

Contents

| | |
|---|----------|
| Book Revision | 1 |
| Bug Reports | 1 |
| Chat With The Community! | 1 |
| Be notified of updates via Twitter | 1 |
| We'd love to hear from you! | 1 |
| Writing your First Angular 2 Web Application | 1 |
| Simple Reddit Clone | 1 |
| Getting started | 4 |
| TypeScript | 4 |
| angular-cli | 4 |
| Example Project | 5 |
| Writing Application Code | 9 |
| Running the application | 9 |
| Making a Component | 10 |
| Importing Dependencies | 12 |
| Component Annotations | 12 |
| Adding a template with templateUrl | 13 |
| Adding a template | 13 |
| Adding CSS Styles with styleUrls | 14 |
| Loading Our Component | 15 |
| Adding Data to the Component | 16 |
| Working With Arrays | 19 |
| Using the User Item Component | 22 |
| Rendering the User ItemComponent | 23 |
| Accepting Inputs | 24 |
| Passing an Input value | 25 |
| Bootstrapping Crash Course | 26 |
| Expanding our Application | 28 |
| Adding CSS | 30 |
| The Application Component | 30 |
| Adding Interaction | 32 |
| Adding the Article Component | 36 |
| Rendering Multiple Rows | 45 |

CONTENTS

| | |
|--|-----------|
| Creating an Article class | 45 |
| Storing Multiple Articles | 49 |
| Configuring the ArticleComponent with inputs | 50 |
| Rendering a List of Articles | 52 |
| Adding New Articles | 54 |
| Finishing Touches | 55 |
| Displaying the Article Domain | 55 |
| Re-sorting Based on Score | 56 |
| Full Code Listing | 57 |
| Wrapping Up | 57 |
| Getting Help | 57 |
| TypeScript | 58 |
| Angular 2 is built in TypeScript | 58 |
| What do we get with TypeScript? | 59 |
| Types | 60 |
| Trying it out with a REPL | 61 |
| Built-in types | 62 |
| Classes | 64 |
| Properties | 64 |
| Methods | 65 |
| Constructors | 67 |
| Inheritance | 68 |
| Utilities | 70 |
| Fat Arrow Functions | 70 |
| Template Strings | 72 |
| Wrapping up | 73 |
| How Angular Works | 74 |
| Application | 74 |
| The Navigation Component | 75 |
| The Breadcrumbs Component | 75 |
| The Product List Component | 76 |
| Product Model | 78 |
| Components | 79 |
| Component Annotation | 81 |
| Component selector | 81 |
| Component template | 82 |
| Adding A Product | 82 |
| Viewing the Product with Template Binding | 84 |
| Adding More Products | 85 |
| Selecting a Product | 86 |
| Listing products using <products-list> | 86 |

CONTENTS

| | |
|---|------------|
| The ProductsList Component | 89 |
| Configuring the ProductsList @Component Options | 90 |
| Component inputs | 90 |
| Component outputs | 93 |
| Emitting Custom Events | 95 |
| Writing the ProductsList Controller Class | 96 |
| Writing the ProductsList View Template | 97 |
| The Full ProductsList Component | 99 |
| The ProductRow Component | 101 |
| ProductRow Component Configuration | 101 |
| ProductRow Component Definition Class | 102 |
| ProductRow template | 103 |
| ProductRow Full Listing | 103 |
| The ProductImage Component | 104 |
| The PriceDisplay Component | 105 |
| The ProductDepartment Component | 105 |
| NgModule and Booting the App | 106 |
| Booting the app | 108 |
| The Completed Project | 108 |
| A Word on Data Architecture | 109 |
| Built-in Directives | 110 |
| Introduction | 110 |
| NgIf | 110 |
| NgSwitch | 110 |
| NgStyle | 113 |
| NgClass | 115 |
| NgFor | 118 |
| Getting an index | 123 |
| NgNonBindable | 124 |
| Conclusion | 125 |
| Forms in Angular 2 | 126 |
| Forms are Crucial, Forms are Complex | 126 |
| FormControls and FormGroups | 126 |
| FormControl | 126 |
| FormGroup | 127 |
| Our First Form | 128 |
| Loading the FormsModule | 129 |
| Simple SKU Form: @Component Annotation | 130 |
| Simple SKU Form: template | 130 |
| Simple SKU Form: Component Definition Class | 134 |
| Try it out! | 134 |

