



Campaign Pixel Guide

Installing Campaign Pixel for Digital Marketing

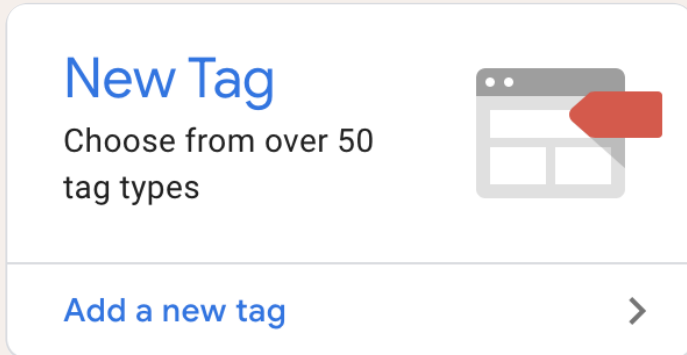


Google Tag Manager

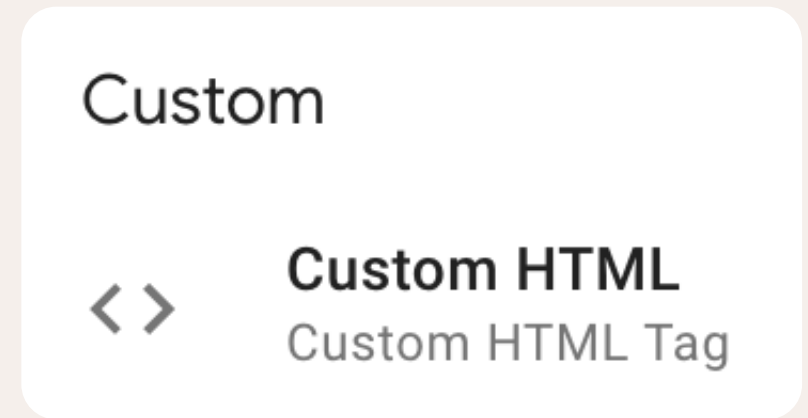
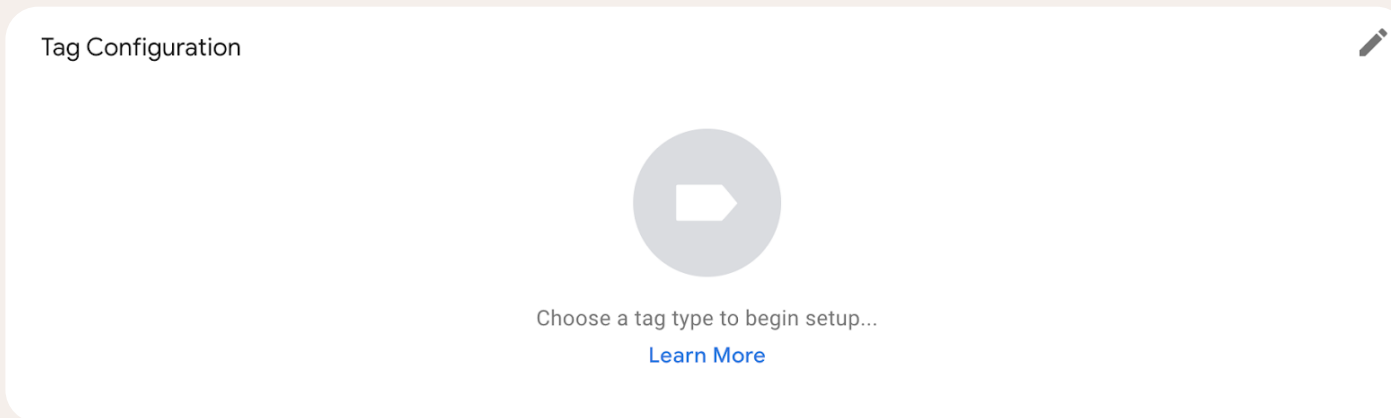
Recommended

