

# Guía para entender el código

---

## 1. App.jsx

- **Router:** Envuelve toda la app para que funcione la navegación entre páginas.
- **Navbar:** Barra de navegación que aparece en todas las páginas.
- **Container:** Contenedor visual con margen para que el contenido no quede pegado arriba.
- **Routes:** Definimos qué componente mostrar según la ruta URL (home, about, skills, certificates).

### Detalle por líneas:

- 1–9: Importaciones de librerías, componentes para rutas, diseño, navegación y páginas.
  - 11: Inicio de la definición del componente principal **App**.
  - 13–14: Envolvemos la app con **Router**.
  - 15: Colocamos la barra de navegación fija y visible siempre.
  - 17–18: Añadimos un contenedor con margen superior.
  - 19–26: Definición de rutas y componentes asociados.
  - 27–29: Cierre de contenedores y final del componente.
  - 31: Exportación del componente **App**.
-

## 2. main.jsx

- 1–5: Importamos React, ReactDOM, el componente `App`, reset CSS y proveedor de tema personalizado.
  - 7: Creamos la raíz en el DOM para montar la app.
  - 8–15: Renderizamos la app dentro de `React.StrictMode`, envolviéndola en el proveedor de tema y `CssBaseline` para estilos consistentes.
- 

## 3. Navbar.jsx

- 1–13: Importamos Material UI, React Router, iconos y el contexto para modo claro/oscuro.
  - 15–20: Definimos el array con las rutas y etiquetas del menú.
  - 22: Función principal `Navbar`.
  - 24–26: Obtenemos tema actual y función para cambiar el modo.
  - 28–47: Renderizado de la barra con botones dinámicos, cada uno usa `NavLink` para navegación y estilo activo. Botón para cambiar tema claro/oscuro con icono.
- 

## 4. ThemeContext.jsx

- 1–5: Importamos React, contexto y funciones para tema Material UI.
  - 7: Creamos el contexto para alternar modo claro/oscuro.
  - 9–28: Componente proveedor que:
    - Guarda estado del modo.
    - Memoriza función para alternar modo.
    - Crea tema Material UI dinámico.
    - Devuelve proveedor de contexto y tema envolviendo a sus hijos.
-

## 5. Home.jsx

- 1–16: Importaciones de React, useState, Material UI, iconos, Framer Motion e imagen avatar.
  - 18–21: Estado `showData` para mostrar/ocultar datos personales.
  - 23–30: Array con datos personales, cada uno con icono y texto.
  - 32–81: Render:
    - Contenedor animado.
    - Avatar con estilo, borde y efecto hover.
    - Título y subtítulo.
    - Botón para mostrar/ocultar datos personales.
    - Animación para sección de datos personales.
    - Listado de datos con iconos y efectos hover.
- 

## 6. About.jsx

- 1–8: Importamos Material UI e iconos para experiencia.
  - 10–30: Array `experiences` con icono, título y descripción de habilidades.
  - 32–51: Render:
    - Contenedor con padding.
    - Título principal.
    - Texto descriptivo.
    - Paper con lista.
    - Mapeo de experiencias en `ListItems` con icono a la izquierda y texto a la derecha.
-

## 7. Skills.jsx

- 1–9: Importamos Material UI y iconos reales para las habilidades.
  - 11–18: Array `skills` con nombre, icono y nivel de dominio.
  - 20–44: Render:
    - Contenedor con padding.
    - Título centrado.
    - Grid responsivo con tarjetas.
    - Cada tarjeta muestra icono, nombre y barra de progreso.
- 

## 8. Certificates.jsx

- 1–14: Importamos React, Material UI, Framer Motion, iconos e imágenes de certificados.
- 16–33: Array `certificates` con título, imagen y detalles.
- 35–39: Variantes de animación para expansión de tarjetas.
- 41–74: Estados para tarjeta expandida, modal abierto y imagen modal. Funciones para abrir/cerrar modal y expandir tarjetas.
- 76–110: Render:
  - Título y grid de tarjetas.
  - Tarjetas clicables que escalan al hover.
  - Click en imagen abre modal con imagen ampliada.
  - Animación de detalles extra al expandir tarjeta.
- 112–138: Modal centrado con imagen grande, sombra, borde redondeado y botón para cerrar.