CONTENTS

| | |
|--|------------|
| Using FormBuilder | 136 |
| Reactive Forms with FormBuilder | 137 |
| Using FormBuilder | 137 |
| Using myForm in the view | 138 |
| Try it out! | 139 |
| Adding Validations | 140 |
| Explicitly setting the sku FormControl as an instance variable | 142 |
| Custom Validations | 147 |
| Watching For Changes | 148 |
| ngModel | 150 |
| Wrapping Up | 152 |
| HTTP | 153 |
| Introduction | 153 |
| Using @angular/http | 154 |
| import from @angular/http | 154 |
| A Basic Request | 155 |
| Building the SimpleHTTPComponent @Component | 156 |
| Building the SimpleHTTPComponent template | 156 |
| Building the SimpleHTTPComponent Controller | 157 |
| Full SimpleHTTPComponent | 159 |
| Writing a YouTubeSearchComponent | 160 |
| Writing a SearchResult | 162 |
| Writing the YouTubeService | 162 |
| Writing the SearchBox | 171 |
| Writing SearchResultComponent | 178 |
| Writing YouTubeSearchComponent | 179 |
| @angular/http API | 183 |
| Making a POST request | 183 |
| PUT / PATCH / DELETE / HEAD | 183 |
| RequestOptions | 184 |
| Summary | 185 |
| Routing | 186 |
| Why Do We Need Routing? | 186 |
| How client-side routing works | 187 |
| The beginning: using anchor tags | 188 |
| The evolution: HTML5 client-side routing | 188 |
| Writing our first routes | 189 |
| Components of Angular 2 routing | 189 |
| Imports | 189 |
| Routes | 190 |
| Installing our Routes | 191 |

CONTENTS

| | |
|--|------------|
| RouterOutlet using <router-outlet> | 192 |
| RouterLink using [routerLink] | 193 |
| Putting it all together | 194 |
| Creating the Components | 196 |
| HomeComponent | 196 |
| AboutComponent | 196 |
| ContactComponent | 196 |
| Application Component | 197 |
| Configuring the Routes | 199 |
| Routing Strategies | 200 |
| Path location strategy | 201 |
| Running the application | 202 |
| Route Parameters | 204 |
| ActivatedRoute | 205 |
| Music Search App | 206 |
| First Steps | 208 |
| The SpotifyService | 209 |
| The SearchComponent | 210 |
| Trying the search | 220 |
| TrackComponent | 222 |
| Wrapping up music search | 224 |
| Router Hooks | 224 |
| AuthService | 225 |
| LoginComponent | 226 |
| ProtectedComponent and Route Guards | 228 |
| Nested Routes | 234 |
| Configuring Routes | 235 |
| ProductsComponent | 235 |
| Summary | 240 |
| Dependency Injection | 241 |
| Injections Example: PriceService | 242 |
| “Don’t Call Us...” | 244 |
| Dependency Injection Parts | 246 |
| Playing with an Injector | 247 |
| Providing Dependencies with NgModule | 249 |
| Providers | 250 |
| Using a Class | 250 |
| Using a Factory | 251 |
| Using a Value | 253 |
| Using an alias | 253 |
| Dependency Injection in Apps | 253 |
| Working with Injectors | 255 |

CONTENTS

| | |
|---|------------|
| Substituting values | 262 |
| NgModule | 266 |
| NgModule vs. JavaScript Modules | 266 |
| The Compiler and Components | 267 |
| Dependency Injection and Providers | 267 |
| Component Visibility | 268 |
| Specifying Providers | 270 |
| Conclusion | 271 |
| Data Architecture in Angular 2 | 272 |
| An Overview of Data Architecture | 272 |
| Data Architecture in Angular 2 | 273 |
| Data Architecture with Observables - Part 1: Services | 274 |
| Observables and RxJS | 274 |
| Note: Some RxJS Knowledge Required | 274 |
| Learning Reactive Programming and RxJS | 274 |
| Chat App Overview | 276 |
| Components | 277 |
| Models | 278 |
| Services | 279 |
| Summary | 279 |
| Implementing the Models | 280 |
| User | 280 |
| Thread | 280 |
| Message | 281 |
| Implementing UserService | 281 |
| currentUser stream | 282 |
| Setting a new user | 283 |
| UserService.ts | 284 |
| The MessagesService | 285 |
| the newMessages stream | 285 |
| the messages stream | 287 |
| The Operation Stream Pattern | 287 |
| Sharing the Stream | 289 |
| Adding Messages to the messages Stream | 290 |
| Our completed MessagesService | 293 |
| Trying out MessagesService | 296 |
| The ThreadsService | 298 |
| A map of the current set of Threads (in threads) | 298 |
| A chronological list of Threads, newest-first (in orderedthreads) | 303 |
| The currently selected Thread (in currentThread) | 303 |
| The list of Messages for the currently selected Thread (in currentThreadMessages) | 305 |

