

secomp 2017

MÓDULO: ANGULAR.JS

Felipe Oliveira



INTRODUÇÃO

Em termos gerais...

ANGULARJS é um **framework** JAVASCRIPT OPEN-SOURCE, MANTIDO PELO **GOOGLE**, que auxilia na execução de **SINGLE-PAGE APPLICATIONS**.



FISOLOFIA

O **Angular.js** é construído sob a crença de que a programação declarativa é melhor do que a programação imperativa quando se trata da **construção de interfaces** com o usuário e da **conexão de componentes software**, enquanto a programação imperativa é excelente para a escrita de regras de negócio.



FISOLOFIA

O Framework adapta e estende o **HTML** tradicional para uma melhor experiência com conteúdo dinâmico, com a ligação direta e bidirecional dos dados que permite sincronização automática de **models e views**.



OBJETIVOS

ABSTRAIR a manipulação DO DOM DA LÓGICA DO APLICATIVO. ISTO MELHORA OS TESTES DO CÓDIGO.

considera os testes DO APLICATIVO TÃO IMPORTANTES QUANTO SEU DESENVOLVIMENTO. **A DIFICULDADE DO TESTE É DIRETAMENTE AFETADA PELA MANEIRA COMO O CÓDIGO É ESTRUTURADO.**

ABSTRAI O ACOPLAMENTO ENTRE O LADO CLIENTE E O LADO SERVIDOR DA APLICAÇÃO. **ISTO PERMITE QUE O DESENVOLVIMENTO DO APLICATIVO EVOLUA EM AMBOS OS LADOS, DE FORMA PARALELA,** e PERMITE O REUSO DE CÓDIGO.

GUIA OS DESENVOLVEDORES ATRAVÉS DA CONSTRUÇÃO DE TODO O APLICATIVO: DESDE O DESIGN DE INTERFACE, PASSANDO PELA ESCRITA DAS REGRAS DE NEGÓCIO, ATÉ CHEGAR AOS TESTES DA APLICAÇÃO.



comunidade sólida

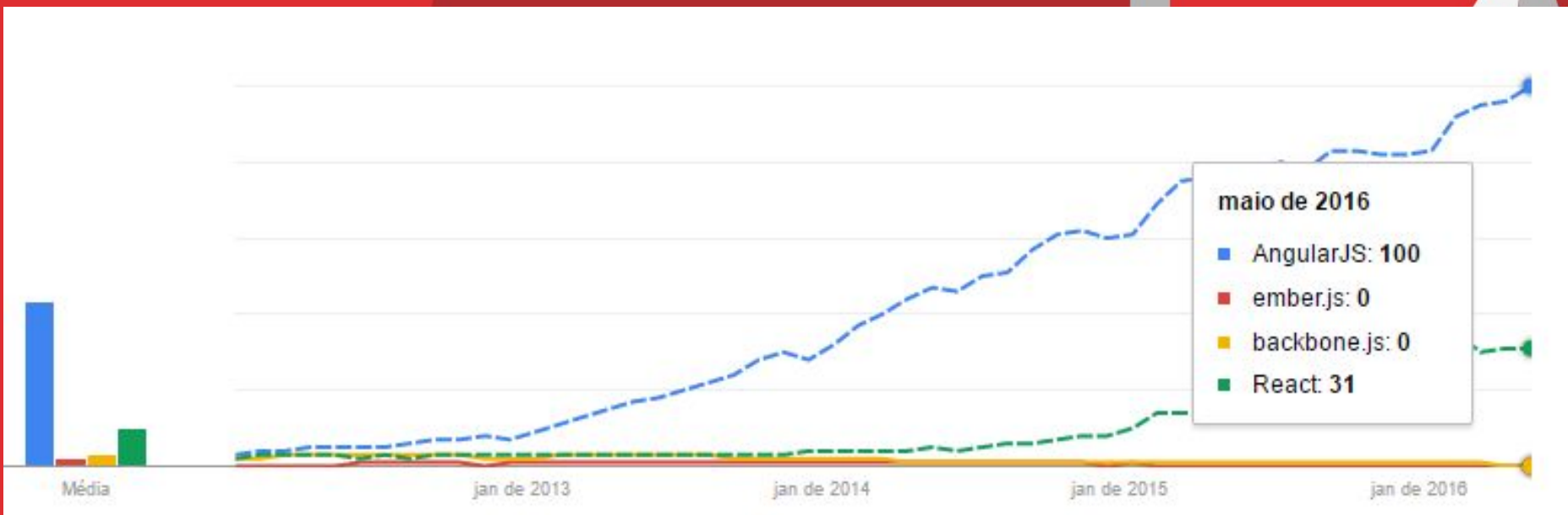
AS métricas DO ANGULAR são impressionantes e refletem a imensa aceitação dele pela comunidade.

O **REPOSITÓRIO DO ANGULAR NO GITHUB TEM 49 MIL ESTRELAS** e mais de MIL contribuintes, além de mais que 150 MIL repositórios com scripts que utilizam a tecnologia.

