



# The Majesty of Vue.js 2

Alex  
Kyriakidis

Kostas  
Maniatis

# The Majesty of Vue.js 2 (Portuguese)

Alex Kyriakidis, Kostas Maniatis and Daniel Schmitz

Esse livro está à venda em <http://leanpub.com/vuejs2-portuguese>

Essa versão foi publicada em 2017-06-19



Esse é um livro [Leanpub](#). A Leanpub dá poderes aos autores e editores a partir do processo de Publicação Lean. [Publicação Lean](#) é a ação de publicar um ebook em desenvolvimento com ferramentas leves e muitas iterações para conseguir feedbacks dos leitores, pivotar até que você tenha o livro ideal e então conseguir tração.

© 2017 Alex Kyriakidis, Kostas Maniatis and Daniel Schmitz

# Conteúdo

<b>Introdução</b>	<b>i</b>
<b>Sobre o Vue.js</b>	<b>ii</b>
Visão Geral	ii
O que as pessoas estão dizendo sobre o Vue.js	ii
<b>Bem vindo</b>	<b>v</b>
Sobre o livro	v
Para quem é este livro	v
Entre em Contato	vi
Tarefa	vi
Código Fonte	vii
Errata	vii
Convenções	vii
<b>I Fundamentos do Vue.js</b>	<b>1</b>
<b>1. Instalar Vue.js</b>	<b>2</b>
1.1    Versão <i>Standalone</i>	2
Download no Site <a href="http://vuejs.org">vuejs.org</a>	2
Incluir do CDN	2
1.2    Download usando NPM	3
1.3    Download usando Bower	3
<b>2. Começando</b>	<b>4</b>
2.1    Olá mundo	4

## CONTEÚDO

2.2	Two-way Binding . . . . .	7
2.3	Comparação com jQuery. . . . .	8
2.4	Tarefa . . . . .	10

# Introdução

# Sobre o Vue.js

## Visão Geral

Vue (pronuncia-se /vju:/, como *view*) é um framework progressivo para criar interfaces. Ao contrário de outros frameworks monolíticos, Vue é projetado a partir do zero para ser adaptado progressivamente. O núcleo da biblioteca está focado na **camada de visualização** e é muito fácil a sua integração com outras bibliotecas ou projetos existentes. Por outro lado, o Vue também é perfeitamente capaz de criar aplicativos sofisticados de uma única página (*Single-Page Applications*) quando usados em combinação com ferramentas modernas e [bibliotecas de suporte](#)<sup>1</sup>.

Se você é um desenvolvedor *front-end* experiente e quer saber como comparar o Vue.js com outras bibliotecas/frameworks, confira o capítulo [Comparação com Outros Frameworks](#).

Se você está interessado em saber mais informações sobre o núcleo do Vue.js, dê uma olhada em [Guia oficial do Vue.js](#)<sup>2</sup>.

## O que as pessoas estão dizendo sobre o Vue.js

*“O Vue.js é o que me faz amar JavaScript. É extremamente fácil e agradável de usar. Ele tem um grande ecossistema de plugins e ferramentas capazes de estender seus serviços básicos. Você pode rapidamente incluí-lo em qualquer projeto, pequeno ou grande, escrever algumas linhas de código e pronto. O Vue.js é rápido, leve e é o futuro do desenvolvimento \*Front-end\*!”*

—Alex Kyriakidis

---

<sup>1</sup><https://github.com/vuejs/awesome-vue#libraries--plugins>

<sup>2</sup><https://br.vuejs.org/v2/guide/>

*“Quando eu comecei a aprender Javascript fiquei empolgado aprendendo uma tonelada de possibilidades, mas quando meu amigo sugeriu aprender Vue.js e eu segui os seus conselhos, as coisas ficaram uma loucura. Enquanto eu lia e assistia tutoriais eu ficava pensando em todas as coisas que tinha feito até agora e como seria mais fácil se eu tivesse investido tempo para aprender Vue mais cedo. Minha opinião é que se você quer fazer seu trabalho de forma rápida, agradável e fácil, Vue é o Framework JS que você precisa.”*

**—Kostas Maniatis**

---

*“Escreva o que te digo: O Vue.js disparará em popularidade em 2016. E isso é bom.”*

**— Jeffrey Way**

---

*“O Vue é o que eu sempre quis em um framework JavaScript. Ele é um framework que cresce na mesma proporção que você como desenvolvedor. Você pode usar um ou outro recurso em uma página ou construir um aplicativo avançado de uma única página com \*Vuex\* e \*Vue Router\*. É realmente o framework JavaScript mais impecável que eu já vi.”*

