

Programación AngularJS

1.2*

Introducción

AngularJS Sección:

- ✦ ngRoute with

- ✦ \$q

- ✦ promise

- ✦ resolves

- ✦ ngRepeat

- ✦ first, middle, last, even, odd

- ✦ Buenas Prácticas

- ✦ ui-router

- ✦ service provider

- ✦ html5 mode on

```
(function () {  
  var app = angular.module('pokedex', [  
    'ngRoute',  
    'pokedex.controllers',  
    'pokedex.directives',  
    'pokedex.filters',  
    'pokedex.services'  
  ]);  
  
  app.config(['$routeProvider', function ($routeProvider) {  
    $routeProvider  
      .when('/', {  
        templateUrl: 'views/pokedex.html',  
        controller: 'PokedexController'  
      })  
      .when('/:type', {  
        templateUrl: 'views/pokedex.html',  
        controller: 'PokedexController'  
      })  
      .when('/pokemon/:name', {  
        templateUrl: 'views/pokemon.html',  
        controller: 'PokemonController'  
      })  
      .otherwise({  
        redirectTo: '/'  
      })  
  });  
})
```

AngularJS **ngRoute**

- ✧ retornamos un *promise*
- ✧ ngRoute se hace cargo del “*then*”

```
$routeProvider
.when('/Book/:bookId', {
  templateUrl: 'book.html',
  controller: 'BookController',
  resolve: {
    // funcion x
    tardar: function($q, $timeout) {
      var delay = $q.defer();
      $timeout(delay.resolve, 1000);
      return delay.promise;
    }
  }
})
```


AngularJS Algunas Buenas Prácticas

1. En \$scope sólo funciones y variables que utilice tu plantilla.
2. No utilizar “\$” para prefijos de variables, propiedades o metodos propios
3. \$rootScope para eventos, **NO** para almacenar Datos. (ocupar servicios)
4. No pasar servicios a la vista
5. No ocupar directivas compartidas

AngularJS Eventos y Resolve

EX_13

Uso de resolve

Ventana 1

Ventana 2

ejemplos: resolve/route/tempscript

AngularJS ngRepeat

- first, middle, last, index.

```
<div ng-init="friends = [
  {name:'John', age:25, gender:'boy'},
  {name:'Jessie', age:30, gender:'girl'},
  {name:'Johanna', age:28, gender:'girl'},
  {name:'Joy', age:15, gender:'girl'},
  {name:'Mary', age:28, gender:'girl'},
  {name:'Peter', age:95, gender:'boy'},
  {name:'Sebastian', age:50, gender:'boy'},
  {name:'Erika', age:27, gender:'girl'},
  {name:'Patrick', age:40, gender:'boy'},
  {name:'Samantha', age:60, gender:'girl'}
]">
  Arreglo: {{friends.length}} length.:
  <ul >
    <span ng-repeat="friend in friends">
      <span ng-show="$first"><li>:::INICIO:::</li></span>
      <li>[{{$index + 1}}] {{friend.name}}, {{friend.age}} años.</li>
      <span ng-show="$last"><li>:::FIN:::</li></span>

    </span>
    <li ng-if="results.length == 0">
      <strong>Sin resultados...</strong>
    </li>
  </ul>
</div>
```


EX_13_2

Arreglo: 10 length:

- ::::INICIO::::
- [1] John, 25 años.
- [2] Jessie, 30 años.
- [3] Johanna, 28 años.
- [4] Joy, 15 años.
- [5] Mary, 28 años.
- [6] Peter, 95 años.
- [7] Sebastian, 50 años.
- [8] Erika, 27 años.
- [9] Patrick, 40 años.
- [10] Samantha, 60 años.
- ::::FIN::::

