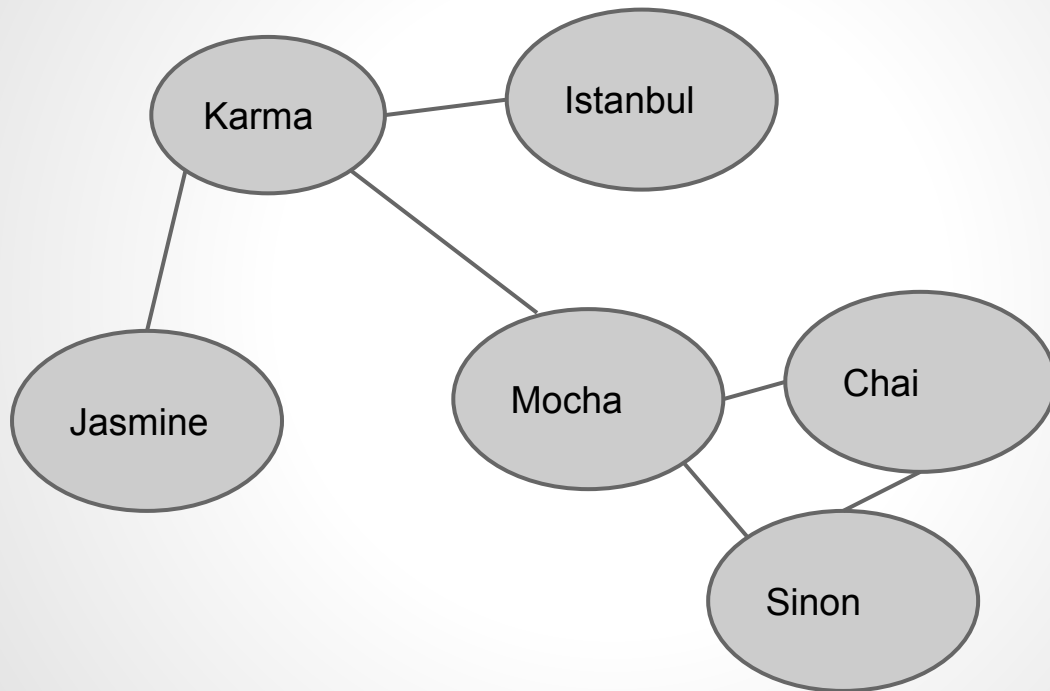


Angular Enterprise

Magnus Gudmundsson CyberDaDa,
Christoffer Noring, Softhouse



Testing Angular - Tools



Karma config + run tests

What it is

How to set it up:

`npm install karma --save-dev // install locally`

Install plugins:

`npm install karma-jasmine karma-chrome-launcher --save-dev`

Install command line interface, cli :

`npm install -g karma-cli`

Install Istanbul (karma-coverage):

`npm install karma-coverage --save-dev`

- Create a `karma.conf.js`, point out where tests are
- Run the tests

Jasmine

service test:

```
describe('login page',function(){
    beforeEach(module('myNameSpace.services'));

    beforeEach(inject(function($injector) {
        userSrv = $injector.get('usrSrv');
    }));

    it('should login user',function(){
        userSrv.login(user,pass);
        expect(userSrv.getLoggedIn()).toBe(true);
    })
})
```

controller test:

```
beforeEach(module('myNameSpace.controllers'));

beforeEach(inject(function($controller, $rootScope,
    _$location_, _restService_) {
    scope = $rootScope.$new();
    $location = _$location_;
    restService = _restService_;

    $controller('ListLibrariesCtrl',
        {$scope: scope, $location: $location,
        restService: restService });

}));

// actual testing
```

Karma Jasmine - http + directive

http test:

```
beforeEach(inject(function($httpBackend, _loginSrv_){
    loginSrv = _loginSrv_
    $httpBackend.expect('POST', 'https://api.mydomain.
com/login')
    .respond(200, "[{ success : 'true', id : 123 }]");
}))

it('should succeed',function(){
    loginSrv.login('test@test.com', 'password')
    .then(function(data) {
        expect(data.success).toBeTruthy();
    });

    $httpBackend.flush();
})
```

directive test:

```
describe("Testing Directives", function() {
    var $compile, $rootScope;

    beforeEach(module('App'));

    beforeEach(inject(
        ['$compile', '$rootScope', function($c, $r) {
            $compile = $c;
            $rootScope = $r;
        }]
    ));

    it("should display the welcome text properly", function() {
        var element = $compile('<div data-app-welcome>User</div>')($rootScope);
        expect(element.html()).toMatch(/Welcome/i);
    })
});
```

Karma Jasmine mocking

mocking:

```
beforeEach(inject(function($provide){  
  
    var mockedSrv = function(){  
        return {  
            a : function(){ }  
        }  
    }  
  
    $provide.service('utilSrv',mockedSrv);  
})
```

Karma - Jasmine

Demo

Angular Testing - Reading

<http://www.smashingmagazine.com/2014/10/07/introduction-to-unit-testing-in-angularjs/>

<http://www.yearofmoo.com/2013/01/full-spectrum-testing-with-angularjs-and-karma.html>

Plato

```
npm install -g plato
```

Demo

Protractor, E2E testing

What it is

Installation:

`npm install -g protractor`

`webdriver-manager update // install needed binaries`

Perform tests:

- 1) `webdriver-manager start // start selenium server`
- 2) `python -m http.server 3000 // start the app somewhere`