**— Taylor Otwell**

---

*“Vue.js é o primeiro framework que encontrei que se torna natural de usar em um aplicativo \*server-rendered\*, assim como um SPA completo tem que ser. Se eu só preciso de um pequeno \*widget\* numa única página ou estou construindo um complexo cliente JavaScript, ele sempre dará conta do recado.”*

**— Adam Wathan**

---

*“Vue.js tem sido capaz de fazer um framework que é tão simples de usar quanto fácil de entender. Ele é um sopro de ar fresco num mundo onde os outros estão lutando para ver quem pode fazer o framework mais complexo.”*

*— Eric Barnes*

---

*“O motivo pelo qual eu gosto do Vue.js é porque sou um híbrido de designer e desenvolvedor. Eu dei uma olhada no \*React\*, \*Angular\* e alguns outros, mas a curva de aprendizado e terminologia tem sempre me deixado de fora. O Vue.js é o primeiro framework JS que eu entendo. Além disso, ele não é apenas fácil de aprender para os desenvolvedores JS menos experientes, como eu, mas tenho reparado desenvolvedores avançados de \*Angular\* e \*React\* tomando notas, e gostando do Vue.js. Isso é muito sem precedentes no mundo JS e por esta razão eu comecei o \*Meetup Vue.js\* em Londres.”*

*— Jack Barham*

---

# Bem vindo

## Sobre o livro

Este livro irá guiá-lo rumo ao aprendizado do já difundido Framework JavaScript chamado Vue.js!

Há algum tempo atrás, nós começamos um novo projeto baseado no Laravel e no Vue.js. Depois de ler todo o guia do Vue.js e alguns tutoriais, nós descobrimos que há muito pouco conteúdo disponível a respeito dele na web. Durante o desenvolvimento do nosso projeto, ganhamos muita experiência, então surgiu a ideia de escrever este livro a fim de compartilhar com o mundo nosso conhecimento adquirido. Agora que o Vue.js 2 foi lançado, decidimos que era hora de atualizar nosso livro para publicar uma segunda versão, onde todos os exemplos e conteúdos foram reescritos.

Este livro está escrito em um formato informal, intuitivo e de fácil acompanhamento, onde todos os exemplos são suficientemente detalhados de forma a fornecer a todos uma orientação adequada.

Vamos começar a partir do básico e através de muitos exemplos trataremos dos recursos mais significativos do Vue.js. Até o final deste livro, você será capaz de criar rapidamente aplicações *front-end* e aumentar o desempenho de seus projetos existentes com a integração do Vue.js 2.

## Para quem é este livro

Todos que acompanham o desenvolvimento web moderno, já estão familiarizados com frameworks Javascript/CSS, tais como *JQuery* e *Bootstrap*. Este livro é para qualquer pessoa interessada em aprender um Framework JavaScript leve e simples. Não é necessário nenhum conhecimento excessivo, embora seja bom estar familiarizado com HTML e JavaScript. Se você não sabe qual a diferença entre uma String e um objeto talvez você precise fazer uma pesquisa antes.

Este livro é útil para desenvolvedores que nunca tiveram contato com o Vue.js, bem como para aqueles que já o usam e querem expandir seus conhecimentos. Ele também é útil para desenvolvedores que querem migrar para o Vue.js 2.

## Entre em Contato

Caso você queira entrar em contato conosco sobre o livro, envie-nos comentários ou outros assuntos que gostaria de chamar a nossa atenção. Não hesite em nos contatar.

Name	Email	Twitter
The Majesty of Vue.js	hello@tmvuejs.com	@tmvuejs
Alex Kyriakidis	alex@tmvuejs.com	@hootlex
Kostas Maniatis	kostas@tmvuejs.com	@kostaskafcas
Daniel Schmitz	danieljfa@gmail.com	@daniel_schmitz

**Nota do tradutor:** Para questões em português, entre em contato com [Daniel Schmitz](#) pelo [Telegram](#)<sup>3</sup>.

## Tarefa

**A melhor maneira de aprender a programar é programando**, então preparamos um exercício no final da maioria dos capítulos para você resolver e realmente testar o que aprendeu. Recomendamos muito que você tente resolver o máximo possível a fim de obter um melhor entendimento do Vue.js. Não tenha medo de testar suas ideias, um pouco de esforço te levará longe! Talvez alguns exemplos ou diferentes formas de resolução te darão uma boa ideia. É claro que não somos intransigentes, sugestões e possíveis soluções serão fornecidas!

Você pode começar sua jornada!