NO STACK OVERFLOW, a maior comunidade de perguntas e respostas do mundo, temos quase **180 MIL PERGUNTAS. CASO O INTERESSE SEJA EM VÍDEOS SOBRE O ASSUNTO, O YOUTUBE NOS DÁ uma marca impressionante de 470 MIL VÍDEOS.**



comunidade sólida



Controller

```
function InvoiceController($scope){  
  $scope.name = "HelloWorld";  
}
```

Scope

```
invoice:  
  new InvoiceController
```

View

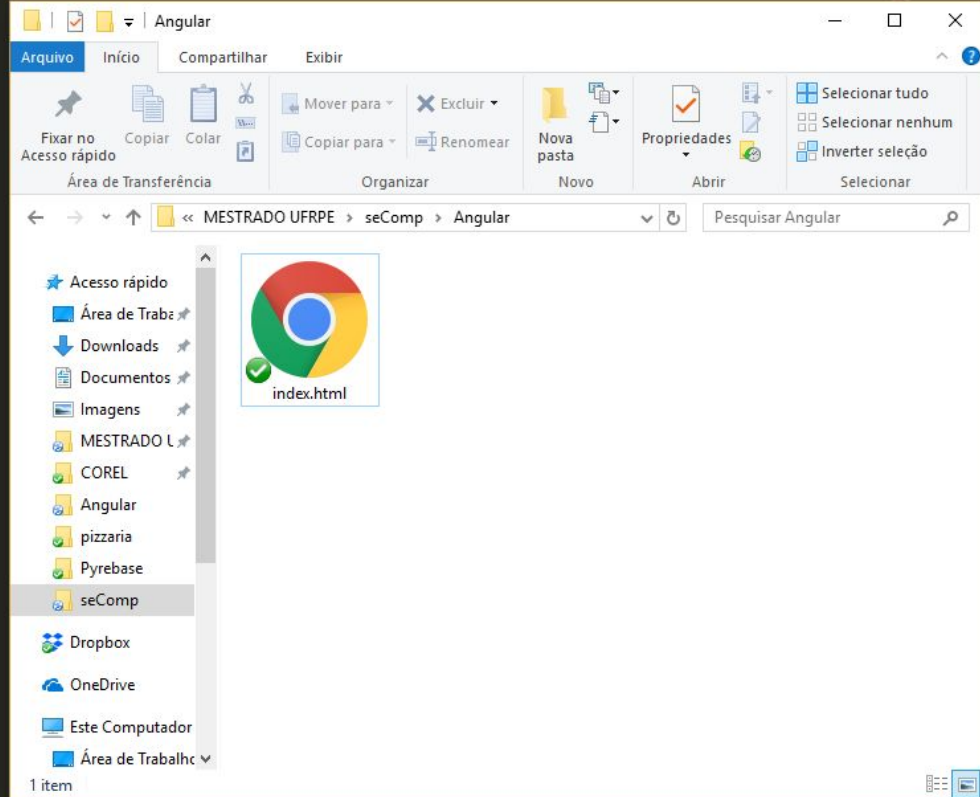
```
<div  
  ng-controller="InvoiceController">  
  
  <h1>{{name}}</h1>
```


PRIMEIROS
PASSOS



index.html

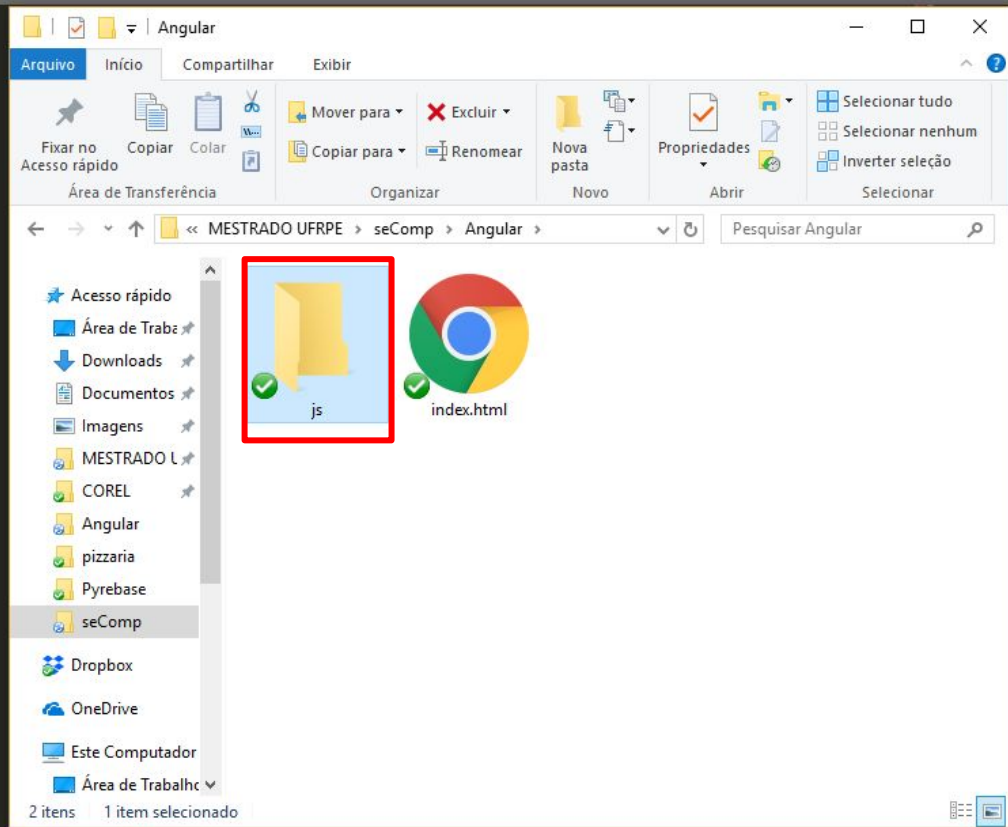
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Pizzaria</title>
5 </head>
6 <body>
7
8   <h1>Hello World!</h1>
9
10 </body>
11 </html>
```



Hello World!

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Pizzaria</title>
5 </head>
6 <body>
7
8   <h1>Hello World!</h1>
9
10 </body>
11 </html>
```



