Colors and Branding

Edit page

One of the main features of theui-svelte components library is its customizable color options, allowing you to maintain brand identity and ensure consistent theming efficiently.

The component library, built with TailwindCSS, offers two custom color options for seamless integration and easy theme customization through CSS variables.

It provides extensive color customization to maintain brand identity and consistent theming. With the component library, you can customize colors with or without shades, giving you flexibility and control over your design.

There are two ways to customize/generate colors in the component library:

1. Generate Color with Shades
2. Generate Color without Shades

Generate Color With Shades

Customizing colors with shades involves generating a cohesive color palette with varying levels of lightness and darkness, similar to the default TailwindCSS color system. You can assign a custom name to your color, generating the following shades: X-50, X-100, X-200, X-300, X-400, X-500, X-600, X-700, X-800, X-900, X-950. This allows for a flexible and comprehensive set of colors for different components and states in your design.

Available Color Shades

By default, the **theui-svelte** includes the following eight (8) shaded colors:

NAME	DESCRIPTION	VALUE	AVAILABILITY	EXAMPLE
`brand-primary`	Main brand color	#536DFE	Everywhere	`.bg-brand-500` , `.text-brand-300` , `.border-brand-200` etc.
`brand-secondary`	Secondary brand color	#FF5722	Everywhere	`.bg-brand-secondary-500` , `.text-brand-secondary-300` , `.border-brand-secondary-200` etc.
`on-brand-primary`	Foreground (Text) color to place on brand	#FAFDFF	Only for text	`.text-on-brand-500` , `.text-on-brand-300` etc.
`on-brand-secondary`	Foreground (Text) color to place on secondary brand color	#FFFDF9	Only for text	`.text-on-brand-secondary-500` , `.text-on-brand-secondary-300` etc.
`error`	Color to represent errors	#E53935	Everywhere	`.bg-error-500` , `.text-error-50` , `.border-error-200` etc.
`info`	Color to represent information	#29B6F6	Everywhere	`.bg-info-500` , `.text-info-50` , `.border-info-200` etc.
`success`	Color to represent success	#00C853	Everywhere	`.bg-success-500` , `.text-success-50` , `.border-success-200` etc.

NAME	DESCRIPTION	VALUE	AVAILABILITY	EXAMPLE
`warning`	Color to represent warning	#FFC107	Everywhere	`.bg-warning-500`, `.text-warning-50`, `.border-warning-200` etc.

Adding New Shaded Colors

To add a new color with shades, use the `twShades` function in your TailwindCSS configuration file. For example, to generate new shades for the `brand-accent` color:

tailwind.config.cjs

```js
/** @type {import('tailwindcss').Config} */
const twShades = require('tw-color-shades');

module.exports = {
    content: [],
    theme: {
        extend: {
            colors: {
                "brand-accent": twShades('#FF5722'), // This generates sha
            }
        }
    },
    plugins: []
}
```

HTML Example

```html
<div class="bg-brand-accent-700 text-brand-accent-100 ...">
  ...
</div>
```

Change/Update a Color

To change a default shaded color, update your TailwindCSS configuration. For example, to update the `brand` color to `#FF0000`:

tailwind.config.cjs

```
/** @type {import('tailwindcss').Config} */
const twShades = require('tw-color-shades');

module.exports = {
    content: [],
    theme: {
        extend: {
            colors: {
                brand: twShades('#FF0000'), // Changing brand color
            }
        }
    },
    plugins: []
}
```

Removing a Color

To remove a color from your palette, set its value to `null` in your TailwindCSS configuration. For example, to remove the `brand-secondary` color:

tailwind.config.cjs

```
/** @type {import('tailwindcss').Config} */

module.exports = {
    content: [],
    theme: {
        extend: {
            colors: {
                "brand-secondary": null // Removing brand-secondary color
            }
        }
    },
    plugins: []
  }
```

This will remove the `brand-secondary` color with all the classes from from the component library.

Generate Color Without Shades

For a non-shaded color, you can use our second method to declare a color. This method allows you to set specific colors for different UI elements.

Available CSS variables

The "theui-svelte" component library provides the following CSS variables, all prefixed with --ui to avoid conflicts with other variables:

CSS variables

```
:root {
  --ui-bg-primary: 250 250 250; /* #FAFAFA */
  --ui-bg-secondary: 238 238 238; /* #EEEEEE */
  --ui-bg-tertiary: 224 224 224; /* #E0E0E0 */
  --ui-bg-primary-alt: 10 10 20; /* #0A0A14 */
  --ui-bg-secondary-alt: 30 30 40; /* #1E1E28 */
  --ui-bg-tertiary-alt: 55 55 65; /* #373741 */

  --ui-text-default: 33 33 33; /* #212121 */
  --ui-text-alt: 189 189 189; /* #BDBDBD */
  --ui-text-muted: 117 117 117; /* #757575 */
}

/* To change any color for dark mode, change the color below */
:root.dark {
  /* Text color on dark mode */
  --ui-bg-primary: 10 10 20; /* #0A0A14 */
  --ui-bg-secondary: 30 30 40; /* #1E1E28 */
  --ui-bg-tertiary: 55 55 65; /* #373741 */
  --ui-bg-primary-alt: 250 250 250; /* #FAFAFA */
```

```
    --ui-bg-secondary-alt: 238 238 238; /* #EEEEEE */
    --ui-bg-tertiary-alt: 224 224 224; /* #E0E0E0 */

    --ui-text-default: 245 245 245; /* #F5F5F5 */
    --ui-text-alt: 75 75 75; /* #4B4B4B */
    --ui-text-muted: 175 175 175; /* #AFAFAF */
  }
```

> The colors prefixed with "--ui-text" are only available for text.
> The colors prefixed with "--ui-bg" are only available for background.

Usages

You can use custom classes to apply these colors in your design. The Components library includes the following classes:

CLASS	DESCRIPTION
`.bg-primary`	Default/primary background color.
`.bg-secondary`	Secondary background color.
`.bg-tertiary`	Tertiary background color.
`.bg-primary-alt`	Default/primary background color.
`.bg-secondary-alt`	Secondary background color.
`.bg-tertiary-alt`	Tertiary background color.
`.text-default`	Default text/foreground color for the component library.
`.text-alt`	Default text/foreground color for the component library.
`.text-muted`	Default text/foreground color for the component library.

CSS Variables Customization

To customize colors, update the corresponding CSS variable. For example, to change the bg-primary color for light and dark modes:

./src/app.postcss

```
:root {
  --ui-bg-primary : 240 252 252;
  --ui-text-default : 15, 15, 15;
}
:root.dark {
  --ui-bg-primary : 15, 8, 8;
  --ui-text-default : 250, 240, 245;
}
```

All components using the `.bg-primary` and `.text-default` classes will now reflect the new colors. Follow the same process to change other colors.