

## **Módulo 1: Introdução ao Vue.js**

- O que é Vue.js?
- Por que usar Vue.js?
- Instalação e configuração básica do Vue.js 2.7.14
- Criando seu primeiro aplicativo Vue.js
- Estrutura básica de um aplicativo Vue.js

## O que é Vue.js?

**VueJS** é uma estrutura JavaScript progressiva usada para desenvolver interfaces web interativas. O foco está mais na parte de visualização, que é o front-end. É muito fácil integrar com outros projetos e bibliotecas. A instalação do VueJS é bastante simples e os iniciantes podem facilmente entender e começar a construir suas próprias interfaces de usuário.

- Vue.js é um framework progressivo JavaScript para construção de interfaces de usuário.
- Ele é projetado desde o início para ser adotável incrementalmente.
- Vue.js se concentra apenas na camada de visualização, facilitando a integração com outras bibliotecas ou projetos existentes.

## Pré-requisitos

Antes de prosseguir com este tutorial, os alunos devem ter um conhecimento básico de HTML, CSS e JavaScript.

## Por que usar Vue.js?

- Vue.js é fácil de aprender e usar.
- Ele oferece uma documentação clara e abrangente.
- Vue.js possui uma comunidade ativa e crescente.
- É altamente flexível e pode ser usado em projetos de qualquer tamanho.

A instalação do VueJS é muito fácil de começar. Qualquer desenvolvedor pode compreender e construir facilmente interfaces web interativas em questão de tempo. VueJS foi criado por Evan You, um ex-funcionário do Google. A primeira versão do VueJS foi lançada em fevereiro de 2014. Recentemente, atingiu 64.828 estrelas no GitHub, o que o torna muito popular.

## Características

A seguir estão os recursos disponíveis com VueJS.

### Virtual DOM

**VueJS** faz uso de DOM virtual, que também é usado por outros frameworks como **React**, **Ember**, etc. As alterações não são feitas no DOM, em vez disso é criada uma réplica do DOM que está presente na forma de estruturas de dados JavaScript. Sempre que alguma alteração for necessária, ela será feita nas estruturas de dados JavaScript e esta será comparada com a estrutura de dados original. As alterações finais são então atualizadas para o DOM real, que o usuário verá sendo alterado. Isso é bom em termos de otimização, é mais barato e as alterações podem ser feitas mais rapidamente.

## Data Binding

O recurso de vinculação de dados ajuda a manipular ou atribuir valores a atributos HTML, alterar o estilo, atribuir classes com a ajuda da diretiva de vinculação chamada **v-bind** disponível com VueJS.

## Components

Os componentes são um dos recursos importantes do **VueJS** que ajudam a criar elementos customizados, que podem ser reutilizados em HTML.

## Manipulação de eventos( Event Handling)

**v-on** é o atributo adicionado aos elementos DOM para ouvir os eventos no VueJS.

## Animação/Transição(Animation/Transition)

VueJS fornece várias maneiras de aplicar transição a elementos HTML quando eles são adicionados/atualizados ou removidos do DOM. VueJS possui um componente de transição integrado que precisa ser envolvido no elemento para obter o efeito de transição. Podemos facilmente adicionar bibliotecas de animação de terceiros e também adicionar mais interatividade à interface.

## Propriedades computadas(Computed Properties)

Este é um dos recursos importantes do VueJS. Ajuda a ouvir as alterações feitas nos elementos da IU e a realizar os cálculos necessários. Não há necessidade de codificação adicional para isso.

## Templates

VueJS fornece templates baseados em HTML que vinculam o DOM aos dados da instância do Vue. Vue compila os modelos em funções virtuais de renderização do DOM. Podemos fazer uso do template das funções render e para isso temos que substituir o template pela função render.

## Diretivas

VueJS possui diretivas integradas como **v-if**, **v-else**, **v-show**, **v-on**, **v-bind** e **v-model**, que são usadas para executar várias ações no frontend.

## Watchers(Observadores)

Os observadores são aplicados aos dados que mudam. Por exemplo, elementos de entrada de formulário. Aqui, não precisamos adicionar nenhum evento adicional. O Watcher se encarrega de lidar com quaisquer alterações de dados, tornando o código simples e rápido.

## Roteamento(Routing)

A navegação entre as páginas é realizada com a ajuda do **vue-router**.