index.html x

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Pizzaria</title>
5 </head>
6 <body>
7
8   <h1>Hello World!</h1>
9
10 </body>
11 </html>
```

The screenshot shows a Windows File Explorer window with the following details:

- Path:** MESTRADO UFRPE > seComp > Angular > js
- Search:** Pesquisas js
- Files:** A table listing files in the directory.

Nome	Data de modificaç...	Tipo	Tamanho
pizza.controller.js	26/10/2017 18:43	Arquivo JavaScript	0 K
pizza.module.js	26/10/2017 18:43	Arquivo JavaScript	0 K

CHAMADA DOS SCRIPTS

AGORA PRECISAMOS CHAMAR
NO **HTML** OS **SCRIPTS** QUE
DESEJAMOS UTILIZAR NA
NOSSA APLICAÇÃO!

```
<Head>
```

```
  #PATH DOS SCRIPTS
```

```
</Head>
```



This site and all of its contents are referring to AngularJS (version 1.x), if you are looking for the latest Angular, please visit angular.io.



TRY THE NEW ANGULAR

DOWNLOAD ANGULARJS



This site and

angular.io

LEARN

DEVELOP

DISCUSS

Download AngularJS

Branch 1.6.x (latest) 1.2.x (legacy)

Build Minified Uncompressed Zip

CTRL + X

CDN

Bower

npm

Extras [Browse additional modules](#)

[Previous Versions](#)

[Download](#)

[VIEW ON GITHUB](#)

Follow +AngularJS on

Follow @angular 249K followers

[DESIGN DOCS & NOTES](#)


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Pizzaria</title>
5
6   <!-- Chamada para os js que precisamos utilizar na aplicacao -->
7   <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.6/angular.min.js"></script>
8   <script src="js/pizza.module.js"></script>
9   <script src="js/pizza.controller.js"></script>
10
11 </head>
12 <body>
13
14   <h1>Hello World!</h1>
15
16 </body>
17 </html>
```

CONFIGURAÇÃO

AGORA VAMOS CONFIGURAR OS SCRIPTS DA NOSSA APLICAÇÃO.

1. CRIAR UM **MÓDULO**
2. CRIAR UM **CONTROLE**
3. LINKAR O MÓDULO AO NOSSO CONTROLE



OPEN FILES

- × index.html
- × pizza.module.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 //Agora vamos criar um módulo (nome do módulo, [dependencias do módulo])
2 angular.module("Pizzaria",[]);
```

OPEN FILES

- × index.html
- × pizza.module.js
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 //vamos criar um controle e linkar ao nosso modulo
2 //Dentro do modulo Pizzaria teremos o controle "PizzariaController"
3 angular.module("Pizzaria").controller("PizzariaController",function(){
4 //controller eh uma funcao que serve como parte logica da aplicacao
5
6
7
8 });
```

ATRIBUIR O CONTROLE

Devemos agora colocar no view o local onde o controle irá funcionar.

Utilizamos a tag:

ng-controller="Nomecontrole"



OPEN FILES

- × index.html
- × pizza.module.js
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
- <> index.html

```
1 <!DOCTYPE html>
2 <html ng-app="Pizzaria">
3 <head>
4   <title>Pizzaria</title>
5
6   <!-- Chamada para os js que precisamos utilizar na aplicacao -->
7   <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.6/
8     angular.min.js"></script>
9   <script src="js/pizza.module.js"></script>
10  <script src="js/pizza.controller.js"></script>
11 </head>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15   <h1>Hello World!</h1>
16
17 </body>
18 </html>
```

vamos Fazer Funcionar!

no nosso **CONTROLE**
Precisamos Passar Por
Parametro nosso **ESCOPO**
Da aplicação, Para que
Finalmente ele consiga
"enxergar" nossa **view**



