

# AngularJS para Zumbis



# O que é?

É um framework MVC(Model-View-Controller) JavaScript mantido pelo Google, que tem como objetivo simplificar o desenvolvimento de aplicações WEB.



# Diretiva: ng-app

A ng-app diz ao AngularJS que este é o elemento raiz da aplicação AngularJS.

```
<!DOCTYPE html>
<html>

<head>
  <title>AngularJS</title>
  <meta charset="utf-8">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
  <link rel="stylesheet" href="css/app.css">
  <script type="application/javascript" src="js/lib/angular-1.6.5/angular.min.js"></script>
  <script type="application/javascript" src="js/lib/angular-1.6.5/angular-route.min.js"></script>
  <script type="application/javascript" src="js/app.js"></script>
  <script type="application/javascript" src="js/route.js"></script>
  <script type="application/javascript" src="js/filters.js"></script>
  <script type="application/javascript" src="js/zumbis.controller.js"></script>
  <script type="application/javascript" src="js/ng-cep-validator.directive.js"></script>
  <script type="application/javascript" src="js/ng-zumbi-submit.directive.js"></script>
  <script type="application/javascript" src="js/ng-blur.directive.js"></script>
  <script type="application/javascript" src="js/cep.service.js"></script>
</head>
<body ng-app="app">
  <div ng-view></div>
</body>

</html>
```



# Diretiva ng-model:

Com a ng-model, você pode vincular o valor de um campo de entrada a uma variável criada no AngularJS.

```
<tbody>
  <tr ng-repeat="ocorrencia in ocorrencias">
    <td>{{ocorrencia.cep}}</td>
    <td>{{ocorrencia.bairro | upper}}</td>
    <td>{{ocorrencia.numero}}</td>
  </tr>
</tbody>
</table>
<form name="formZumbi" novalidate>
  <div class="form-group">
    <label>CEP</label>
    <input ng-blur ng-model="ocorrencia.cep" ng-cep-validator type="text" class="form-control" />
  </div>
  <div class="form-group">
    <label>Bairro</label>
    <input ng-blur ng-model="ocorrencia.bairro" ng-minlength="4" type="text" class="form-control" />
  </div>
  <div class="form-group">
    <label>Número de ocorrências</label>
    <input ng-blur ng-model="ocorrencia.numero" required type="text" class="form-control" />
  </div>
  <ng-zumbi-submit submit="adiciona(ocorrencia)"></ng-zumbi-submit>
</form>
</div>
```

# Validações:

- Required: ele especifica que um campo de entrada deve ser preenchido antes de enviar o formulário.
- Ng-minlength: adicionará um estado "inválido" do campo de entrada se o comprimento do valor for menor do que o especificado.

```
<div class="form-group">
  <label>Bairro</label>
  <input ng-blur ng-model="ocorrencia.bairro" ng-minlength="4" type="text" class="form-control" />
</div>
<div class="form-group">
  <label>Número de ocorrências</label>
  <input ng-blur ng-model="ocorrencia.numero" required type="text" class="form-control" />
</div>
<ng-zumbi-submit submit="adiciona(ocorrencia)"></ng-zumbi-submit>
</form>
</div>
```



# Criação de diretiva:

- Ng-cep-validator: foi criada uma diretiva contendo uma função especificando a entrada de valores.

```
angular.module('app').directive('ngCepValidator', function(CepService, $rootScope) {  
  return {  
    restrict: 'A',  
    require: 'ngModel',  
    link: function($scope, $element, $attrs, ngModel) {  
      $scope.$watch($attrs.ngModel, function(value) {  
        if (value) {  
          if (value.match(/^([0-9]{5})-([0-9]{3})$/)) {  
            CepService.get(value).then(function(response) {  
              $rootScope.$broadcast('cep', response.data);  
            });  
            ngModel.$setValidity($attrs.ngModel, true);  
          }  
          else {  
            ngModel.$setValidity($attrs.ngModel, false);  
          }  
        }  
        else {  
          ngModel.$setValidity($attrs.ngModel, false);  
        }  
      });  
    }  
  });  
});
```



# Criação de diretiva:

- Ng-zumbi-submit foi criado uma diretiva contendo a função que define a validade do formulário. Caso seja inválido, será apresentando um alerta ao usuário.

```
angular.module('app').directive('ngZumbiSubmit', function() {  
  return {  
    restrict: 'E',  
    require: '^form',  
    scope: {  
      submit: '&',  
    },  
    template: '<button type="submit" class="btn btn-primary">Salvar</button>',  
    link: function($scope, $element, $attrs, formCtrl) {  
      $element.on('click', function() {  
        if (formCtrl.$valid) {  
          $scope.submit();  
        }  
        else {  
          alert('formulário inválido');  
        }  
      });  
    }  
  };  
});
```



# Criação de diretiva:

- Ng-blur: foi criada uma diretiva com a funcionalidade de quando o formulário for preenchido invalidamente terá sua borda colorida de vermelho.

```
angular.module('app').directive('ngBlur', function() {  
  return {  
    restrict: 'A',  
    require: 'ngModel',  
    link: function($scope, $element, $attrs, ngModel) {  
      $element.on('blur', function() {  
        if (ngModel.$invalid) {  
          $element.addClass('error');  
        }  
        else {  
          $element.removeClass('error');  
        }  
      });  
    }  
  }  
});
```





# Controllers:

- \$scope: O escopo é a parte vinculativa entre o HTML (view) e o JavaScript (controlador).
- Ng-repeat: repete um conjunto de HTML, um dado número de vezes.
- Ng-if: foi utilizado para mostrar a tabela quando o array for > 0.

```
<table ng-if="ocorrencias.length > 0" class="table">
  <thead>
    <th>CEP</th>
    <th>Bairro</th>
    <th>Número</th>
  </thead>
  <tbody>
    <tr ng-repeat="ocorrencia in ocorrencias">
      <td>{{ocorrencia.cep}}</td>
      <td>{{ocorrencia.bairro | upper}}</td>
      <td>{{ocorrencia.numero}}</td>
    </tr>
  </tbody>
</table>
```



# Rotas:

- Criação de módulo:

```
angular.module('app', ['ngRoute', 'filters']);
```

- ngRoute é um módulo que roteia seu aplicativo para páginas diferentes sem recarregar o aplicativo inteiro.

```
angular.module('app').config(function ($routeProvider) {  
    $routeProvider  
        .when('/', {  
            templateUrl: 'zumbis.html',  
            controller: 'zumbisCtrl'  
        });  
});
```



# Comunicando com o back-end (\$HTTP):

- \$HTTP: faz uma solicitação ao servidor e retorna uma resposta. Ele faz uso do .get() que é um método de atalho do serviço \$http, ele solicita dados de um recurso especificado.

```
angular.module('app').service('CepService', function($http) {  
  this.get = function(cep) {  
    return $http.get('http://api.postmon.com.br/v1/cep/' + cep);  
  };  
});
```



# Ocorrências de zumbis

CEP	Bairro	Número
58155-000	SOLEDADE	5
58388-000	ALAGOA GRANDE	10

CEP

58388-000

Bairro

Alagoa Grande

Número de ocorrências

10

Salvar

