

# Actividad en clase/casa

## Estilado aislado con Styled Components en una arquitectura de Microfrontends

### Objetivo de aprendizaje

Que el estudiante:

- Comprenda el problema del **aislamiento de estilos** en microfrontends.
- Implemente estilos encapsulados usando **Styled Components**.
- Analice cómo Styled Components evita conflictos de CSS entre microfrontends.
- Integre múltiples microfrontends en una aplicación contenedora.

### Contexto del problema

Una empresa está desarrollando una plataforma web compuesta por **microfrontends**, donde cada equipo es responsable de un módulo independiente.

El reto principal es **evitar conflictos de estilos CSS** cuando los microfrontends se integran en una misma aplicación.

Para resolver este problema, se decide utilizar **Styled Components** como solución de **CSS-in-JS**, aprovechando su encapsulamiento automático de estilos.

### Arquitectura propuesta

- **Microfrontend 1 (React)**  
Módulo de *Productos*
- **Microfrontend 2 (React)**  
Módulo de *Usuarios*
- **Aplicación Contenedora (Shell)**  
Integra ambos microfrontends

Cada microfrontend:

- Usa **React**
- Usa **Styled Components**
- Define sus propios estilos sin afectar a los demás

### Tecnologías

- React
- Styled Components
- Module Federation (Webpack) (*o carga independiente por rutas si se desea simplificar*)
- Node.js / npm o yarn

# Actividades a realizar

## Configuración inicial

1. Crear dos aplicaciones React independientes.
2. Instalar Styled Components en cada una:

```
npm install styled-components
```

ó

```
npm i @emotion/styled @emotion/react
```

```
yarn add @emotion/styled @emotion/react
```

## Desarrollo de componentes estilizados

Cada microfrontend debe crear al menos:

### Microfrontend Productos

- Un CardProducto
- Un BotonComprar

Ejemplo:

```
import styled from "styled-components";
```

```
export const BotonComprar = styled.button`  
  background-color: #007bff;  
  color: white;  
  padding: 10px;  
  border-radius: 6px;  
`;  
`;
```

### Microfrontend Usuarios

- Un PerfilUsuario
- Un BotonEditar

Ambos microfrontends pueden usar **nombres de componentes o estilos similares**, pero **no deben generar conflictos visuales**.

## Integración en el Shell

1. Integrar ambos microfrontends en la aplicación contenedora.
2. Verificar que:
  - Los estilos se mantienen aislados.

- No hay colisiones de CSS.
- Cada microfrontend conserva su identidad visual.

## **Pruebas y análisis**

Los estudiantes deben responder:

- ¿Qué problema resuelve Styled Components en microfrontends?
- ¿Qué pasaría si se usara CSS global?
- ¿Cómo ayuda el hash de clases generado automáticamente?
- ¿Styled Components favorece la independencia entre equipos?

## **Entrega**

Enviar un enlace al repositorio de todos los microfrontends.

Enviar en pdf capturas de pantalla de la aplicación contenedora y las respuestas a las preguntas formuladas.