Running Unit and End to End Tests

John Papa

PRINCIPAL DEVELOPER ADVOCATE

@john_papa www.johnpapa.net

Testing

ANGULAR CLI

Execute unit tests

Watch for file changes

Execute end to end tests

Explore the various options

Testing Angular

Unit Tests

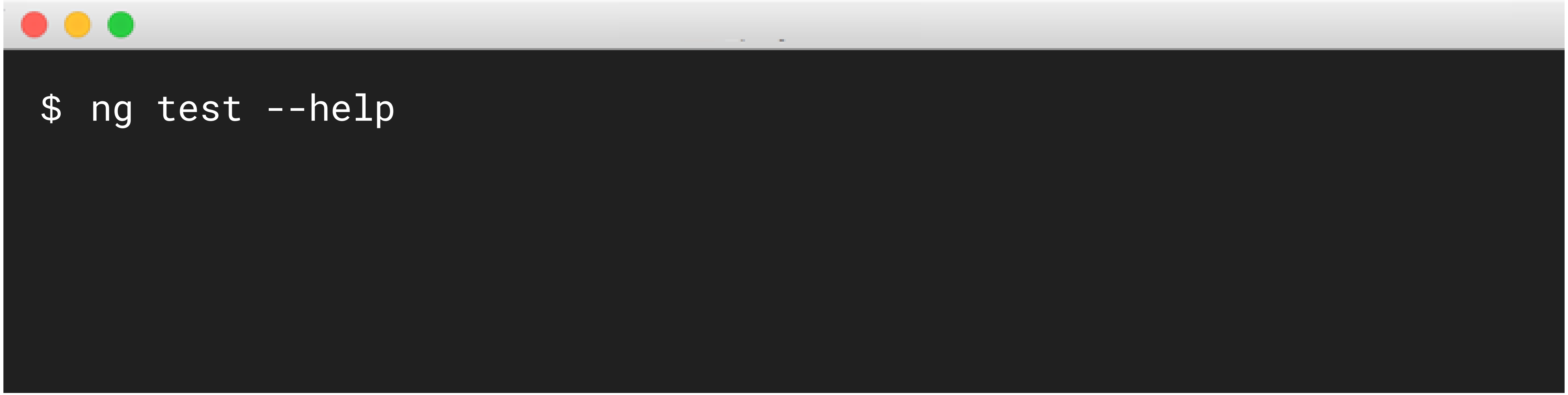
Test discrete functionality

Isolate the feature under test

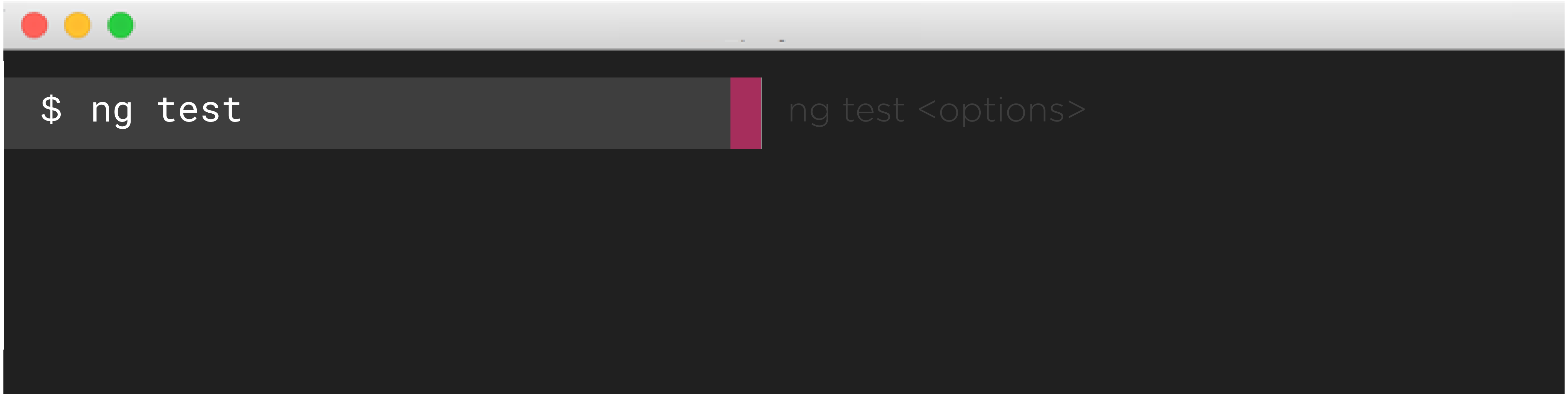
Valuable part of the dev and CI lifecycles

