

Recommended by you, Éverton Roberto Auler, and 19 others



Éverton Roberto Auler

Developer/Entrepreneur

Mar 27

Angular 4 universal app with @angular/cli

Based on Rob Wormald's [Demo Repository](#), I decided to build it using @angular/cli.

Start the project

```
npm install -g @angular/cli@latest
ng new --style=scss cli-universal-demo
cd cli-universal-demo
npm install -D ts-node
npm install -S @angular/platform-server
@angular/animations
```

Update src/app/app.module.ts

```
1  import { BrowserModule } from '@angular/platform-browser';
2  import { NgModule } from '@angular/core';
3  import { FormsModule } from '@angular/forms';
4  import { HttpClientModule } from '@angular/http';
5
6  import { AppComponent } from './app.component';
7
8  @NgModule({
9    declarations: [
10     AppComponent
11   ],
12   imports: [
13     BrowserModule.withServerTransition({appId: 'cli-univers
14     FormsModule
```

Create src/app/app.server.module.ts

```
1  import { NgModule } from '@angular/core';
2  import { ServerModule } from '@angular/platform-server';
3  import { AppModule } from './app.module';
4  import { AppComponent } from './app.component';
5
6  @NgModule({
7    imports: [
8      ServerModule,
9      AppModule
```

Update src/tsconfig.app.json

```
1  {
2    "extends": "../tsconfig.json",
3    "compilerOptions": {
4      "outDir": "../out-tsc/app",
5      "module": "es2015",
6      "baseUrl": "",
7      "types": []
8    },
9    "exclude": [
10     "server.ts",
```

Update tsconfig.json

```
1  {
2    "compileOnSave": false,
3    "compilerOptions": {
4      "outDir": "./dist/out-tsc",
5      "baseUrl": "src",
6      "sourceMap": true,
7      "declaration": false,
8      "moduleResolution": "node",
9      "emitDecoratorMetadata": true,
10     "experimentalDecorators": true,
11     "target": "es5",
12     "typeRoots": [
13       "node_modules/@types"
14     ],
15     "lib": [
16       "es2016",
```

Create a express.js server: src/server.ts

```

1  import 'reflect-metadata';
2  import 'zone.js/dist/zone-node';
3  import { platformServer, renderModuleFactory } from '@angular/platform-server';
4  import { enableProdMode } from '@angular/core';
5  import { AppServerModuleNgFactory } from '../dist/ngfactory';
6  import * as express from 'express';
7  import { readFileSync } from 'fs';
8  import { join } from 'path';
9
10 const PORT = 4000;
11
12 enableProdMode();
13
14 const app = express();
15
16 let template = readFileSync(join(__dirname, '..', 'dist', 'index.html'), 'utf-8');
17
18 app.engine('html', (_, options, callback) => {
19     const opts = { document: template, url: options.req.url };
20
21     renderModuleFactory(AppServerModuleNgFactory, opts)
22         .then(html => callback(null, html));
23 });

```

Update scripts on package.json

```

1  {
2    "name": "cli-universal-demo",
3    "version": "0.0.0",
4    "license": "MIT",
5    "scripts": {
6      "prestart": "ng build --prod && ngc",
7      "start": "ts-node src/server.ts"
8    },
9    "private": true,
10   "dependencies": {
11     "@angular/animations": "^4.0.0",
12     "@angular/common": "^4.0.0",
13     "@angular/compiler": "^4.0.0",
14     "@angular/core": "^4.0.0",
15     "@angular/forms": "^4.0.0",
16     "@angular/http": "^4.0.0",
17     "@angular/platform-browser": "^4.0.0",
18     "@angular/platform-browser-dynamic": "^4.0.0",
19     "@angular/platform-server": "^4.0.0",
20     "@angular/router": "^4.0.0",
21     "core-js": "^2.4.1",
22     "rxjs": "^5.1.0",
23     "zone.js": "^0.8.4"
24   },
25   "devDependencies": {
26     "@angular/cli": "1.0.0",
27     "@angular/compiler-cli": "^4.0.0",
28     "@types/jasmine": "2.5.38",
29     "@types/node": "^6.0.0"

```

Try it out

```
npm run start
```