AngularJS \$http => resolve

EX_15

Sin Resolve Con Resolve	
Numero	Titulo
12259	TROUBLESHOOT ANGULAR MATERIAL BREAKAGE WITH "DO NOT WRITE @-BOU
12248	SYNCHRONOUS CLASSES IN NGANIMATE
12265	PASS BY VALUE NOT WORKING IN SELECT
12260	PARENT SCOPE NOT INHERITED WITHIN DIRECTIVE USING NGREPEAT AND REF
12255	ORIGIN NOT BEING ADDED TO REQUEST HEADERS IN SOME CIRCUMSTANCES
12222	OPTIONALY FILTER ARGUMENTS OF XSS-VULNERABILITY-PRONE JQLITE METH
12266	NGANIMATE MESSING WITH VALIDATION CLASSES
12249	NG-FORM CANNOT SHOW MESSAGE & REMOVE ITEM FROM ARRY
12233	NESTED HEADERS ARE NOT SERIALIZED CORRECTLY
12218	MOVE SCE VALIDATION FROM COMPILE/ROUTE/
12271	LEAVE ANIMATION CALLBACK OVERRIDDEN BY NEXT ANIMATION
12263	INPUT VALIDITY REFLECTED ON FORM AFTER '\$REMOVECONTROL'
12241	INFINITE DIGEST ON LOCATION CHANGE ON IOS 9 BETA 2.
12245	FIXES TO BODY ELEMENT ANIMATION

AngularJS \$http distintos estilos

EX_17

\$http llamadas anidadas

Ejecutar 3

Result:

Peter

Albert

Diana

\$q.all

Ejecutar 3

Result:

Peter

Albert

Diana

Eventos - keydown

un texto, y otro más

un texto

y otro más

AngularJS **ui-router**

- Librería externa al core, posiblemente incorporada al core en el futuro
- Diferencia con ngRoute?
 - ui-router tiene vistas anidada
 - ui-router tiene múltiples vistas
 - ui-router tiene ui-sref en donde podemos manejar de mejor forma las rutas para proyectos grandes
 - ui-router se enfoca en estados y ngroute en URLs
 - Capacidad de hacer rutas dinámicas

AngularJS ui-router

```
// app.js
var routerApp = angular.module('routerApp', ['ui.router']);

routerApp.config(function($stateProvider, $urlRouterProvider) {

    $urlRouterProvider.otherwise('/home');

    $stateProvider

        // ESTADO HOME =====
        .state('home', {
            url: '/home',
            templateUrl: 'partial-home.html'
        })

        // OTRO =====
        .state('about', {
            //configs....
        });

});
```

```
<!-- MAIN CONTENT -->
<div class="container">

    <!-- AQUÍ INYECTAMOS CONTENIDO ===== -->
    <div ui-view></div>

</div>
```


AngularJS ui-router - nested views

```
<!-- partial-home.html -->

<div class="jumbotron text-center">
  <h1>Home</h1>
  <p>Vistas anidadas, dentro de otra vista</p>

  <a ui-sref=".list" class="btn btn-primary">List</a>
  <a ui-sref=".paragraph" class="btn btn-danger">Paragraph</a>
</div>

<div ui-view></div>
```

```
$stateProvider
  .state('home', {
    url: '/home',
    templateUrl: 'partial-home.html'
  })
  .state('home.list', {
    url: '/list',
    templateUrl: 'partial-home-list.html'
  })
  .state('home.paragraph', {
    url: '/paragraph',
    template: 'otra más.'
  })
```


AngularJS ui-router ejemplos

EX_18

AngularUI NAV Inicio Nosotros

Pagina inicio

Esta página muestra vistas **anidadas**.

Lista

Parrafo

Texto simple en home.paragraph.["Gato","Oso","Perro"]

AngularJS HTML5 Mode True

```
.config(function ($routeProvider, $locationProvider) {  
  $routeProvider  
    .when('/', {  
      templateUrl: 'views/main.html',  
      controller: 'MainCtrl'  
    })  
    .when('/about', {  
      templateUrl: 'views/about.html',  
      controller: 'AboutCtrl'  
    })  
  
    .otherwise({  
      redirectTo: '/'  
    });  
  |  
  $locationProvider.html5Mode(true);  
});
```


AngularJS PHP Server example config

```
RewriteEngine On
Options FollowSymLinks

RewriteBase /

RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ /#/$1 [L]
```


AngularJS Utilizando Links

```
<base href="/sub_carpeta/" />
```

```
<base href="/" />
```

```
<a href="#/home/contacto">link</a> to <a href="/home/contacto">link</a>
```

```
<a href="#/home/contacto">link</a> to <a href="/home/otro">link</a>
```


AngularJS Service->Provider

- El servicio más sofisticado
- Posibilidad de crear funciones y variables
- Similar a factory
- Al igual que constant puede inyectarse en la fase “config”
- No es posible inyectarlo en run, controllers, etc.
- Se puede inyectar servicios en \$get
- Un ejemplo es \$routeProvider o \$stateProvider