`protractor conf.js // run the tests`

Protractor conf.js

Protractor supports the two latest major versions of **Chrome**, **Firefox**, **Safari**, and **IE**, **needs extra install**

```
capabilities: {  
  'browserName': 'firefox'  
}
```

Test two or more browsers concurrently:

```
multiCapabilities: [{  
  'browserName': 'firefox'  
}, {  
  'browserName': 'chrome'  
}]
```

Interacting with 2 different browsers:

// scenarios where you want to test chat like capabilities

Inside a spec file

```
var browser2 = browser.forkNewDriverInstance();  
var element2 = browser2.element();  
element2(by.model("sendChatButton")).click();
```

Protractor - selectors

find controls:

element() returns ElementFinder, is *lazy*, only returns when action is called example **sendKeys()**

element(by.model('test')); // \$scope.test

element(by.id('saveButton')) // <button id="saveButton">

element(by.binding('person.email')) or element(by.binding('email')) //

element(by.css('cssClass')) or \$('cssClass')

list elements:

element.all(by.css('css')) or element.all(by.repeater('todo in todos'))

helper methods: **first()**, **last()**, **count()**

action methods:

click, getText, clear - returns a promise, so if you wanna do something after it is done then, example

```
element(by.id('test')).getText().then(val){ console.log(val); }
```

Protractor demo

First test

Page Objects, make it nicer

Protractor - further reading

Locators in detail , <https://github.com/angular/protractor/blob/master/docs/locators.md>

Page objects, <http://www.thoughtworks.com/insights/blog/using-page-objects-overcome-protractors-shortcomings>

Protractor API, <http://angular.github.io/protractor/#/api>

Templates

first app :

js/controllers.js
js/services.js
js/app.js
js/models.js
js/directives.js
js/specs
index.html
partials/user.html
partials/products.html
partials/login.html

changes

second app :

js/controllers/userController.js
productsController.js
loginController.js
js/services/userService.js
productsService.js
loginService.js
userService.js
js/app.js
js/models/userModel.js
productsModel.js
index.html
partials/user.html
products.html
login.html

Enterprise app :

js/user/userController.js
userService.js
userModel.js
user.html
user-directive.js
js/products/productsController.js
productsService.js
productsModel.js
products.html
products-directive.js
js/common/login/loginService.js
login.html
menu
menuService.js
js/app.js
index.html

per
feature

merge conflict
waiting to
happen

Taskrunners:

Gulp

Grunt

Broccoli

The world according to Gulp

Gulping the globs

1360 plugins

Gulpstreams



Gulpwatching

```
gulp.task('watch', function () {
```

```
  gulp.watch('templates/*.tpl.html', ['build']);
```

```
});
```

Take a big Gulp...

Demo

Bower

- Before
 - locate file / files
 - download file or refer to CDN
 - upgrade as well as remove was a **pain** too
- After
 - bower install <package name>, bower uninstall, bower update
- Package manager for the frontend / web
- npm install -g bower
- example packages angular, jquery, lodash etc..

Bower commands

install :

bower install <package name>

install with version :

bower install <package>#<versionNumber>

list all versions:

bower info <package name>

uninstall:

bower uninstall <package name>

list installed packages :

bower list

search for package (title) :

bower search <part of package name>

web page search (title + description) :

www.bower.io/search

update all packages:

bower update

update 1 package:

bower install <package name>

list everything in cache:

bower cache list

clean cache:

bower cache clean

Bower package

- installed under bower_components
- **bower.json**, similar to package.json
- **bower init**, create bower.json
- **.bowerrc**, overrides install directory

Example { "directory" : "js/lib" }

Bower demo

- Install and update packages
 - Cached vs not cached
- Override install path
- Create your own package with dependency
- Install from local directory

Bower further reading

bower homepage ,<http://bower.io/>

jspm - new kid on the block, jspm.io

Load any module format (ES6, AMD, CommonJS and globals) directly from any endpoint such as npm and github

Communication (Breakdown)

\$rootscope

“Channels”

Other?

Yeoman

is a scaffolding tool, to help you create a starter template for your project

has generators, which speeds up development

Example

Angular generator scaffolds a project which includes **directories**, **files** and **configuration** , you get

- **Angular** project skeleton
- **Tests** for different components
- **Grunt** configuration and setup
- **Bower** configuration and setup
- **Generators** for working with the angular projects for things such as **controllers**, **factories** / **services**, **modules**, **routes** , makes you faster !

Yeoman - get started

Prerequisites:

npm install -g **grunt** **bower** **yo** **generator-karma** **generator-angular**

yo angular [app-name] // scaffolding

run with :
grunt or **grunt serve**

Other generators under angular :

angular:controller

angular:directive

angular:filter

angular:route

angular:service

angular:provider

angular:factory

angular:value

angular:constant

angular:decorator

angular:view

Tests don't work but add this:

npm install grunt-karma --save-dev

npm install karma-phantomjs-launcher --save-dev

Yeoman demo

- Download angular generator
- Scaffolding yo app
- Show
 - `yo angular:route` ,adding routes
 - `yo angular:controller`, service, factory, filter
 - starting grunt tasks like `grunt serve`, `test`, `build`

Yeoman further reading

- Yeoman homepage, <http://yeoman.io/>

Find the generator that suits your needs for example :

- angular project generator, <https://github.com/yeoman/generator-angular>
- angular enterprise generator, <https://github.com/cgross/generator-cg-angular>

Or build your own

- Create your own generator <http://code.tutsplus.com/tutorials/build-your-own-yeoman-generator--cms-20040>

Questions?

Feel free to contact / connect with us on either
linked in or twitter