Website Visits

- Log in to Google Tag Manager. Click to Add New Tag

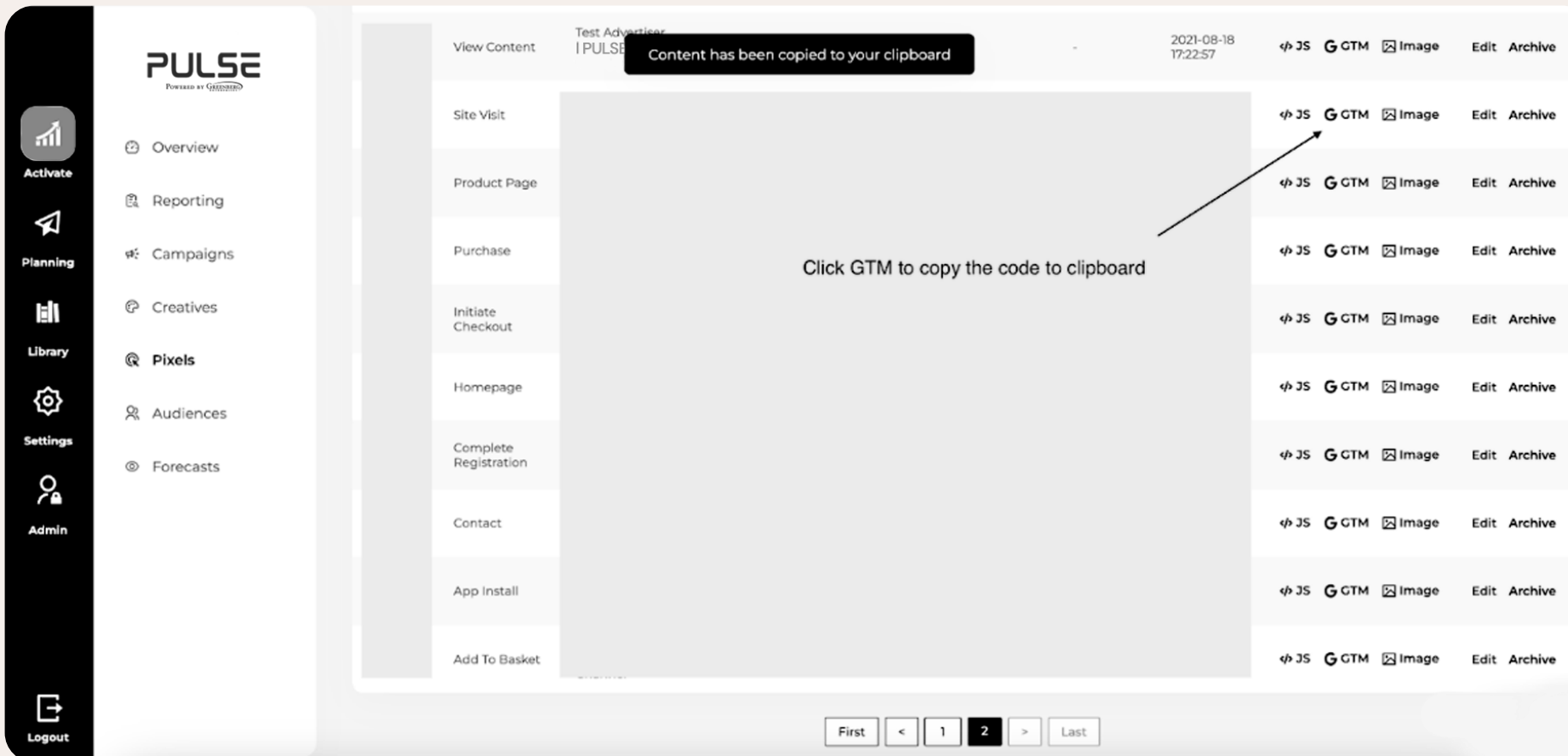


- In Tag Configuration, choose Custom HTML.



Website Visits

- Navigate to PULSE Activate Pixels, and click to “Get Pixel” for Site Visit as GTM as shown in the screenshot below:



The screenshot displays the PULSE Activate Pixels interface. On the left is a sidebar with navigation options: Activate, Planning, Library, Settings, and Admin. The main content area shows a table of website visits. The 'Site Visit' row is highlighted, and an arrow points to the 'GTM' button in its row. A tooltip above the arrow reads 'Click GTM to copy the code to clipboard'. A notification at the top of the table states 'Content has been copied to your clipboard'. The table columns include icons for JS, GTM, and Image, along with 'Edit' and 'Archive' links. The bottom of the interface features pagination controls with 'First', '<', '1', '2', '>', and 'Last' buttons.

View Content	Test Advertising I PULSE	2021-08-18 17:22:57	JS	GTM	Image	Edit	Archive
Site Visit			JS	GTM	Image	Edit	Archive
Product Page			JS	GTM	Image	Edit	Archive
Purchase			JS	GTM	Image	Edit	Archive
Initiate Checkout			JS	GTM	Image	Edit	Archive
Homepage			JS	GTM	Image	Edit	Archive
Complete Registration			JS	GTM	Image	Edit	Archive
Contact			JS	GTM	Image	Edit	Archive
App Install			JS	GTM	Image	Edit	Archive
Add To Basket			JS	GTM	Image	Edit	Archive

Website Visits

- Add HTML code to the Google Tag Manager as shown below (use the pixel that you copied from the PULSE dashboard in the previous step):

Tag Configuration

Tag Type

< > Custom HTML
Custom HTML Tag

HTML ⓘ

```
1 <script async src="https://pxl.greenbergenterprises.com/s/35267d5c-2e1b-4f59-887e-61575ba6669b"></script>
2
```

☐ Support document.write ⓘ

Website Visits

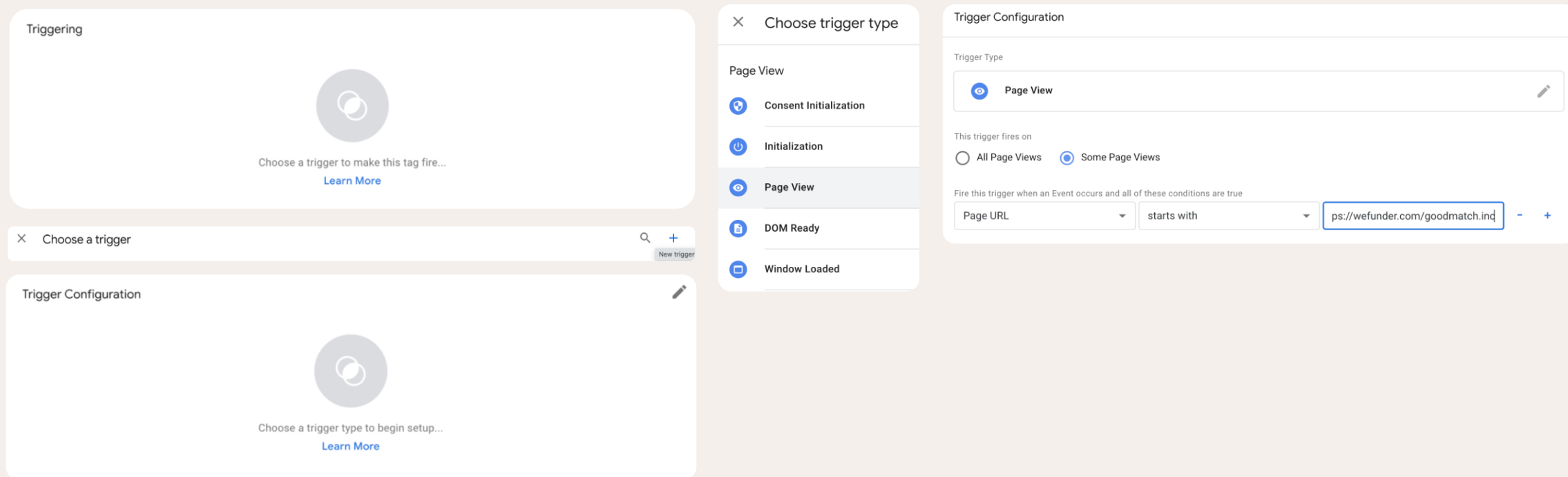
- In Triggering, choose “All page views” if you want to apply the remarketing to the whole website.

✕ Choose a trigger			
☐	Name ↑	Type	Filter
	All Pages	Page View	--
	Consent Initialization - All Pages	Consent Initialization	--
	Initialization - All Pages	Initialization	--

Website Visits

- If you only want to apply remarketing to select pages, choose those instances in the list of predefined triggers, or to create a new trigger, follow the steps below:

New trigger > Trigger configuration > Page view > Some page visits > Page URL starts with [enter url]



The screenshot displays the Pulse interface for configuring a website visit trigger. It is divided into three main sections:

- Triggering:** A large white box with a circular icon and the text "Choose a trigger to make this tag fire..." and a "Learn More" link.
- Choose a trigger:** A horizontal bar with a search icon, a plus icon, and a "New trigger" button.
- Trigger Configuration:** A large white box with a circular icon and the text "Choose a trigger type to begin setup..." and a "Learn More" link.

