . . 125
- Simple SKU Form: @Component Annotation . . . 126
- Simple SKU Form: `template` . . . 127
- Simple SKU Form: Component Definition Class . . . 130
- Try it out! . . . 130
- Using `FormBuilder` . . . 132
- Using `FormBuilder` . . . 132
- Using `myForm` in the view . . . 134
- Try it out! . . . 135
- Adding Validations . . . 136
- Explicitly setting the `sku` `Control` as an instance variable . . . 137

Explicitly setting the `sku` `Control` as an instance variable . . . 142
Custom Validations . . . 146
Watching For Changes . . . 148
ngModel . . . 149
Wrapping Up . . . 151

Data Architecture in Angular 2 . . . 152
An Overview of Data Architecture . . . 152
Data Architecture in Angular 2 . . . 153

Data Architecture with Observables - Part 1: Services . . . 154
Observables and RxJS . . . 154
Note: Some RxJS Knowledge Required . . . 154
Learning Reactive Programming and RxJS . . . 154
Chat App Overview . . . 156
Components . . . 157
Models . . . 158
Services . . . 159
Summary . . . 159
Implementing the Models . . . 160
`User` . . . 160
`Thread` . . . 160
`Message` . . . 161
Implementing `UserService` . . . 162
`currentUser` stream . . . 162
Setting a new user . . . 163
`UserService.ts` . . . 164
The `MessagesService` . . . 165
the `newMessages` stream . . . 165
the `messages` stream . . . 167
The Operation Stream Pattern . . . 167
Sharing the Stream . . . 169
Adding `Messages` to the `messages` Stream . . . 170
Our completed `MessagesService` . . . 173
Trying out `MessagesService` . . . 176
The `ThreadsService` . . . 178
A map of the current set of `Threads` (in `threads`) . . . 178
A chronological list of `Threads`, newest-first (in `orderedthreads`) . . . 182
The currently selected `Thread` (in `currentThread`) . . . 183
The list of `Messages` for the currently selected `Thread` (in `currentThreadMessages`) . . . 185
Our Completed `ThreadsService` . . . 187
Data Model Summary . . . 189

Data Architecture with Observables - Part 2: View Components 190
Building Our Views: The ChatApp Top-Level Component 190
The ChatThreads Component 192
ChatThreads Controller 193
ChatThreads template 193
The Single ChatThread Component 195
ChatThread Controller and ngOnInit 196
ChatThread template 197
ChatThread Complete Code 197
The ChatWindow Component 199
The ChatMessage Component 209
Setting incoming 210
The ChatMessage template 211
The Complete ChatMessage Code Listing 212
The ChatNavBar Component 214
The ChatNavBar @Component 214
The ChatNavBar Controller 215
The ChatNavBar template 216
The Completed ChatNavBar 217
Summary 218
Next Steps 220

HTTP 221
Introduction 221
Using angular2/http 222
import from angular2/http 222
A Basic Request 223
Building the SimpleHTTPComponent @Component 223
Building the SimpleHTTPComponent template 223
Building the SimpleHTTPComponent Controller 224
Full SimpleHTTPComponent 226
Writing a YouTubeSearchComponent 227
Writing a SearchResult 229
Writing the YouTubeService 229
Writing the SearchBox 234
Writing SearchResultComponent 240
Writing YouTubeSearchComponent 241
angular/http API 245
Making a POST request 245
PUT / PATCH / DELETE / HEAD 246
RequestOptions 247
Summary 247

Routing **248**
Why routing? 248
How client-side routing works 249
The beginning: using anchor tags 250
The evolution: HTML5 client-side routing 250
Writing our first routes 251
Components of Angular 2 routing 251
RouteConfig 251
RouterOutlet 253
RouterLink 254
Putting it all together 254
Creating the components 256
HomeComponent 256
AboutComponent 256
ContactComponent 257
Application component 257
Routing strategies 260
Path location strategy 262
Running the application 262
Route Parameters 264
`RouteParams` 265
Music Search App 266
First Steps 268
The `SpotifyService` 269
The `SearchComponent` 270
Trying the search 279
`TrackComponent` 281
Wrapping up music search 282
Router Lifecycle Hooks 283
`AuthService` 284
`LoginComponent` 286
`ProtectedComponent` 288
Nested routes 294
Configuring Routes 295
`ProductsComponent` 295
Summary 299