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 //vamos criar um controle e linkar ao nosso modulo
2 //Dentro do modulo Pizzaria teremos o controle "PizzariaController"
3 angular.module("Pizzaria").controller("PizzariaController",function($scope){
4 //controller eh uma funcao que serve como parte logica da aplicacao
5
6
7
8 });
```


vamos fazer funcionar!

AGORA podemos criar uma variável e testar sua chamada no view

```
$SCOPE.nomevariavel = valor;
```

no **HTML**

```
<H1>{{nomevariavel}}</H1>
```



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 //vamos criar um controle e linkar ao nosso modulo
2 //Dentro do modulo Pizzaria teremos o controle "PizzariaController"
{ 3 angular.module("Pizzaria").controller("PizzariaController",function($scope){
4 //controller eh uma funcao que serve como parte logica da aplicacao
5
6 $scope.titulo = "Eh o Troiiinhaaaa!";
7
8 });
```

OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 <!DOCTYPE html>
2 <html ng-app="Pizzaria">
3 <head>
4   <title>Pizzaria</title>
5
6   <!-- Chamada para os js que precisamos utilizar na aplicacao -->
7   <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.6/
8     angular.min.js"></script>
9   <script src="js/pizza.module.js"></script>
10  <script src="js/pizza.controller.js"></script>
11 </head>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15   <!-- chamada da variavel que declaramos no controller -->
16   <h1>{{titulo}}</h1>
17
18 </body>
19 </html>
```

Eh o Troiinhaaaa!



percorrer
uma LISTA
ng-repeat



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- ▼ Angular
 - ▼ js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
1 //vamos criar um controle e linkar ao nosso modulo
2 //Dentro do modulo Pizzaria teremos o controle "PizzariaController"
3 angular.module("Pizzaria").controller("PizzariaController",function($scope)
4 //controller eh uma funcao que serve como parte logica da aplicacao
5
6 $scope.variavel = "Eh o Troiinhaaaa!";
7
8 //Declaracao de uma lista
9 $scope.pedido = ['Pizza de Calabria',
10                 'Batata frita',
11                 'Coxinha de Bacon',
12                 'Sorvete de Flocos',
13                 'Coca-Cola zero']
14
15 });
```



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
- <> index.html

```
7 <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.6/
8 angular.min.js"></script>
9 <script src="js/pizza.module.js"></script>
10 <script src="js/pizza.controller.js"></script>
11 </head>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15 <!-- chamada da variavel que declaramos no controller -->
16 <h1>{{variavel}}</h1>
17 <ul>
18 <!-- tag utilizada para percorrer uma listas -->
19 <li ng-repeat="item in pedido">{{item}}</li>
20 </ul>
21
22 </body>
23 </html>
```


Eh o Troiinhaaaa!

- Pizza de Calabresa
- Batata frita
- Coxinha de Bacon
- Sorvete de Flocos
- Coca-Cola zero



VALOR
DINÂMICO
ng-model



OPEN FILES

- index.html
- pizza.controller.js

FOLDERS

- Angular
 - js
 - pizza.controller.js
 - pizza.module.js
 - index.html

```
7 <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.6/
8 angular.min.js"></script>
9 <script src="js/pizza.module.js"></script>
10 <script src="js/pizza.controller.js"></script>
11 </head>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15 <!-- chamada da variavel que declaramos no controller -->
16 <h1>{{variavel}}</h1>
17 <p>Ola Sr {{name}}!</p>
18
19 <ul>
20 <!-- tag utilizada para percorrer uma listas -->
21 <li ng-repeat= "item in pedido">{{item}}</li>
22 </ul>
23 <input type="text" ng-model="name">
24
25 </body>
26 </html>
```



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
- <> index.html

```
1 //vamos criar um controle e linkar ao nosso modulo
2 //Dentro do modulo Pizzaria teremos o controle "PizzariaController"
3 angular.module("Pizzaria").controller("PizzariaController",function($scope)
4 //controller eh uma funcao que serve como parte logica da aplicacao
5
6 $scope.variavel = "Eh o Troiinhaaaa!";
7 $scope.name = $scope.name;
8
9 //Declaracao de uma lista
10 $scope.pedido = ['Pizza de Calabresa',
11                 'Batata frita',
12                 'Coxinha de Bacon',
13                 'Sorvete de Flocos',
14                 'Coca-Cola zero']
15
16 });
```

Eh o Troiinhaaaa!

Ola Sr Wesley Safadão!

