

# “Empowering HTML with angularJs”

mg-presenter="EmmanuelAtakem as  
atacraft"

# Prérequis

- HTML
- CSS
- JAVASCRIPT

# Introduction

- HTML

- C'est quoi? (Hypertext Markup Language)
- Quelles en sont les limites? ( "Build for docs")
- Les solutions connues ... ( js, jquery, FrameWorks)

- AngularJS

- "C'est qui encore celui là?!?" (un FrameworkJS...)
- Il est le seul dans ce genre? (EmberJS, Backbone...)
- "heu ... je fais quoi finalement?!?"
- "Qui utilise ton histoire là même?" (vevo, plunker...)



# Quelques indicateurs

6,395 commits



19 branches

125 releases

1,160 contributors

 branch: master **angular.js** / + 

fix(openPlunkr): enable cmd+click for us mac users :>



 caitp authored 13 hours ago latest commit 301663e734 

8,601 commits



39 branches

101 releases

451 contributors

 branch: master **ember.js** / + 

Merge pull request #10276 from ef4/fix-unregister-waiter ...



 rwjblue authored 7 hours ago latest commit d2d1490d0d 

2,747 commits



2 branches

21 releases

246 contributors

 branch: master **backbone** / + 

Merge pull request #3461 from jridgewell/listenToOnce-noop ...

 akre54 authored 16 hours ago latest commit 7315c56d2e 

# Quelques indicateurs

angularjs

Search term

emberjs

Search term

backbonejs

Search term

+ Add term

Interest over time ?

☐ Compare to category ? ☐ News headlines ? ☐ Forecast ?



</>

# Quelques indicateurs

- Comparaison sur openhub : [https://www.openhub.net/p/compare?project\\_0=Ember.js&project\\_1=Backbone.js&project\\_2=AngularJS](https://www.openhub.net/p/compare?project_0=Ember.js&project_1=Backbone.js&project_2=AngularJS)
- Exemples sur todomvc: <http://todomvc.com/>

# Environnement de travail

- NodeJS
- AngularJS
- Un éditeur de texte ou IDE
- “Et le navigateur alors? tu as parlé de JS non? ...”

# Principales notions (Module)

- Module : Un conteneur pour les différentes parties d'une application

```
1 <!doctype html>
2 <html lang="fr" ng-app="drugDemo">
3 <head>
4 <title>Drug demo</title>
5 </head>
6 <body>
7
8 <script src="bower_components/jquery/dist/jquery.js"></script>
9 <script src="bower_components/angular/angular.js"></script>
10 <script src="assets/js/app.js"></script>
11 </body>
12 </html>
```



# Principales notions (Module)

```
1 <!doctype html>
2 <html lang="fr" >
3 <head>
4 <title>Drug demo</title>
5 </head>
6 <body>
7 <div ng-app="drugDemo">
8
9 </div>
10 <script src="bower_components/jquery/dist/jquery.js"></script>
11 <script src="bower_components/angular/angular.js"></script>
12 <script src="assets/js/app.js"></script>
13 </body>
14 </html>
```

```
1 angular.module('drugDemo', ['dependances']);
2
```

# Principales notions (Scope)

- Scope: Objet définissant le contexte d'exécution d'une partie ou, de l'application entière.
  - Un seul "rootScope"
  - Zéro ou plusieurs scope fils
  - Un scope fils hérite du scope parent

# Principales notions (Scope)

```
1 <!doctype html>
2 <html lang="fr" ng-app="drugDemo">
3 <head>
4 <title>Drug demo</title>
5 </head>
6 <body>
7 <div>
8   <p>Some math operation inside the rootScope: 1 + 2 = {{ 1 + 2 }}</p>
9 </div>
10
11 <div>
12   i can have a rootScope's child here!
13 </div>
14
15 <script src="bower_components/jquery/dist/jquery.js"></script>
16 <script src="bower_components/angular/angular.js"></script>
17 <script src="assets/js/app.js"></script>
18 </body>
19 </html>
```

rootScope

currentScope is undefined here!

Expression in bindings

# Principales notions (Contrôleur)

- Contrôleur: constructeur javascript encapsulant une certaine logique métier et utilisée pour enrichir le scope du framework.

# Principales notions (Contrôleur)

```
1 angular.module('drugDemo').controller('DrugDemoCtrl', [
2   'dépendances',
3   function(dependances) {
4     console.log('Hello DRUGS!');
5   }
6 ]);
```