---

<sup>3</sup><http://t.me/DanielSchmitz>

## Código Fonte

Você pode encontrar a maioria dos exemplos de código usados no livro no *GitHub*. Você pode acessar o código [aqui](#)<sup>4</sup>.

Se preferir fazer o download, você encontrará um arquivo .zip [aqui](#)<sup>5</sup>.

Isto te poupará de copiar e colar coisas do livro, o que provavelmente seria horrível.

## Errata

Embora todos os cuidados tenham sido tomados para garantir a precisão do nosso conteúdo, erros acontecem. Se você encontrar um erro no livro, ficaremos gratos se puder nos avisar. Ao fazer isso, você ajudará outros leitores e a melhorar as versões subsequentes deste livro. Caso encontre algum erro, por favor envie uma *issue* em nosso repositório [GitHub](#)<sup>6</sup>.

## Convenções

As seguintes convenções de notação são usadas em todo o livro.

Um bloco de código é definido da seguinte forma:

### JavaScript

```
1 function(x, y){  
2     // isto é um comentário  
3 }
```

Um código no texto é exibido da seguinte forma: “Use **.container** para um contêiner de largura fixa responsiva”

---

<sup>4</sup><https://github.com/hoottlex/the-majesty-of-vuejs-2>

<sup>5</sup><https://github.com/hoottlex/the-majesty-of-vuejs-2/archive/master.zip>

<sup>6</sup><https://github.com/hoottlex/the-majesty-of-vuejs-2>

**Termos novos e palavras importantes** são mostrados em negrito.

**Dicas, notas e avisos** são mostrados assim:



## Isto é um aviso

Este elemento indica um aviso ou atenção.



## Isto é uma dica

Este elemento significa uma dica ou sugestão.



## Isto é uma caixa de informações

Algumas informações especiais aqui.



## Isto é uma nota

Uma nota sobre o assunto



## Isto é uma dica

Uma dica sobre o assunto.



## Isto é um Comando de Terminal

Comandos a serem executados no terminal.



## Isto é um texto de comparação

Um pequeno texto comparando coisas relativas ao assunto.



## Isto é um link para o Github.

Os links para o repositório deste livro, onde você pode encontrar os códigos fonte e as potenciais soluções dos exercícios de cada capítulo.

# I Fundamentos do Vue.js

# 1. Instalar Vue.js

Quando se trata da instalação do Vue.js, você tem algumas opções a escolher.

## 1.1 Versão *Standalone*

### Download no Site [vuejs.org](http://vuejs.org)

Para instalar o Vue você pode simplesmente realizar o download e incluí-lo com a tag script. Vue será registrado como uma variável global.

Você pode realizar o download de duas versões do Vue.js:

1. Versão de desenvolvimento: <http://vuejs.org/js/vue.js><sup>1</sup>.
2. Versão de produção: <http://vuejs.org/js/vue.min.js><sup>2</sup>.



Dica: Não use a versão de produção durante o desenvolvimento. Você vai perder todos os *warnings* amigáveis para erros comuns.

### Incluir do CDN

[Vue.js.org](http://vuejs.org)<sup>3</sup> recomenda [unpkg](https://unpkg.com)<sup>4</sup>, o qual reflete a versão mais recente assim que for publicado no npm.

---

<sup>1</sup><http://vuejs.org/js/vue.js>

<sup>2</sup><http://vuejs.org/js/vue.min.js>

<sup>3</sup><https://vuejs.org/v2/guide/installation.html#CDN>

<sup>4</sup><https://unpkg.com/vue/dist/vue.js>

O Vue.js também pode ser encontrado em [jsdelivr](https://cdn.jsdelivr.net/npm/vue@2.0.1)<sup>5</sup> ou [cdnjs](https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.1/vue.min.js)<sup>6</sup>.



Leva algum tempo para sincronizar a versão mais recente, então você deve verificar atualizações frequentemente.

## 1.2 Download usando NPM

NPM é o método de instalação recomendado quando for desenvolver aplicações em larga escala com Vue.js. Este método combina bem com um *CommonJS module bundler* como o [Webpack](#)<sup>7</sup> ou [Browserify](#)<sup>8</sup>.

```
1 # Última versão estável
2 $ npm install vue
3 # Última versão estável + CSP-compliant
4 $ npm install vue@csp
5 # versão dev (diretamente do GitHub):
6 $ npm install vuejs/vue#dev
```

## 1.3 Download usando Bower

```
1 # Última versão estável
2 $ bower install vue
```



Para maiores informações sobre a instalação e atualizações dê uma olhada a mais no [Guia de Instalação do Vue.js](#)<sup>9</sup>

Na maioria dos exemplos estamos incluindo o Vue.js pelo CDN, mas você pode instalar por qualquer método que achar melhor.

