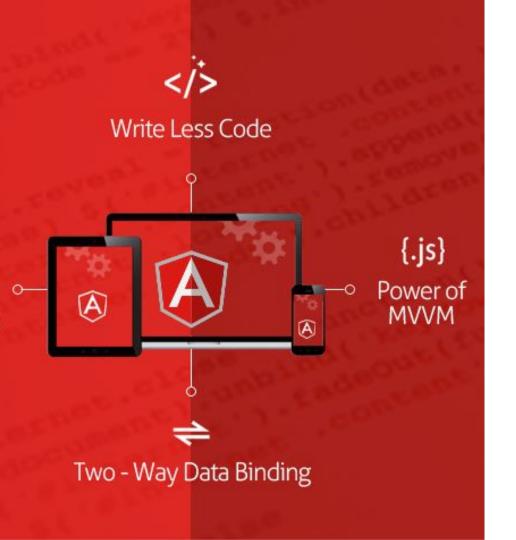


AngularJS



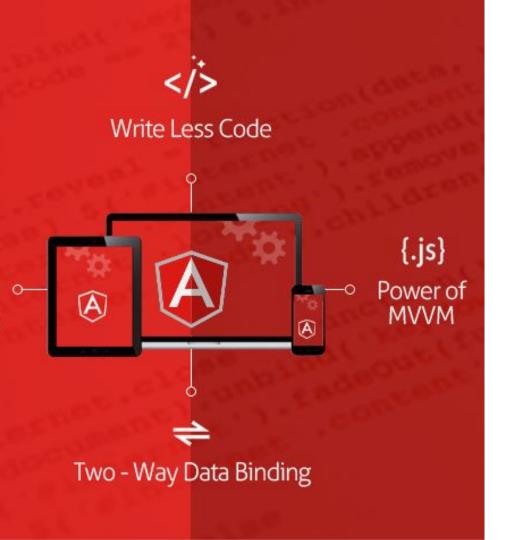
AngularJS é um framework SPA (Single Page Applications), isto significa que você pode criar aplicações web (webapps) com ele. Fazer páginas que funcionam sem precisar de reload e que se comporte verdadeiramente como uma aplicação.



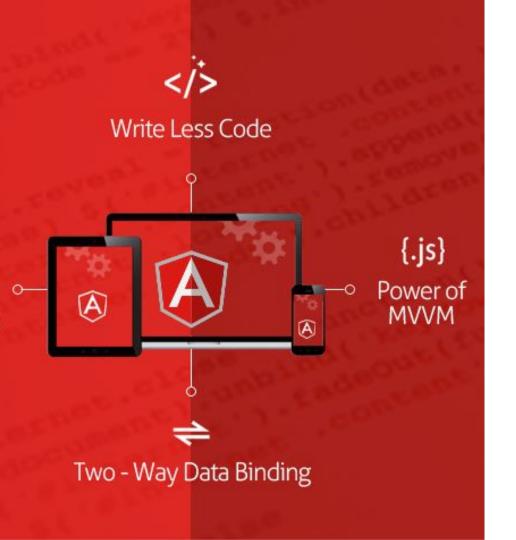
Angular JS é ótimo para estruturar uma aplicação em camadas bem definidas.



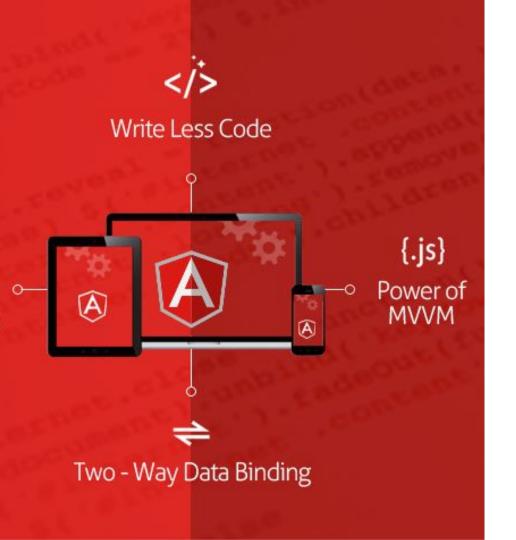
Permite a criação de componentes reusáveis e modulares.



