

# Migrando da v3 para v4

Sim, v4 foi lançada!

Procurando pelos documentos da v3? [Encontre-os aqui](#).

*Este documento está em constante evolução. Você atualizou seu site e encontrou algo que não é abordado aqui?*  
[Adicione suas alterações no GitHub](#).

## Introdução

Esta é uma referência para atualizar seu site de Material-UI v3 para v4. Embora haja muita coisa coberta por aqui, você provavelmente não precisará fazer tudo no seu site. Faremos o nosso melhor para manter as coisas fáceis de seguir e tão sequenciais quanto possível, para que você possa rapidamente agitar na v4!

## Por que você deve migrar

Esta página de documentação cobre o **como** migrar da v3 para a v4. O **porque** é abordado na [postagem no blog do Medium](#).

## Atualizando suas dependências

A primeira coisa que você precisa fazer é atualizar suas dependências.

### Atualize a versão do Material-UI

Você precisa atualizar seu `package.json` para usar a versão mais recente do Material-UI.

```
"dependencies": {  
  "@material-ui/core": "^4.0.0"  
}
```

Ou execute

```
npm install @material-ui/core  
  
ou  
  
yarn add @material-ui/core
```

### Atualize a versão do React

A versão mínima necessária do React foi incrementada de `react@^16.3.0` para `react@^16.8.0`. Isso nos permite a utilizar [Hooks](#) (não usamos mais a API class).

### Atualize a versão do Material-UI Styles

Se você estava usando anteriormente `@material-ui/styles` com a versão 3, precisa atualizar o `package.json` para usar a última versão de Material-UI Styles.

```
"dependencies": {
  "@material-ui/styles": "^4.0.0"
}
```

Ou execute

```
npm install @material-ui/styles

ou

yarn add @material-ui/styles
```

## Tratamento de alterações recentes

### Núcleo

- Cada componente encaminha seu ref. Isso é implementado usando `React.forwardRef()`. Isso afeta a árvore interna do componente e o nome de exibição, portanto, pode quebrar testes superficiais ou instantâneos. `innerRef` não retornará mais um ref à instância (ou nada se o componente interno for um componente de função), mas uma referência ao seu componente raiz. Os documentos da API correspondentes listam o componente raiz.

### Estilos

- ⚠ Material-UI depende do JSS v10. JSS v10 não é compatível com a v9. Certifique-se de que o JSS v9 não esteja instalado em seu ambiente. (Remover `react-jss` do seu `package.json` pode ajudar). O componente `StylesProvider` substitui o componente `JssProvider`.
- Remova a primeira opção de argumento do `withTheme()`. (O primeiro argumento é um espaço reservado para uma opção futura potencial que nunca existiu.)

It matches the [emotion API](#) and the [styled-components API](#).

```
-const DeepChild = withTheme()(DeepChildRaw);
+const DeepChild = withTheme(DeepChildRaw);
```

- Renomeie `convertHexToRGB` para `hexToRgb`.

```
-import { convertHexToRgb } from '@material-ui/core/styles/colorManipulator';
+import { hexToRgb } from '@material-ui/core/styles';
```

- Escopo da [keyframes API](#). Você deve aplicar as seguintes alterações na sua base de código. Ele ajuda a isolar a lógica da animação:

```
rippleVisible: {
  opacity: 0.3,
  - animation: 'mui-ripple-enter 100ms cubic-bezier(0.4, 0, 0.2, 1)',
  + animation: '$mui-ripple-enter 100ms cubic-bezier(0.4, 0, 0.2, 1)',
},
'@keyframes mui-ripple-enter': {
```

```

    '0%': {
      opacity: 0.1,
    },
    '100%': {
      opacity: 0.3,
    },
  },
},

```

## Tema

- O método `theme.palette.augmentColor()` não produz mais um efeito colateral em sua cor de entrada. Para usá-lo corretamente, agora você precisa usar o valor retornado.

```

-const background = { main: color };
-theme.palette.augmentColor(background);
+const background = theme.palette.augmentColor({ main: color });

console.log({ background });

```

- Você pode remover com segurança a opção `useNextVariants` do tema:

```

typography: {
  - useNextVariants: true,
},

```

- `theme.spacing.unit` está com o uso obsoleto, você pode usar a nova API:

```

label: {
  [theme.breakpoints.up('sm')]: {
    - paddingTop: theme.spacing.unit * 12,
    + paddingTop: theme.spacing(12),
  },
}

```

Dica: você pode fornecer mais de 1 argumento: `theme.spacing(1, 2) // = '8px 16px' *`.