Getting Help

Add the --help or -h flag



```
$ ng test --help
```



```
$ ng test
```

ng test <options>

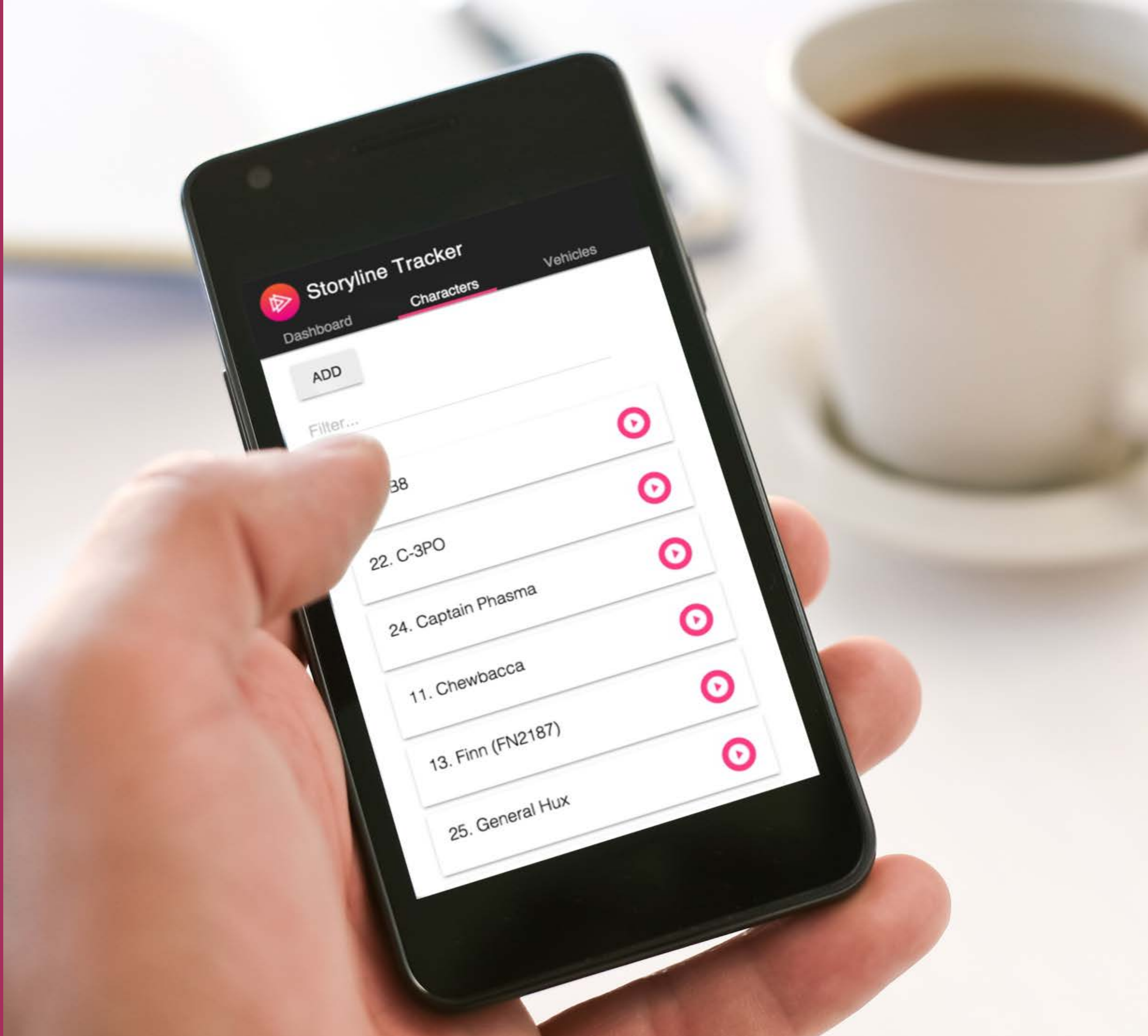
Test During Development

Execute tests on all *.spec.ts files

Watch for file changes, then test again

Run in its own terminal process

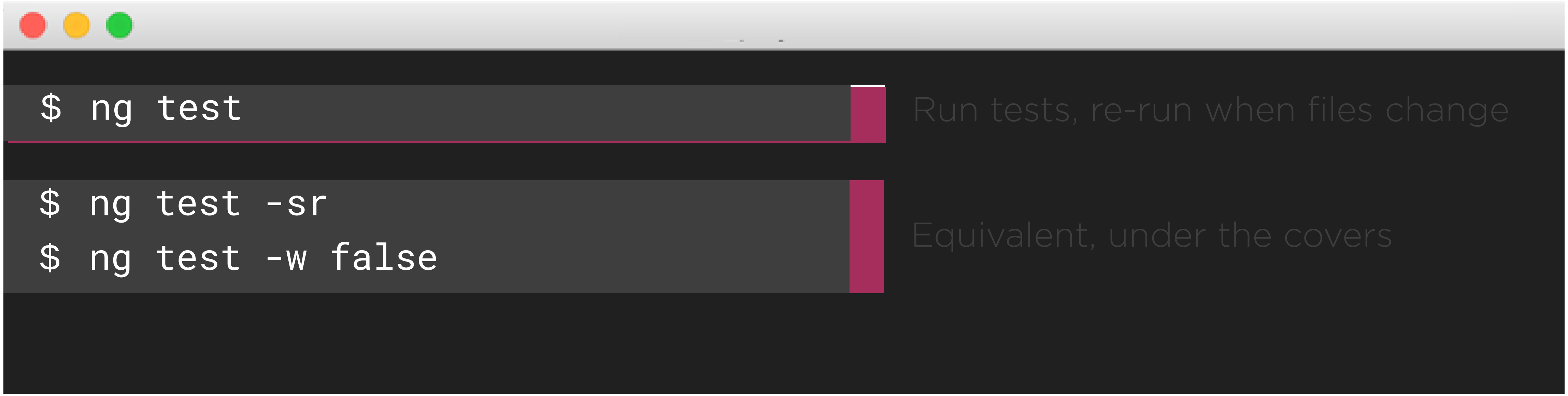
Demo


Storyline Tracker
Dashboard
Characters
Vehicles
ADD
Filter...
38
22. C-3PO
24. Captain Phasma
11. Chewbacca
13. Finn (FN2187)
25. General Hux

Testing Options

Common ng test Options

Options	Alias	Description
--code-coverage	-cc	Generate code coverage report (default false)
--colors		Enable/disable output colors (default true)
--single-run	-sr	Execute tests but no watching (default false)
--progress		Log progress to console (default true)
--sourcemaps	-sm	Generate sourcemaps (default true)
--watch	-w	Run test once, and watch (default true)



```
$ ng test
```