- Pizza de Calabresa
- Batata frita
- Coxinha de Bacon
- Sorvete de Flocos
- Coca-Cola zero



Digite aqui para pesquisar



POR
PTB2 23:26
26/10/2017



evento
BOTAO
ng-click



OPEN FILES

× index.html
× pizza.controller.js

FOLDERS

Angular
js
/* pizza.controller.js
/* pizza.module.js
index.html

```
11 </neaa>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15     <!-- chamada da variavel que declaramos no controller -->
16     <h1>{{variavel}}</h1>
17     <p>Ola Sr {{name}}!</p>
18
19     <ul>
20         <!-- tag utilizada para percorrer uma listas -->
21         <li ng-repeat="item in pedido">{{item}}</li>
22     </ul>
23     <input id="texto" type="text" ng-model="name">
24
25     <!-- criando um botao e adicionando a tag ng-click -->
26     <button type="button" ng-click="send(name)">CLICA AQUI</button>
27
28 </body>
29 </html>
```

OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
7 $scope.name = $scope.name;
8
9 //Declaracao de uma lista
10 $scope.pedido = ['Pizza de Calabresa',
11                 'Batata frita',
12                 'Coxinha de Bacon',
13                 'Sorvete de Flocos',
14                 'Coca-Cola zero']
15
16
17 //Funcao send recebe o valor vindo da view a adiciona na lista pedido
18 $scope.send = function(valor){
19     $scope.pedido.push(valor)
20 }
21
22 });
```


Eh o Troiinhaaaa!

Ola Sr Vai Safadão!

- Pizza de Calabresa
- Batata frita
- Coxinha de Bacon
- Sorvete de Flocos
- Coca-Cola zero
- Cabeça de Ladrão
- #ForaTemer
- Vai Safadão

Vai Safadão

CLICA AQUI



Digite aqui para pesquisar



POR
PTB2 23:27
26/10/2017



FILTRO DE LISTA DINÂMICO

ITEM IN LISTA | FILTER : nomeFILTRO



OPEN FILES

- × index.html
- × pizza.controller.js

FOLDERS

- Angular
 - js
 - /* pizza.controller.js
 - /* pizza.module.js
 - <> index.html

```
11 </neao>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15     <!-- chamada da variavel que declaramos no controller -->
16     <h1>{{variavel}}</h1>
17     <p>Ola Sr {{name}}!</p>
18
19     <!-- campo para receber o valor para filtragem -->
20     <input type="text" placeholder="Filtrar" ng-model="search">
21     <ul>
22         <!-- tag utilizada para percorrer uma listas -->
23         <li ng-repeat= "item in pedido| filter: search">{{item}}</li>
24     </ul>
25     <input id="texto" type="text" ng-model="name">
26
27     <!-- criando um botao e adicionando a tag ng-click -->
28     <button type="button" ng-click="send(name)">CLICA AQUI</button>
29
30
31 </body>
32 </html>
```

OPEN FILES

× index.html
× pizza.controller.js

FOLDERS

Angular
js
/* pizza.controller.js
/* pizza.module.js
index.html

