# How to measure code coverage for Angular JS unit test

**TypeScript**

App.component.ts

App.component.spec.ts

**JavaScript**

App.component.js

App.component.spec.js

**tsc compile**

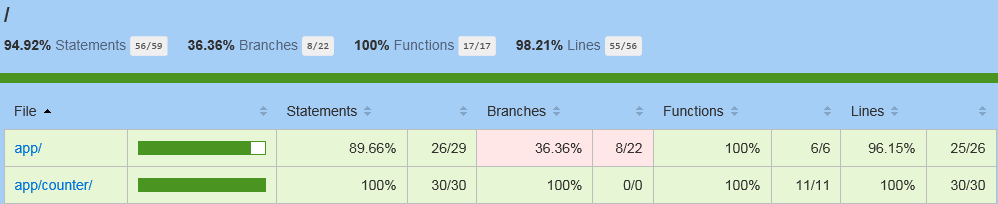
Jasmine Framework

**Karma**

Karma Chrome Launcher

Karma Coverage

Karma Jasmine



## Test Runner

[Karma](https://karma-runner.github.io/1.0/index.html) is a test runner that recommended by the AngularJS team which aims to bring developers a productive testing environment.

[**Note: The recommended approach is to install Karma (and all the plugins) locally in the project's directory.**](http://karma-runner.github.io/1.0/intro/installation.html)

The NPM dependencies for Karma are listed below:

|  |  |
| --- | --- |
| Name | Descriptions |
| Karma | Karma NPM package |
| Karma-jasmine | Adapter between Jasmine-Core & Karma |
| Karma-coverage | Measures the code coverage information about the executed JavaScript files |
| Karma-chrome-launcher | Adapter between Chrome browser & Karma |

## Test Framework

[Jasmine](http://jasmine.github.io/2.4/introduction.html) is a behavior-driven development framework for testing JavaScript code. It is recommended in the AngularJS official site and does not depend on any other JavaScript frameworks.

The NPM dependencies for Jasmine are listed below:

|  |  |
| --- | --- |
| Name | Descriptions |
| jasmine-core | Jasmine NPM package |
| @types/jasmine | Typescript type definition for Jasmine |

## Environment Setup

1. The highlighted NPM modules in package.json below are the necessary dependencies for unit testing.
2. To trigger the UT, we’ve defined 2 tasks test & test-once. The difference between “test” & “test-once” is that “test” will monitor the UT & source code so if someone modified the source code or unit test cases karma will rerun all the unit test cases.

**Package.json**

|  |
| --- |
| {  "name": "angular2-testing",  "version": "1.0.0",  "description": "",  "main": "index.js",  "scripts": {  "start": "tsc --p tsconfig.test.json && concurrently \"tsc -w\" \"lite-server\" ",  "test": "tsc --p tsconfig.test.json && concurrently \"tsc -w\" \"karma start karma.conf.js\"",  "test-once": "tsc --p tsconfig.test.json && karma start karma.conf.js --single-run",  "tsc": "tsc --p tsconfig.test.json"  },  "devDependencies": {  "@types/jasmine": "^2.5.38",  "jasmine-core": "^2.5.2",  "karma": "^1.3.0",  "karma-chrome-launcher": "^2.0.0",  "karma-coverage": "^1.1.1",  "karma-jasmine": "^1.1.0",  "karma-jasmine-html-reporter": "^0.2.2",  "concurrently": "^3.1.0",  "lite-server": "^2.2.2",  "typescript": "^2.1.4"  },  "dependencies": {  "@angular/common": "^2.3.1",  "@angular/compiler": "^2.3.1",  "@angular/core": "^2.3.1",  "@angular/platform-browser": "^2.3.1",  "@angular/platform-browser-dynamic": "^2.3.1",  "core-js": "^2.4.1",  "reflect-metadata": "^0.1.8",  "rxjs": "^5.0.1",  "systemjs": "^0.19.41",  "zone.js": "^0.7.2"  }  } |