CONTENTS

| | |
|---|------------|
| Our Completed ThreadsService | 308 |
| Data Model Summary | 310 |
| Data Architecture with Observables - Part 2: View Components | 311 |
| Building Our Views: The ChatApp Top-Level Component | 311 |
| The ChatThreads Component | 314 |
| ChatThreads Controller | 314 |
| ChatThreads template | 315 |
| The Single ChatThread Component | 316 |
| ChatThread Controller and ngOnInit | 317 |
| ChatThread template | 318 |
| ChatThread Complete Code | 319 |
| The ChatWindow Component | 320 |
| The ChatMessage Component | 330 |
| Setting incoming | 331 |
| The ChatMessage template | 332 |
| The Complete ChatMessage Code Listing | 333 |
| The ChatNavBar Component | 338 |
| The ChatNavBar @Component | 338 |
| The ChatNavBar Controller | 338 |
| The ChatNavBar template | 340 |
| The Completed ChatNavBar | 340 |
| Summary | 342 |
| Next Steps | 343 |
| Introduction to Redux with TypeScript | 344 |
| Redux | 345 |
| Redux: Key Ideas | 345 |
| Core Redux Ideas | 346 |
| What's a <i>reducer</i> ? | 346 |
| Defining Action and Reducer Interfaces | 347 |
| Creating Our First Reducer | 348 |
| Running Our First Reducer | 348 |
| Adjusting the Counter With <i>actions</i> | 349 |
| Reducer switch | 350 |
| Action "Arguments" | 352 |
| Storing Our State | 353 |
| Using the Store | 354 |
| Being Notified with subscribe | 354 |
| The Core of Redux | 358 |
| A Messaging App | 359 |
| Messaging App state | 359 |
| Messaging App actions | 360 |

CONTENTS

| | |
|---|------------|
| Messaging App reducer | 361 |
| Trying Out Our Actions | 364 |
| Action Creators | 365 |
| Using Real Redux | 367 |
| Using Redux in Angular | 369 |
| Planning Our App | 370 |
| Setting Up Redux | 370 |
| Defining the Application State | 371 |
| Defining the Reducers | 371 |
| Defining Action Creators | 372 |
| Creating the Store | 372 |
| CounterApp Component | 374 |
| Providing the Store | 375 |
| Bootstrapping the App | 376 |
| The CounterComponent | 377 |
| imports | 377 |
| The template | 377 |
| The constructor | 379 |
| Putting It All Together | 380 |
| What's Next | 382 |
| References | 383 |
| Intermediate Redux in Angular | 384 |
| Context For This Chapter | 384 |
| Chat App Overview | 385 |
| Components | 386 |
| Models | 386 |
| Reducers | 387 |
| Summary | 387 |
| Implementing the Models | 388 |
| User | 388 |
| Thread | 388 |
| Message | 389 |
| App State | 389 |
| A Word on Code Layout | 390 |
| The Root Reducer | 390 |
| The UsersState | 391 |
| The ThreadsState | 391 |
| Visualizing Our AppState | 393 |
| Building the Reducers (and Action Creators) | 394 |
| Set Current User Action Creators | 394 |
| UsersReducer - Set Current User | 395 |
| Thread and Messages Overview | 396 |

