

Plan de mejoras - Proyecto Taura Birra

Mejoras propuestas y objetivos

Refactorización a React.js

Mejorar la mantenibilidad del código, permitir la creación de componentes reutilizables y facilitar futuras ampliaciones del proyecto.

Implementación de criterios de accesibilidad

Aumentar la accesibilidad del sitio web según estándares WCAG, asegurando que sea usable por personas con distintas capacidades (lectores de pantalla, navegación por teclado, contraste adecuado, etc.).

Integración de sistema de gestión de contenido

Permitir que el contenido del sitio (descripciones de cervezas, imágenes, información de contacto, etc.) pueda actualizarse sin modificar el código, reduciendo la dependencia del desarrollador.

Implementación de formulario de pedidos vía WhatsApp

Facilitar que los clientes puedan realizar pedidos directamente desde el sitio web de manera rápida y simple, mejorando la experiencia del usuario y potencialmente aumentando las ventas.

Etapas de desarrollo y entregas

Etaapa 1: Análisis de la situación Actual y diseño estructural

Duración estimada: 2 semanas

Actividades Detalladas

- Revisar toda la estructura de archivos HTML, CSS y JS existentes.
- Mapear las secciones y funcionalidades actuales del sitio (inicio, sobre nosotros, cervezas, bar, contacto).
- Identificar elementos que se pueden convertir en componentes React reutilizables (tarjetas de cerveza, header, footer, menús, separadores decorativos).
- Evaluar contenido que debería ser dinámico y gestionable mediante CMS (textos, imágenes, descripciones de cervezas, horarios, datos de contacto).
- Revisar la accesibilidad actual:
 - Contraste de colores y tipografía.
 - Navegación mediante teclado.

- Uso de etiquetas alt y roles ARIA.
- Definir la estructura de carpetas y archivos en React (components, pages, assets, services).
- Diseñar un plan de flujo de datos y props entre componentes.
- Crear un documento de planificación con prioridades para refactorización y accesibilidad.

Entrega Informe con análisis de situación actual, diseño estructural propuesto en React y plan de mejoras de accesibilidad.

Etapla 2: Refactorización a React e Implementación Inicial de criterios de accesibilidad

Duración estimada: 2 semanas

Actividades detalladas

- Crear el proyecto base en React con la estructura definida en la etapa 1.
- Transformar páginas y secciones HTML existentes en componentes React.
- Integrar CSS existente en los componentes o migrarlo a CSS modular/Styled Components según corresponda.
- Implementar navegación y rutas con React Router (si aplica).
- Incorporar accesibilidad inicial:
 - Añadir roles ARIA y etiquetas adecuadas en elementos interactivos.
 - Revisar y ajustar navegación por teclado.
 - Revisar y mejorar contraste de colores y tipografía.
- Validar que cada componente se comporta correctamente y mantiene la responsividad en distintos dispositivos.
- Realizar pruebas iniciales de funcionamiento de los componentes y la navegación.

Entrega Sitio web funcional en React con componentes reutilizables y accesibilidad inicial implementada.

Etapla 3: Integración de CMS headless

Duración estimada: 2 semanas

Actividades Detalladas

- Seleccionar y configurar el Headless CMS (ej. Netlify CMS).
- Crear colecciones y definir campos para cervezas, secciones del sitio, contacto y otros contenidos dinámicos.
- Conectar React con el CMS para renderizar contenido dinámico en los componentes correspondientes.
- Ajustar los componentes React para mostrar correctamente datos provenientes del CMS.
- Probar la creación, edición y eliminación de contenido en el CMS sin modificar código.
- Validar que el sitio sigue siendo responsive y accesible con contenido dinámico.

Entrega Sitio web React con contenido dinámico gestionable desde CMS y componentes preparados para futuras actualizaciones.

Etap 4: Implementación del formulario de pedidos vía WhatsApp y ajustes finales

Duración estimada: 2 semanas

Actividades Detalladas

- Diseñar y maquetar el formulario de pedidos dentro del sitio React.
- Configurar el enlace de WhatsApp para que envíe los datos ingresados en el formulario automáticamente.
- Implementar validaciones de formulario (campos requeridos, formatos correctos de datos).
- Ajustar accesibilidad en el formulario (roles ARIA, navegación por teclado, etiquetas de campos).
- Realizar pruebas de usabilidad del formulario y del sitio completo: interacciones, responsividad, carga de datos dinámicos.
- Corregir errores detectados durante pruebas y optimizar el flujo de interacción del usuario.

Entrega Sitio web completo con formulario de pedidos vía WhatsApp, accesibilidad optimizada y contenido dinámico gestionable desde CMS.