Você pode usar o [auxiliar de migração] (<https://github.com/mui/material-ui/tree/master/packages/material-ui-codemod/README.md#theme-spacing-api>) em seu projeto para tornar isso mais suave.

## Leiaute

- [Grid] Para suportar valores de espaçamento arbitrários e para remover a necessidade de contar mentalmente por 8, estamos mudando a API de espaçamento:

```

/**
 * Define o espaço entre o tipo` componente do item.
 * Só pode ser usado em um componente do tipo `container`.
 */

```

```
- spacing: PropTypes.oneOf([0, 8, 16, 24, 32, 40]),  
+ spacing: PropTypes.oneOf([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]),
```

Indo além, você pode usar o tema para implementar [uma função de transformação de espaçamento de grade customizada](#).

- Você pode remover com segurança a próxima variante da criação de temas:

```
-import Container from '@material-ui/lab/Container';  
+import Container from '@material-ui/core/Container';
```

## TypeScript

### Tipo value

Tipo da propriedade `value` normalizado para os componentes de entrada utilizarem `unknown`. Isso afeta `InputBase`, `NativeSelect`, `OutlinedInput`, `Radio`, `RadioGroup`, `Select`, `SelectInput`, `Switch`, `TextArea`, and `TextField`.

```
function MySelect({ children }) {  
- const handleChange = (event: any, value: string) => {  
+ const handleChange = (event: any, value: unknown) => {  
    // valor manipulado  
  };  
  
  return <Select onChange={handleChange}>{children}</Select>  
}
```

Esta alteração é explicada em mais detalhes no [guia TypeScript](#)

### Button

- [Button] Remova as variantes descontinuadas (flat, raised e fab):

```
-<Button variant="raised" />  
+<Button variant="contained" />
```

```
-<Button variant="flat" />  
+<Button variant="text" />
```

```
-import Button from '@material-ui/core/Button';  
-<Button variant="fab" />  
+import Fab from '@material-ui/core/Fab';  
+<Fab />
```

```
-import Button from '@material-ui/core/Button';  
-<Button variant="extendedFab" />
```

```
+import Fab from '@material-ui/core/Fab';  
+<Fab variant="extended" />
```

- [ButtonBase] O componente passado para a propriedade `component` precisa ser capaz de lidar com ref. O [guia de composição](#) explica a estratégia de migração.

Isso também se aplica a `BottomNavigationAction`, `Button`, `CardActionArea`, `Checkbox`, `ExpansionPanelSummary`, `Fab`, `IconButton`, `MenuItem`, `Radio`, `StepButton`, `Tab`, `TableSortLabel` bem como `ListItem` se a propriedade `button` for `true`.

## Cartão

- [CardActions] Renomeie a propriedade `disableActionSpacing` para `disableSpacing`.
- [CardActions] Remova a classe CSS `disableActionSpacing`.
- [CardActions] Renomeie a classe CSS `action` para `spacing`.

## ClickAwayListener

- [ClickAwayListener] Esconda propriedades react-event-listener.

## Diálogo

- [DialogActions] Renomeie a propriedade `disableActionSpacing` para `disableSpacing`.
- [DialogActions] Renomeie a classe CSS `action` para `spacing`.
- [DialogContentText] Use a variante de tipografia `body1` em vez de `subtitle1`.
- [Dialog] O elemento filho precisa ser capaz de lidar com ref. O [guia de composição](#) explica a estratégia de migração.