In the center, a "Choose trigger type" modal is open, showing a list of trigger types under the "Page View" category:

- Consent Initialization
- Initialization
- Page View (highlighted)
- DOM Ready
- Window Loaded

To the right, the "Trigger Configuration" panel is visible, showing the following settings:

- Trigger Type:** Page View
- This trigger fires on:** ☐ All Page Views ☒ Some Page Views
- Fire this trigger when an Event occurs and all of these conditions are true:**
 - Page URL starts with ps://wefunder.com/goodmatch.ind

- Name the trigger, i.e. Wefunder page visit and click save.

Website Visits

- Name the tag, i.e. PULSE DSP – WeFunder page visit and click save.

× PULSE DSP – WeFunder page visit Save

Tag Configuration

Tag Type

<> Custom HTML
Custom HTML Tag

HTML ⓘ

```
1 <script async src="https://pxl.greenbergenterprises.com/s/35267d5c-2e1b-4f59-887e-61575ba6669b"></script>
2
```

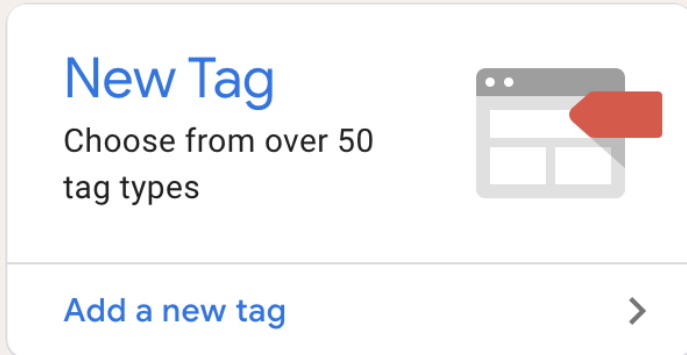
Triggering

Firing Triggers

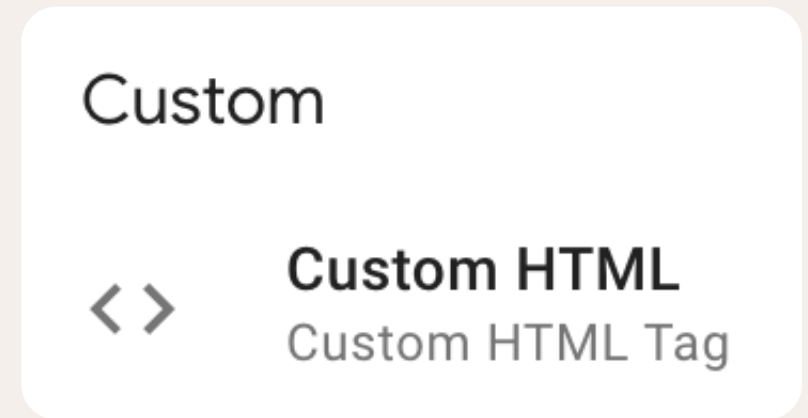
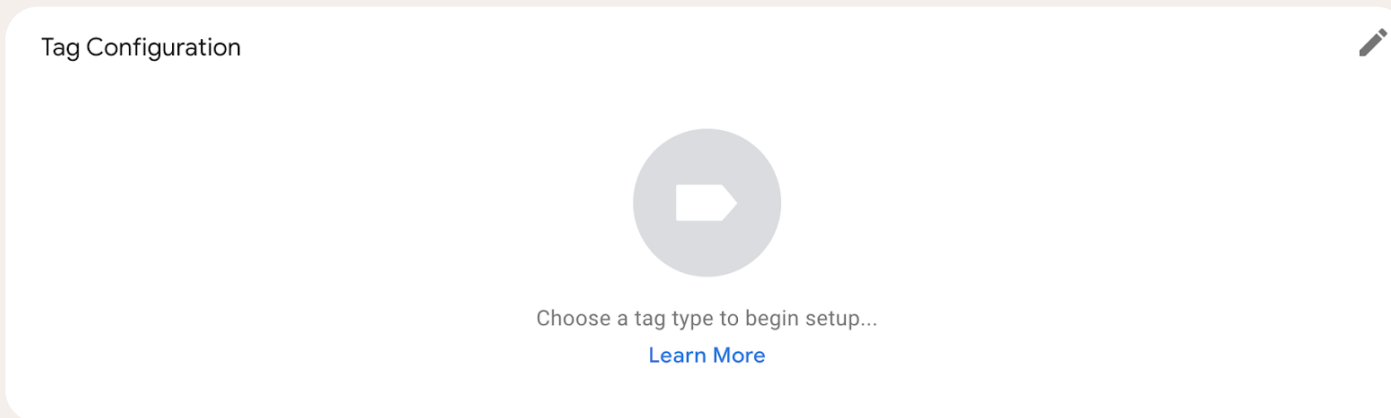
 Wefunder page visit
Page View