AngularJS Service->Provider

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

app.provider('usuario', function () {
  var _login;
  return {
    setLogin: function (login) {
      _login = login;
    },
    $get: function ($http, ...) {
      return {
        isOk: _login ? true:false,
        getTexto: function(){
          return "Mi Funcion"
        }
      }
    }
  }
});

app.config(function (usuarioProvider) {
  usuarioProvider.setLogin("mbarria");
});

app.controller('ctrl', function (usuario) {
  console.log(usuario.isOk)
  console.log(usuario.getTexto())
});
```


AngularJS Lluvia de conceptos

Modules ✓

Directives

Dependency Injection

Providers

Scopes

Factories

Two-way Data Binding

Expressions

Controllers

Filters

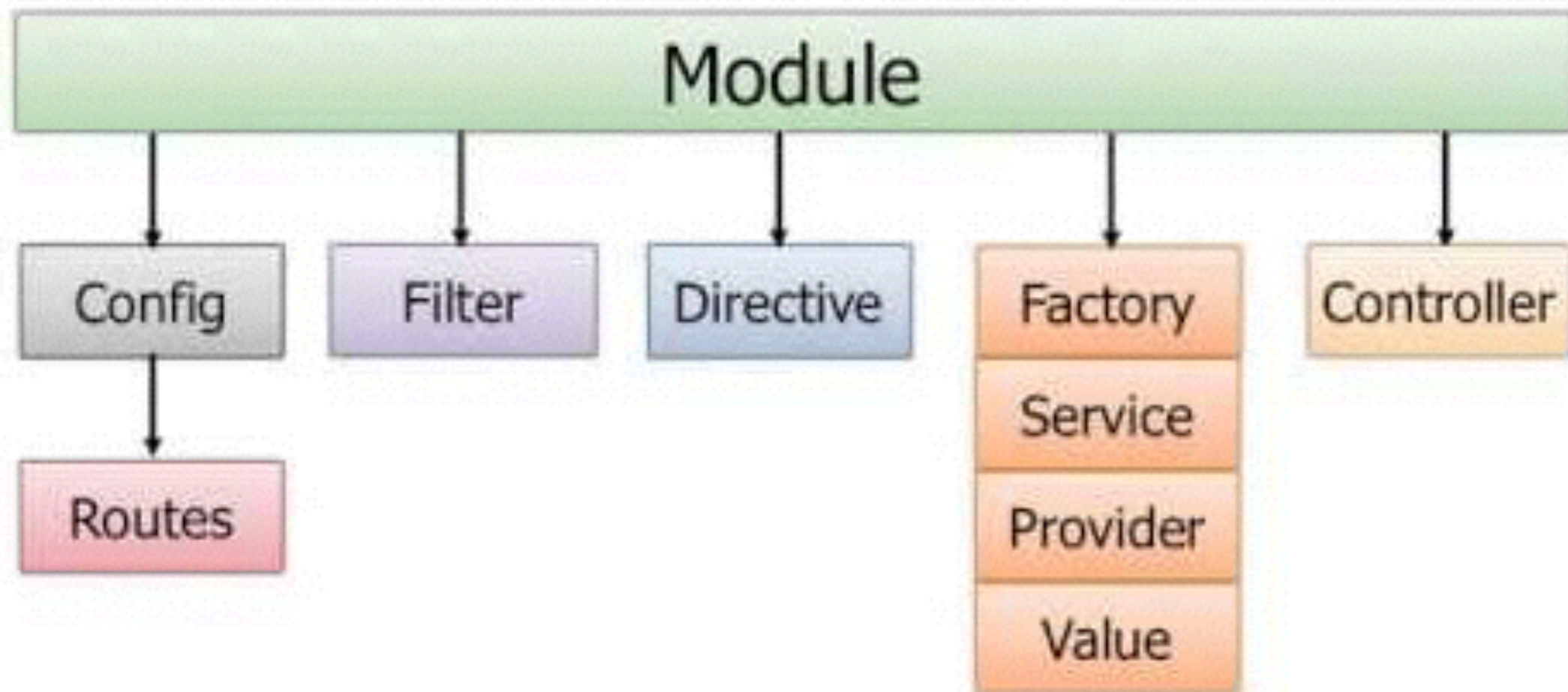
Validators

Services

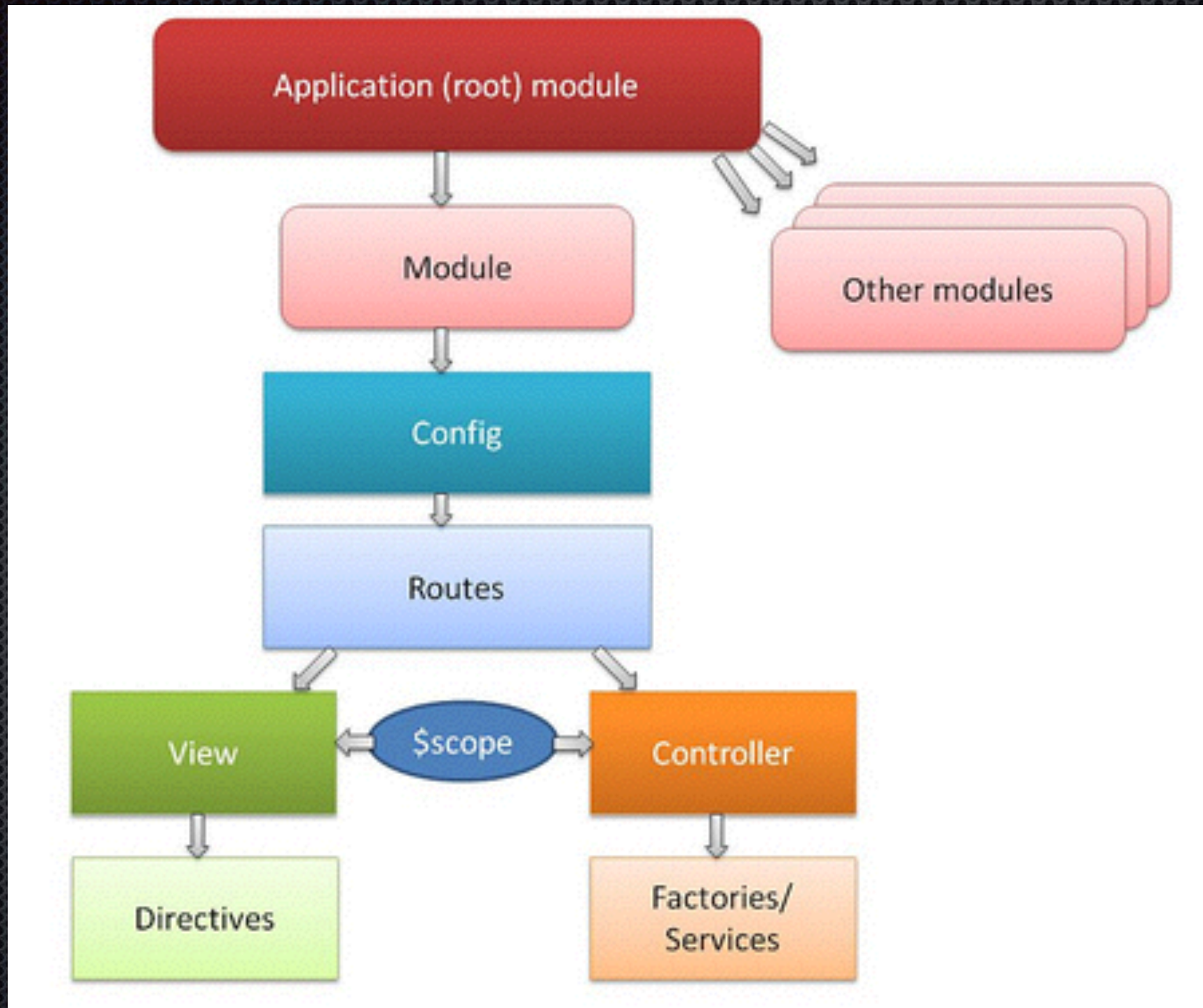
AngularJS Lluvia de conceptos

Modules are Containers