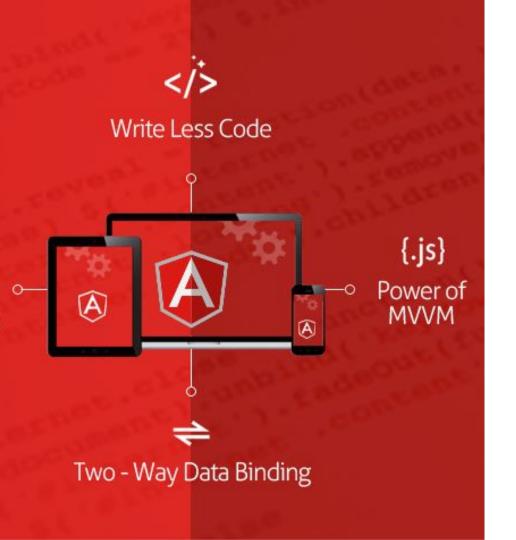
Angular facilita muito a criação de testes e automação.



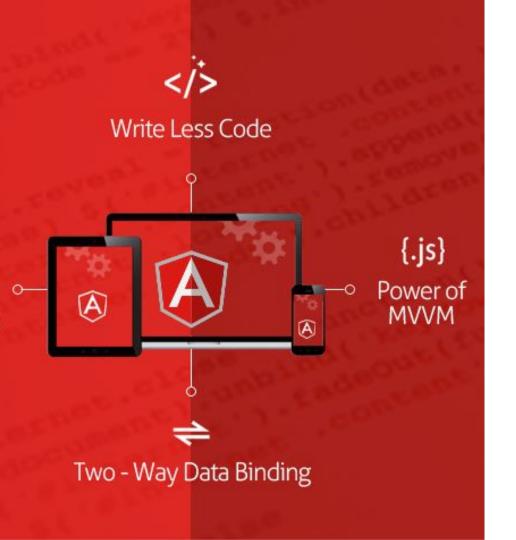
Atualmente o projeto Angular é mantido pelo Google.



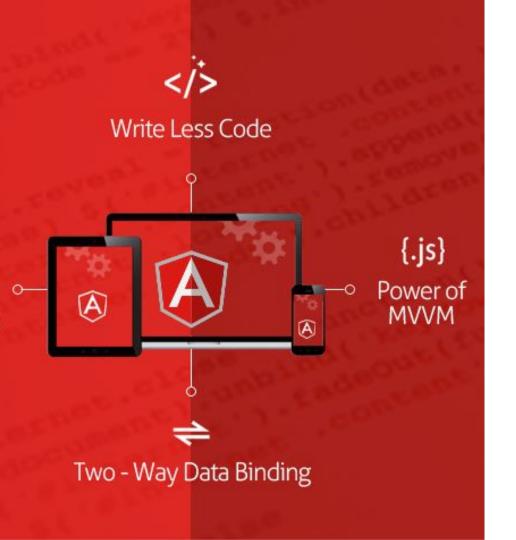
Atualmente o projeto Angular é mantido pelo Google.