Website Conversions

- Log in to Google Tag Manager. Click to Add New Tag

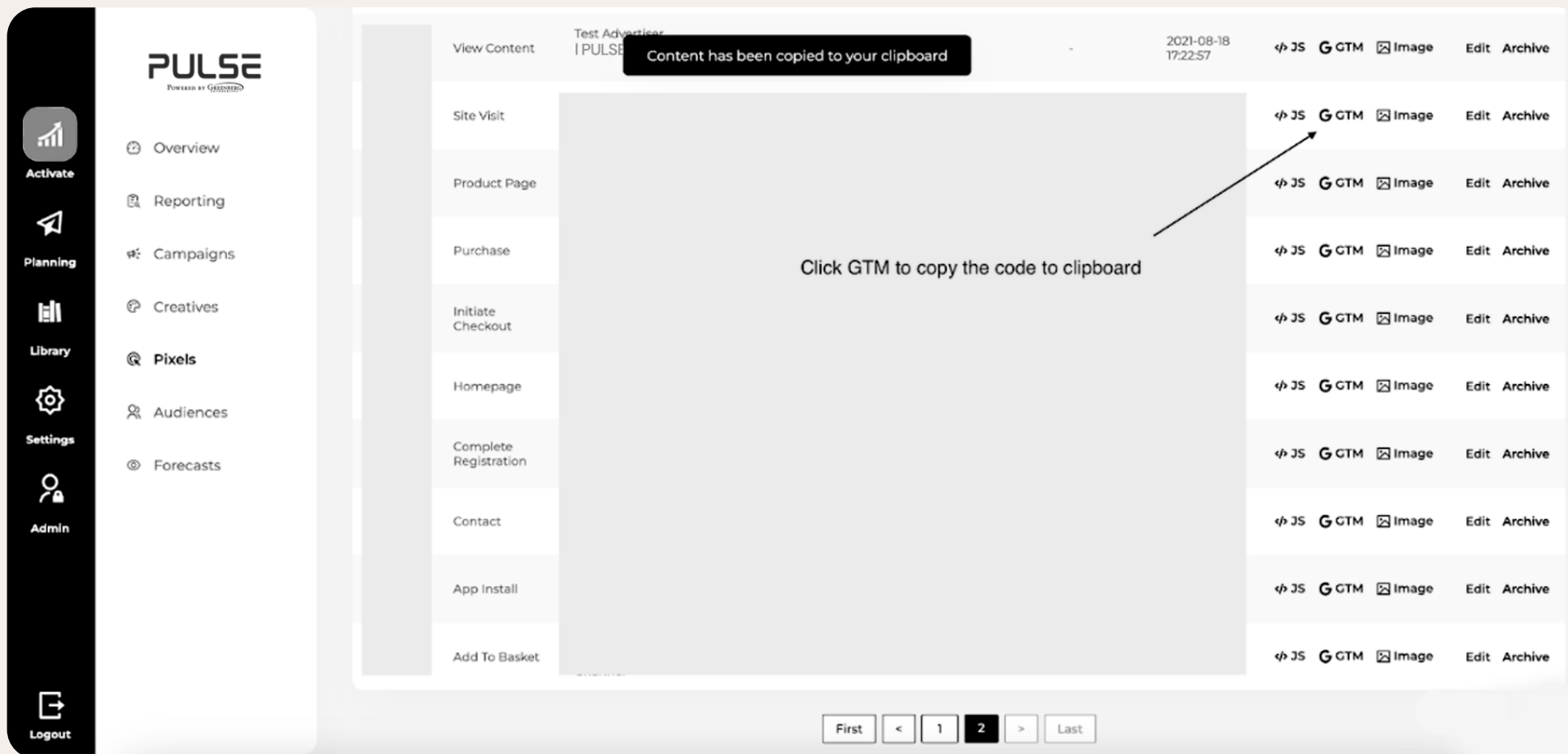


- In Tag Configuration, choose Custom HTML.



Website Conversions

- Navigate to PULSE Activate Pixels and click to “Get Pixel” for Conversion pixel type GTM.



The screenshot displays the PULSE Activate Pixels interface. On the left is a sidebar with navigation options: Activate, Planning, Library, Settings, and Admin. The main content area shows a list of conversion events. A table lists these events with columns for event type, pixel type, and actions. The 'GTM' column is highlighted, and an arrow points to the 'GTM' button for the 'Site Visit' event. A tooltip indicates 'Content has been copied to your clipboard' after clicking the GTM button. A text overlay says 'Click GTM to copy the code to clipboard'.

Event	Pixel Type	Actions
Site Visit	GTM	Edit Archive
Product Page	GTM	Edit Archive
Purchase	GTM	Edit Archive
Initiate Checkout	GTM	Edit Archive
Homepage	GTM	Edit Archive
Complete Registration	GTM	Edit Archive
Contact	GTM	Edit Archive
App Install	GTM	Edit Archive
Add To Basket	GTM	Edit Archive

Website Conversions

- Add HTML code to the Google Tag Manager as shown below
(use the pixel that you copied from the PULSE dashboard in the previous step):

Tag Configuration

Tag Type

< >

Custom HTML

Custom HTML Tag

HTML

1

<script async src="https://pxl.greenbergenterprises.com/s/90aeec97-2cdc-4248-babf-68ef385e8d9f"></script>

2

☐

Support document.write

Website Conversions

- In Triggering, choose the instances in the list of predefined triggers for conversion, or to create a new trigger, follow the steps below:

All form submissions

Choose Trigger type – All form submissions from the list.

× Choose trigger type 🔍

Page View



Consent Initialization



Initialization



Page View



DOM Ready



Window Loaded

Click



All Elements



Just Links

User Engagement



Element Visibility



Form Submission



Scroll Depth

Website Conversions

- Name the form, then choose either to trigger on all form submissions or specific forms. Click Save once complete.

× Untitled Trigger 📁 Save ⋮

Trigger Configuration

Trigger Type

 **Form Submission** 

☐ Wait for Tags 

☐ Check Validation 

This trigger fires on

☒ All Forms ☐ Some Forms

Website Conversions

Thank you page

New trigger > Trigger configuration > Page view > Some page visits > Page URL starts with [enter thank-you page]

Triggering



Choose a trigger to make this tag fire...
[Learn More](#)

Choose a trigger

New trigger

Trigger Configuration



Choose a trigger type to begin setup...
[Learn More](#)

Choose trigger type

Page View

Consent Initialization

Initialization

Page View

DOM Ready

Window Loaded

Trigger Configuration

Trigger Type

Page View

This trigger fires on

All Page Views

Some Page Views

Fire this trigger when an Event occurs and all of these conditions are true

Page URL

starts with

/m/goodmatchinc/invest?amount=

- Name the trigger, i.e. Clicked to invest and Click Save.

Website Conversions



- Name the tag, i.e. Pulse DSP – clicked to invest and Click Save.

Tag Configuration

Tag Type

< >

Custom HTML

Custom HTML Tag

HTML ?

1

<script async src="https://pxl.greenbergenterprises.com/s/90aeec97-2cdc-4248-babf-68ef385e8d9f"></script>