```
11 </neaa>
12 <!-- Chamada para controle da aplicacao -->
13 <body ng-controller="PizzariaController">
14
15     <!-- chamada da variavel que declaramos no controller -->
16     <h1>{{variavel}}</h1>
17     <p>Ola Sr {{name}}!</p>
18
19     <!-- campo para receber o valor para filtragem -->
20     <input type="text" placeholder="Filtrar" ng-model="search">
21     <ul>
22         <!-- tag utilizada para percorrer uma listas -->
23         <li ng-repeat= "item in pedido| filter: search">{{item}}</li>
24     </ul>
25     <input id="texto" type="text" ng-model="name">
26
27     <!-- criando um botao e adicionando a tag ng-click -->
28     <button type="button" ng-click="send(name)">CLICA AQUI</button>
29
30
31 </body>
32 </html>
```

Eh o Troiinhaaaa!

Ola Sr !

- Pizza de Calabresa
- Batata frita
- Coxinha de Bacon
- Sorvete de Flocos
- Coca-Cola zero



Digite aqui para pesquisar



POR
PTB2

23:42

26/10/2017



Eh o Troiinhaaaa!

Ola Sr !

- Coxinha de Bacon
- Sorvete de Flocos
- Coca-Cola zero



Digite aqui para pesquisar



Eh o Troiinhaaaa!

Ola Sr !

- #ForaTemer

CLICA AQUI

Eh o Troiinhaaaa!

Ola Sr Vai Safadão!

- Vai Safadão



Digite aqui para pesquisar



POR
PTB2

23:43

26/10/2017



HTML + CSS +
ANGULAR.JS



EXPLORAR FUNÇÕES

BAIXE ESSE TEMPLATE:

<https://drive.google.com/drive/folders/0BzZ-bvEmEI1WY0hCLUIVd2VHUVk?usp=sharing>



Pizzaria da SeComp



Mesa: 12

- **Quantidade:** 2
- **Sabor:** Calabresa

Adicionar Pedido

Mesa

Quantidade

Sabor

Limpar

Salvar



vamos fazer funcionar!

AGORA VAMOS APLICAR A
TECNOLOGIA **ANGULAR.JS**
PARA FORNECER
FUNCIONALIDADES AO NOSSO
PROJETO COM **HTML E CSS**
PURO



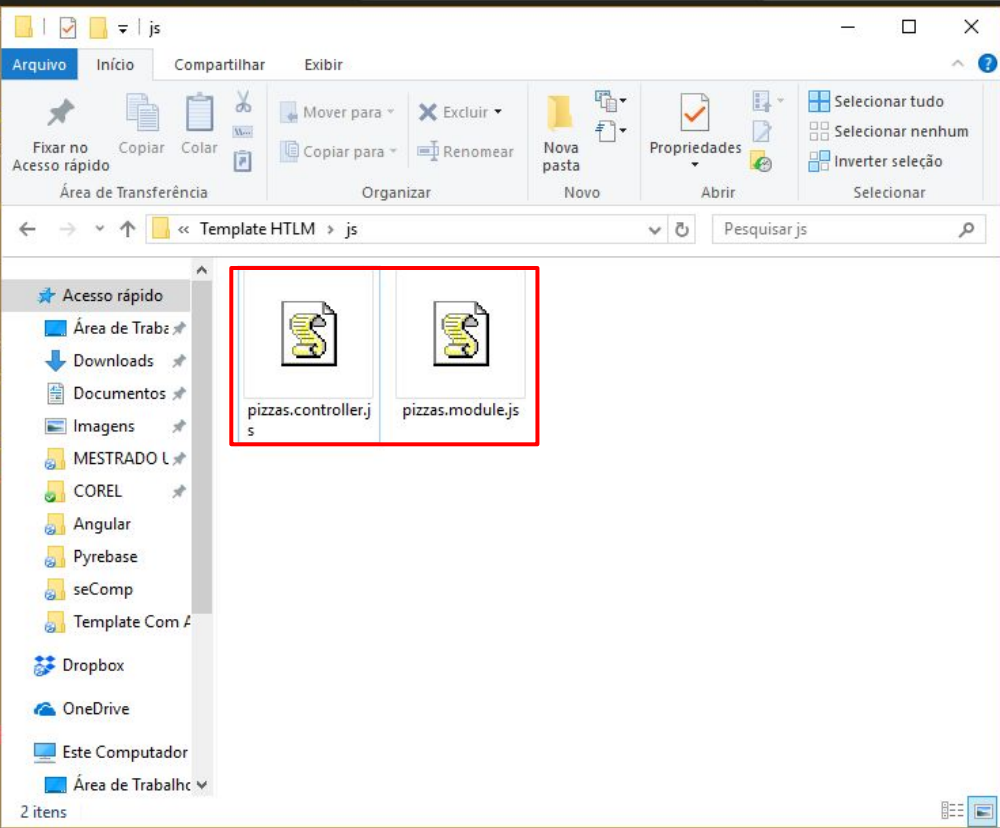
```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charsets="UTF-8">
5 <title>Pizza
6
7 <link rel="stylesheet" href="css/index.css">
8
9 </head>
10 <body>
11 <div class="container">
12 <font class="font">
13
14 <div class="card">
15 <div class="card-body">
16
17
18
19
20
21
22
23 </div>
24 </div>
25
26 <hr>

```

The screenshot shows a Windows File Explorer window titled 'Template HTML'. The address bar shows the path: 'Template HTML-20171030T000228Z-001 > Template HTML'. The ribbon includes 'Arquivo', 'Início', 'Compartilhar', and 'Exibir'. The main area displays four items: 'css' folder, 'img' folder, 'js' folder (highlighted with a red box), and 'index.html' file. The left sidebar shows 'Acesso rápido' with various locations like 'Área de Trabalho', 'Downloads', 'Documentos', etc.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Pizzas
6
7 <link rel="stylesheet" href="css/bootstrap.min.css">
8
9 </head>
10 <body>
11 <div class="container">
12 <font color="red">
13
14 <div class="col-md-6">
15 <div class="card">
16
17
18
19
20
21
22
23 </div>
24 </div>
25
26 <hr>
```



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="UTF-8">
5     <title>Pizzeria da SeComp</title>
6
7     <link rel="stylesheet" href="css/style.css">
8
9     <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/angular.min.js"></script>
10    <script src="js/pizzas.module.js"></script>
11    <script src="js/pizzas.controller.js"></script>
12
13 </head>
14 <body>
15     <div class="container">
16         <font color="#ffffff"><h1>Pizzeria da SeComp</h1></font>
17
18         <div class="lista-pedidos">
19             <div class="pedido com-cartaz">
20                 <h1>Mesa: 12</h1>
21                 <button class="del" title="Excluir"></button>
22                 
23                 <ul>
24                     <li><strong>Quantidade:</strong> 2</li>
25                     <li><strong>Sabor:</strong> Calabresa</li>
26                 </ul>
```



module()




```
1 //Defina um nome para o seu modulo  
2 angular.module("Pizzaria", []);
```



Digite aqui para pesquisar



controller()



```
1 //Chamando o modulo "Pizzaria" e adicionando a ele um controle
2 angular.module("Pizzaria").controller("PizzariaController", function(){
3
4 //Os códigos back-end da nossa aplicação ficam aqui
5
6
7
8 });
```