CONTENTS

| | |
|--|------------|
| Adding a New Thread Action Creators | 396 |
| Adding a New Thread Reducer | 397 |
| Adding New Messages Action Creators | 398 |
| Adding A New Message Reducer | 399 |
| Selecting A Thread Action Creators | 401 |
| Selecting A Thread Reducer | 402 |
| Reducers Summary | 403 |
| Building the Angular Chat App | 403 |
| The top-level ChatApp | 405 |
| The ChatPage | 406 |
| Container vs. Presentational Components | 407 |
| Building the ChatNavBar | 408 |
| Redux Selectors | 410 |
| Threads Selectors | 411 |
| Unread Messages Count Selector | 412 |
| Building the ChatThreads Component | 413 |
| ChatThreads Controller | 414 |
| ChatThreads template | 415 |
| The Single ChatThread Component | 416 |
| ChatThread @Component and template | 417 |
| Building the ChatWindow Component | 418 |
| The ChatMessage Component | 426 |
| Setting incoming | 426 |
| The ChatMessage template | 427 |
| Summary | 428 |
| Advanced Components | 430 |
| Styling | 430 |
| View (Style) Encapsulation | 432 |
| Shadow DOM Encapsulation | 436 |
| No Encapsulation | 438 |
| Creating a Popup - Referencing and Modifying Host Elements | 441 |
| Popup Structure | 441 |
| Using ElementRef | 443 |
| Binding to the host | 445 |
| Adding a Button using exportAs | 448 |
| Creating a Message Pane with Content Projection | 450 |
| Changing the host CSS | 451 |
| Using ng-content | 451 |
| Querying Neighbor Directives - Writing Tabs | 453 |
| Tab Component | 454 |
| Tabset Component | 454 |
| Using the Tabset | 456 |

CONTENTS

| | |
|---|------------|
| Lifecycle Hooks | 458 |
| OnInit and OnDestroy | 459 |
| OnChanges | 463 |
| DoCheck | 469 |
| AfterContentInit, AfterViewInit, AfterContentChecked and AfterViewChecked | 481 |
| Advanced Templates | 489 |
| Rewriting ngIf - ngBookIf | 490 |
| Rewriting ngFor - ngBookRepeat | 492 |
| Change Detection | 498 |
| Customizing Change Detection | 502 |
| Zones | 509 |
| Observables and OnPush | 510 |
| Summary | 514 |
| Testing | 515 |
| Test driven? | 515 |
| End-to-end vs. Unit Testing | 515 |
| Testing Tools | 516 |
| Jasmine | 516 |
| Karma | 517 |
| Writing Unit Tests | 517 |
| Angular Unit testing framework | 517 |
| Setting Up Testing | 518 |
| Testing Services and HTTP | 521 |
| HTTP Considerations | 522 |
| Stubs | 522 |
| Mocks | 523 |
| Http MockBackend | 524 |
| TestBed.configureTestingModule and Providers | 524 |
| Testing getTrack | 525 |
| Testing Routing to Components | 531 |
| Creating a Router for Testing | 532 |
| Mocking dependencies | 535 |
| Spies | 535 |
| Back to Testing Code | 538 |
| fakeAsync and advance | 540 |
| inject | 541 |
| Testing ArtistComponent's Initialization | 541 |
| Testing ArtistComponent Methods | 543 |
| Testing ArtistComponent DOM Template Values | 544 |
| Testing Forms | 547 |
| Creating a ConsoleSpy | 549 |
| Installing the ConsoleSpy | 550 |

CONTENTS

| | |
|---|------------|
| Configuring the Testing Module | 551 |
| Testing The Form | 551 |
| Refactoring Our Form Test | 554 |
| Testing HTTP requests | 557 |
| Testing a POST | 558 |
| Testing DELETE | 559 |
| Testing HTTP Headers | 560 |
| Testing YouTubeService | 562 |
| Conclusion | 568 |
| Converting an Angular 1 App to Angular 2 | 569 |
| Peripheral Concepts | 569 |
| What We're Building | 570 |
| Mapping Angular 1 to Angular 2 | 571 |
| Requirements for Interoperability | 573 |
| The Angular 1 App | 573 |
| The ng1-app HTML | 575 |
| Code Overview | 576 |
| ng1: PinsService | 576 |
| ng1: Configuring Routes | 578 |
| ng1: HomeController | 579 |
| ng1: / HomeController template | 579 |
| ng1: pin Directive | 580 |
| ng1: pin Directive template | 580 |
| ng1: AddController | 582 |
| ng1: AddController template | 584 |
| ng1: Summary | 587 |
| Building A Hybrid | 587 |
| Hybrid Project Structure | 587 |
| Bootstrapping our Hybrid App | 592 |
| What We'll Upgrade | 593 |
| A Minor Detour: Typing Files | 596 |
| Writing ng2 PinControlsComponent | 602 |
| Using ng2 PinControlsComponent | 603 |
| Downgrading ng2 PinControlsComponent to ng1 | 604 |
| Adding Pins with ng2 | 606 |
| Upgrading ng1 PinsService and \$state to ng2 | 608 |
| Writing ng2 AddPinComponent | 609 |
| Using AddPinComponent | 615 |
| Exposing an ng2 service to ng1 | 615 |
| Writing the AnalyticsService | 616 |
| Downgrade ng2 AnalyticsService to ng1 | 616 |
| Using AnalyticsService in ng1 | 617 |