## Leve

O script VueJS é muito leve e o desempenho também é muito rápido.

## Vue-CLI

O VueJS pode ser instalado na linha de comando usando a interface de linha de comando vue-cli. Ajuda a construir e compilar o projeto facilmente usando **vue-cli**.

## Instalação e configuração básica do Vue.js 2.7.14

Para começar, você pode incluir o Vue.js em seu projeto via CDN ou usando um gerenciador de pacotes como npm ou yarn.

Se estiver usando npm, você pode instalar Vue.js executando o seguinte comando no terminal:

# NPM

```
npm install vue
```

Depois de instalar o Vue.js, você pode incluí-lo em seu projeto HTML da seguinte maneira:

## Script

The screenshot shows the Vue.js v2 guide page. The browser address bar shows `v2.vuejs.org/v2/guide/`. A red box highlights the address bar, and a red arrow points to it. Below the address bar, a yellow banner states: "Vue 2 has reached EOL and is no longer actively maintained. Upgrade to Vue 3 or learn more about Vue 2 EOL." The main content area has a red box around the development version script: `<!-- development version, includes helpful console warnings -->` and `<script src="https://cdn.jsdelivr.net/npm/vue@2/dist/vue.js"></script>`. A red arrow points to this script. The page also includes a sidebar with navigation links and a search bar.

## VUE CLI

E futuramente para aplicações mais complexas utilizamos o **VUE CLI**:

<https://cli.vuejs.org/>

```
npm install --global vue-cli
```



## Vue CLI

Standard Tooling for Vue.js Development

Get Started →

Neste momento não abordaremos este conceito, quando terminarmos os conceitos iniciais, iremos aprender sobre VUE CLI.

Mais sobre a instalação e sobre o vue js em sua documentação

<https://v2.vuejs.org/>

**Agora você está pronto para começar a usar o Vue.js em seu aplicativo!**

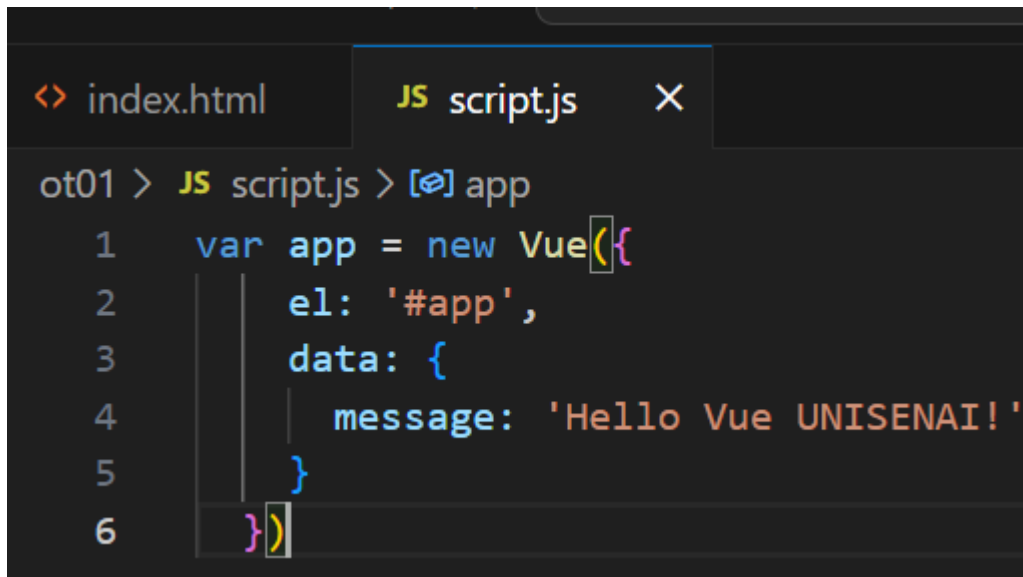
## Criando seu primeiro aplicativo Vue.js

Crie um arquivo HTML básico para o seu aplicativo.

