

Página web personal  
[erjieisheree.com](http://erjieisheree.com)



**LinkiaFP**

Formación Profesional Oficial

ERJIE XIA

DISEÑO DE APLICACIONES MULTIPLATAFORMA

SEGUNDO AÑO

15 DE DICIEMBRE

## **ÍNDICE**

1.	Objetivos de la fase.....	3
2.	Desarrollo de la fase.....	3
2.1.	Cambio de MD2 a MUI.....	3
2.2.	Desarrollo de las páginas con MUI.....	4
2.3.	Desarrollo de las páginas dinámicas.....	5
3.	Conclusión.....	6
4.	Webgrafía.....	6

## **1. OBJETIVOS DE LA FASE**

Esta es una de las principales grandes fases que contiene el proyecto entero, la interfaz del usuario.

En él se encuentra todo el contenido principal que el cliente principal necesita para lanzar la app a producción, el MVP (minimum viable product) del proyecto.

Durante la segunda entrega ya se explicó qué es el “*estilado*”, pues en esta fase se aplica todo lo aprendido en ese momento, haciendo que la UI quede bonita y preparada para mostrar al público.

Cabe destacar que durante esta fase se decidió cambiar el framework de MD2 (Material Design 2) a MUI (Material UI), debido a la naturaleza de MUI, que está desarrollado con similares componentes de MD2, pero centrado en el entorno de Next.js.

## **2. DESARROLLO DE LA FASE**

### **2.1. Cambio de MD2 a MUI**

El cambio de MD2 a MUI supone un gran cambio debido a que cada uno ofrece distintas herramientas para una misma finalidad, lo que supone bastantes cambios, sobre todo en la sintaxis.

Antes de ligar este framework al proyecto, se desinstalan todos los archivos con relación a MD2 y sus paquetes.

La implementación de MUI con el proyecto realizado con Next.js, tras la instalación de los paquetes necesarios, se debe envolver todo el proyecto con MUI mediante un componente “*AppRouterCacheProvider*” y en este caso, al desearse implementar estilos propios, también se envolverá con “*ThemeProvider*”.

```

export default function RootLayout({ children }: Readonly<{ children: React.ReactNode; }>) {
  return (
    <html lang="en">
      <body>
        <AppRouterCacheProvider>
          <ThemeProvider theme={AppTheme}>
            <header />
            <Box component="main" sx={{ minHeight: "100vh", bgcolor: "background.default" }}>
              {children}
            </Box>
            <Footer />
          </ThemeProvider>
        </AppRouterCacheProvider>
      </body>
    </html>
  );
}

```

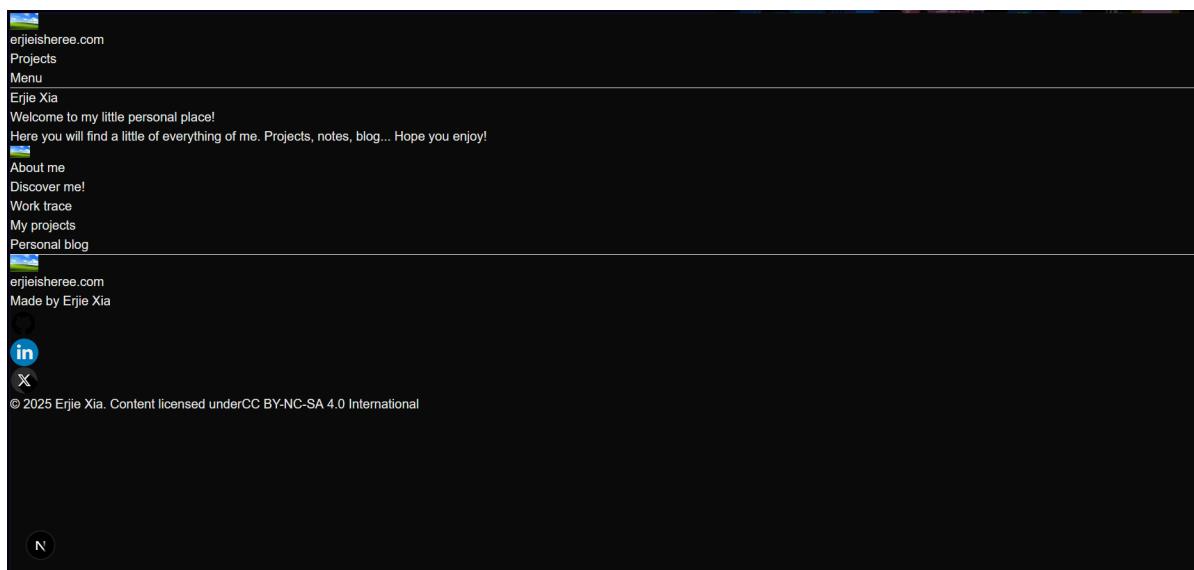
### *Ejemplo de implementación*

De esta forma desde el fichero que contiene los datos del objeto “*AppTheme*” podemos editar el “*estilado*” de forma universal para todo el proyecto, brindando un gran dinamismo al proyecto y creando mayor facilidad para dar paso a nuevas actualizaciones.

## **2.2. Desarrollo de las páginas con MUI**

El desarrollo de cada página con MUI es un proceso de aprendizaje e implementación. Los componentes, al principio desconocidos, fueron estudiados mientras que se fueron usando.

Tras implementar completamente los componentes que ofrece MUI, se ha logrado convertir el esqueleto de la página web, en una interfaz más atractiva para el usuario.



### *Esqueleto de la página web*



*La misma página tras la implementación*

Se puede comprobar la gran diferencia. Es cierto que aún hay imágenes, contenidos textuales... que requieren de un toque más creativo.

### 2.3. Desarrollo de las páginas dinámicas

A pesar de tener la misma esencia de construcción de interfaz que el anterior tema, este requiere tocar unos matices más complicados, por lo que es interesante darle un espacio en el que mostrarlo al completo.

Durante la primera entrega se habla sobre páginas cuya información es recogida de la base de datos de Supabase. Estos datos, con el paso del tiempo se espera que vayan aumentando, de forma que la página web sea más rica.

Estas páginas son publicaciones que el propietario de la página va subiendo. Para no crear nuevos ficheros por cada publicación nueva, se implementó el sistema de páginas dinámicas.

La teoría de esta práctica es simple, desde la página que contiene el listado de todas las publicaciones, al desear abrir una de las publicaciones, se dirigirá a una página cuya url contenga información sobre la publicación, en este caso, el id. La página nueva, deberá recoger este id, buscar en la base de datos el contenido de la publicación cuya id es coincidente y generar la página con el contenido obtenido.

### **3. CONCLUSIÓN**

Durante esta fase se ha logrado terminar el la interfaz del usuario, por tanto, el proyecto ya está casi finalizado, solo faltan un pocos cambios en cuanto al contenido.

Durante la siguiente fase ya se comenzará con el lanzamiento de la página web y el desarrollo de funciones secundarias como las de administrador, modo oscuro/claro...

### **4. WEBGRAFÍA**

Las fuentes de información usadas durante esta fase son las siguientes.

Documentos de Next.js: <https://nextjs.org/>

Documentos de Material Design:

<https://github.com/material-components/material-web/discussions/categories/announcements>

Documentos de Material UI: <https://mui.com/material-ui/getting-started/>

ChatGPT: <https://chatgpt.com/>