Triggering

Firing Triggers



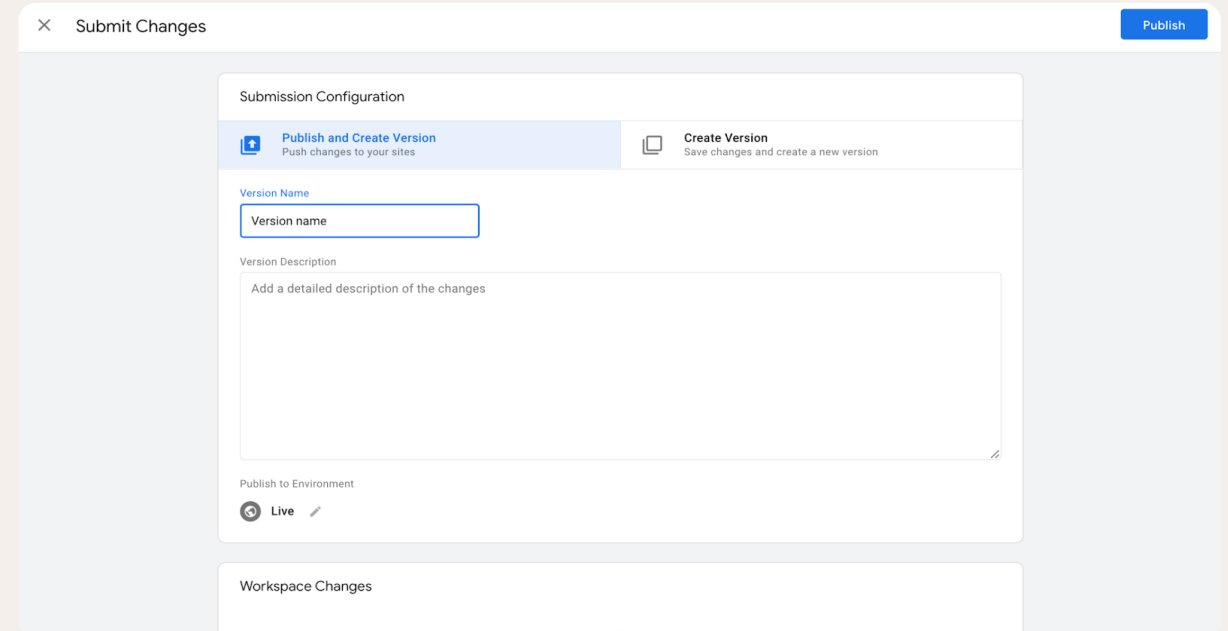
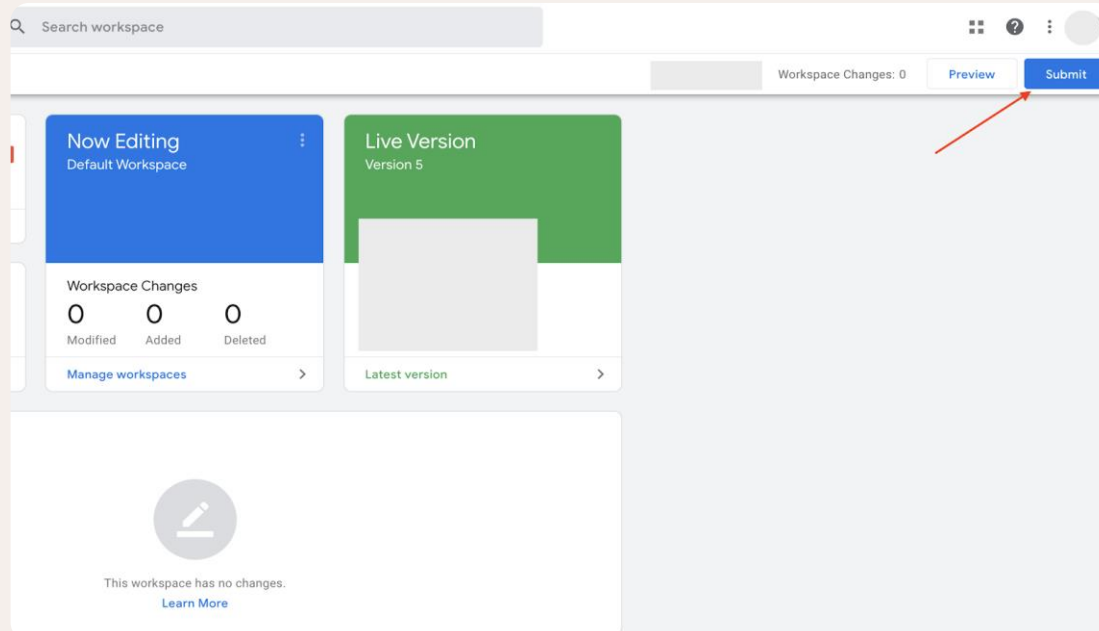
Clicked to invest

Page View

Add Exception

Website Conversions

- Submit changes and Publish Directly to the website code





Directly to the Website Code



Website Visits



- Navigate to PULSE Activate Pixels, and click on “Get Pixel” Image Tracker as shown in the screenshot below for Site Visit:

The screenshot displays the PULSE Pixels management interface. On the left is a sidebar with navigation options: Overview, Reporting, Campaigns, Creatives, Pixels, and Audiences. The main area shows a table of pixels. The table has columns for ID, Name, Advertiser, Unique Users (Yesterday), Total Loads (Yesterday), Unique Users (All Time), Total Loads (All Time), Created At, and Get Pixel. The 'Get Pixel' column contains a dropdown menu with options for JS, GTM, and Image. The 'Image' option is highlighted with a red box, indicating the correct selection for the 'Get Pixel' button.

ID ↓	Name ↑↓	Advertiser ↑↓	Unique Users (Yesterday) ↑↓	Total Loads (Yesterday) ↑↓	Unique Users (All Time) ↑↓	Total Loads (All Time) ↑↓	Created At ↑↓	Get Pixel
								</> JS G GTM Image Edit Archive
								</> JS G GTM Image Edit Archive
								</> JS G GTM Image Edit Archive
								</> JS G GTM Image Edit Archive

- The code will be copied to the clipboard once clicked.
- Add HTML code to the head tag of the page/s where you want the remarketing to trigger and save changes



Website Conversions



- Navigate to PULSE Activate Pixels, and click on “Get Pixel” Image Tracker, as shown in the screenshot below for Conversion:

PULSE
POWERED BY GREENBERG ENTERPRISES

Pixels Agency: PULSE DSP Advertiser: Test Advertiser Search: Keyword **NEW PIXEL**

ID ↓	Name ↑↓	Advertiser ↑↓	Unique Users (Yesterday) ↑↓	Total Loads (Yesterday) ↑↓	Unique Users (All Time) ↑↓	Total Loads (All Time) ↑↓	Created At ↑↓	Get Pixel
								JS GTM Image Edit Archive
								JS GTM Image Edit Archive
								JS GTM Image Edit Archive
								JS GTM Image Edit Archive

- Add HTML code to the thank-you page/s or the event triggers (i.e. button clicks) where you want the conversion to trigger.
- Save changes and publish



Testing



The PULSE Pixels dashboard is refreshed daily, and you should see instances populate with load numbers:

