# Complete Angular Developer Course Guide

For more courses, resources and workshop, visit https://zerotomastery.io



# 2. Bootstrapping Angular

**The Angular CLI**
- Angular CLI - https://angular.io/cli

**Everything Else**
- section-3-16-COMPLETE.zip

# 3. Optional: TypeScript

**Installing TypeScript**
- TypeScript - https://www.typescriptlang.org/

**Variables**
- Primitive Types - https://www.typescriptlang.org/docs/handbook/2/everyday-types.html#the-primitives-string-number-and-boolean

**Why Decorators?**
- TC39 Proposals - https://github.com/tc39/proposals

**TypeScript Configuration**
- TSConfig - https://www.typescriptlang.org/tsconfig

**Writing Decorators**
- Section-2-13-COMPLETE.zip

# 4. Component Fundamentals

**Property Binding**
- Dog Image - https://picsum.photos/id/237/500/500

**Type Assertions**
- Lorem Picsum - https://picsum.photos/images
- EventTarget Interface - https://microsoft.github.io/PowerBI-JavaScript/interfaces/_node_modules_typedoc_node_modules_typescript_lib_lib_dom_d_.eventtarget.html

**Input Aliases**
- Style Guide - https://angular.io/guide/styleguide#avoid-aliasing-inputs-and-outputs

**Emitting Events**
- Prefix Guideline - https://angular.io/guide/styleguide#dont-prefix-output-properties

**Discovering Lifecycle Hooks**
- Lifecycle Hooks - https://angular.io/guide/lifecycle-hooks

**Scoped CSS**
- section-4-12-COMPLETE.zip

# 5. Transforming Content

**Pipes Basic**
- Pipes - https://angular.io/api?type=pipe

**Angular Dev Tools**
- Angular Devtools - https://angular.io/guide/devtools

**Pipe Parameters**
- Custom Format Options - https://angular.io/api/common/DatePipe#custom-format-options

**Dealing with Numbers**
- Currency Codes - https://en.wikipedia.org/wiki/ISO_4217

**Debugging with Pipes**
- JSON Pipe - https://angular.io/api/common/JsonPipe

**Understanding Directives**

- Directives - https://angular.io/api?type=directive

**The ngIf Directive**
- Shorthand Examples - https://angular.io/guide/structural-directives#shorthand-examples

**The ngFor Directive**
- section-5-12-COMPLETE.zip

# 6. Master Project: Component Design

**Introduction to Master Project**
- HTML Template.zip

**What is Tailwind?**
- TailwindCSS - https://tailwindcss.com/
- Purge CSS - https://purgecss.com/

**Configuring Tailwind**
- Tailwind Configuration - https://tailwindcss.com/docs/configuration

**Loading Static Assets**
- Autoplay Policy Changes - https://developers.google.com/web/updates/2017/09/autoplay-policy-changes

**Creating a User Module**
- Common Module - https://angular.io/api/common/CommonModule

**Preventing the Default Behavior**
- section-6-34-COMPLETE.zip

# 7. Master Project: Reactive Forms

**Form Validation**
- Validation Functions - https://angular.io/api/forms/Validators

**Form Controller Status**
- AbstractControl Class - https://angular.io/api/forms/AbstractControl

**Validating Emails**
- Email Validator - https://angular.io/api/forms/Validators#email

**Validating Numbers**
- Validators Min - https://angular.io/api/forms/Validators#min

**Input Masking**
- NGX Mask - https://www.npmjs.com/package/ngx-mask

**Disabling Buttons**
- FormGroup Class - https://angular.io/api/forms/FormGroup
- Tailwind States -
  https://tailwindcss.com/docs/hover-focus-and-other-states

**Designing an Alert Component**
- section-7-19-COMPLETE.zip

# 8. Master Project: Template Forms

**Registering a New Form**
- NgForm - https://angular.io/api/forms/NgForm

**Two-way Binding**
- NgModel - https://angular.io/api/forms/NgModel

**Attribute Validation**
- Constraint Validation -
  https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Constraint_validation