<sup>5</sup><https://cdn.jsdelivr.net/npm/vue@2.0.1>

<sup>6</sup><https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.1/vue.min.js>

<sup>7</sup><https://webpack.github.io/>

<sup>8</sup><http://browserify.org/>

<sup>9</sup><http://br.vuejs.org/v2/guide/installation.html>

# 2. Começando

Vamos iniciar com um *tour* rápido pelas funcionalidades de *data binding* do Vue.js.

Vamos fazer uma simples aplicação que nos permitirá inserir uma mensagem e exibi-la na página em tempo real. Ela vai mostrar o poder do *two-way data binding* que o Vue.js possui.

Para criar a nossa aplicação Vue, precisamos configurar algumas coisas que envolvem apenas a criação de uma página HTML.

Neste processo você terá uma ideia de quanto de tempo e esforço é poupado ao usar o Vue.js em relação à outra ferramenta, como a biblioteca jQuery.

## 2.1 Olá mundo

Vamos criar um novo arquivo e inserir um código padrão (chamado de *boilerplate*). Você pode nomeá-lo da forma que quiser, como por exemplo `hello.html`.

```
1 <html>
2 <head>
3   <title>Hello Vue</title>
4 </head>
5 <body>
6   <h1>Greetings your Majesty!</h1>
7 </body>
8 </html>
```

Este é um simples arquivo HTML com uma mensagem de boas vindas.

Agora vamos continuar e fazer a mesma coisa usando o Vue.js. Primeiro, vamos incluir o Vue.js e criar uma nova instância.

```
1 <html>
2 <head>
3     <title>Hello Vue</title>
4 </head>
5 <body>
6     <div id="app">
7         <h1>Greetings your majesty!</h1>
8     </div>
9 </body>
10 <script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.1/vue.mi\
11 n.js"></script>
12 <script>
13     new Vue({
14         el: '#app',
15     })
16 </script>
17 </html>
```

Para iniciarmos, nós incluímos o Vue.js do [cdnjs](https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.1/vue.min.js)<sup>1</sup> e dentro da tag **script** teremos a instância do Vue disponível para uso.

Usamos uma **div** com o parâmetro **id** tendo o valor **#app**, que é o elemento de referência, então o Vue irá saber para onde “olhar”. Tente pensar nisso como um container no qual o Vue irá trabalhar. Vue não irá reconhecer nada fora do seu contêiner. Use a opção **el** para indicar qual elemento será observado pelo Vue.

Agora nós iremos atribuir a mensagem que queremos exibir a uma variável dentro do objeto chamado de **data**. Em seguida, passaremos o objeto **data** como uma propriedade do construtor do Vue.

---

<sup>1</sup><https://cdnjs.cloudflare.com/ajax/libs/vue/2.0.1/vue.min.js>

```
1  var data = {  
2      message: 'Greetings your majesty!'  
3  };  
4  new Vue({  
5      el: '#app',  
6      data: data  
7  })
```

Para exibir a mensagem na página, precisamos apenas envolver a variável `message` em colchetes duplos.

Então, o que estiver dentro da mensagem, aparecerá automaticamente na tag `h1`.

```
1 <div id="app">  
2     <h1>{{ message }}</h1>  
3 </div>
```

*É simples assim!*

Outra maneira de definir uma mensagem em uma variável é fazer isso diretamente dentro do construtor Vue, no objeto `data`.

```
1 new Vue({  
2     el: '#app',  
3     data: {  
4         message: 'Greetings your Majesty!'  
5     }  
6 });
```

Ambos os caminhos têm o mesmo resultado, então você pode escolher livremente a sintaxe que preferir.



## Informação

Os colchetes duplos não são código HTML, são scripts, qualquer coisa dentro dele é chamado de *binding expression*. O Javascript irá “executar” estas expressões. A expressão `{{ message }}` irá trazer o valor da variável. Este código `{{1+2}}` irá mostrar o número 3.

## 2.2 Two-way Binding

O legal do Vue é que ele torna tudo mais fácil. Assumindo que precisamos mudar a mensagem em um campo de texto, como poderíamos fazer isso de forma bem fácil?