Activate

Planning

Settings

Admin

Overview

Reporting

Campaigns

Creatives

Pixels

Audiences

Forecasts

Pixels

Agency: PULSE DSP Advertiser: Test Advertise

Search: Keyword

Download as CSV

UUID ↑↓	Pixel Name ↑↓	Advertiser ↑↓	Total's Fire ↑↓	24 Hour Fires ↑↓	Time Since Last Fire ↑↓	Created At ↓	Get Pixel		
fe0751a1-fb37-4321-aa4b-b77c9e158ac0	GC Pricing	Test Advertiser PULSE DSP	932	9	13h 0m	2025-01-28 16:36:08	⚡ Get pixel code	Edit	Delete
e8d2ae50-b66b-4fe8-a3c2-73addf53b91f	Test Advertiser PULSE DSP	Test Advertiser PULSE DSP	100,454	1	7h 20m	2025-01-28 16:28:13	⚡ Get pixel code	Edit	Delete
036e5fc5-22a6-4ce2-a6f8-36d238773cc5	GC - Site visit	Test Advertiser PULSE DSP	144,517	181	37m 26s	2024-11-15 22:39:28	⚡ Get pixel code	Edit	Delete
41a1368e-8aba-43fa-b0f3-350b1dbbfda8	Visited DSP	Test Advertiser PULSE DSP	850	12	2h 29m	2022-12-29 04:46:09	⚡ Get pixel code	Edit	Delete

First

<

1

>

Last



Direct Website Integrations



Direct Website Integration



The guides below explain how you can add the pixels directly to the website on popular platforms such as HubSpot & Shopify.

HubSpot (Native Integration)

HubSpot often provides a native connection to major ad networks (e.g., Google Ads, Facebook) to simplify pixel deployment and allow audience syncing. The Pulse DSP Pixel must be added via Site Settings.

To obtain the tracking code for the site-wide tracking from the HubSpot Account, follow the steps below:

Login to HubSpot → Settings → Website → Pages → Tracking Code

HubSpot (Native Integration)

Reports & Analytics Tracking

Tracking code

Advanced Tracking

Analytics views

✓ Set up tracking by embedding code

Copy and paste this tracking code into every page of your site, just before the `</body>` tag.

Embed code

Detailed instructions for installing tracking code can be found [here](#). Tips for validating your tracking code installation can be found [here](#).

```
<!-- Start of HubSpot Embed Code -->
<script type="text/javascript" id="hs-script-loader" async defer
src="//js.hs-scripts.com/[account_id].js"></script>
<!-- End of HubSpot Embed Code -->
```

Copy

Email to developer

The above step is essential for HubSpot's functions. To deploy the pixel through HubSpot, ensure the tracking code is implemented across all pages.

HubSpot (Native Integration)



To place the pixels, follow the steps below:

- **Navigate to Settings → Website → Pages → Site Header/Footer HTML**

HTML

Site header HTML

Add custom code snippets to the HTML head section across all of your pages.

1

Site footer HTML

Add javascript code snippets right before the end of the body tag, across all of your pages.

1

- Place the remarketing pixel (PULSE DSP Remarketing Pixel) in the **Header HTML** for all-page fire.
- Place conversion pixels in the **Thank You Page Footer HTML** or via **Form Options → Redirect to thank-you page**.

Form Configuration (Redirect)

- **Forms → Marketing (or Content) → Lead Capture → Forms.**
- **Edit Form → Go to Options → Set Action → Choose Page → Thank You Page / External URL → Publish.**

Thank You Page HTML (Pixel Placement)

- **Pages → Content → Landing Pages/Website Pages → Thank You Page.**
- **Settings → Advance → Paste your conversion pixel code into the Footer HTML field → Publish**
- Paste the PULSE DSP Conversion Pixel code into the Header or Footer HTML section of the specific page.

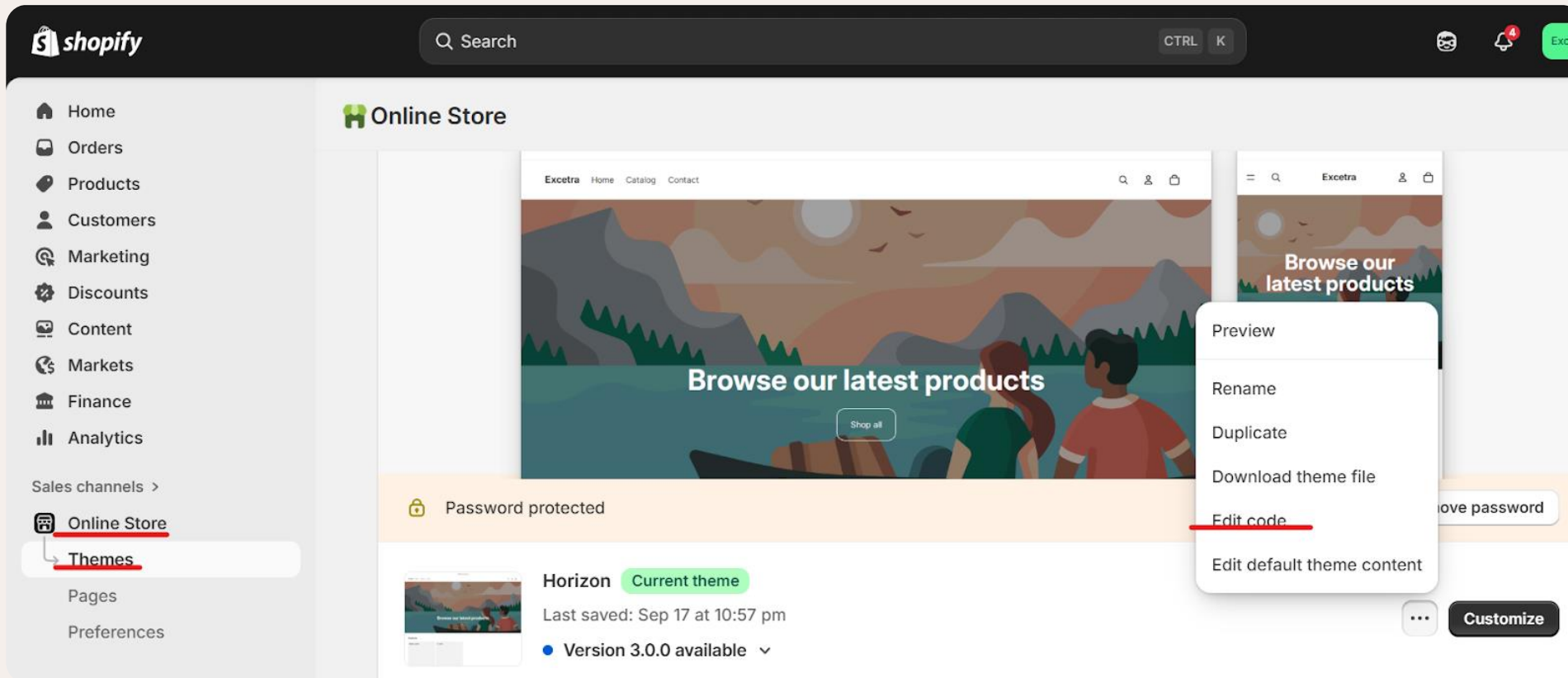
Troubleshooting:

