



## Revisão 3

Importamos scripts de terceiros baixados através do npm e nossos scripts da mesma forma que aprendemos a importar arquivos `css`. No entanto, se um desses scripts declarar variáveis globais que são acessadas por outros scripts importados é necessário um passo a mais. Esse foi o caso da importação de `bootstrap.js`. Ele depende do jQuery que na verdade é uma variável que mora no escopo global como `$` ou `jQuery`. Nesse caso, além de importar o `jquery` através do npm, substituímos sua importação via `import` pelo carregamento através do plugin `webpack.ProvidePlugin`. Neste plugin indicamos o caminho do módulo que desejamos importar e seu apelido, ou seja, como ele será referenciado na aplicação. Veja um trecho da configuração que fizemos:

```
plugins: [  
  new webpack.ProvidePlugin({  
    $: 'jquery/dist/jquery.js',  
    jQuery: 'jquery/dist/jquery.js'  
  })  
]
```

[COPIAR CÓDIGO](#)

Dessa forma, quando o script `bootstrap.js` solicitar o jQuery ele já estará disponível, evitando assim qualquer erro.

Por fim, aprendemos que é dentro da pasta `static`, pasta que não foi criada por padrão pelo Vue CLI, que ficam todas as imagens da nossa aplicação. Para acessá-las, basta usarmos o endereço `/static/nome-do-arquivo.extensao`. Só não podemos esquecer de enviar essa pasta quando formos fazer o deploy da aplicação.