No exemplo a seguir usamos `v-model`, uma diretiva do Vue (iremos falar sobre diretivas em um próximo capítulo). Então usamos o *two-way data binding* para dinamicamente alterar o valor da mensagem enquanto o usuário altera a mensagem dentro da caixa de texto. Os dados são sincronizados à medida que o usuário digita o texto.

```
1 <div id="app">
2   <h1>{{ message }}</h1>
3   <input v-model="message">
4 </div>
```

```
1 new Vue({
2   el: '#app',
3   data: {
4     message: 'Greetings your Majesty!'
5   }
6 })
```

**É isso aí!** Agora a nossa mensagem e a caixa de texto estão ligadas. Usando `v-model` dentro da tag `input` dizemos ao Vue qual variável deverá estar ligada a qual `input`, neste caso `message`.



### Two-way data binding

O termo *Two-way data binding* significa que se você alterar o valor do modelo no formulário, tudo será atualizado.

## 2.3 Comparação com jQuery.

Provavelmente, todos vocês tem um pouco de experiência com jQuery. Se não, tudo bem, o uso de jQuery neste livro é mínimo. Quando nós o usamos, é apenas para demonstrar como as coisas podem ser feitas com o Vue ao invés do jQuery e nos certificaremos que tudo será entendível.

De qualquer forma, a fim de entender melhor como o *data-binding* ajuda a criar aplicações, tire um tempo e pense em como seria o exemplo anterior usando jQuery. Você provavelmente iria criar a caixa de texto e dar um valor a um **id** ou **class**, então você poderia obtê-la e modificá-la.

Depois disso, você deveria chamar uma função que muda o elemento para combinar com o valor de entrada, sempre que o evento *keyup* for executado. É assim que funciona, sério!

Seu código se pareceria mais ou menos com o código a seguir.

```
1  <html>
2  <head>
3      <title>Hello Vue</title>
4  </head>
5  <body>
6  <div id="app">
7      <h1>Greetings your Majesty!</h1>
8      <input id="message">
9  </div>
10 </body>
11 <script src="https://code.jquery.com/jquery-2.1.4.min.js"></script>
12 <script type="text/javascript">
13     $('#message').on('keyup', function(){
14         var message = $('#message').val();
15         $('h1').text(message);
16     })
17 </script>
18 </html>
```

Este é um exemplo simples de comparação e, como você pode ver, o Vue parece ser bem mais bonito, mais rápido de escrever e mais fácil de entender.

Claro que, o jQuery é uma poderosa biblioteca para manipular o *Documento Object Model* (DOM), mas tudo tem os seus prós e contras.



## Código fonte

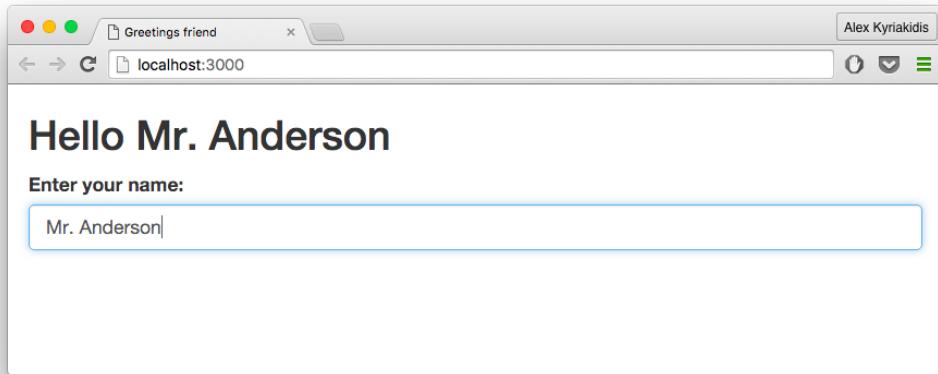
Você pode encontrar estes exemplos no [GitHub](#)<sup>2</sup>.

---

<sup>2</sup><https://github.com/hootlex/the-majesty-of-vuejs-2/blob/master/codes/chapter2.html>

## 2.4 Tarefa

Um bom e super exercício de introdução é criar um arquivo com uma mensagem de *Hello* no cabeçalho com a variável `{{name}}`. Adicione uma caixa de texto e ligue-a a variável `name`. Como pode-se imaginar, o título será alterado instantaneamente enquanto o usuário digita algo. Boa sorte e divirta-se!



Exemplo



### Nota

O exemplo exibido faz uso do *Bootstrap*. Se você não é familiar ao que o *Bootstrap* faz, pode ignorar por enquanto, ele será visto em um [próximo capítulo](#).



### Solução sugerida

Você pode encontrar uma solução sugerida deste exercício [aqui](#)<sup>3</sup>.

<sup>3</sup><https://github.com/hootlex/the-majesty-of-vuejs-2/blob/master/homework/chapter2.html>