- Use HubSpot Form Analytics to ensure submissions are tracked.
- Test in an incognito browser with pixel helper tools.
- Validate reporting in the HubSpot Ads dashboard.

shopify (Native Integration)

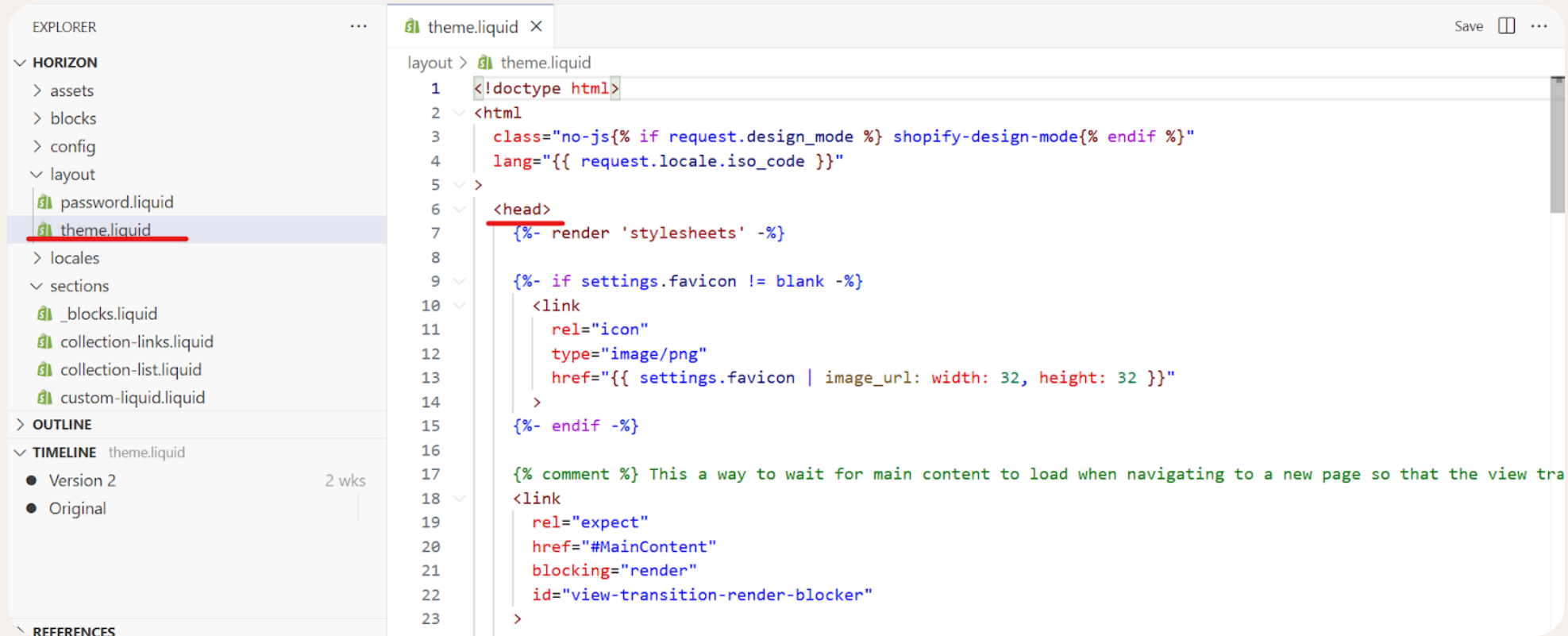
Remarketing (Site Visits)

- In Shopify Admin, go to **Online Store** → **Themes** → **Edit Code**.



Remarketing (Site Visits)

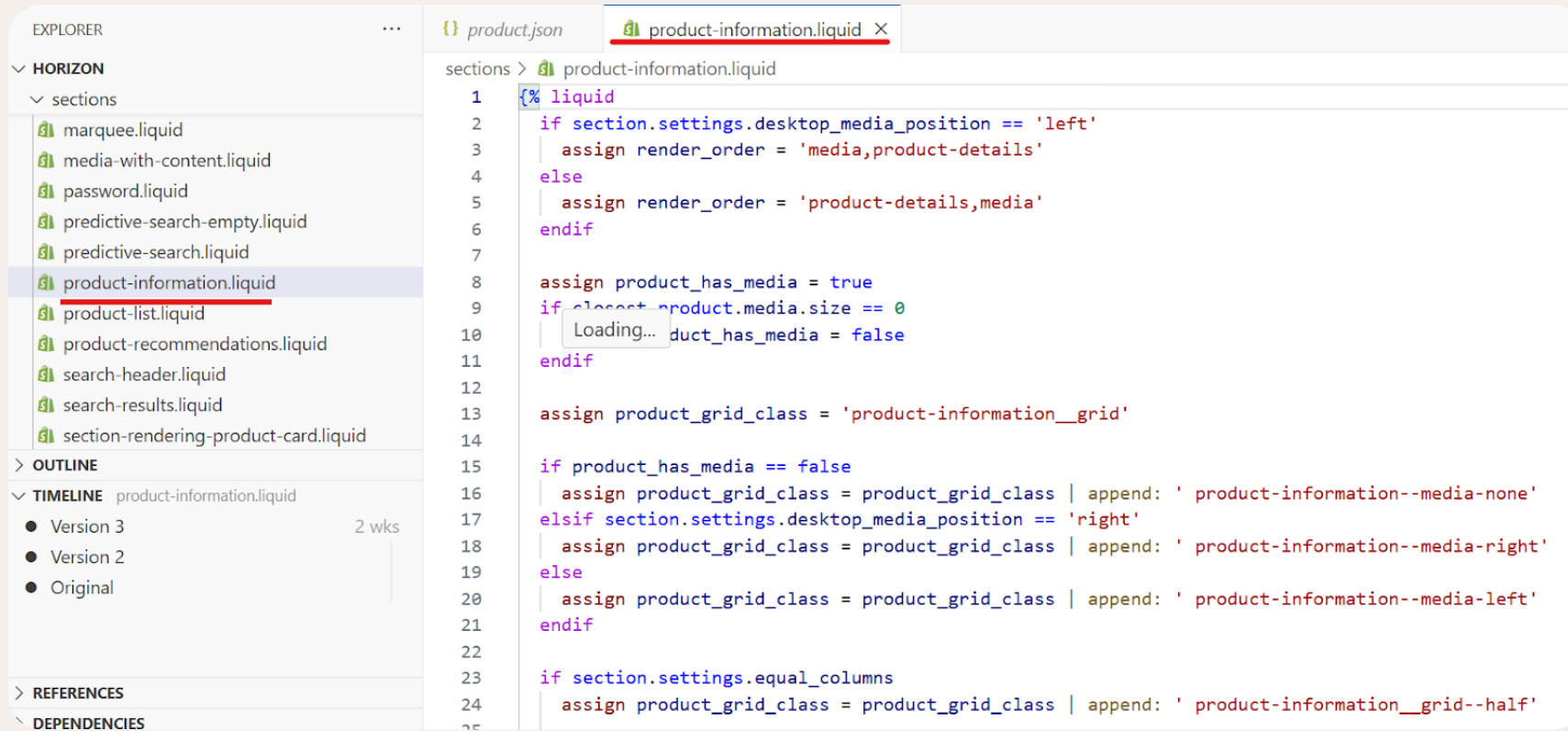
- Paste the remarketing pixel in the theme.liquid → <head> section for site-wide tracking and save changes.



```
1 <!doctype html>
2 <html
3   class="no-js{% if request.design_mode %} shopify-design-mode{% endif %}"
4   lang="{{ request.locale.iso_code }}"
5 >
6 <head>
7   {% render 'stylesheets' -%}
8
9   {% if settings.favicon != blank -%}
10     <link
11       rel="icon"
12       type="image/png"
13       href="{{ settings.favicon | image_url: width: 32, height: 32 }}"
14     >
15   {% endif -%}
16
17   {% comment %} This a way to wait for main content to load when navigating to a new page so that the view tra
18 <link
19   rel="expect"
20   href="#MainContent"
21   blocking="render"
22   id="view-transition-render-blocker"
23 >
```