Run tests, re-run when files change

```
$ ng test -sr
$ ng test -w false
```

Equivalent, under the covers

Continuous Testing vs Single Test Cycle

Tests execute after a build is executed via Karma

Automatically watch your files for changes, by default with ng test

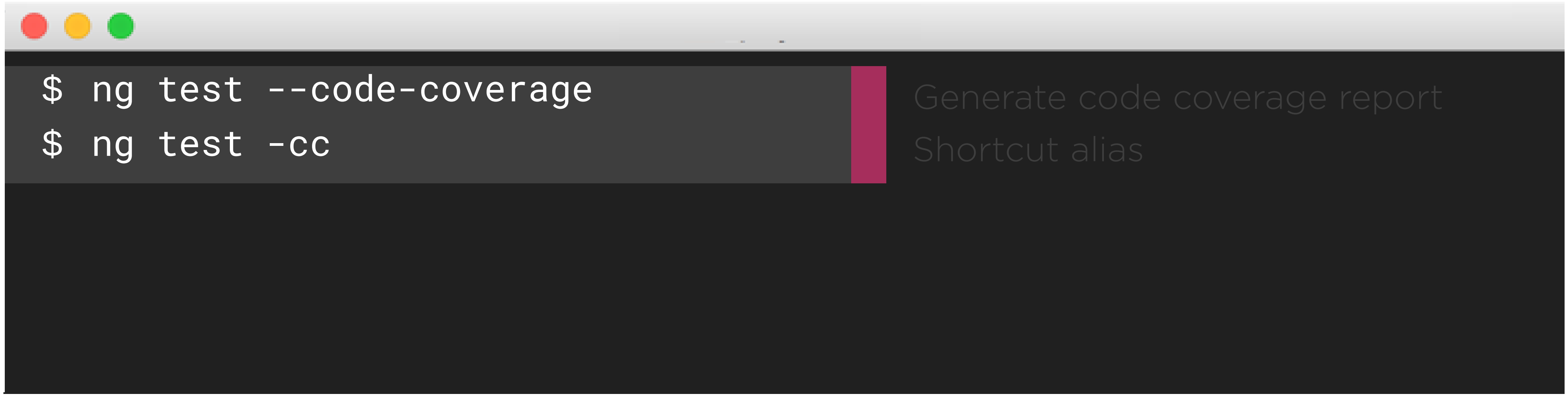
Run a test a single time with ng test -sr or ng test –w false