## Divisor

- [Divider] Remova a propriedade obsoleta `inset`.

```
-<Divider inset />  
+<Divider variant="inset" />
```

## Painel de expansão

- [ExpansionPanelActions] Renomeie a classe CSS `action` para `spacing`.
- [ExpansionPanel] Aumente a especificidade CSS das regras de estilo `disabled` e `expanded`.
- [ExpansionPanel] Renomeie a propriedade `CollapseProps` para `TransitionProps`.

## Lista

- [List] Refaça a lista de componentes para coincidir com a especificação:
  - O componente `ListItemAvatar` é necessário ao usar um avatar.
  - O componente `ListItemIcon` é necessário ao usar uma caixa de seleção à esquerda.
  - A propriedade `edge` deve ser definida para botões de ícone.
- [List] `dense` não reduz mais o espaçamento superior e inferior do elemento `List`.
- [ListItem] Aumente a especificidade CSS das regras de estilo `disabled` e `focusVisible`.

## Menu

- [Menuitem] Remova a altura fixa do Menuitem. O preenchimento e a altura da linha são usados pelo navegador para calcular a altura.

## Modal

- [Modal] O elemento filho precisa ser capaz de lidar com ref. O [guia de composição](#) explica a estratégia de migração.

Isso também se aplica aos componentes `Dialog` e `Popover`.

- [Modal] Remova a API de customização de classes para o componente Modal (redução do tamanho do pacote -74% quando usado de forma independente).
- [Modal] `event.defaultPrevented` é agora ignorado. A nova lógica fecha o Modal mesmo se `event.preventDefault()` é chamado no evento down da tecla escape (Esc). `event.preventDefault()` destina-se a impedir comportamentos padrão, como clicar em uma caixa de seleção para verificá-lo, apertar um botão para enviar um formulário e pressionar a seta para a esquerda para mover o cursor em uma entrada de texto, etc. Apenas elementos HTML especiais possuem esses comportamentos padrão. Only special HTML elements have these default behaviors. Você deve usar `event.stopPropagation()` se você não quer acionar o evento `onClose` no modal.

## Paper

- [Paper] Reduza a elevação padrão. Altere a elevação padrão de Paper, para corresponder ao cartão e ao painel de expansão:

```
-<Paper />
+<Paper elevation={2} />
```

Isso afeta o componente `ExpansionPanel` também.

## Portal

- [Portal] O elemento filho precisa ser capaz de lidar com ref, quando a propriedade `disablePortal` é usada. O [guia de composição](#) explica a estratégia de migração.

## Slide

- [Slide] O elemento filho precisa ser capaz de lidar com ref. O [guia de composição](#) explica a estratégia de migração.

## Slider

- [Slider] Mova de `@material-ui/lab` para `@material-ui/core`.

```
-import Slider from '@material-ui/lab/Slider'
+import Slider from '@material-ui/core/Slider'
```

## Interruptor

- [Switch] Refatore a implementação para torná-la mais fácil de sobrescrever os estilos. Renomeie os nomes das classes para corresponder ao texto da especificação:

```
-icon
-bar
+thumb
+track
```

## Snackbar

- [Snackbar] Coincide com a nova especificação.
  - Modificado as dimensões
  - Modificado a transição padrão de `Slide` para `Grow`.

## SvgIcon

- [SvgIcon] Renomeie `nativeColor` -> `htmlColor`. React resolveu o mesmo problema com o atributo HTML `for`, eles decidiram chamar um propriedade `htmlFor`. Essa mudança segue o mesmo raciocínio.

```
-<AddIcon nativeColor="#fff" />
+<AddIcon htmlColor="#fff" />
```

## Abas

- [Tab] Remova as chaves de classe `labelContainer`, `label` e `labelWrapped` para simplificar. Isso nos permitiu remover 2 elementos DOM intermediários. Você deve conseguir mover os estilos customizados para chave de classe `root`.

```
▼<button class="MuiButtonBase-root-7hb5p9 MuiTab-root-1vz4akf MuiTab-textColorInherit-1bavt5r" tabindex="0" type="button" role="tab" aria-selected="false">
  <span class="MuiTab-wrapper-zx603k">Item One</span>
  <span class="MuiTouchRipple-root-11z6ptj"></span>
</button>
```

