GUÍA PASO A PASO

Eventos de ecommerce en GA4

El código correcto de view_item, add_to_cart, begin_checkout y purchase, con gtag y con dataLayer, listo para adaptar a tu tienda. Al final, el orden para validarlo en la consola, en Tag Assistant y en DebugView antes de confiar en un reporte.

PARA	Tiendas con gtag.js o Google Tag Manager
PROBADO CON	Chromium 151 (Playwright) · dataLayer local · red de Google bloqueada
ACTUALIZADA	30 de octubre de 2026
DOCUMENTACIÓN	developers.google.com/analytics/devguides/collection/ga4/ecommerce
ARTÍCULO	ecomlabs.dev/g/ga4
AUTOR	Luis Morales · @luismoralesdev

Índice

Antes de empezar	Qué vas a configurar
Paso 01	view_item: ven un producto
Paso 02	add_to_cart: lo agregan al carrito
Paso 03	begin_checkout: empiezan a pagar
Paso 04	purchase: compran
Paso 05	Configura Tag Manager
Paso 06	Valida en este orden
Al terminar	Checklist

ANTES DE EMPEZAR

Qué vas a configurar

Cuatro eventos cubren el recorrido principal de una compra: `view_item` (ven un producto), `add_to_cart` (lo agregan), `begin_checkout` (empiezan a pagar) y `purchase` (compran). Con ellos GA4 puede mostrarte en qué paso se va la gente.

Usa una sola forma de envío por sitio. Si cargas `gtag.js` directamente, llama a `gtag("event", …)`. Si usas Google Tag Manager, haz push al `dataLayer` y deja que una etiqueta de GA4 lo envíe. Esta guía usa dataLayer; los productos y precios son ilustrativos: cambia los valores, no los nombres de los campos.

PASO 01

view_item: ven un producto

En la página de producto, con los datos del producto que se muestra:

```
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({ ecommerce: null });
window.dataLayer.push({
  event: "view_item",
  ecommerce: {
    currency: "MXN",
    value: 349,
    items: [{ item_id: "TAZA-001", item_name: "Taza de cerámica", price: 349, quantity: 1 }],
  },
});
```

La primera línea crea el data layer sin borrar lo que ya tiene (nunca lo reasignes con `= []`), y `{ ecommerce: null }` limpia el evento anterior para que no se mezcle con este.

PASO 02

add_to_cart: lo agregan al carrito

Con dataLayer, cuando el carrito confirma que el producto entró:

```
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({ ecommerce: null });
window.dataLayer.push({
  event: "add_to_cart",
  ecommerce: {
    currency: "MXN",
    value: 698,
    items: [{ item_id: "TAZA-001", item_name: "Taza de cerámica", price: 349, quantity: 2 }],
  },
});
```

Con gtag.js, los mismos `currency`, `value` e `items` van directo en `gtag("event", "add_to_cart", { … })`, sin el objeto `ecommerce` y sin la línea de limpieza. El ejemplo completo está en el artículo.

> **CUIDADO**
>
> Dispara add_to_cart cuando tu carrito confirma que el producto entró (la respuesta de tu API o de tu tema), no en el clic del botón. Si el producto no se agrega por falta de stock o un error de red, no debe contarse.

PASO 03

begin_checkout: empiezan a pagar

Con el contenido completo del carrito. `value` es la suma de todas las líneas: 349 × 2 + 189 × 1 = 887.

```
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({ ecommerce: null });
window.dataLayer.push({
  event: "begin_checkout",
  ecommerce: {
    currency: "MXN",
    value: 887,
    items: [
      { item_id: "TAZA-001", item_name: "Taza de cerámica", price: 349, quantity: 2 },
      { item_id: "PLATO-002", item_name: "Plato hondo", price: 189, quantity: 1 },
    ],
  },
});
```

PASO 04

purchase: compran

En la página de confirmación, una sola vez por pedido:

```
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({ ecommerce: null });
window.dataLayer.push({
  event: "purchase",
  ecommerce: {
    transaction_id: "MX-1001",
    currency: "MXN",
    value: 887,
    shipping: 99,
    items: [
      { item_id: "TAZA-001", item_name: "Taza de cerámica", price: 349, quantity: 2 },
      { item_id: "PLATO-002", item_name: "Plato hondo", price: 189, quantity: 1 },
    ],
  },
});
```

- `transaction_id` es obligatorio: usa el número de pedido real de tu tienda.
- `shipping` va aparte. La referencia pide que `value` no incluya envío ni impuestos.

