Developer Documentation: GA4 ViewItem Event on Shopify using Google Tag Manager

by ShahzadaAliHassan - 6th November 2023

Introduction

This documentation guides developers through the process of configuring the 'view_item' event for GA4 using Google Tag Manager on a Shopify store.

> 🚀 **Expand Your Skills:**
>
> Dive deeper into analytics with my course: **Analytics Mastery for Shopify using Google Tag Manager**. Perfect for those looking to master the integration of Shopify with GA4 via GTM.

Shopify Store Configuration for the Event

To successfully trigger the "view_item" event on Shopify, follow these steps:

1. Access your Shopify store backend.

2. Navigate to the online store and themes section.

3. Before any changes, duplicate your theme for backup.

4. Integrate the provided JavaScript code.

5. Include the code snippet in your main theme file.

6. Refresh a product page to ensure the custom event triggers correctly.

DataLayer Code Snippets Overview

The code aims to integrate the dataLayer object with Shopify, sending an event named `custom_view_item` when a product page is visited.

Head DataLayer Code

```
<script>
    window.dataLayer = window.dataLayer || [];
```

```
</script>

{%- if template contains 'product' -%}
  {%- render 'datalayer-product', product: product -%}
{%- endif -%}
```

Product DataLayer Code

```
{%- liquid assign current_variant = product.selected_or_first_available_variant -%}

<script>
  console.log("datalayer-product snippet is loaded")
  var itemObject = {
      item_id: {{ current_variant.id | json }},
      id: {{ current_variant.id | json }},
      item_name: {{ product.title | json }},
      currency: {{ shop.currency | json }},
      item_brand: {{ product.vendor | json }},
      item_category: {{ product.type | json }},
      item_variant: {{ current_variant.title | json }},
      price: {{ current_variant.price }} / 100,
      google_business_vertical: 'retail',
      quantity: 1
  }

  document.addEventListener('DOMContentLoaded', () => {
      dataLayer.push({ ecommerce: null });
      dataLayer.push({
          event: "custom_view_item",
          ecommerce: {
              currency: {{ shop.currency | json }},
              items: [itemObject]
          }
      });
  })
</script>
```

Theme DataLayer Code

```
{% render 'head-datalayer' %}
```

Steps to Integrate DataLayer Code in Shopify:

1. **Backup**: Always back up your current theme.

2. **Access Theme Files**:

   - Login to Shopify Admin.

- Go to Online Store > Themes.

- Click "Actions" > "Edit code".

3. **Insert Head DataLayer Code**:

- Click "Add new snippet".

- Name it `head-datalayer.liquid`.

- Insert the "Head DataLayer Code" into this file.

- Save changes.

4. **Integrate Product DataLayer Code**:

- Click "Add new snippet".

- Name it `datalayer-product.liquid`.

- Insert the "Product DataLayer Code" into this file.

- Save changes.

5. **Add Theme DataLayer Code**:

- Insert the "Theme DataLayer Code" in the desired location (usually within `<head>` tags in `theme.liquid` after the google tag manager snippet).

- Save changes.

6. **Test**:

- Open a product page on Shopify.

- In the browser's developer console, confirm the `custom_view_item` event and verify dataLayer details. You can type `dataLayer` in the console window to see the object.

Note: These instructions assume a standard Shopify setup. Custom themes may require adjustments. Always test before and after applying changes.

Hey, I'm Hassan 👋

I'm a **Top Rated Freelancer** on Upwork, here's what you should know about me:

- 📚 **Top Rated Web Analytics Expert:** With a 98% Job Success on Upwork, I specialize in web analytics, providing expertise in tools like GA4, UA, FB Pixel, and CAPI server-side tracking. I've had the privilege of working with renowned clients like BookOnline.com and TruelyBeauty.com.

- 🌎 **Global Impact:** My work has spanned across 230 diverse clients and has been showcased in major corporations. This exposure allows me to navigate and comprehend various industry terrains and cultural nuances in business.

- 🔧 **251 Total Jobs Completed:** Over the course of my career, I've tackled a variety of challenges in web analytics and tracking setups. I've successfully completed 251 jobs, always focusing on delivering value and concrete solutions to clients across several sectors.

- ⌚ **523 Total Hours Worked:** I've dedicated 523 hours to working closely with my clients. My goal is always to ensure their web analytics are set up correctly, their data is precise, and they can draw strategic insights from this data to make informed business decisions.