```
index.html — Desktop\Template HTML-20171030T000228Z-001\Template HTML x pizzas.module.js x pizzas.controller.js x index.html — Dropbox\...\Template Com Angular x
1 <!DOCTYPE html>
2 <html ng-app= "Pizzeria">
3 <head>
4     <meta charset="UTF-8">
5     <title>Pizzeria da SeComp</title>
6
7     <link rel="stylesheet" href="css/style.css">
8
9     <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/
angular.min.js"></script>
10    <script src="js/pizzas.module.js"></script>
11    <script src="js/pizzas.controller.js"></script>
12
13 </head>
14 <body>
15     <div class="container">
16         <font color="#ffffff"><h1>Pizzeria da SeComp</h1></font>
```

```
index.html — Desktop\Template HTML-20171030T000228Z-001\Template HTML x pizzas.module.js x pizzas.controller.js x index.html — Dropbox\...\Template Com Angular x

angular.min.js"></script>
10 <script src="js/pizzas.module.js"></script>
11 <script src="js/pizzas.controller.js"></script>
12
13 </head>
14 <body>
15 <div class="container" ng-controller="_PizzariaController">
16   <font color="#ffffff"><h1>Pizzaria da SeComp</h1></font>
17
18   <div class="lista-pedidos">
19     <div class="pedido com-cartaz">
20       <h1>Mesa: 12</h1>
21       <button class="del" title="Excluir"></button>
22       
23       <ul>
24         <li><strong>Quantidade:</strong> 2</li>
25         <li><strong>Sabor:</strong> Calabresa</li>
26       </ul>
27     </div>
```

ANGULARJS

DATA BINDING



```
1 //Chamando o modulo "Pizzaria" e adicionando a ele um controle
2 angular.module("Pizzaria").controller("PizzariaController", function($scope){
3
4 //Os códigos back-end da nossa aplicação ficam aqui
5
6
7
8 });
```



```
1 //Chamando o modulo "Pizzaria" e adicionando a ele um controle
2 angular.module("Pizzaria").controller("PizzariaController", function($scope){
3 //Os códigos back-end da nossa aplicação ficam aqui
4
5 $scope.titulo = 'Pizzaria da SeComp';
6
7
8
9 });
```




```
angular.min.js"></script>
10 <script src="js/pizzas.module.js"></script>
11 <script src="js/pizzas.controller.js"></script>
12
13 </head>
14 <body>
15 <div class="container" ng-controller="PizzariaController">
16 <font color="#ffffff"><h1>{{ titulo }}</h1></font>
17
18 <div class="lista-pedidos">
19 <div class="pedido com-cartaz">
20 <h1>Mesa: 12</h1>
21 <button class="del" title="Excluir"></button>
22 
23 <ul>
24 <li><strong>Quantidade:</strong> 2</li>
25 <li><strong>Sabor:</strong> Calabresa</li>
26 </ul>
27 </div>
```

LISTar()
ng-repeat



```
1 //Chamando o modulo "Pizzaria" e adicionando a ele um controle
2 angular.module("Pizzaria").controller("PizzariaController", function($scope){
3 //Os códigos back-end da nossa aplicação ficam aqui
4
5 $scope.titulo = 'Pizzaria da SeComp';
6
7 //Definindo uma lista de pedidos contendo 3 pedidos
8 $scope.pedidos = [{id:'123',
9                   mesa: '1',
10                  quantidade: '2',
11                  sabor: 'Calabresa'},
12
13                  {id:'456',
14                   mesa: '2',
15                  quantidade: '4',
16                  sabor: 'Peperoni'},
17
18                  {id:'789',
19                   mesa: '3',
20                  quantidade: '1',
21                  sabor: 'Portuguesa'}]
```



```
script>
10 <script src="js/pizzas.module.js"></script>
11 <script src="js/pizzas.controller.js"></script>
12
13 </head>
14 <body>
15 <div class="container" ng-controller="PizzariaController">
16 <font color="#ffffff"><h1>{{ titulo }}</h1></font>
17
18 <div class="lista-pedidos" ng-repeat="pedido in pedidos">
19 <div class="pedido com-cartaz">
20 <h1>Mesa: {{ pedido.mesa }}</h1>
21 <button class="del" title="Excluir"></button>
22 
23 <ul>
24 <li><strong>Quantidade:</strong> {{ pedido.quantidade }}</li>
25 <li><strong>Sabor:</strong> {{ pedido.sabor }}</li>
26 </ul>
27 </div>
28 </div>
29
30 <hr>
31
```



Pizzaria da SeComp



Mesa: 1

- **Quantidade:** 2
- **Sabor:** Calabresa



Mesa: 2

- **Quantidade:** 4
- **Sabor:** Peperoni



Mesa: 3

- **Quantidade:** 1
- **Sabor:** Portuguesa





cadastrar()
ng-submit com
ng-model



```
19     mesa: '3',
20     quantidade: '1',
21     sabor: 'Portuguesa' }]}
22
23 //Declarando uma funcao para realizar o cadastro dos pedidos
24 $scope.cadastrarPedido = function(){
25
26
27 }
28 });
```




```
index.html — Desktop\Template HTML-20171030T000228Z-001\Template HTML x index.html — Dropbox\...Template Com Angular x pizzas.controller.js — Desktop\...js x pizzas.controller.js — Dropbox\...js x
```