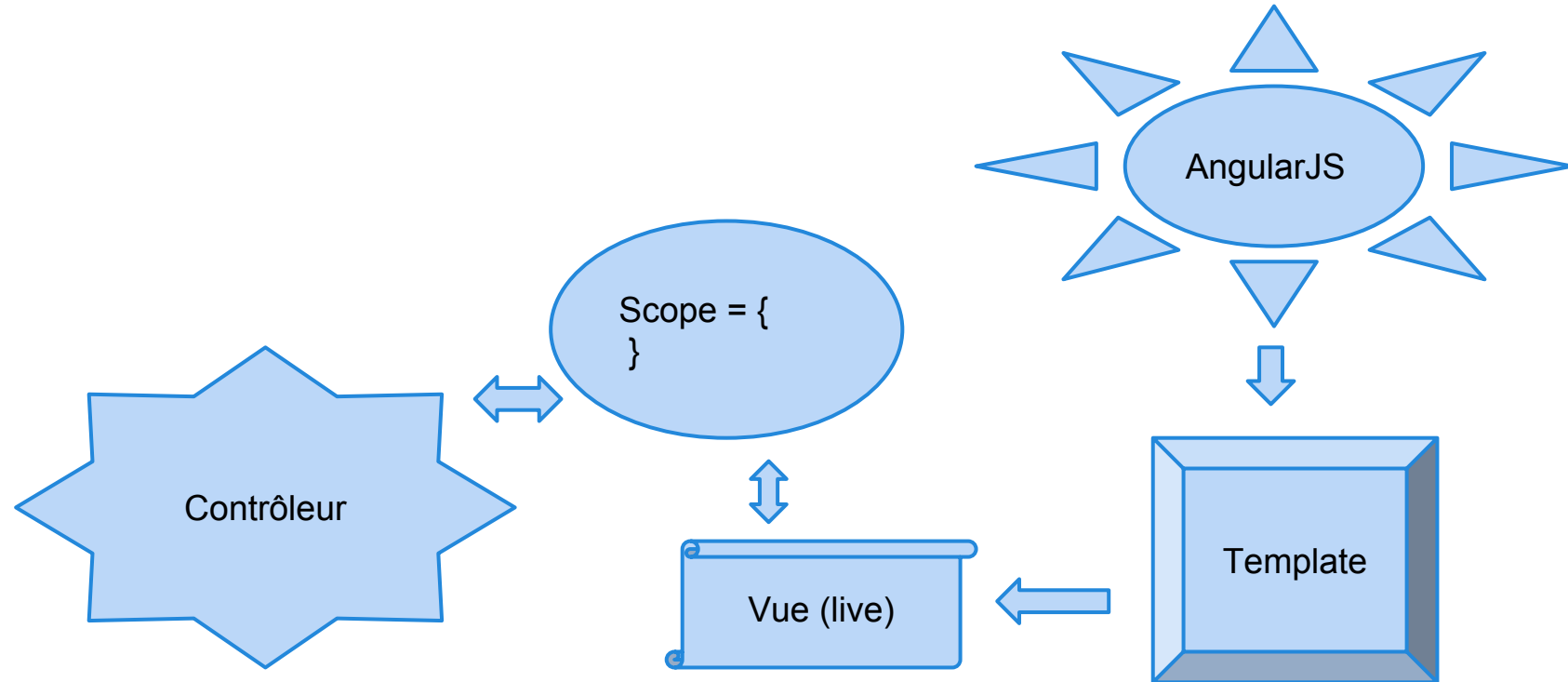
```
<script src="bower_components/angular/angular.js"></script>
<script src="assets/js/app.js"></script>
<script src="demo_notions/demo_controller.js"></script>
</body>
</html>
```

```
<div ng-controller='DrugDemoCtrl'>
  i extended the rootScope and i can populate my own scope!
</div>
```

# Principales notions (Two-ways ...)

- Two-ways binding: Synchronisation permanente des données de la vue et celles du modèle.

# Principales notions (Two-ways ...)



# Principales notions (Directive)

- Directive: Marqueur sur le DOM
  - directives de AngularJS

```
<div ng-controller='DrugDemoCtrl'>  
  i extended the rootScope and i can populate my own scope!  
</div>
```

- directives personnelles

```
1 angular.module('demoDrug').directive('mgValidateForm', function() {  
2   return {  
3   }  
4 }));
```



# Principales notions (service)

- Service: Objet remplaçable par un autre (similaire), et utilisé pour organiser et partager du code. Il est toujours un singleton.

```
1 angular.module('drugDemo').factory('drugDemoService', function() {  
2  
3 });
```

# Principales notions (Routage)

- Routage: selection du template et du contrôleur (si nécessaire) associés au traitement d'une demande donnée.

```
1 var drugDemo = angular.module('drugDemo', ['ngRoute']);
2
3 drugDemo.config(['$routeProvider', function($routeProvider) {
4     $routeProvider.
5     when('/some_url', {
6         templateUrl: 'path to template',
7         controller: 'controller name'
8     }).
9     otherwise({
10        redirectTo: '/'
11    });
12 });
```

# Tests (Unitaires)

- Unitaires: validation de la conformité de certaines instructions et fonctions. (karma, jasmine)

```
4 describe('context', function() {  
5     it('does some particular stuffs', function() {  
6         expect(some var).toEqual('some value');  
7     });  
8 });
```

# Tests (End-to-end)

- End-to-end: simulation de l'utilisation du produit par une personne physique.  
(protractor)

```
3 describe('a feature', function() {  
4     it('should guarantee some visible results', function() {  
5         element(by.css(':button')).click();  
6     });  
7 });
```

# En somme ... “angular”

- + une forte communauté
- + two ways data binding
- + facile (template entièrement en HTML)
- + facile a tester
- - Performance

# Sources

- NETICOA Sénégal ( :-o )
- codeSchool ( <http://campus.codeschool.com/courses/shaping-up-with-angular-js/contents> )
- angularjs website ( <https://angularjs.org/> )
- ng-book ( ... not free :p ... google-it )

# Remerciements

- Sir MBACKE
- Master Hassane MOUSTAPHA
- Mame Cheikh CAMARA
- ... “ Way yow!!!?! Et moi?!?” ...

**MERCI || ARIGATO || THANKS ||  
DIEUREDIEUF**

- `$scope.$watch(Questions && Contributions);`