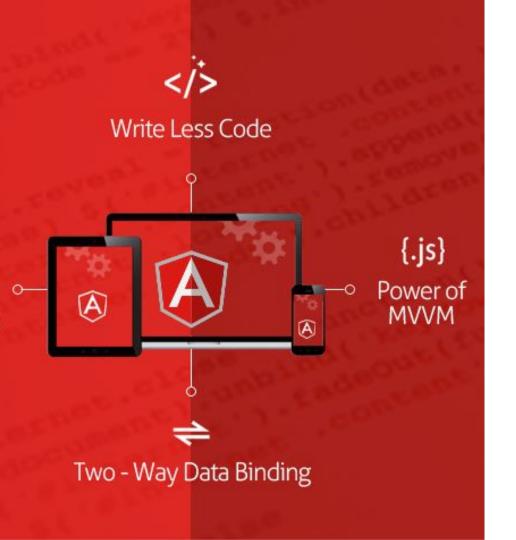
O interesse pelo framework cresce cada vez mais.



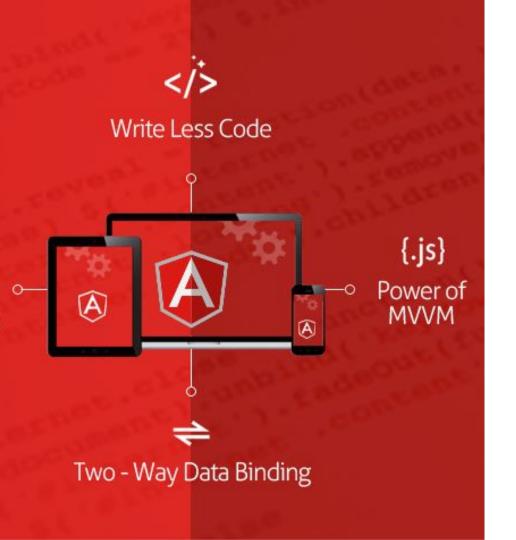
O interesse pelo framework cresce cada vez mais.



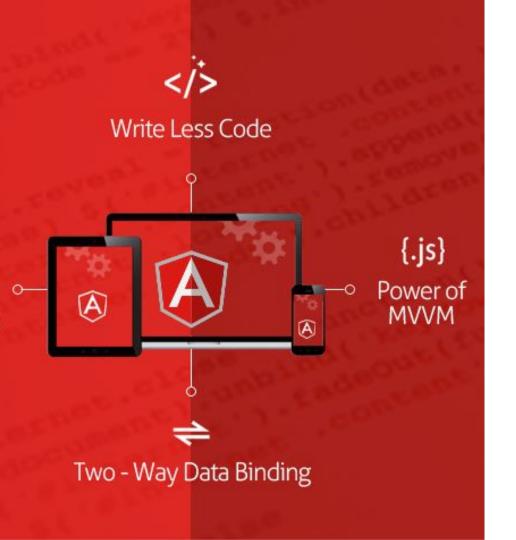
8 motivos para utilizar o AngularJS



8 motivos para utilizar o AngularJS



O AngularJS possui suporte a MVC (Model, View, Controller) e MVVM (Model, View, Viewmodel), assim há uma boa separação de responsabilidades do código escrito pois o MVC guia o desenvolvedor a fazer a coisa do jeito certo.

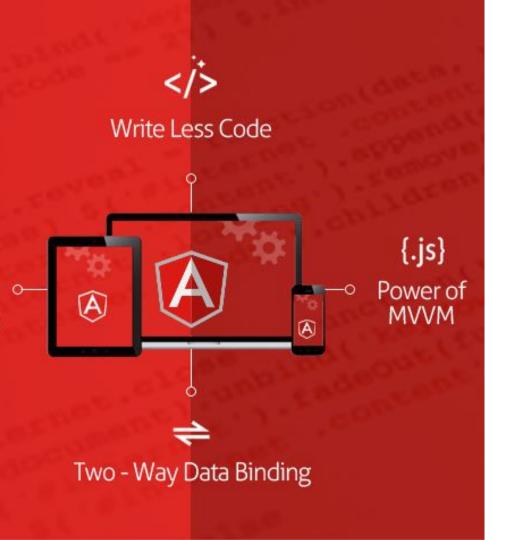


Com o MVVM há uma facilidade enorme de ligar a View com os dados do Model, pois utiliza o modelo de binding bem parecido com o que temos no Knockout



