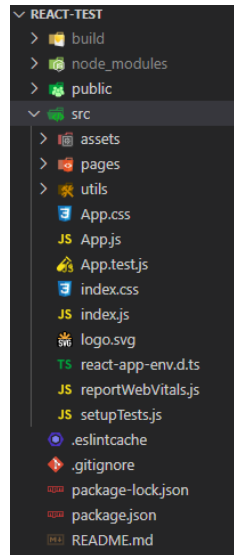


# Componente Slider.

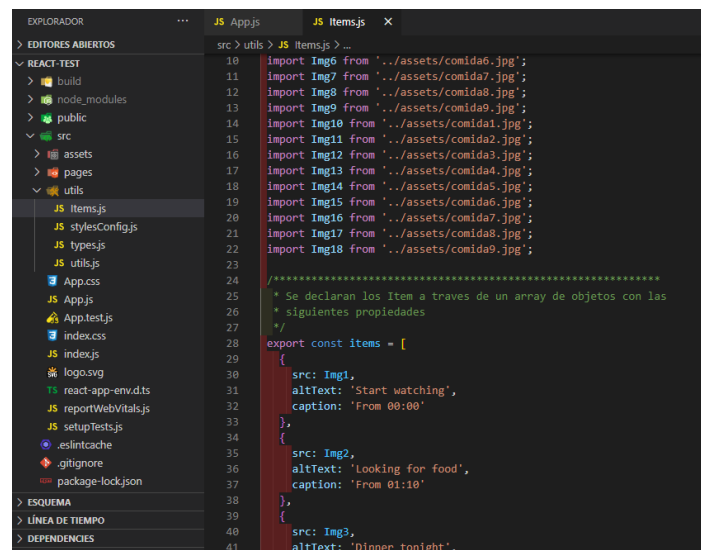
## 1- Estructura del proyecto



Los directorios con nombre “pages” y “utils” son los que debería de copiar para el proyecto que pretende utilizar el componente.

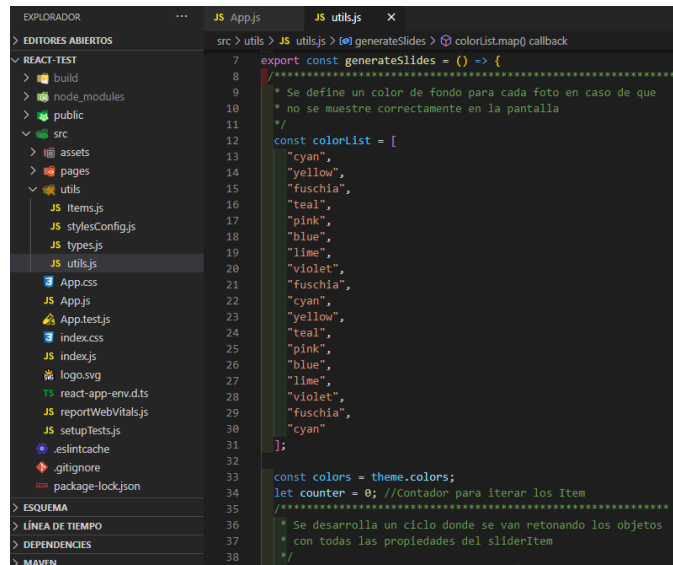
En el directorio “assets” se almacenan todas las imágenes que se quieren mostrar.

En “pages” están los componentes que se renderizan y en “utils” están los ficheros que sirven de base y configuración.



En el directorio “utils” hay un fichero “Item.js” que contiene los “import” de cada imagen. Si el directorio donde se encuentran las imágenes cambia, se deben de actualizar las rutas de los “import”.

La información de los Slider se conforma en el array de objetos que se muestra mas abajo en la imagen de arriba.



```
7 export const generateSlides = () => {
8
9
10  * Se define un color de fondo para cada foto en caso de que
11  * no se muestre correctamente en la pantalla
12  */
13  const colorList = [
14    "cyan",
15    "yellow",
16    "teal",
17    "pink",
18    "blue",
19    "lime",
20    "violet",
21    "fuschia",
22    "cyan",
23    "yellow",
24    "teal",
25    "pink",
26    "blue",
27    "lime",
28    "violet",
29    "fuschia",
30    "cyan"
31  ];
32
33  const colors = theme.colors;
34  let counter = 0; //Contador para iterar los Item
35  /* Se desarrolla un ciclo donde se van retonando los objetos
36  * con todas las propiedades del sliderItem
37  */
38  return [...Array(12)].map((_, index) => {
39    const color = colors[counter % colors.length];
40    return {
41      id: index + 1,
42      color: color,
43      image: `./assets/images/sliderItem${index + 1}.jpg`
44    };
45  });
46 }
```

En el fichero “utils.js” hay un array de colores que debe de tener la misma cantidad de elementos que tiene el array de ítems, de tal manera que exista un color para cada imagen a mostrar en un Slider.

Hasta aquí, se ha completado los pasos de la carga inicial que se requiere para poder crear los Slider.

## 2- Dependencias npm que debes de instalar

"react": "^16.8.6",

"react-dom": "^16.8.6",

"react-id-swiper": "^4.0.0",

"swiper": "^6.5.2",

"@emotion/core": "^10.1.1",

"@react-yuki/ui": "^0.0.38",

"@emotion/react": "^11.1.4",

## 3- Como usar el componente Slider.

Se deben de copiar los componentes “pages” y “utils” para el proyecto. Para agregar el componente Slider se debe de importar lo siguiente:

```
import {ThemeProvider } from '@emotion/react'
import { theme, globalStyles } from "../utils/stylesConfig";
import { Global } from "@emotion/core";
import Slider from "../pages/Slider";
import './App.css'; | Find related code in react-test
```

En “App.css” lo que existen son dos clases css que le dan estilo a los textos que se muestran en el Slider. Estos estilos puede incorporarlos a su fichero css, que debe de trabajar sin problemas.

Donde desee incorporar el componente debe de copiar los siguiente:

```
<ThemeProvider theme={theme}>
  <>
    <Global styles={globalStyles} />
    <Slider
      params={{
        grabCursor: true,
        /**
         * Agregar puntos de ruptura segun la el tipo de resolucio
         * del dispositivo. Existe otra variante para el ratio
         * en swiper-api.
         */
        breakpoints: {
          // when window width is >= 320px
          320: {
            slidesPerView: 2.44,
            spaceBetween: 12,
          },
          // when window width is >= 360px
          360: {
            slidesPerView: 2.65,
            spaceBetween: 12,
          },
          // when window width is >= 375px
          375: {
            slidesPerView: 2.77,
            spaceBetween: 12,
          },
          // when window width is >= 390px
          390: {
            slidesPerView: 2.89,
            spaceBetween: 12,
          },
        },
      }}
    />
  </>
</ThemeProvider>
```

Los componentes “ThemeProvider” y “Global” se encargan de brindar los estilos que se requiere para el componente Slider. Los “breakpoints” se ajustaron para que se pudiera cumplir con el requisito de mantener el width muy cerca de los 124px para cada resolución que se pudiera presentar.