Advanced Components **300**
Styling 300
View (Style) Encapsulation 302
Shadow DOM Encapsulation 306
No Encapsulation 308
Creating a Popup - Referencing and Modifying Host Elements 311

Popup Structure 311
Using `ElementRef` 313
Binding to the `host` 315
Adding a Button using `exportAs` 318
Creating a Message Pane with Transclusion 320
Changing the `host` CSS 321
Using `ng-content` 321
Querying Neighbor Directives - Writing Tabs 323
`Tab` Component 324
`Tabset` Component 325
Using the `Tabset` 326
Lifecycle Hooks 328
`OnInit` and `OnDestroy` 329
`OnChanges` 333
`DoCheck` 339
AfterContentInit, AfterViewInit, AfterContentChecked and AfterViewChecked 352
Advanced Templates 360
Rewriting `ngIf` - `ngBookIf` 361
Rewriting `ngFor` - `ngBookRepeat` 363
Change Detection 369
Customizing Change Detection 373
Zones 380
Observables and OnPush 381
Summary 385

Converting an Angular 1 App to Angular 2 386
Peripheral Concepts 386
What We’re Building 387
Mapping Angular 1 to Angular 2 388
Requirements for Interoperability 390
The Angular 1 App 390
The ng1-app HTML 392
Code Overview 393
ng1: `PinsService` 393
ng1: Configuring Routes 395
ng1: `HomeController` 396
ng1: `/ HomeController` template 396
ng1: `pin` Directive 397
ng1: `pin` Directive template 397
ng1: `AddController` 399
ng1: `AddController` template 401
ng1: Summary 404
Building A Hybrid 404

Hybrid Project Structure 404
Bootstrapping our Hybrid App 409
What We'll Upgrade 410
A Minor Detour: Typing Files 412
Writing ng2 `PinControlsComponent` 418
Using ng2 `PinControlsComponent` 419
Downgrading ng2 `PinControlsComponent` to ng1 421
Adding Pins with ng2 422
Upgrading ng1 `PinsService` and `$state` to ng2 424
Writing ng2 `AddPinComponent` 425
Using `AddPinComponent` 431
Exposing an ng2 service to ng1 431
Writing the `AnalyticsService` 432
Downgrade ng2 `AnalyticsService` to ng1 432
Using `AnalyticsService` in ng1 433
Summary 434
References 434

Testing **435**
Test driven? 435
End-to-end vs. Unit Testing 435
Testing tools 436
Jasmine 436
Karma 437
Writing unit tests 437
Angular2 Unit testing framework 437
Setting Up Testing 438
Testing Services and HTTP 441
HTTP Considerations 442
Stubs 442
Mocks 443
`Http` `MockBackend` 444
`beforeEachProviders` Hooks 444
Testing `getTrack` 445
Testing Components 451
Mocking dependencies 452
Mocking the router classes 455
Testing Forms 464
Testing HTTP requests 476
Conclusion 486

Changelog **487**
Revision 31 - `2016-04-28` 487

Revision 30 - 2016-04-20 487
Revision 29 - 2016-04-08 487
Revision 28 - 2016-04-01 487
Revision 27 - 2016-03-25 487
Revision 26 - 2016-03-24 487
Revision 25 - 2016-03-21 487
Revision 24 - 2016-03-10 487
Revision 23 - 2016-03-04 488
Revision 22 - 2016-02-24 488
Revision 21 - 2016-02-20 488
Revision 20 - 2016-02-11 489
Revision 19 - 2016-02-04 489
Revision 18 - 2016-01-29 489
Revision 17 - 2016-01-28 489
Revision 16 - 2016-01-14 489
Revision 15 - 2016-01-07 490
Revision 14 - 2015-12-23 490
Revision 13 - 2015-12-17 490
Revision 12 - 2015-11-16 491
Revision 11 - 2015-11-09 491
Revision 10 - 2015-10-30 492
Revision 9 - 2015-10-15 492
Revision 8 - 2015-10-08 492
Revision 7 - 2015-09-23 492
Revision 6 - 2015-08-28 493
Revision 5 493
Revision 4 493
Revision 3 493
Revision 2 493
Revision 1 493