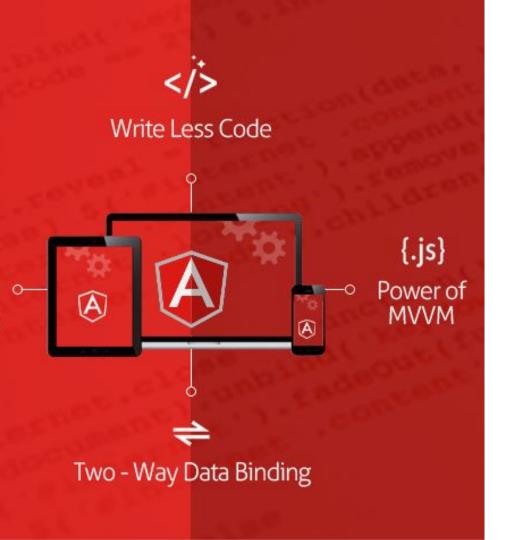
Produtividade

O AngularJS foi pensando para que o desenvolvedor possa escrever o mínimo de código possível, assim desenvolver uma aplicação com ele requer bem menos código do que usando <u>¡Query</u>, por exemplo.



Produtividade

Esta característica se deve muito ao fato do AngularJS ser extremamente orientado a componentes.



Produtividade

Se você precisa de um componente HTML muito provavelmente encontrará algum já pronto para usar. Existe até um site que centraliza esses componentes chamado <u>Angular Modules</u>.



Foco no Importante

A maior parte do código escrito em Angular é relativo a lógica de negócio, graças a estrutura direta e objetiva na qual ele foi concebido.



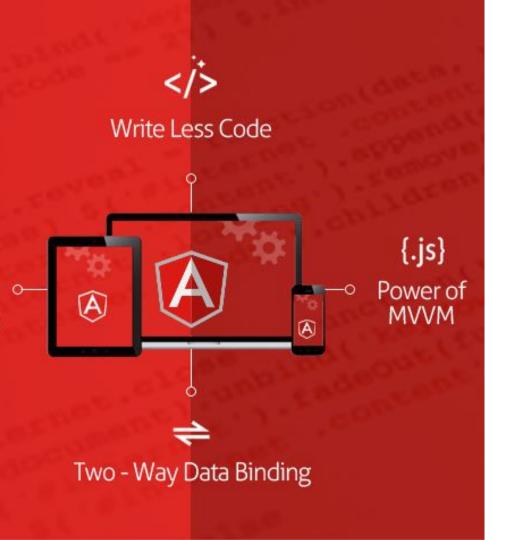
Foco no Importante

Não é necessário criar extensos códigos de configuração, metadados ou contratos dentro da aplicação, apenas o que realmente é necessário dentro do contexto de negócio.



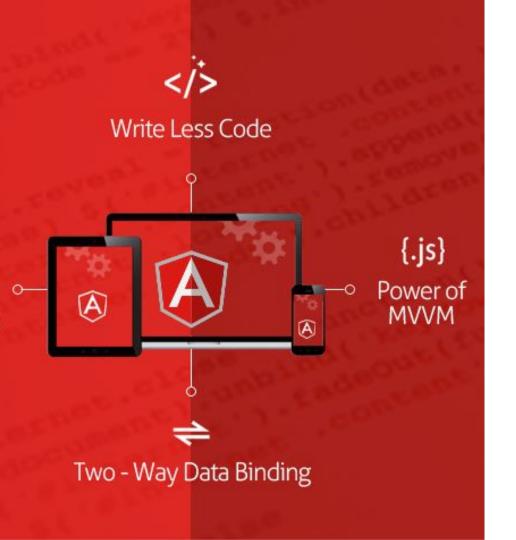
Foco no Importante

Não é necessário criar extensos códigos de configuração, metadados ou contratos dentro da aplicação, apenas o que realmente é necessário dentro do contexto de negócio.

