

# Tailwind CSS

U-TAD

# Ventajas

- Manejo de estados hover, focus.
- Ahorro de tiempo eligiendo clases.
- Cambios de forma segura.
- Personalizable (a diferencia de Bootstrap)
- Buena documentación:
  - <https://tailwindcss.com/docs/installation>

# Instalación

```
npx create-next-app estilos
```

Selecciona todas las opciones por defecto, y asegúrate que:

```
? Would you like to use Tailwind CSS? › No / Yes
```

```
cd estilos && npm run dev
```

y vamos a **app/page.js** (donde usamos clases de tailwind como text-3xl, font-bold y underline)

```
export default function Home() {
```

```
  return (
```

```
    <h1 className="text-3xl font-bold underline">
```

```
      Hello world!
```

```
    </h1>
```

```
)
```

```
}
```

# Colores

<https://tailwindcss.com/docs/customizing-colors>

- Background color
  - **bg-**
  - **bg-slate-600**
- Text color
  - **text-**
  - **text-white**
  - **text-[#d2d255]** //personalizado

Puedo personalizar mis clases en **tailwind.config.js**

```
theme: {  
  extend: {  
    colors: {  
      'azul-claro': '#242cff'  
      'azul-oscuro': '#0d1b3e'  
    }  
  }  
}
```

# Colores

```
<p className="bg-azul-oscuro text-azul-claro">Mi clase personalizada</p>
<button className="bg-azul-claro text-white p-2 rounded-lg w-24 mx-auto block my-8">
  Suscríbete
</button>
```

- **p-2** es el padding (en la documentación busca padding en Quick Search)
  - <https://tailwindcss.com/docs/padding>
- **rounded-lg** es border-radius (buscar en la doc)
  - <https://tailwindcss.com/docs/border-radius>
- **mx-auto block** (centrar en el eje x automáticamente en elementos de bloque, es decir, que se apilan verticalmente ocupando todo el ancho, como div, p, h1, li, etc.)
- **my-8** (margen de 8 en el eje y)
- **w-24** (establece el ancho del elemento)

# Colores

```
<p className="bg-azul-oscuro text-azul-claro">Mi clase personalizada</p>  
<button className="bg-azul-claro/50 hover:bg-azul-oscuro text-white p-2 rounded-lg  
w-24 mx-auto block my-8">  
    Suscríbete  
</button>
```

- **hover:bg-azul-oscuro** (cambia el color cuando se sitúa el ratón sobre él)
- **opacidad:** /50

# Colores - Gradientes

<https://tailwindcss.com/docs/gradient-color-stops>

starting-color

ending-color

middle-color

```
<div className="bg-gradient-to-r from-azul-claro to-azul-oscuro border-red-500 border-2">DIV</div>
```

# Colores - Cropping to text

<https://tailwindcss.com/docs/background-clip#cropping-to-text>

```
<div className="text-4xl font-extrabold">  
  <span className="bg-clip-text text-transparent bg-gradient-to-r from-pink-500  
    to-violet-500">  
    Hello world  
  </span>  
</div>
```

# Medidas (sizing)

Anchos y altos de los elementos

- Width: <https://tailwindcss.com/docs/width>

```
<div className="bg-red-400 w-40">Medidas en Tailwind</div>
```

O puedo personalizar el ancho con *w-[170px]*

O mejor, en tailwind.config.js

spacing: { '42': '170px' } //aplica al ancho y al alto

width: { '42': '170px' } //solo aplica al ancho

```
<div className="bg-red-400 w-42">Medidas en Tailwind</div>
```

# Medidas (sizing)

Anchos y altos de los elementos

- Height: <https://tailwindcss.com/docs/height>

```
<div className="bg-red-400 h-[100px]">  
  <div className="bg-green-400 h-1/3">Altura proporcional en Tailwind</div>  
</div>
```

# Estados

```
<input className="border border-gray-300 w-full px-3 py-2 rounded-md focus:outline-none  
focus:ring-1 focus:ring-purple-600 invalid:focus:ring-red-400 peer"  
        type="email"  
        placeholder="E-mail"  
    />  
<p className="text-red-400 hidden peer-invalid:block">El correo es incorrecto</p>
```

# Pseudo-classes

<https://tailwindcss.com/docs-hover-focus-and-other-states#before-and-after>

- after
- marker
- selection
- first-line y first-letter

(ver código en estilos/components/PseudoClases)

# Diseño Responsive

<https://tailwindcss.com/docs/responsive-design>

Ver ejemplo en *components/Responsive.jsx*

También se puede personalizar los breakpoints en *tailwind.config.js*

<https://tailwindcss.com/docs/responsive-design#using-custom-breakpoints>

# Flexbox

Creo una clase *card* personalizada en globals.css

```
@layer components {  
  .card {  
    @apply bg-red-500 w-40 h-40 grid place-content-center rounded-lg text-white font-bold border-2 border-red-600  
text-4xl;  
  }  
}
```

Ver en components/Flexbox

# Grid

Indicamos el grid y las columnas que queremos con grid-cols

Ver en components/Grid.jsx

# Dark Mode

```
<body className=dark:bg-slate-900>
```

Habría que setear el modo oscuro en el navegador