

Introducción

Empezando

Componentes

[Alert](#)

[Badge](#)

[Button](#)

[Detail](#)

[Dialog](#)

[Dropdown](#)

[Errors](#)

[Filters](#)

[Input](#)

[Loading](#)

[Pagination](#)

[ProgressBar](#)

[Select](#)

[Tab](#)

[Table](#)

[TextareaResize](#)

[Toast](#)

Utilidades

[useAction](#)

[useFormat](#)

[useInputMask](#)

[useKeyPress](#)

[useTable](#)

[useTheme](#)

[useTooltip](#)

[useValidator](#)

Composables

[useFilters](#)

[useLogos](#)

Alert

Documentación y ejemplos del componente `alert`. Para definir **mensajes contextuales** en las distintas partes de la interfaz, con distintos colores y estados.

Props

message

Type: `String`

Define el **mensaje a mostrar**. Soporta lenguaje `HTML` dentro del mensaje.



Una alerta simple, se puede agregar lenguaje `HTML`.

```
<AlertComponent
  data-eit-variant="info"
  message="Una alerta simple, se puede agregar lenguaje <strong>HTML</strong>."
/>
```

html

icon

Type: `String`

Define el **ícono a mostrar** dentro del componente `alert`.



Una alerta simple, donde fue modificado el **ícono**.

```
<AlertComponent
  data-eit-variant="info"
  icon="fa-solid fa-circle-info"
  message="Una alerta simple, donde fue modificado el <strong>ícono</strong>."
/>
```

html

Introducción

Empezando

Componentes

Alert

Badge

Button

Detail

Dialog

Dropdown

Errors

Filters

Input

Loading

Pagination

ProgressBar

Select

Tab

Table

TextareaResize

Toast

Utilidades

useAction

useFormat

useInputMask

useKeyPress

useTable

useTheme

useTooltip

useValidator

Composables

useFilters

useLogos

Define el **color del ícono** dentro del componente `alert`.



Una alerta simple, fue modificada la clase del ícono.

```
<AlertComponent
  data-eit-variant="info"
  icon="fa-solid fa-circle-user"
  iconClass="eit-color--green"
  message="Una alerta simple, fue modificada la clase del <strong>ícono</strong>"
/>
```

html

Styles

colors

Para definir el **color dentro del componente**, puedes usar `class` o `data-attributes`.

class	data-attributes
<code>.eit-alert--primary</code>	<code>data-eit-variant="primary"</code>
<code>.eit-alert--secondary</code>	<code>data-eit-variant="secondary"</code>
<code>.eit-alert--tertiary</code>	<code>data-eit-variant="tertiary"</code>
<code>.eit-alert--success</code>	<code>data-eit-variant="success"</code>
<code>.eit-alert--warning</code>	<code>data-eit-variant="warning"</code>
<code>.eit-alert--error</code>	<code>data-eit-variant="error"</code>
<code>.eit-alert--info</code>	<code>data-eit-variant="info"</code>
<code>.eit-alert--light</code>	<code>data-eit-variant="light"</code>



Una alerta simple, se define con el tipo **primary**.



Una alerta simple, se define con el tipo **secondary**.



Una alerta simple, se define con el tipo **success**.

Introducción

Empezando

Componentes

Alert

Badge

Button

Detail

Dialog

Dropdown

Errors

Filters

Input

Loading

Pagination

ProgressBar

Select

Tab

Table

TextareaResize

Toast

Utilidades

useAction

useFormat

useInputMask

useKeyPress

useTable

useTheme

useTooltip

useValidator

Composables

useFilters

useLogos



Una alerta simple, se define con el tipo **warning**.



Una alerta simple, se define con el tipo **info**.



Una alerta simple, se define con el tipo **light**.

Ejemplo

```

<!--class-->
<AlertComponent
  class="eit-alert--primary"
  message="Una alerta simple, se define con el tipo <strong>primary</strong>."
/>

<!--data-attributes-->
<AlertComponent
  data-eit-variant="primary"
  message="Una alerta simple, se define con el tipo <strong>primary</strong>."
/>

```

animation

Crea una **animación de entrada** para el componente `alert`. Para utilizar la animación debes crear una constante boolean que controle su estado.

class	data-attributes
<code>.eit-alert--animation</code>	<code>data-eit-animation="hide"</code>
<code>.eit-alert--animation--show</code>	<code>data-eit-animation="show"</code>

Mostrar alerta

Ejemplo

```

<script setup lang="ts">
  import { ref } from 'vue'

  const alertShow = ref(false)

  const controlToggleAlert = computed(() => {
    return alertShow.value ? 'show' : 'hide'
  })

  const handleToggleAnimation = () => {

```

Introducción

Empezando

Componentes

Alert

Badge

Button

Detail

Dialog

Dropdown

Errors

Filters

Input

Loading

Pagination

ProgressBar

Select

Tab

Table

TextareaResize

Toast

Utilidades

useAction

useFormat

useInputMask

useKeypress

useTable

useTheme

useTooltip

useValidator

Composables

useFilters

useLogos

```
<template>
  <AlertComponent
    data-eit-variant="error"
    :data-eit-animation="controlToggleAlert"
    icon="fa-solid fa-ban"
    message="Una alerta animada, fue modificada con <strong>alertShow</strong>."
  />
</template>
```

Previous page
[Empezando](#)

Next page
[Badge](#)