```
20     quantidade: '1',
21     sabor: 'Portuguesa'}}];
22
23 $scope.novoPedido = {};
24 //Declarando uma funcao para realizar o cadastro dos pedidos
25 $scope.cadastrarPedido = function(){
26
27     //Faz uma cópia de um novoPedido para servir como modelo de dados
28     var pedido = angular.copy($scope.novoPedido);
29     //Pega a exata hora para servir como id do pedido
30     pedido.id = Date.now();
31     //Adiciona um novo pedido da lista de pedidos
32     $scope.pedidos.push(pedido);
33
34     //Restaura a variável novoPedido que serviu como modelo de dados
35     $scope.novoPedido = {};
36 }
37
38
39
40 }));
```



Remover()
ng-click



```
32 $scope.pedidos.push(pedido);
33
34 //Restaura a variável novoPedido que serviu como modelo de dados
35 $scope.novoPedido = {};
36 }
37
38 //Declarando uma funcao para remover um pedido da lista de pedidos
39 $scope.removerPedido = function(id){
40
41
42 }
43
44
45
46 }));
```

```
index.html x pizzas.controller.js — Desktop\...js x pizzas.controller.js — Dropbox\...js x
14 <body>
15   <div class="container" ng-controller="PizzariaController">
16     <font color="#ffffff"><h1>{{ titulo }}</h1></font>
17
18     <div class="lista-pedidos" ng-repeat="pedido in pedidos">
19       <div class="pedido com-cartaz">
20         <h1>Mesa: {{ pedido.mesa }}</h1>
21         <button class="del" title="Excluir" ng-click="removePedido(pedido.id)"></button>
22         
23         <ul>
24           <li><strong>Quantidade:</strong> {{ pedido.quantidade }}</li>
25           <li><strong>Sabor:</strong> {{ pedido.sabor }}</li>
26         </ul>
27       </div>
28     </div>
29
30     <hr>
31
32     <form class="form-pedidos" ng-submit="cadastrarPedido()">
33       <h1>Adicionar Pedido</h1>
34       <div class="form-campo pequeno">
35         <label for="titulo">Mesa</label>
36         <input type="number" id="titulo" ng-model = "novoPedido.mesa">
37       </div>
38       <div class="form-campo pequeno">
39         <label for="ano">Quantidade</label>
40         <input type="number" id="ano" ng-model = "novoPedido.quantidade">
```



```
index.html x pizzas.controller.js — Desktop\...js x pizzas.controller.js — Dropbox\...js x
32 $scope.pedidos.push(pedido);
33
34 //Restaura a variável novoPedido que serviu como modelo de dados
35 $scope.novoPedido = {};
36 }
37
38 //Declarando uma função para remover um pedido da lista de pedidos
39 $scope.removerPedido = function(id){
40
41 alert('Estou enviando o ID com Sucesso! [ID]:'+id);
42
43 }
44
45
46
47 });
```

Pizzaria

Essa página diz:

Estou enviando o ID com Sucesso! [ID]:456

OK



Mesa: 1

- **Quantidade:** 2
- **Sabor:** Calabresa



Mesa: 2

- **Quantidade:** 4
- **Sabor:** Peperoni



Mesa: 3

- **Quantidade:** 1
- **Sabor:** Portuguesa



```
index.html x pizzas.controller.js — Desktop\...js x pizzas.controller.js — Dropbox\...js x
31 //Adiciona um novo pedido na lista de pedidos
32 $scope.pedidos.push(pedido);
33
34 //Restaura a variável novoPedido que serviu como modelo de dados
35 $scope.novoPedido = {};
36 }
37
38 //Declarando uma funcao para remover um pedido da lista de pedidos
39 $scope.removePedido = function(id){
40
41     //Definir uma funcao para percorrer a lista de pedidos e remover o pedido que desejamos
42     angular.forEach($scope.pedidos, function(pedido, indice){
43
44         //Checar se o pedido que desejamos excluir existe na lista
45         if(pedido.id == id){
46             //funcao para remover um pedido da lista de pedidos
47             $scope.pedidos.splice(indice,1);
48         }
49     });
50
51 }
52
53
```

Pizzaria da SeComp



Mesa: 1

- **Quantidade:** 2
- **Sabor:** Calabresa



Mesa: 2

- **Quantidade:** 4
- **Sabor:** Peperoni



Mesa: 3

- **Quantidade:** 1
- **Sabor:** Portuguesa



Pizzaria da SeComp



Mesa: 1

- Quantidade: 2
- Sabor: Calabresa



Mesa: 2

- Quantidade: 4
- Sabor: Peperoni

Adicionar Pedido

Mesa

Quantidade



Pizzaria da SeComp



Mesa: 2

- Quantidade: 4
- Sabor: Peperoni

Adicionar Pedido

Mesa

Quantidade

Sabor

Limpar

Salvar



Digite aqui para pesquisar



POR
PTB

22:30
29/10/2017





FIM DO MÓDULO - [1]

PRÓXIMO:
MÓDULO [2] - APPINVENTOR
MIT - ANDROID

