

## 1. O que é o Bootstrap e quais são suas principais vantagens ao ser utilizado no desenvolvimento web com Vue.js?

**Bootstrap** é um framework CSS que fornece estilos prontos e componentes responsivos para criação de interfaces web.

### Principais vantagens com Vue.js:

- Estilização rápida e padronizada.
- Sistema de grid responsivo para layout.
- Compatível com bibliotecas como BootstrapVue.
- Reduz tempo de desenvolvimento e facilita a criação de interfaces modernas.

## 2. Quais são as principais formas de integrar o Bootstrap a um projeto Vue.js? Explique as diferenças entre utilizar CDN e instalar via NPM.

As duas formas principais são:

- **Via CDN:** adiciona os links do Bootstrap diretamente no HTML.  
**Vantagem:** rápido e simples para testes ou projetos pequenos.  
**Limitação:** menos controle e customização.
- **Via NPM:** instala o Bootstrap como dependência do projeto.  
**Vantagem:** permite importações diretas no código, uso com SCSS e integração com bibliotecas como BootstrapVue.  
**Melhor para:** projetos maiores e com build personalizado.

## 3. Explique o conceito de classes utilitárias no Bootstrap. Como elas ajudam a estilizar elementos sem a necessidade de CSS personalizado?

As **classes utilitárias** do Bootstrap são classes pré-definidas que aplicam estilos diretamente aos elementos, como margens, cores, tamanhos, alinhamentos e espaçamentos.

**Exemplo de uso:** mt-3 (margem superior), text-center (texto centralizado), bg-primary (fundo azul padrão).

Essas classes evitam a criação de CSS manual, agilizando o desenvolvimento e mantendo o código mais limpo e padronizado.

#### **4. Descreva o funcionamento do Grid System do Bootstrap. Como ele facilita a criação de layouts responsivos?**

O **Grid System** do Bootstrap é baseado em **12 colunas flexíveis** que se adaptam ao tamanho da tela. Ele usa classes como `.row` e `.col` para criar layouts alinhados e responsivos.

**Funciona com breakpoints**, como `col-sm-6`, `col-md-4`, ajustando automaticamente o layout em diferentes dispositivos.

**Facilita a criação de layouts responsivos** porque permite dividir o espaço visual de forma simples e previsível, sem escrever CSS complexo.

#### **5. Qual é a importância das classes `col-`, `row`, `container` e `container-fluid` no sistema de grid do Bootstrap?**

Essas classes são fundamentais para estruturar o layout com o sistema de grid:

- **container**: define uma largura fixa e centralizada com base no tamanho da tela.
- **container-fluid**: ocupa 100% da largura da tela, útil para layouts que precisam de espaço total.
- **row**: cria uma linha horizontal para agrupar colunas.
- **col-**: define colunas dentro da linha, controlando largura e responsividade.

Juntas, essas classes permitem criar layouts bem organizados, responsivos e consistentes.

#### **6. Explique a diferença entre os componentes do Bootstrap Navbar, Card e Modal. Quando utilizar cada um deles?**

- **Navbar:** é uma barra de navegação que geralmente fica no topo da página.  
**Usar quando:** precisa de menus, links, login ou pesquisa em uma navegação principal.
- **Card:** é um contêiner com bordas, usado para exibir conteúdo organizado (texto, imagem, botões).  
**Usar quando:** deseja apresentar blocos de informação como perfis, produtos ou artigos.
- **Modal:** é uma janela flutuante sobre a interface principal.  
**Usar quando:** precisa exibir mensagens, formulários ou confirmações sem sair da página atual.

7. O que são breakpoints no Bootstrap e como eles afetam o comportamento dos layouts responsivos?

Breakpoints são larguras específicas da tela onde o layout do site se adapta automaticamente.

No Bootstrap, eles definem pontos de corte para aplicar estilos diferentes em tamanhos como:

sm ( $\geq 576\text{px}$ )

md ( $\geq 768\text{px}$ )

lg ( $\geq 992\text{px}$ )

xl ( $\geq 1200\text{px}$ )

xxl ( $\geq 1400\text{px}$ )

Eles afetam o comportamento dos layouts ao permitir que colunas, elementos e classes utilitárias mudem conforme o dispositivo, garantindo uma experiência responsiva e otimizada.

## 8. O Bootstrap permite personalizar seus estilos utilizando SCSS. Qual a vantagem de utilizar SCSS ao invés de CSS puro para modificar estilos no Bootstrap?

A principal vantagem de usar **SCSS** é a **flexibilidade e organização**. Com SCSS, é possível:

- Usar **variáveis** para cores, tamanhos e fontes.
- Aproveitar **mixin** e **funções** reutilizáveis.
- **Sobrescrever estilos do Bootstrap** sem alterar os arquivos originais.
- Criar código mais limpo, modular e fácil de manter.

Isso facilita a personalização e acelera o desenvolvimento de interfaces consistentes.

## 9. Qual a importância dos modais em uma aplicação Vue.js? Cite um exemplo prático de uso em um sistema web

Os **modais** são importantes em aplicações Vue.js porque permitem exibir informações ou interações sem sair da página atual, mantendo o fluxo do usuário.

**Exemplo prático:** em um sistema de gestão, um modal pode ser usado para **editar dados de um usuário** sem precisar navegar para outra tela, tornando a experiência mais rápida e intuitiva.

## 10. Como o Vue.js pode ser combinado com o Bootstrap para criar interfaces mais dinâmicas e interativas?

Vue.js pode ser usado com Bootstrap para aplicar **estilo visual com componentes prontos** (como botões e cards) e controlar a **lógica reativa e interativa** (como formulários, modais e filtros).

A combinação permite criar **interfaces modernas, responsivas e interativas**, onde o Bootstrap cuida do visual e o Vue.js da atualização automática dos dados e eventos do usuário.