# 9. Master Project: Intro to RXJS

**Introduction to RXJS**
- section-9-1-STARTER.zip

**Unsubscribing from Observables**
- Subscription Object - https://rxjs.dev/guide/subscription

**Declarative Programming with Operators**
- RXJS Reference - https://rxjs.dev/api

**The of and from Operators**
- JSON Placeholder - https://jsonplaceholder.typicode.com/

**Marble Diagram**
- Map Operator - https://rxjs.dev/api/operators/map

**Understanding Flattening Operators**
- JSON Placeholder URL -
  https://jsonplaceholder.typicode.com/todos/1

**Handling Errors**
- section-9-23-COMPLETE.zip

# 10. Master Project: Authentication

**Setting up Firebase**
- Firebase - https://firebase.google.com/

**Reviewing the Rules**
- Firebase Rules - https://firebase.google.com/docs/rules

**Installing AngularFire**
- Firebase Web Setup - https://firebase.google.com/docs/web/setup
- AngularFire - https://firebaseopensource.com/projects/angular/angularfire2/

**User Registration**
- AngularFire Authentication - https://github.com/angular/angularfire/blob/master/docs/auth/getting-started.md

**Handling the Response**
- Authentication SDK - https://firebase.google.com/docs/reference/js/firebase.auth.Auth

**Connecting the User with their Data**
- Create Account Function - https://firebase.google.com/docs/reference/js/auth#createuserwithemailandpassword

**Database Rules**
- Firebase Request Auth - https://firebase.google.com/docs/reference/rules/rules.firestore.Request#auth

**The Async Pipe**
- Async Pipe - https://angular.io/api/common/AsyncPipe

**Initializing Firebase First**
- Firebase Installation - https://firebase.google.com/docs/web/setup

**Setting up the Login**

- signInWithEmailAndPassword - https://firebase.google.com/docs/reference/js/firebase.auth.Auth.html#signinwithemailandpassword

**Sidebar: JSON Web Tokens**
- JWT - https://jwt.io/
- Base64- https://www.base64decode.org/
- section-10-23-COMPLETE.zip

# 11. Master Project: Custom Validators

**Validating Unique Emails**
- AsyncValidator - https://angular.io/api/forms/AsyncValidator
- fetchSignInMethodsForEmail - https://firebase.google.com/docs/reference/js/auth#fetchsigninmethodsforemail

**Finalizing the Validator**
- section-11-8-COMPLETE.zip

# 12. Master Project: Routing

**Understanding Routing**
- Angular Router - https://angular.io/api/router

**Registering Routes**
- Route - https://angular.io/api/router/Route

**Active Links**
- RouterLinkActive Directive - https://angular.io/api/router/RouterLinkActive

**Forcing Redirection**
- Router - https://angular.io/api/router/Router

**Learning more about Query Parameters**
- Navigate Function - https://angular.io/api/router/Router#navigate

# 13. Master Project: Uploading Files

**Handling Files**

- HTML5 Video - https://en.wikipedia.org/wiki/HTML5_video#Browser_support

**Uploading Files with Firebase**
- UUID - https://www.npmjs.com/package/uuid

**Firebase Rules and Validation**
- Firebase Secure Files - https://firebase.google.com/docs/storage/security/secure-files

**Handling Errors and Successful Uploads**
- Error Code - https://firebase.google.com/docs/storage/web/handle-errors

**Fallback Upload**
- Drag and Drop Support - https://caniuse.com/dragndrop

**Redirection after Upload**
- Document Reference - https://firebase.google.com/docs/reference/node/firebase.firestore.DocumentReference

**Deleting a Clip from the Storage/Database**
- Delete Files - https://firebase.google.com/docs/storage/web/delete-files
- Storage Security Rules - https://firebase.google.com/docs/firestore/security/rules-structure#granular_operations