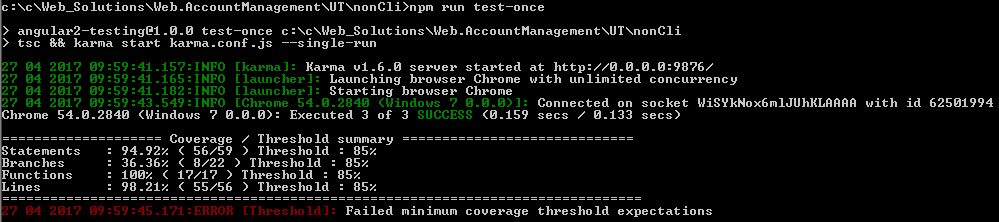
**Karma.conf.js**

|  |
| --- |
| module.exports = function(config) {  config.set({  basePath: '', // Current directory  // Unit test framework  frameworks: ['jasmine'],  plugins: [  require('karma-jasmine'),  require('karma-chrome-launcher'),  require('karma-coverage'),  require('karma-jasmine-html-reporter'),  require('karma-threshold-reporter')  ],  // list of files / patterns to load in the browser  files: [  'node\_modules/systemjs/dist/system.src.js',  // Polyfills  'node\_modules/core-js/client/shim.js',  'node\_modules/reflect-metadata/Reflect.js',    // zone.js  'node\_modules/zone.js/dist/zone.js',  'node\_modules/zone.js/dist/long-stack-trace-zone.js',  'node\_modules/zone.js/dist/proxy.js',  'node\_modules/zone.js/dist/sync-test.js',  'node\_modules/zone.js/dist/jasmine-patch.js',  'node\_modules/zone.js/dist/async-test.js',  'node\_modules/zone.js/dist/fake-async-test.js',  //Include: true = included in the browser using <script> tag; false = load them manually (eg. using Require.js)  { pattern: 'node\_modules/@angular/\*\*/\*.js', included: false, watched: false }  ,{ pattern: 'node\_modules/@angular/\*\*/\*.js.map', included: false, watched: false }  ,{ pattern: 'node\_modules/rxjs/\*\*/\*.js', included: false, watched: false }  ,{ pattern: 'node\_modules/rxjs/\*\*/\*.js.map', included: false, watched: false }  ,{ pattern: 'systemjs.config.js', included: false, watched: false }  ,'karma-test-shim.js'  ,{ pattern: 'build/app/\*\*/\*.js', included: false, watched: true },  //{ pattern: 'src/app/\*\*/\*.ts', included: false, watched: true },  //{ pattern: 'build/app/\*\*/\*.js.map', included: false, watched: true }  ],  // list of files to exclude  exclude: [  './build/\*\*/polyfills.js'  ,'./build/\*\*/vendor.js'  ],  // preprocess matching files before serving them to the browser  // available preprocessors: https://npmjs.org/browse/keyword/karma-preprocessor  preprocessors: {  "build/app/\*\*/\*.js":["coverage"]  },  // test results reporter to use  // possible values: 'dots', 'progress'  // available reporters: https://npmjs.org/browse/keyword/karma-reporter  reporters: ['progress', 'coverage', 'threshold'],  coverageReporter:{  type:'html',  dir:'test-coverage/'  },  port: 9876,  colors: true,  logLevel: config.LOG\_INFO,  autoWatch: true,  browsers: ['Chrome'],  singleRun: false  });  }; |

## Run Unit Test Cases

1. Type command npm rum test-once and press enter.

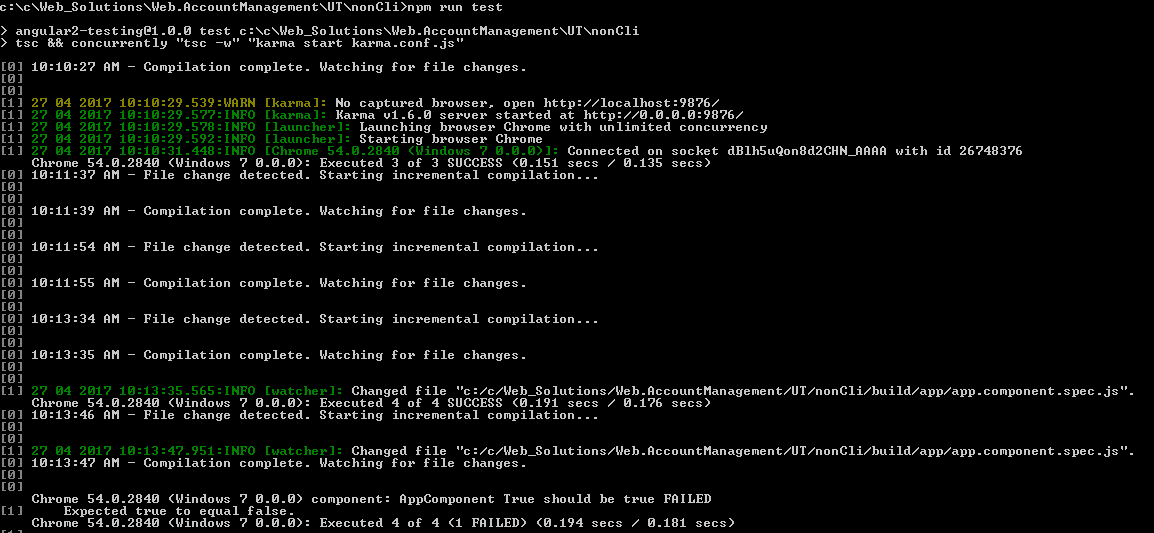
*All the test cases were passed successfully. However, the test run still failed due to insufficient branch coverage (36.36% < 85%).*



1. Type command npm run test and press enter.

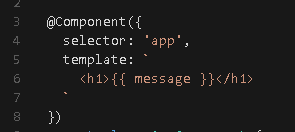
*Refer to the screenshot below. Typescript is monitoring the typescript source code and unit test cases so the modification will trigger rebuilding immediately.*

*Karma is monitoring the compiled JavaScript files so if any typescript build result generated, it will rerun all the unit test cases.*



## Known Issues

1. Some of the auto-generated js code logic cannot be covered which will impact the overall coverage ratio.



The highlighted content below is uncovered branches & statements which are compiled from the Angular Component decorator.



1. Karma cannot run test cases outside of its root directory directly.

The folder structure like below is not supported by default, further research is needed if we planned to separate UT out of the source code project directory.

|  |
| --- |
| Web.AccountManagement   * Web.AccountManagement * 🡺src * 🡺package.json (angular, webpack, typescript) * Web.AccountManagement.Test (spec: angular, karma, jasmine) * 🡺test * 🡺package.json (angular, karma, jasmine) * Web.AccountManagement.sln |