Test Code Coverage

Calculate and report test code coverage with ng test --code-coverage

Report is generated in the /coverage folder

Report folder is configurable in .angular-cli.json

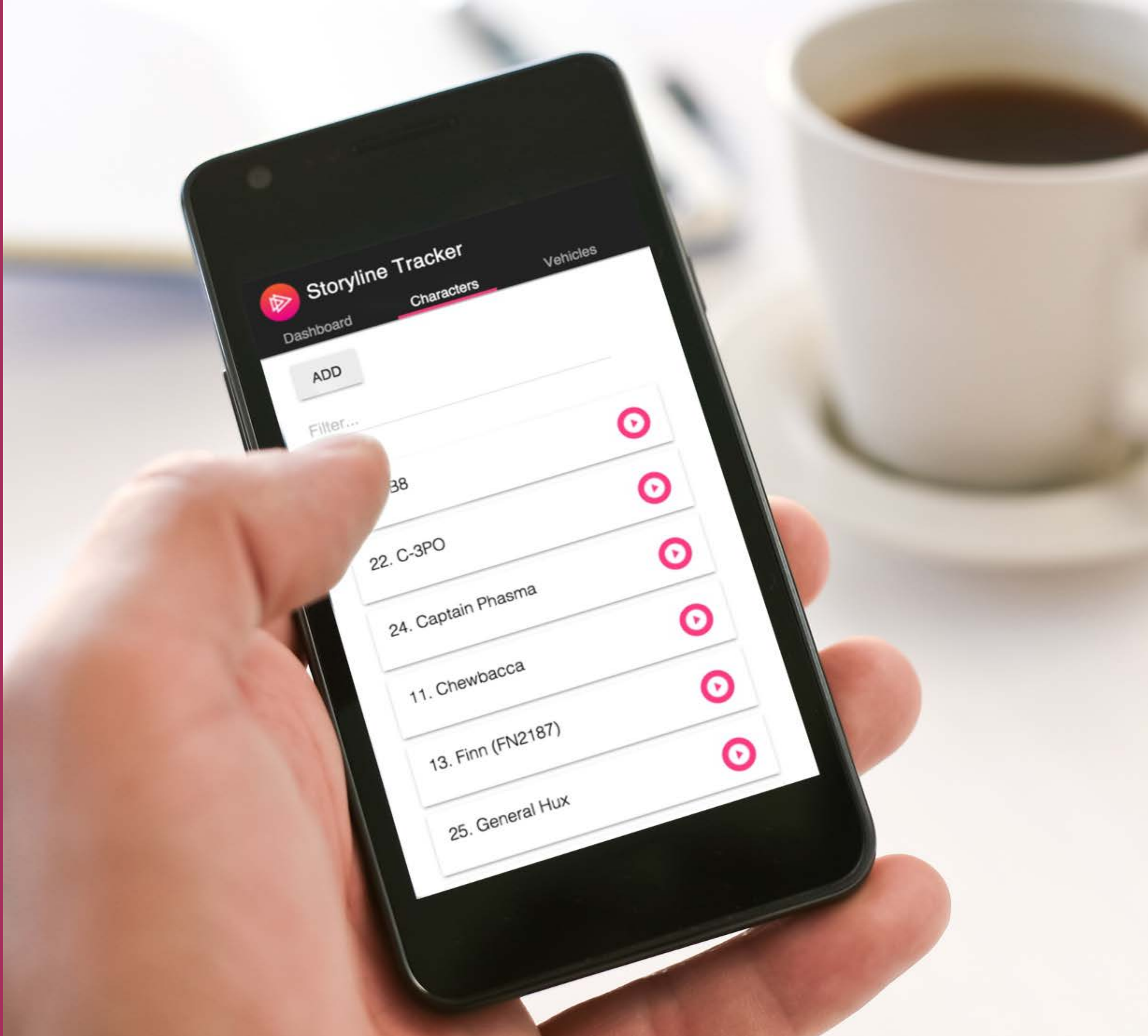


```
$ ng test --code-coverage
$ ng test -cc
```

Generate code coverage report

Shortcut alias

Demo



B8

22. C-3PO

24. Captain Phasma

11. Chewbacca

13. Finn (FN2187)

25. General Hux

End to End Tests

End to End Tests

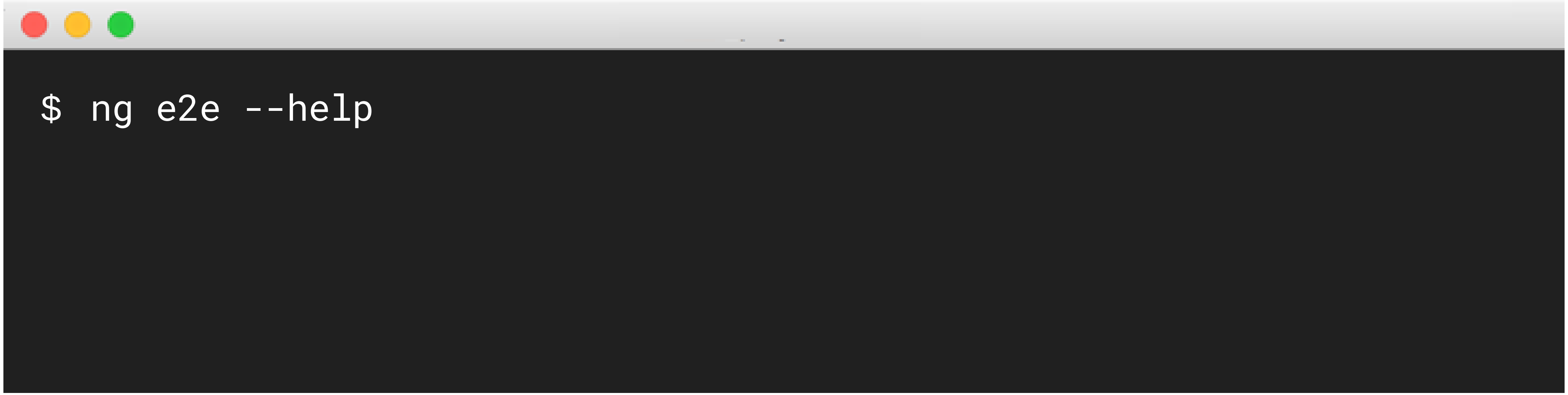
Test app behavior through user interaction