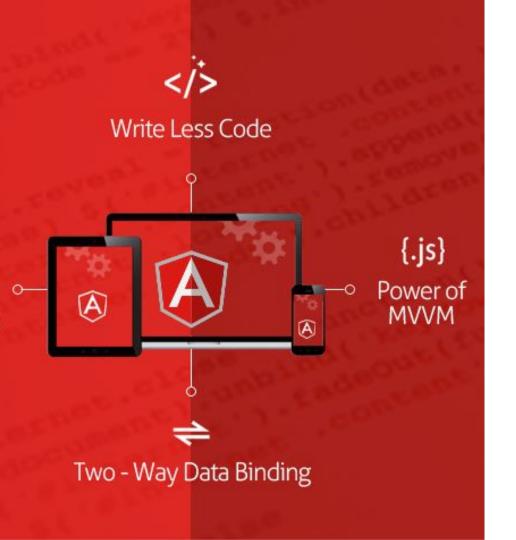


Fácil de aprender e manter

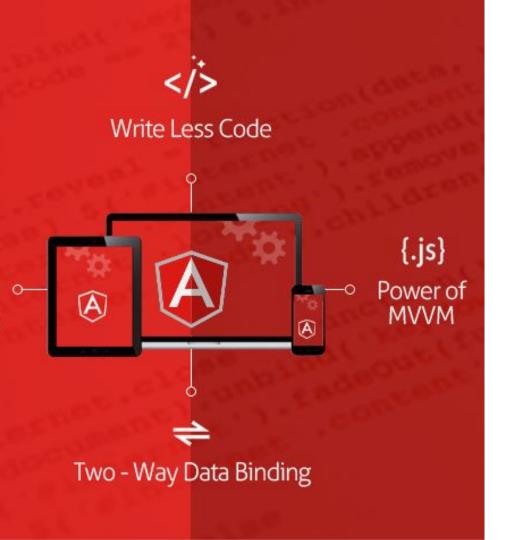
O AngularJS é extremamente declarativo, sendo muito fácil entender o funcionamento das aplicações lendo apenas o HTML.



Isto é uma grande vantagem para manter a produtividade do time alta, quando precisamos de um novo componente criamos (ou reutilizamos) uma diretiva que funciona como uma tag HTML comum.



Isto é uma grande vantagem para manter a produtividade do time alta, quando precisamos de um novo componente criamos (ou reutilizamos) uma diretiva que funciona como uma tag HTML comum.

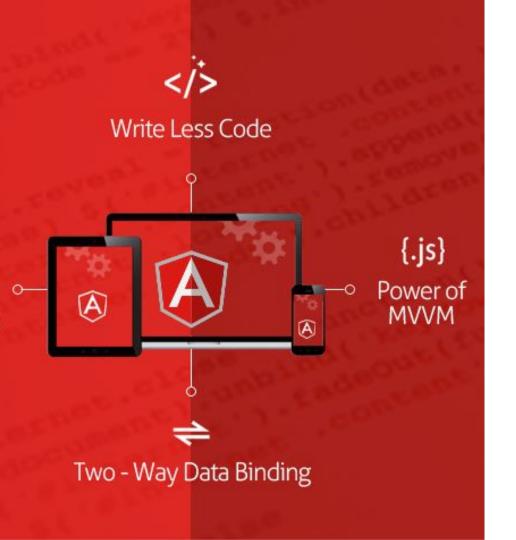


Componentização

Os componentes (diretivas) do
AngularJS podem ser estilizados
facilmente usando HTML e CSS de
forma independente das regras de
negócio ou da lógica de
comportamento da tela.

