



Kurt Wiersma

Jumpstart Your App With Angular CLI

Agenda



- ▶ Getting Started
- ▶ Generate
- ▶ Develop & Debug
- ▶ Build
- ▶ Test
- ▶ Customize

What is Angular CLI?



- ▶ An easy and effective way to setup your Angular app's build, test, and development configuration
- ▶ Derived from ember-cli
- ▶ Uses the Angular Style Guide so you don't have to type out any boilerplate code

Getting Started



- ▶ Install node and npm
- ▶ Install Angular CLI
 - ▶ `$ npm install -g @angular/cli`

Generate

Generate an App



- ▶ To generate a new app:
- ▶ `$ ng new my-new-app --skip-install`
- ▶ `--style (String)` (Default: `css`) The style file default extension. Possible values: `css`, `scss`, `less`, `sass`, `styl` (stylus). You can later change the value in `".angular-cli.json"`
- ▶ `--routing (Boolean)` (Default: `false`) Generate a routing module.

Generate Components



- ▶ Generate a component you can control which module it gets imported into.
 - ▶ `$ ng generate component my-module-a/my-subcomponent-a`
- ▶ `--dry-run` Run through without making any changes.

Generate Classes



- ▶ Module
- ▶ Service
- ▶ Class
- ▶ Directive
- ▶ Enum
- ▶ Guard
- ▶ Interface

Develop & Debug



- ▶ `$ ng serve -open`
- ▶ Watches source path and rebuilds automatically
- ▶ Live reload enabled by default
- ▶ `--aot` Build using Ahead of Time compilation

Proxy API Requests



proxy.conf.json

```
1  {
2    "/api/*": {
3      "target": "http://localhost:8000",
4      "secure": false,
5      "logLevel": "debug"
6    },
7    "/api2/*": {
8      "target": "http://localhost:8000",
9      "secure": false,
10     "logLevel": "debug"
11   }
12 }
```

.angular-cli.json

```
1  {
2    "$schema": "./node_modules/@angular/cli/lib/config/schema.json",
3    "project": {"name": "djleague-ng2"...},
4    "apps": [...],
5    "e2e": {...},
6    "lint": [...],
7    "test": {...},
8    "defaults": {
9      "styleExt": "css",
10     "component": {},
11     "serve": {
12       "proxyConfig": "proxy.conf.json"
13     }
14   }
15 }
```



Debugging



- ▶ Source maps are enabled by default in ng serve

- ▶ Uses Codelyzer and the Angular Style Guide
- ▶ `$ ng lint`
- ▶ Validity of the directive and component selectors according to the style guide.
- ▶ Best practices related to the declaration of the metadata of our projects.
- ▶ Proper implementation of life-cycle hooks.
- ▶ Proper component and directive naming.
- ▶ Detection of dead CSS.

Build

- ▶ Webpack powered!
- ▶ `$ ng build -prod`
- ▶ `--aot` Build using Ahead of Time compilation
- ▶ Vendor chunk default = true
- ▶ You can “eject” your webpack config

Test

Unit Testing



- ▶ Unit test runner (karma)
- ▶ `$ ng test --watch`
- ▶ `--code-coverage` Coverage report will be in the `coverage/` directory.

End-to-End Testing



- ▶ End-to-end test runner (protractor)
- ▶ `$ ng e2e`
- ▶ `--element-explorer` (Default: false) Start Protractor's Element Explorer for debugging.

Customize & Upgrade

Customize



- ▶ .angular-cli.json
- ▶ [Angular CLI config JSON schema](#)
- ▶ ng eject

Upgrading



- ▶ When angular-cli releases a new version I use the new version to generate a new empty project.
- ▶ Then I compare the new empty project with my project using a tool like FileMerge or BeyondCompare
- ▶ Pay particular attention to config files and package.json
- ▶ You can usually upgrade angular to the latest version although I tend to stick with what the cli generates

Resources



- ▶ [Angular CLI Ref Card](#)
- ▶ [Angular Timesheet](#)
- ▶ [Angular Real World Example App](#)
- ▶ [John Papa's Event-View CLI Example App](#)
- ▶ [Angular Style Guide](#)
- ▶ [Creating modules and components with Angular CLI](#)
- ▶ [Why Angular CLI Doesn't Expose Webpack](#)
- ▶ [Angular Universal and Angular CLI](#)



- ▶ Slides & Code:

- ▶ <https://github.com/kwiersma/djleague-ng2>