```
<html ng-app="moduleName">
```



AngularJS Lluvia de conceptos



AngularJS Directivas

Key AngularJS Directives

Templates

- ng-disabled
- ng-cloak
- ng-hide
- ng-if
- ng-repeat
- ng-show
- ng-switch
- ng-view

Data Binding

- ng-bind
- ng-href
- ng-init
- ng-model
- ng-src
- ng-style

Forms

- ng-maxlength
- ng-minlength
- ng-pattern
- ng-required
- ng-submit

Application

- ng-app
- ng-controller

Behavior

- ng-blur
- ng-change
- ng-checked
- ng-click
- ng-key*
- ng-mouse*

AngularJS Directivas Scope

```
angular.module('moduleName')
  .directive('myDirective', function () {
    return {
      restrict: 'EA', //E = element, A = attribute, C = class, M = comment
      scope: {
        //@ reads the attribute string value,
        //= provides two-way binding,
        //& works with functions
        title: '@'
      },
      template: '<div>{{ myVal }}</div>',
      templateUrl: 'mytemplate.html',
      controller: controller,
      link: function ($scope, element, attrs) { } //DOM manipulation
    }
  });
```

@: TEXTO

= : DataBinding

& : FUNCTION

AngularJS Directivas Scope compartido

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

app.controller('Controller', ['$scope', function ($scope) {
  $scope.customer = {
    name: 'David',
    street: '1234 Anywhere St.'
  };
}]);

app.directive('sharedScope', function () {
  return {
    template: 'Name: {{customer.name}} Street: {{customer.street}}'
  };
});
```

Scope is inherited

`<div shared-scope></div>`

“**scope**: es buena práctica no compartir el scope, más bien pasarlo a través de atributos (para reutilizar)”

AngularJS Directivas link(manejo del DOM)

```
angular.module('directivesModule')
.directive('domDirective', function () {
  return {
    restrict: 'A',
    link: function ($scope, element, attrs) {
      element.bind('click', function () {
        element.html('You clicked me!');
      });
      element.bind('mouseenter', function () {
        element.css('background-color', 'yellow');
      });
      element.bind('mouseleave', function () {
        element.css('background-color', 'white');
      });
    }
  };
});
```

```
<div dom-directive>Click Me!</div>
```

“**element**: es como si tuviéramos con jquery el elemento \$('#id)’”

“**attrs**: acceso a todos los atributos del elemento”

AngularJS Directivas con controlador propio

```
function cajaBusqueda() {  
  return {  
    restrict: 'E',  
    transclude: false,  
    scope: { textoABuscar: "=" },  
    controller: function($scope) {  
      $scope.textoLocal = "";  
      $scope.borrarBusqueda = function() {  
        $scope.textoABuscar = "";  
        $scope.textoLocal = "";  
      };  
      $scope.buscar = function() {  
        $scope.textoABuscar = $scope.textoLocal;  
      };  
    },  
    replace: true,  
    template:  
      "<form>" +  
      "<div>" +  
        "<input ng-model='textoLocal' type='text' />" +  
      "</div>" +  
      "<div>" +  
        "<button ng-click='borrarBusqueda()'>Borrar</button>" +  
        "<button ng-click='buscar()'>Buscar</button>" +  
      "</div>" +  
      "</form>"  
  };  
};
```

“transclude: para preservar código del contenido”

AngularJS LOGIN

```
angular.module('Authentication', []);  
angular.module('Home', []);  
  
angular.module('BasicHttpAuthExample', [  
    'Authentication',  
    'Home',  
    'ngRoute',  
    'ngCookies'  
])
```

Username: test

Password: test

Username

Username is required

Password

Password is required

Login

AngularJS Tarea - Wizard

```
app.factory("user",function(){  
    return {};  
});
```

- Utilizar servicio para las sub-vistas
- Validaciones
- Siguiente - Atrás
- 1 vista -> 3 sub-vistas -> Controllers
- Utilizar ui-route (ex_18)
- Extra: Utilizar directiva para mostrar el último cuadro
- Progressbar -> Bootstrap

The image displays three sequential screenshots of a wizard form, each with a progress bar at the top indicating the current step (Paso 1, Paso 2, or Paso 3).

Paso 1: The first step contains three input fields: "Nombre", "Apellido" (marked "Requerido!" in red), and "Sexo" (a dropdown menu). A "Siguiente" button is at the bottom right.

Paso 2: The second step contains four input fields: "Numero contacto", "Universidad" (marked "Requerido!" in red), "Edad" (marked "Formato incorrecto" in orange), and a checkbox for "Servicio Militar". "Atras" and "Siguiente" buttons are at the bottom.

Paso 3: The third step displays a summary of the entered data: "Juan Perez, 50 años", "Masculino", "+568432745", "Santo Tomas", and "Servicio Militar al dia". "Atras" and "ok" buttons are at the bottom.