Great way to prove out use cases

Use protractor with Angular

Getting Help

Add the --help or -h flag



```
$ ng e2e --help
```

Common ng e2e Options

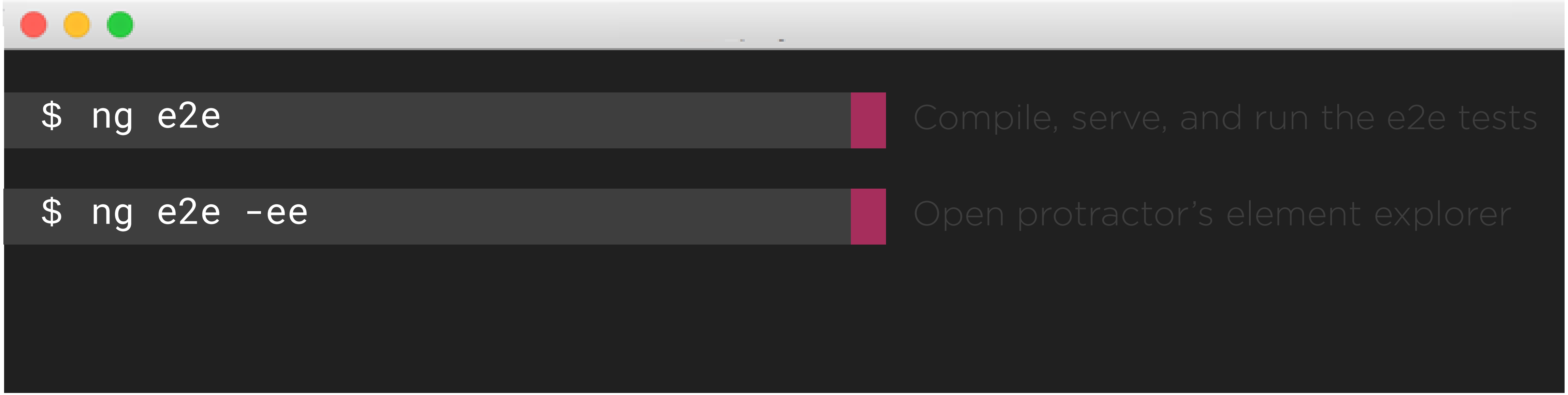
Most serve options are available, plus the following …

Options	Alias	Description
--config	-c	Point to a config file. Defaults to protractor.conf.js
--element-explorer	-ee	Opens protractor’s element explorer
--serve	-s	Compile and serve on a random port (default true)
--specs	-sp	Override specs to test (defaults to all)
--webdriver-update	-wu	Try to update webdriver (default true)

