

Angular JS



Journées Mathrice - 31 Mars / 2 Avril 2015 - Olivier Labbe - Mathdoc

Angular - Objectifs de la présentation

~~Formation~~ Vue d'ensemble

Angular - Objectifs de la présentation

Formation Vue d'ensemble

Utilisé par le portail math

The screenshot shows the 'portail M4T BETA' website interface. At the top, there is a navigation bar with links for 'Bibliothèque Numérique', 'Annuaire', 'Agenda', 'Services Numériques', and 'Sites utiles', along with a 'Connexion' button. Below the navigation bar, there are filters for 'Par mois' (set to 'Mai'), 'Par région' (set to 'Toutes'), and 'Par séminaire' (with a search box). The main content area displays a list of events for May 2015:

- 4 - 7 MAI 2015**: Spectral and Analytic Inverse Problems, Paris, IHP Colloques et Conférences: Colloques de l'IHP.
- 5 MAI 2015**: Exposé de Olivier Piltant, Versailles, LMV Séminaires: AG 2014-2015, by Olivier Piltant.
- 6 MAI 2015**: Motivic periods and the cosmic Galois group (1/4).

On the right side of the page, there is a map of France with a blue pin indicating the location of Le Havre and a yellow circle highlighting Paris.

Angular - C'est quoi ?

Framework Javascript

- S'exécute côté client
- Structure le code et impose une architecture
- Model View Controller (MVC)

Angular - Les services

Les services Angular permettent de partager du code dans toute l'application

`$location`: interagir avec l'URL de votre navigateur

`$route`: changer de vue en fonction de l'URL

`$http`: communiquer avec les serveurs

Angular - Le modèle

Données côté serveur

```
// Simple GET request example :
$http.get('/someUrl').
  success(function(data, status, headers, config) {
    // this callback will be called asynchronously
    // when the response is available
  }).
  error(function(data, status, headers, config) {
    // called asynchronously if an error occurs
    // or server returns response with an error status.
  });
```

Angular - Le modèle - \$resource

Permet de traiter les informations
provenant d'un serveur REST

```
// Format {"id": 123,"name": "Utilisateur 1"}  
var User = $resource('/users/:userId', {userId:'@id'});
```

```
// query  
var the_users = User.query();
```

```
// get  
var the_user = User.get({userId:123});
```

```
// new
```

Angular - Le controleur

Récupère les données du modèle et les met à disposition de la vue

```
var myApp = angular.module('myApp', []);

myApp.controller('UserCtrl', ['$scope', function ($scope) {
  $scope.user = {};
  $scope.user.details = {
    "username": "Todd Motto",
    "id": "89101112"
  };
};
```


Angular - La vue

Code HTML

```
<div ng-app="myApp">
  <div ng-controller="UserCtrl">
    <p class="username">Welcome, {{ user.details.username }}</p>
    <p class="id">User ID: {{ user.details.id }}</p>
  </div>
</div>
```

Angular - Le controleur

1 controleur par vue

```
var myApp = angular.module('myApp', []);  
  
myApp.controller('MainCtrl', ['$scope', function ($scope) {...}]);  
myApp.controller('NavCtrl', ['$scope', function ($scope) {...}]);  
myApp.controller('UserCtrl', ['$scope', function ($scope) {...}]);
```

Angular - Les filtres

Filtrer une collection de données

Filtrer la saisie d'un utilisateur

```
myApp.controller('MainCtrl', ['$scope', function ($scope) {  
  $scope.numbers = [10, 25, 35, 45, 60, 80, 100];  
  $scope.lowerBound = 42;  
  
  $scope.greaterThanNum = function (item) {  
    return item > $scope.lowerBound;  
  };  
}]);
```

Angular - Binding de données à double
Si le modèle est mis à jour, la vue est
mise à jour
Si vue est mise à jour, le modèle est
automatiquement mis à jour

```
<div ng-app="myApp">  
  <div ng-controller="MainCtrl">  
    <input type="text" ng-model="myModel" placeholder="Start typing..." />  
    <p>My model data: {{ myModel }}</p>  
  </div>  
</div>
```

Angular - Les directives

ng-hide, ng-show

```
<!-- Affiche le div quand $scope.myValue est vrai -->  
<div ng-show="myValue"></div>
```

On peut créer ses propres directives

```
<!-- 1: déclaration en tant qu'attribut -->  
<a custom-button>Click me</a>
```

```
<!-- 2: en tant que nouvel élément -->  
<custom-button>Click me</custom-button>
```

Angular - Routage

Insérer des pages dynamiquement

```
// Dans index.html  
<div ng-view></div>
```

Routage

```
$routeProvider  
  .when('/agenda', {  
    templateUrl: 'partials/agenda.html',  
    controller: 'agendaCtrl'  
  })  
  .when('/contact', {  
    templateUrl: 'partials/contact.html',  
    controller: 'contactCtrl'  
  })  
  .when('/', {  
    templateUrl: 'partials/home.html',  
    controller: 'homeCtrl'  
  })  
  .otherwise({  
    templateUrl: 'partials/404.html',  
    controller: '404Ctrl'  
  })
```

Angular - Compatible avec les autres OpenStreetMap (leafletJS)

```
markers.on( 'click', function (a) {  
    ...  
    var the_position = a.latlng;  
    the_locations.push( { lat: the_position.lat, lon: the_position.lon } );  
  
    $rootScope.$broadcast(MATH_EVENTS.markerClicked, { locations: the_locations } );  
}  
  
$scope.$on(MATH_EVENTS.markerClicked, function(event, args) {  
    $scope.clickedMarkers = args.locations;
```

Angular - Tendance

angularJS

Terme de recher...

emberjs

Terme de recher...

backbon...

Terme de recher...

Knockou...

Software

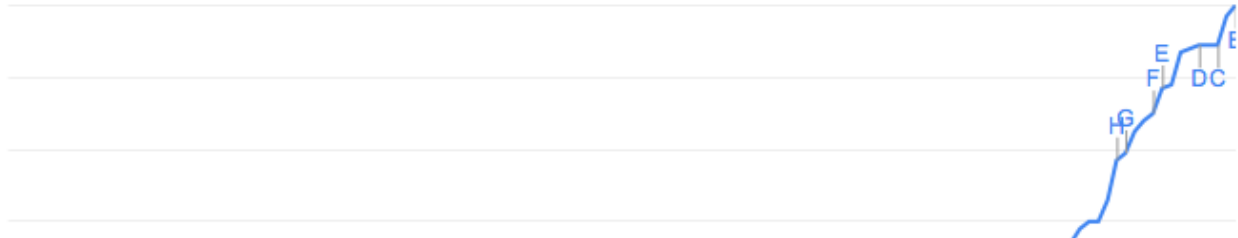
react js

Terme de recher...

Bêta : l'analyse statistique des *sujets* de recherche qui intéressent le plus les internautes est une fonctionnalité bêta qui permet d'analyser rapidement et avec précision les centres d'intérêt des utilisateurs dans la recherche sur Internet en général. Pour connaître le volume de recherche pour une *requête* en particulier, sélectionnez l'option "Terme de recherche". [?](#)

Évolution de l'intérêt pour cette recherche [?](#)

Titres des actualités Prévisions [?](#)



Angular - Exemples de sites

[youtube.com](https://www.youtube.com)

[netflix.com](https://www.netflix.com)

[ovh.com](https://www.ovh.com)

[weather.com](https://www.weather.com)