> **CUIDADO**
>
> Si tus precios ya incluyen IVA, decide un criterio (con o sin IVA) y úsalo igual en `price` y `value` de los cuatro eventos. Si cada evento usa uno distinto, los totales de GA4 no cuadran entre sí.

PASO 05

Configura Tag Manager

Si usas dataLayer, cada evento necesita una etiqueta y un activador en Tag Manager:

DÓNDE	QUÉ PONER
Tipo de etiqueta	Google Analytics: GA4 Event
Nombre del evento	`add_to_cart` (o el evento que corresponda)
More Settings › Ecommerce	Marca *Send Ecommerce data* y elige *Data Layer* como fuente
Activador	*Custom Event* con el mismo nombre: `add_to_cart`

Repite para `view_item`, `begin_checkout` y `purchase`. El nombre del activador y el del push tienen que coincidir letra por letra: GA4 y Tag Manager distinguen mayúsculas.

PASO 06

Valida en este orden

1. En la consola

Agrega un producto al carrito y pega esto en la consola del navegador. Encuentra tanto los push de dataLayer como las llamadas de gtag:

```
dataLayer.filter((e) => e.event === "add_to_cart" || e[1] === "add_to_cart");
```

Si devuelve un arreglo vacío, el problema está en tu sitio. No sigas a GA4 hasta que aparezca.

2. En Tag Assistant

Abre tu sitio desde `tagassistant.google.com` o con la vista previa de Tag Manager. Confirma que la etiqueta de GA4 se dispara con el evento y lleva `currency`, `value` e `items`.

3. En DebugView

Tag Assistant y la vista previa ya activan el modo de depuración en tu dispositivo. Con gtag también puedes activarlo así:

```
gtag("config", "G-XXXXXXXXXX", { debug_mode: true });
```

En GA4 entra a *Admin › Data display › DebugView*, haz clic en el evento y revisa sus parámetros.

> **CUIDADO**
> Para apagar el modo de depuración, quita `debug_mode`. Ponerlo en `false` no lo desactiva. Y si tu sitio usa modo de consentimiento, acepta las cookies de Analytics mientras pruebas o los eventos no aparecen en DebugView.

4. En los reportes

El reporte de tiempo real tarda unos minutos. Los reportes estándar pueden tardar de 24 a 48 horas: no juzgues el cambio el mismo día.

AL TERMINAR

Checklist

- [ ] Los cuatro nombres están en minúsculas y copiados de la referencia: `view_item`, `add_to_cart`, `begin_checkout`, `purchase`.
- [ ] Cada push de ecommerce va precedido de `{ ecommerce: null }`.
- [ ] `currency` es un código ISO de tres letras y aparece siempre que hay `value`.
- [ ] `value` es un número igual a la suma de `price × quantity`, sin envío ni impuestos.
- [ ] Cada artículo lleva `item_id` o `item_name` (mejor los dos).
- [ ] `purchase` lleva el `transaction_id` real y se envía una sola vez por pedido.
- [ ] add_to_cart se dispara cuando el carrito confirma, no en el clic.
- [ ] La consola encuentra el evento y Tag Assistant muestra la etiqueta disparada.
- [ ] DebugView muestra los cuatro eventos con sus parámetros.
- [ ] No hay otra integración (una app o un canal de ventas) enviando los mismos eventos.
- [ ] Quitaste `debug_mode` antes de publicar.