

Documento de Alcance Funcional: VTEX Pixel App

Product Owner: Leo

Developer: Karen

1. Objetivo del Proyecto

Implementar una Pixel App nativa en VTEX para capturar eventos de comportamiento de usuario y transaccionales, asegurando la integridad de los datos enviados desde el front-end hacia GA4.

La app será server-side y se utilizará stape.io para alojar el contenedor. Es. Decir Pixel App que actúe como puente entre el pixel-manager de VTEX y el contenedor Server-Side de Stape.io. La app capturará los eventos del lado del cliente y los enviará mediante una solicitud fetch o el script de transporte de Stape al endpoint de nuestro servidor.

2. Definición Técnica de Eventos (Mapping)

Los eventos a configurar serán los siguientes:

ProductViewData	// ← Vista de producto
ProductClickData	// ← Click en producto
OrderPlacedData	// ← Orden completada
OrderPlacedTrackedData	// ← Orden trackeada
PageViewData	// ← Vista de página
ProductImpressionData	// ← Impresión de productos
AddToCartData	// ← Agregar al carrito
RemoveToCartData	// ← Remover del carrito
CartChangedData	// ← Cambio en carrito
HomePageInfo	// ← Info página home
ProductPageInfoData	// ← Info página product

- | SearchPageInfoData // ← Info página búsqueda
- | UserData // ← Datos de usuario
- | CartIdData // ← ID del carrito
- | PromoViewData // ← Vista de promoción
- | PromotionClickData // ← Click en promoción
- | NewsletterSubscriptionData // ← Suscripción newsletter

Detalle:

A. Eventos de Navegación y Usuario

Evento	Disparador (Trigger)	Propiedades Críticas a Enviar
PageViewData	Carga de cualquier página.	pageTitle, pageUrl, referrer
UserData	Identificación de usuario (Login/Checkout).	email, firstName, userId
HomePageInfo	Carga específica de la Home.	pageUrl

B. Ciclo de Vida del Producto (Funnel)

Evento	Disparador (Trigger)	Data Schema Sugerido
ProductViewData	Visualización de PDP.	productName, productId, brand, category, price
ProductImpressionData	Producto aparece en vitrina/shelf.	listName, position, productId, productName
ProductClickData	Click en un producto de lista.	productId, productName, position

C. Carrito y Transacción (Críticos)

Evento	Disparador (Trigger)	Detalles
AddToCartData	Click en el botón de compra.	skuId, quantity, price, variant
RemoveFromCartData	Eliminación de item del mini-cart.	skuId, quantity
CartChangedData	Cualquier update en el carrito.	items[] (array completo con IDs y cantidades)
OrderPlacedData	Página de confirmación (Order Placed).	orderId, transactionTotal, transactionProducts[], coupon

Diccionario de Eventos y Data Schema

Cada evento enviado a Stape debe seguir esta estructura mínima en el body del JSON:

Evento VTEX	Evento Destino (Stape)	Data obligatoria para Karen
ProductViewData	view_item	id, name, price, brand, category
AddToCartData	add_to_cart	id, name, price, quantity
OrderPlacedData	purchase	transaction_id, value, currency, items[], coupon
ProductClickData	select_item	id, name, position

Eventos de Usuario (User ID Resolution)

Para que Stape identifique al usuario entre sesiones, la Pixel App debe mapear:

- **UserData:** se debe capturar el email (hasheado si es necesario por privacidad) o el userId y enviarlo como user_data en los headers o el body del evento hacia Stape.

Especificación de Lenguaje y Entorno

Lenguaje: JavaScript (ES6+). **Entorno:** VTEX Pixel App (vtex.pixel-manager).

La implementación se realizará exclusivamente en **JavaScript**.

Mapeo de Variables GA4 (Stape)

Propiedad en VTEX	Nombre para Stape/GA4	Ejemplo / Formato
productId o skuId	item_id	"12345"
productName	item_name	"Zapatillas Running"
transactionTotal	value	150.00 (Number, no String)
transactionProducts	items	Array de objetos

Consideraciones de Performance y Seguridad

- Manejo de Scripts: Los scripts de tracking deben inyectarse de forma asíncrona para no afectar el LCP (Largest Contentful Paint) del sitio.
- Persistencia: En eventos como OrderPlaced, se debe asegurar que si el usuario refresca la página de agradecimiento, el evento no se dispare por duplicado
- Fetch asíncrono: Usar el método fetch con keepalive: true para asegurar que el evento de purchase se envíe incluso si el usuario cierra la pestaña justo después de comprar.