- [Tabs] Remova as propriedades descontinuadas `fullWidth` e `scrollable`:

```
-<Tabs fullWidth scrollable />
+<Tabs variant="scrollable" />
```

## Tabela

- [TableCell] Remova a propriedade descontinuada `numeric`:

```
-<TableCell numeric>{row.calories}</TableCell>
+<TableCell align="right">{row.calories}</TableCell>
```

- [TableRow] Remova a propriedade CSS de altura fixa. A altura da célula é calculada pelo navegador usando o preenchimento e a altura da linha.
- [TableCell] Movemos o modo `dense` para uma propriedade diferente:

```
-<TableCell padding="dense" />
+<TableCell size="small" />
```

- [TablePagination] O componente já não tenta corrigir as combinações de propriedades inválidas ( `page` , `count` , `rowsPerPage` ). Em vez disso, emite um aviso.

## Campo de texto

- [InputLabel] Você deve conseguir sobrescrever todos os estilos do componente FormLabel usando a API CSS do componente InputLabel. A propriedade `FormLabelClasses` foi removida.

```
<InputLabel
- FormLabelClasses={{ asterisk: 'bar' }}
+ classes={{ asterisk: 'bar' }}
>
Foo
</InputLabel>
```

- [InputBase] Modificado o modelo padrão de box sizing. Ele usa o seguinte CSS agora:

```
box-sizing: border-box;
```

Isso resolve problemas com a propriedade `fullWidth`.

- [InputBase] Remova a classe `inputType` do `InputBase`.

## Dica

- [Tooltip] O elemento filho precisa ser capaz de lidar com ref. O [guia de composição](#) explica a estratégia de migração.
- [Tooltip] Aparece somente após o foco ser "focus-visible" em vez de qualquer foco.

## Tipografia

- [Typography] Remova as variantes de tipografia descontinuadas. Você pode atualizar executando as seguintes substituições:
  - `display4` => `h1`
  - `display3` => `h2`
  - `display2` => `h3`
  - `display1` => `h4`
  - `headline` => `h5`
  - `title` => `h6`
  - `subheading` => `subtitle1`
  - `body2` => `body1`
  - `body1` (padrão) => `body2` (padrão)



- [Typography] Remova o padrão opinativo do estilo da tipografia `display: block` . Você pode usar a nova propriedade `display?: 'initial' | 'inline' | 'block';` .
- [Typography] Renomeie a propriedade `headlineMapping` para `variantMapping` , se alinha melhor com a sua finalidade.

```
-<Typography headlineMapping={headlineMapping}>
+<Typography variantMapping={variantMapping}>
```

- [Typography] Modifique a variante padrão de `body2` para `body1` . Um tamanho de fonte de 16px é um padrão melhor que 14px. Bootstrap, material.io e até a documentação usam 16px como tamanho de fonte padrão. 14px como o Ant Design usa, é compreensível, já que os usuários chineses têm um alfabeto diferente. Recomenda-se 12px como o tamanho de fonte padrão para japonês.
- [Typography] Remova a cor padrão das variantes de tipografia. A cor deve herdar a maior parte do tempo. É o comportamento padrão da web.
- [Typography] Renomeie `color="default"` para `color="initial"` seguindo a lógica [desta discussão](#). O uso de *default* deve ser evitado, isso perde semântica.

## Node

- [Removemos suporte ao node 6](#), você deve atualizar para o node 8.

## UMD

- Essa alteração facilita o uso de Material-UI com uma CDN:

```
const {
  Button,
  TextField,
-} = window['material-ui'];
+} = MaterialUI;
```

É consistente com outros projetos React:

- material-ui => MaterialUI
- react-dom => ReactDOM
- prop-types => PropTypes