# angular / quickstart

### Angular QuickStart - source from the documentation

165 commits	<b>₽ 4</b> branches	<b>25</b> releases	<b>23</b> cc	ontributors		<b>ناِ</b> ه MIT
Branch: master ▼ New pull request			Create new file	Upload files	Find File	Clone or download ▼
Foxandxss chore: separate deprecation not from old content  Latest commit abf8486						bf8486 on Oct 31, 2017
e2e	chore: add src folder	(#362)				2 years ago
src	chore: fix systemjs-a	ngular-loader.js for external tem	plates (#444) (#	•••		2 years ago
editorconfig .editorconfig	chore(editorconfig):	remove end_of_line setting (#26	57)			3 years ago
:gitignore	chore: add modulelo	d removal loader (#410)				2 years ago
:travis.yml	chore: add src folder	(#362)				2 years ago
CHANGELOG.md	fix(systemjs-angular-	-loader): replace w/ version that	works in IE			2 years ago
LICENSE	chore: correct license	е				3 years ago
README.md	chore: separate depr	recation not from old content				2 years ago
bs-config.e2e.json	chore: add src folder	(#362)				2 years ago
bs-config.json	chore: add src folder	(#362)				2 years ago
karma-test-shim.js	chore: add src folder	(#362)				2 years ago
karma.conf.js	chore: add src folder	(#362)				2 years ago
non-essential-files.osx.txt	chore: add bs-config	g.e2e.json to non-essential files				2 years ago
non-essential-files.txt	chore: add bs-config	g.e2e.json to non-essential files				2 years ago

package.json	chore: update angular version from 4.0.0 to 4.3.4 (#488)	2 years ago
protractor.config.js	chore: add src folder (#362)	2 years ago
tslint.json	chore: extend with testing support (preliminary)	3 years ago

README.md

# Angular QuickStart Source

build passing

This repository is now deprecated. The Angular Quickstart project was a nice starting point for creating Angular applications. Now we recommend using the Angular CLI to create new Angular projects.

Starting from 1 November 2017, all the Angular documentation, at angular.io, is based on the Angular CLI.

Let's get started

This repository holds the TypeScript source code of the angular.io quickstart, the foundation for most of the documentation samples and potentially a good starting point for your application.

It's been extended with testing support so you can start writing tests immediately.

This is not the perfect arrangement for your application. It is not designed for production. It exists primarily to get you started quickly with learning and prototyping in Angular

We are unlikely to accept suggestions about how to grow this QuickStart into something it is not. Please keep that in mind before posting issues and PRs.

# Updating to a newer version of the Quickstart Repo

From time to time the QuickStart will be enhanced with support for new features or to reflect changes to the official Style Guide.

You can update your existing project to an up-to-date QuickStart by following these instructions:

- Create a new project using the instructions below
- Copy the code you have in your project's main.ts file onto src/app/main.ts in the new project
- Copy your old app folder into src/app
- Delete src/app/main.ts if you have one (we now use src/main.ts instead)
- Copy your old index.html, styles.css and tsconfig.json into src/
- Install all your third party dependencies
- Copy your old e2e/ folder into e2e/
- Copy over any other files you added to your project
- Copy your old .git folder into your new project's root

Now you can continue working on the new project.

### **Prerequisites**

Node.js and npm are essential to Angular development.

Get it now if it's not already installed on your machine.

**Verify that you are running at least node v4.x.x and npm 3.x.x** by running node -v and npm -v in a terminal/console window. Older versions produce errors.

We recommend nvm for managing multiple versions of node and npm.

# Create a new project based on the QuickStart

Clone this repo into new project folder (e.g., my-proj ).

```
git clone https://github.com/angular/quickstart my-proj
cd my-proj
```

We have no intention of updating the source on angular/quickstart. Discard the .git folder..

```
rm -rf .git # OS/X (bash)
rd .git /S/Q # windows
```

### Delete non-essential files (optional)

You can quickly delete the *non-essential* files that concern testing and QuickStart repository maintenance (*including all git-related artifacts* such as the .git folder and .gitignore!) by entering the following commands while in the project folder:

OS/X (bash)

```
xargs rm -rf < non-essential-files.osx.txt
rm src/app/*.spec*.ts
rm non-essential-files.osx.txt</pre>
```

#### Windows

```
for /f %i in (non-essential-files.txt) do del %i /F /S /Q
rd .git /s /q
rd e2e /s /q
```

### Create a new git repo

You could start writing code now and throw it all away when you're done. If you'd rather preserve your work under source control, consider taking the following steps.

Initialize this project as a *local git repo* and make the first commit:

```
git init
git add .
git commit -m "Initial commit"
```

Recover the deleted .gitignore from the QuickStart repository if you lost it in the *Delete non-essential files* step.

Create a remote repository for this project on the service of your choice.

Grab its address (e.g. https://github.com/<my-org>/my-proj.git ) and push the local repo to the remote.

```
git remote add origin <repo-address>
git push -u origin master
```

# Install npm packages

See npm and nvm version notes above

Install the npm packages described in the package.json and verify that it works:

```
npm install
npm start
```

Doesn't work in Bash for Windows which does not support servers as of January, 2017.

The npm start command first compiles the application, then simultaneously re-compiles and runs the lite-server. Both the compiler and the server watch for file changes.

Shut it down manually with Ctrl-C.

You're ready to write your application.

### npm scripts

We've captured many of the most useful commands in npm scripts defined in the package.json:

- npm start runs the compiler and a server at the same time, both in "watch mode".
- npm run build runs the TypeScript compiler once.
- npm run build:w runs the TypeScript compiler in watch mode; the process keeps running, awaiting changes to TypeScript files and re-compiling when it sees them.
- npm run serve runs the lite-server, a light-weight, static file server, written and maintained by John Papa and Christopher Martin with excellent support for Angular apps that use routing.

Here are the test related scripts:

- npm test compiles, runs and watches the karma unit tests
- npm run e2e compiles and run protractor e2e tests, written in Typescript (\*e2e-spec.ts)

### **Testing**

The QuickStart documentation doesn't discuss testing. This repo adds both karma/jasmine unit test and protractor end-to-end testing support.

These tools are configured for specific conventions described below.

It is unwise and rarely possible to run the application, the unit tests, and the e2e tests at the same time. We recommend that you shut down one before starting another.

### **Unit Tests**

TypeScript unit-tests are usually in the src/app folder. Their filenames must end in .spec.ts.

Look for the example src/app/app.component.spec.ts. Add more .spec.ts files as you wish; we configured karma to find them.

Run it with npm test

That command first compiles the application, then simultaneously re-compiles and runs the karma test-runner. Both the compiler and the karma watch for (different) file changes.

Shut it down manually with Ctrl-C.

Test-runner output appears in the terminal window. We can update our app and our tests in real-time, keeping a weather eye on the console for broken tests. Karma is occasionally confused and it is often necessary to shut down its browser or even shut the command down (Ctrl-C) and restart it. No worries; it's pretty quick.

### End-to-end (E2E) Tests

E2E tests are in the e2e directory, side by side with the src folder. Their filenames must end in .e2e-spec.ts.

Look for the example e2e/app.e2e-spec.ts . Add more .e2e-spec.js files as you wish (although one usually suffices for small projects); we configured Protractor to find them.

Thereafter, run them with npm run e2e.

That command first compiles, then simultaneously starts the lite-server at localhost:8080 and launches Protractor.

The pass/fail test results appear at the bottom of the terminal window. A custom reporter (see protractor.config.js) generates a ./ test-output/protractor-results.txt file which is easier to read; this file is excluded from source control.

Shut it down manually with ctrl-c.