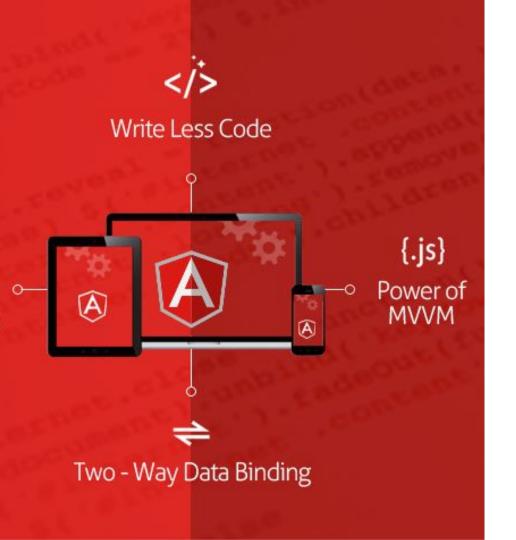


Sendo possível mudar todo o estilo das páginas sem precisar mexer em uma linha sequer de JavaScript. Assim os designers conseguem manter o mesmo modus operandi que já estão acostumados e as mesmas ferramentas onde já dominam.

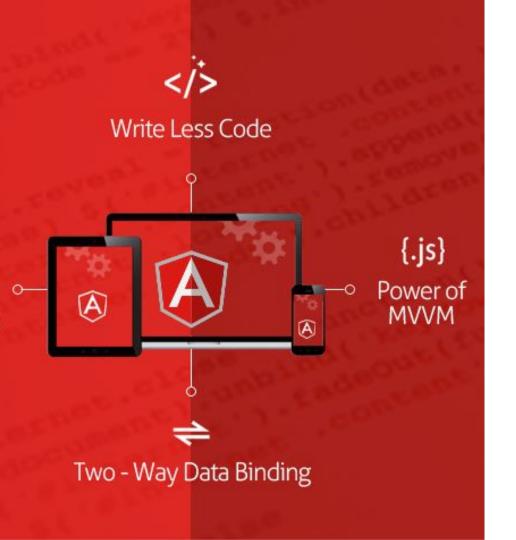


Criação de Frameworks

Os criadores do AngularJS o classificam como *meta-framework*, isto significa que podemos criar nosso próprio framework a partir dele.



E isto é bastante comum hoje em dia, há diversos frameworks baseados em AngularJS, o mais famoso deles é o <u>lonic</u>, que permite a criação de aplicativos mobile híbridos usando AngularJS e <u>Cordova</u>.

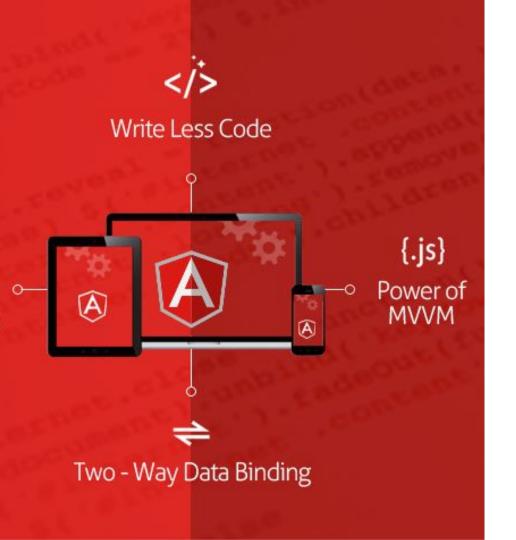


E isto é bastante comum hoje em dia, há diversos frameworks baseados em AngularJS, o mais famoso deles é o <u>lonic</u>, que permite a criação de aplicativos mobile híbridos usando AngularJS e <u>Cordova</u>.