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Aprendendo Vue</title>
7 </head>
8 <body>
9
10  <div id="app">
11    {{ message }}
12  </div>
13
14
15
16  <!-- link de desenvolvimento vuejs -->
17  <script src="https://cdn.jsdelivr.net/npm/vue@2/dist/vue.js"></script>
18  <!-- link do script que tem o vuejs estanciado -->
19  <script src="script.js"></script>
20 </body>
21 </html>
```

Dentro do arquivo HTML, crie uma estrutura básica e inclua um elemento com um ID único para o seu aplicativo Vue.js, por exemplo:

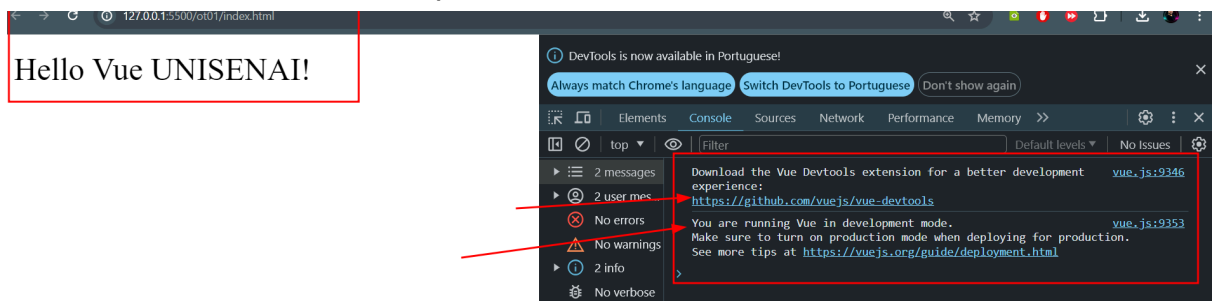
No final do seu arquivo HTML, inclua um link para o script inicializar o Vue.js e associá-lo ao elemento com o ID "app":



```
ot01 > JS script.js > [🔍] app
1  var app = new Vue({
2    el: '#app',
3    data: {
4      message: 'Hello Vue UNISENAI!'
5    }
6  })
```

Agora, quando você abrir o arquivo HTML em um navegador, verá "Olá, Vue!" renderizado na tela.

E também ele dá o link para baixar a extensão para debugar no navegador **vue-devtools**, recomendo que instalem para mais a frente aprendermos a utilizarmos a mesma, e também mostra um console, informando que o **vue** está rodando em modo development.



### Estrutura básica de um aplicativo Vue.js

- Em um aplicativo Vue.js, você geralmente terá um HTML básico para a estrutura da página.
- Os componentes Vue.js podem ser incluídos dentro do HTML para controlar partes específicas da página.
- O Vue.js usa o sistema de reatividade para rastrear as dependências entre os dados e a interface do usuário, garantindo que as atualizações de dados sejam refletidas automaticamente na interface do usuário.

# Atividade do Módulo 1: Criando um Aplicativo de Saudação com Vue.js

Objetivo:

- Aplicar os conceitos aprendidos no Módulo 1 para criar um aplicativo Vue.js simples que exiba uma mensagem de saudação.

Instruções:

Crie um novo arquivo HTML chamado **index.html**.

Dentro do arquivo HTML, inclua o CDN do Vue.js para importar a biblioteca:

# CDN

For prototyping or learning purposes, you can use the latest version with:

```
<script src="https://cdn.jsdelivr.net/npm/vue@2.7.16/dist/vue.js"></script>
```

HTML

Abaixo do CDN do Vue.js, adicione o seguinte código para criar a estrutura básica do aplicativo Vue.js:

```
<div id="app">
  <p>{{ greeting }}</p>
</div>

<script>
  var app = new Vue({
    el: '#app',
    data: {
      greeting: 'Olá, Vue!'
    }
  });
</script>
```



Abra o arquivo `index.html` em seu navegador e verifique se a mensagem "Olá, Vue!" está sendo exibida corretamente.

Agora, sua tarefa é modificar o aplicativo para que ele exiba uma saudação personalizada com base no nome fornecido pelo usuário, como por exemplo:



Olá, Uchôa!!

Teste o aplicativo digitando diferentes nomes de entrada e verificando se a mensagem de saudação é atualizada corretamente.

**Bônus: Estilize o aplicativo com CSS para torná-lo mais atraente visualmente.**

Dicas:

- Utilize interpolação (`{{ }}`) para exibir o valor do nome fornecido pelo usuário na mensagem de saudação.

Conclusão:

Ao concluir esta atividade, você terá praticado a criação de um aplicativo Vue.js simples, bem como a manipulação de dados e atualização dinâmica da interface do usuário. Isso ajudará a consolidar os conceitos aprendidos no Módulo 1.