**Composite Indexes**
- section-13-30-COMPLETE.zip

# 14. Master Project: WebAssembly & Rust

**What is WebAssembly**

- Figma and WebAssembly -

  https://www.figma.com/blog/webassembly-cut-figmas-load-time-by-3x/

- WebAssembly Languages -

  https://github.com/appcypher/awesome-wasm-langs

**Getting Started with Rust**

- Rust Installation Page- https://www.rust-lang.org/tools/install

- Rustup Book - https://rust-lang.github.io/rustup/

- Crates - https://crates.io/

**Exploring the Starter Project**

- Cargo Manifest File -

  https://doc.rust-lang.org/cargo/reference/manifest.html

**Data Types**

- Rust Data Types -

  https://www.codingame.com/playgrounds/365/getting-started-with-rust/primitive-data-types

**Control Flow**

- Operators and Symbols -

  https://doc.rust-lang.org/book/appendix-02-operators.html

**Vectors**

- section-14-10-COMPLETE.zip

**Results**

- Error Handling -

  https://doc.rust-lang.org/book/ch09-00-error-handling.html

- section-14-13-COMPLETE.zip

**Installing Webpack**

- Webpack - https://webpack.js.org/

**Configuring Webpack**

- rust-image-starter.zip

**Reading Files**

- File Reader Methods -
  https://developer.mozilla.org/en-US/docs/Web/API/FileReader#methods

**Compiling Rust with Webpack**

- WASM Pack - https://rustwasm.github.io/wasm-pack/

**Importing WebAssembly**

- wasm-bindgen Crate - https://crates.io/crates/wasm-bindgen

**Logging Files**

- web-sys Crate - https://crates.io/crates/web-sys

**Base64 Decoding**

- Base64 Crate - https://crates.io/crates/base64

**Loading an Image from Memory**

- Image Crate - https://crates.io/crates/image

**Grayscaling an Image**

- load_from_memory() Function -
  https://docs.rs/image/0.19.0/image/fn.load_from_memory.html

**App Deployment**

- Vercel - https://vercel.com/


# 15. Master Project: Processing Video with WebAssembly

**Understanding FFmpeg**
- FFmpeg - https://www.ffmpeg.org/
- FFmpeg WASM - https://github.com/ffmpegwasm/ffmpeg.wasm

**Custom Asset Paths**
- Node Glob - https://github.com/isaacs/node-glob

**Initializing FFmpeg**
- Material Icons - https://fonts.google.com/icons

- Tailwind Animations - [https://tailwindcss.com/docs/animation](https://tailwindcss.com/docs/animation)

**Generating a Screenshot**
- FFmpeg - [https://www.ffmpeg.org/](https://www.ffmpeg.org/)

**Deleting Screenshots**
- section-15-18-COMPLETE.zip

# 16. Master Project: Playing Videos

**Fixing Cross Origin Issues**
- Gsutil - [https://cloud.google.com/storage/docs/gsutil_install](https://cloud.google.com/storage/docs/gsutil_install)

**Installing Videojs**
- Videojs - [https://videojs.com/](https://videojs.com/)

**Copying Links to the Clipboard**
- Clipboard API - [https://caniuse.com/clipboard](https://caniuse.com/clipboard)
- execCommand() - [https://caniuse.com/mdn-api_document_execcommand](https://caniuse.com/mdn-api_document_execcommand)

**Lazy Loading Modules**
- section-16-19-COMPLETE.zip

# 17. Master Project: Deployment

**Production Budgets**
- Configuring Budgets - [https://angular.io/guide/build#configuring-size-budgets](https://angular.io/guide/build#configuring-size-budgets)

**Deploying an App with Vercel**
- Vercel - [https://vercel.com](https://vercel.com)
- Project Configuration - [https://vercel.com/docs/cli#project-configuration](https://vercel.com/docs/cli#project-configuration)
- GitHub Desktop - [https://desktop.github.com/](https://desktop.github.com/)
- section-17-3-COMPLETE.zip

*For more courses, resources and workshop, visit [https://zerotomastery.io](https://zerotomastery.io)*