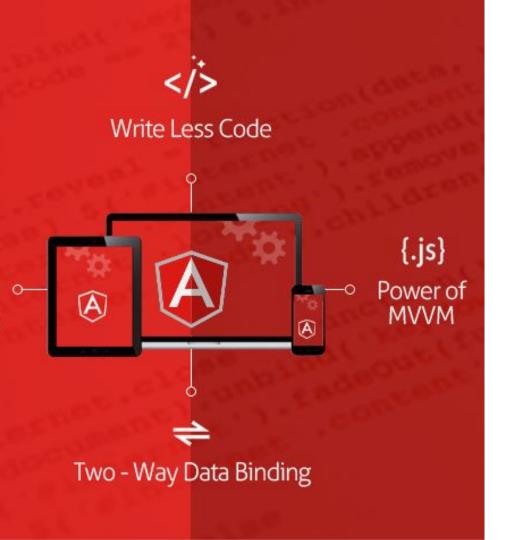


Testes unitários

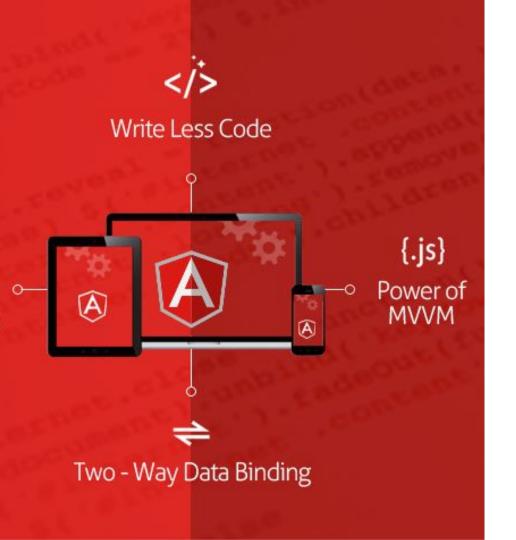
Criar e manter testes unitários para AngularJS é relativamente simples.

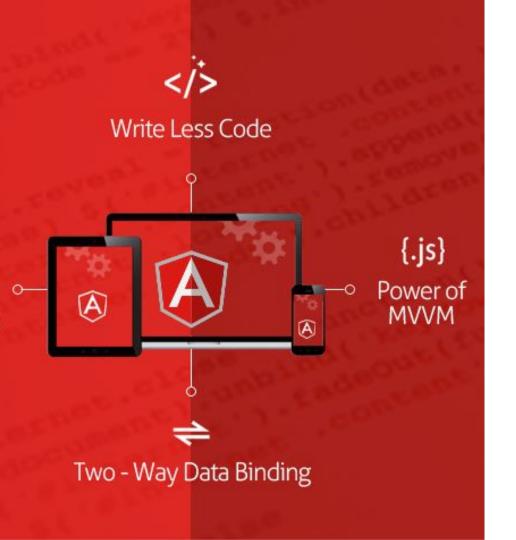


Toda a estrutura é desacoplada e as dependências são injetadas, facilitando a criação de *fakes*, *stubs*, *spies* e *mocks*, melhorando—e muito—todo o processo de teste de controllers, services e directives.

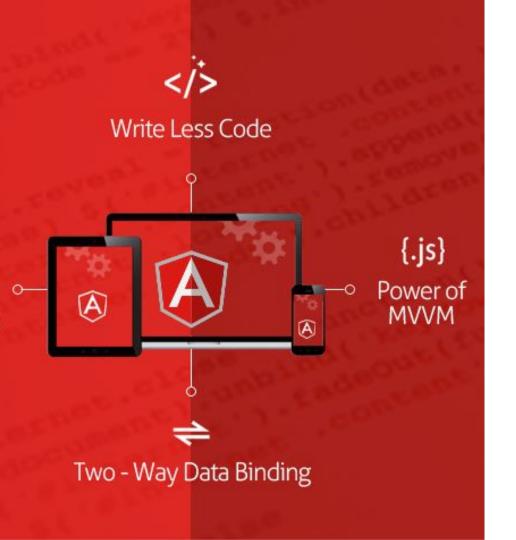


É muito produtivo desenvolver com o AngularJS e a formação de pessoas com este skill não tem sido um problema, pois é possível capacitar os profissionais em no máximo 3 dias dada a facilidade e intuitividade do desenvolvimento com AngularJS.