CONTENTS

| | |
|--|------------|
| Summary | 618 |
| References | 619 |
| NativeScript: Mobile Applications for the Angular Developer | 620 |
| What is NativeScript? | 620 |
| Where NativeScript Differs from Other Popular Frameworks | 621 |
| What are the System and Development Requirements for NativeScript? | 622 |
| Creating your First Mobile Application with NativeScript and Angular | 624 |
| Adding Build Platforms for Cross Platform Deployment | 624 |
| Building and Testing for Android and iOS | 624 |
| Installing JavaScript, Android, and iOS Plugins and Packages | 625 |
| Understanding the Web to NativeScript UI and UX Differences | 626 |
| Planning the NativeScript Page Layout | 626 |
| Adding UI Components to the Page | 628 |
| Styling Components with CSS | 629 |
| Developing a Geolocation Based Photo Application | 630 |
| Creating a Fresh NativeScript Project | 631 |
| Creating a Multiple Page Master-Detail Interface | 632 |
| Creating a Flickr Service for Obtaining Photos and Data | 636 |
| Creating a Service for Calculating Device Location and Distance | 641 |
| Including Mapbox Functionality in the NativeScript Application | 645 |
| Implementing the First Page of the Geolocation Application | 646 |
| Implementing the Second Page of the Geolocation Application | 652 |
| Try it out! | 653 |
| NativeScript for Angular Developers | 654 |
| Changelog | 655 |
| Revision 49 - 2017-01-18 | 655 |
| Revision 48 - 2017-01-13 | 655 |
| Revision 47 - 2017-01-06 | 655 |
| Revision 46 - 2017-01-03 | 655 |
| Revision 45 - 2016-12-05 | 655 |
| Revision 44 - 2016-11-17 | 655 |
| Revision 43 - 2016-11-08 | 655 |
| Revision 42 - 2016-10-14 | 656 |
| Revision 41 - 2016-09-28 | 656 |
| Revision 40 - 2016-09-20 | 656 |
| Revision 39 - 2016-09-03 | 656 |
| Revision 38 - 2016-08-29 | 656 |
| Revision 37 - 2016-08-02 | 657 |
| Revision 36 - 2016-07-20 | 657 |
| Revision 35 - 2016-06-30 | 657 |
| Revision 34 - 2016-06-15 | 657 |

CONTENTS

| | |
|------------------------------------|-----|
| Revision 33 - 2016-05-11 | 657 |
| Revision 32 - 2016-05-06 | 657 |
| Revision 31 - 2016-04-28 | 658 |
| Revision 30 - 2016-04-20 | 658 |
| Revision 29 - 2016-04-08 | 658 |
| Revision 28 - 2016-04-01 | 658 |
| Revision 27 - 2016-03-25 | 659 |
| Revision 26 - 2016-03-24 | 659 |
| Revision 25 - 2016-03-21 | 659 |
| Revision 24 - 2016-03-10 | 659 |
| Revision 23 - 2016-03-04 | 659 |
| Revision 22 - 2016-02-24 | 660 |
| Revision 21 - 2016-02-20 | 660 |
| Revision 20 - 2016-02-11 | 660 |
| Revision 19 - 2016-02-04 | 660 |
| Revision 18 - 2016-01-29 | 660 |
| Revision 17 - 2016-01-28 | 660 |
| Revision 16 - 2016-01-14 | 661 |
| Revision 15 - 2016-01-07 | 661 |
| Revision 14 - 2015-12-23 | 661 |
| Revision 13 - 2015-12-17 | 662 |
| Revision 12 - 2015-11-16 | 662 |
| Revision 11 - 2015-11-09 | 662 |
| Revision 10 - 2015-10-30 | 663 |
| Revision 9 - 2015-10-15 | 663 |
| Revision 8 - 2015-10-08 | 664 |
| Revision 7 - 2015-09-23 | 664 |
| Revision 6 - 2015-08-28 | 664 |
| Revision 5 | 664 |
| Revision 4 | 664 |
| Revision 3 | 665 |
| Revision 2 | 665 |
| Revision 1 | 665 |