Common ng e2e Options

By default it compiles, serves on a random port, and runs tests

Add the --element-explorer or -ee flag to debug



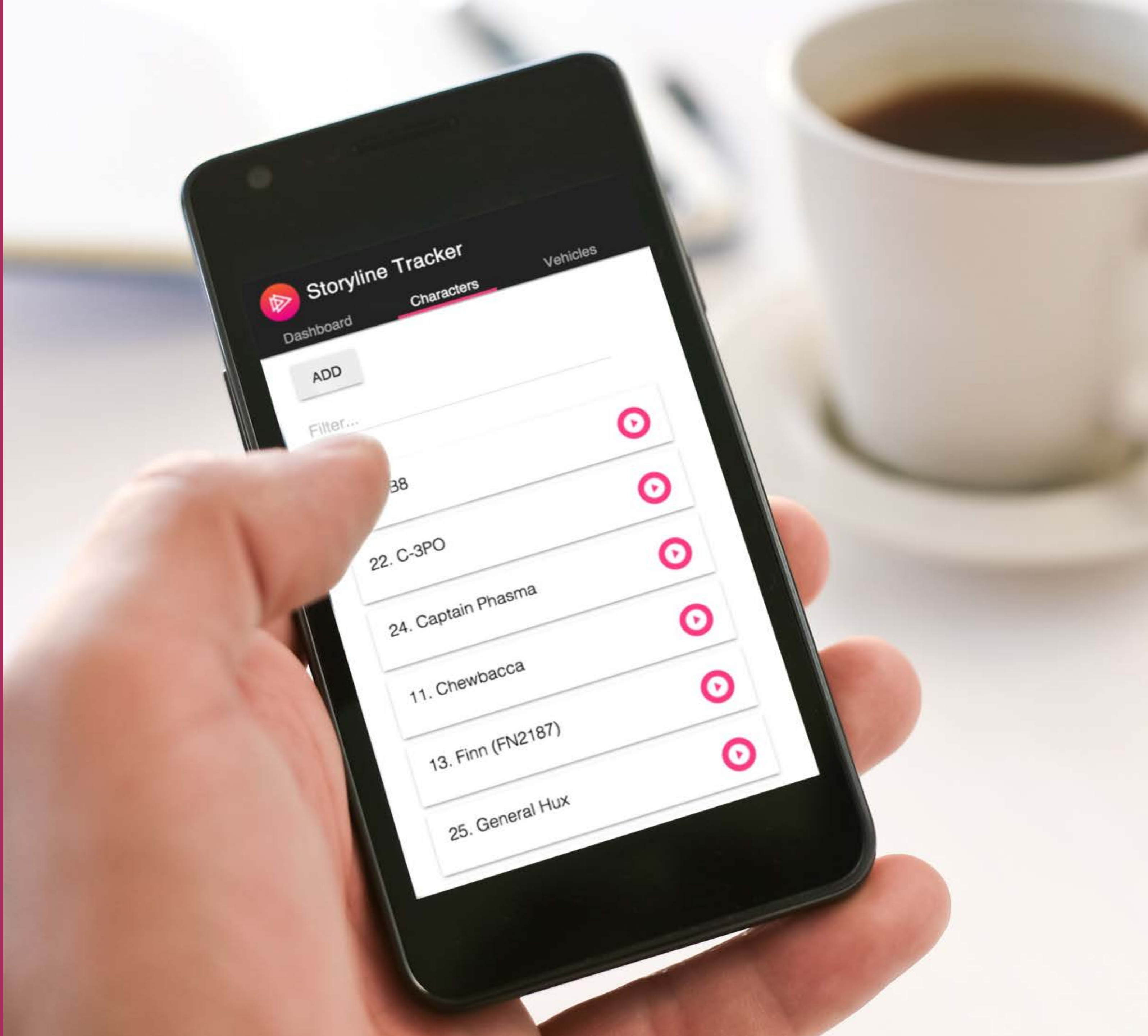
```
$ ng e2e
```

Compile, serve, and run the e2e tests

```
$ ng e2e -ee
```

Open protractor’s element explorer

Demo



Storyline Tracker

Running and Debugging Tests

ANGULAR CLI

Execute unit tests

Debug tests in the browser

Execute end to end tests

Debug end to end tests with protractor

Review

ANGULAR CLI

```
ng new customer-app
ng g c customer
ng g s sales-data
ng lint
ng build
ng serve
ng test
ng e2e
```