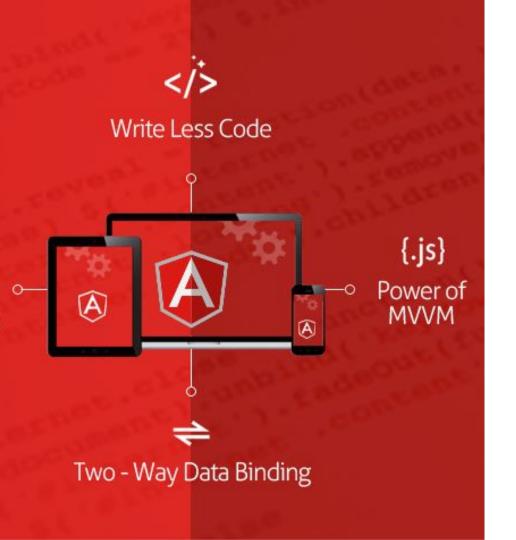




```
<a href="helloWorld">
<head>
```



```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
```



```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
```

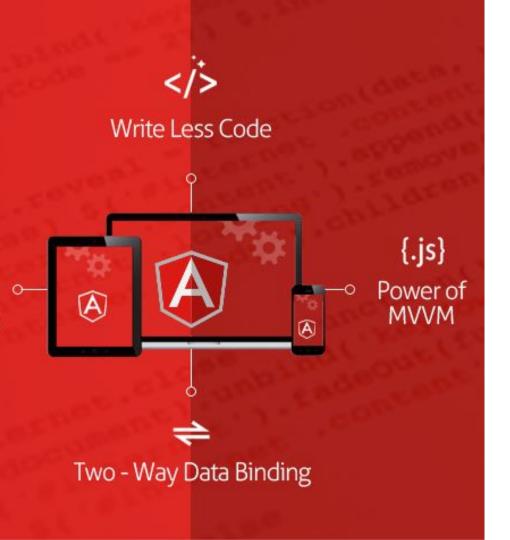
<script src="angular.js"></script>



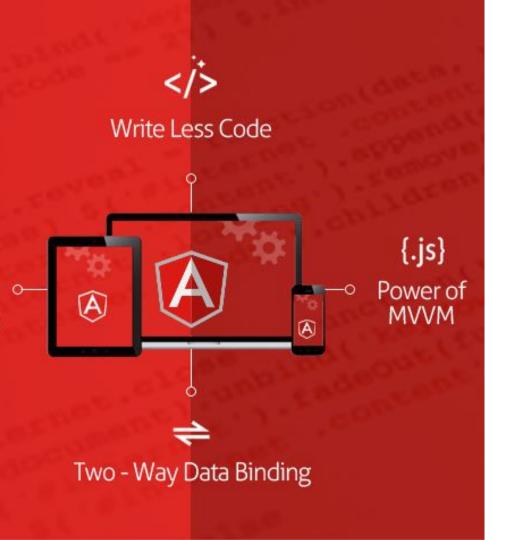
```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
<script src="angular.js"></script>
<script>
```



```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
<script src="angular.js"></script>
<script>
angular.module("helloWorld", []);
```



```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
<script src="angular.js"></script>
<script>
angular.module("helloWorld", []);
angular.module("helloWorld").controll
er("helloWorldCrt", function ($scope ){
$scope.message="hello";
});
</script>
</head>
<body>
<div ng-controller="helloWorldCrt">
{message}}
 </div>
</body>
</html>
```



```
<html ng-app="helloWorld">
<head>
<title>Hello World</title>
<script src="angular.js"></script>
<script>
angular.module("helloWorld", []);
angular.module("helloWorld").controll
er("helloWorldCrt", function ($scope ){
$scope.message="hello";
});
```