Remarketing (Site Visits)

- In Themes → Edit Code → Sections → product-information.liquid (or the equivalent product detail section file in your theme).



The screenshot displays the Shopify theme editor interface. On the left, the 'EXPLORER' sidebar shows the file structure under 'HORIZON' > 'sections'. The file 'product-information.liquid' is selected and highlighted. Below it, the 'OUTLINE' section shows the 'TIMELINE' for 'product-information.liquid', listing 'Version 3' (2 wks), 'Version 2', and 'Original'. The main editor area shows the code for 'product-information.liquid' with the following content:

```
1 {% liquid
2   if section.settings.desktop_media_position == 'left'
3     | assign render_order = 'media,product-details'
4   else
5     | assign render_order = 'product-details,media'
6   endif
7
8   assign product_has_media = true
9   if product.media.size == 0
10    | Loading... product_has_media = false
11  endif
12
13  assign product_grid_class = 'product-information__grid'
14
15  if product_has_media == false
16    | assign product_grid_class = product_grid_class | append: ' product-information--media-none'
17  elsif section.settings.desktop_media_position == 'right'
18    | assign product_grid_class = product_grid_class | append: ' product-information--media-right'
19  else
20    | assign product_grid_class = product_grid_class | append: ' product-information--media-left'
21  endif
22
23  if section.settings.equal_columns
24    | assign product_grid_class = product_grid_class | append: ' product-information__grid--half'
25  endif
26 %}
```



For Product Page Visits

- Add the product pixel, passing dynamic values:

```
<script>
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({
  event: "product_view",
  product_id: "{{ product.id }}",
  product_name: "{{ product.title }}",
  product_price: "{{ product.price | money_without_currency }}"
});
</script>
```

- Save changes.



For Add to Cart

- In Themes → Edit Code → main-cart.liquid. Save and test with Pixel Helper tools.

```
1  
2  
3 <cart-items-component  
4   class="cart-items-component"  
5   data-section-id="{{ section.id }}"  
6 >  
7 <div class="section-background color-{{ section.settings.color_scheme }}"></div>  
8 <div  
9   class="section color-{{ section.settings.color_scheme }} section-{{ section.settings.section_width }}"  
10 >  
11   <div  
12     class="cart-page spacing-style{% if cart.empty? %} cart-page--empty{% endif %}"  
13     style="{% render 'spacing-style', settings: section.settings %}"  
14   >  
15     <div class="cart-page_title">  
16       {% content_for 'block', id: 'cart-page-title', type: 'cart-title' %}  
17     </div>  
18  
19     <div class="cart-page_items">  
20       {% content_for 'block', id: 'cart-page-items', type: 'cart-products' %}  
21     </div>  
22  
23     {% unless cart.empty? -%}  
24     <div class="cart-page_summary">  
25       {% content_for 'block', id: 'cart-page-summary', type: 'cart-summary' -%}
```

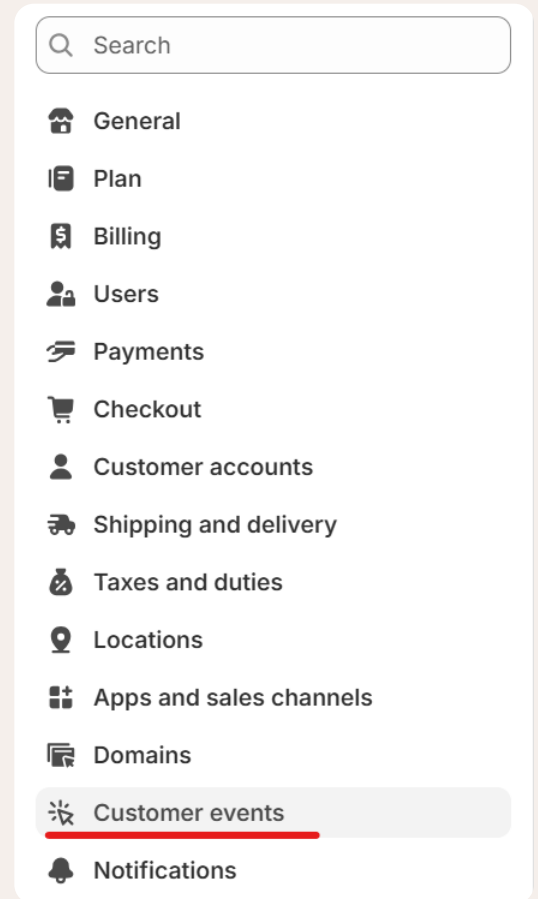
shopify (Native Integration)



For Checkout

Note: The legacy checkout.liquid and “Additional Scripts” are deprecated for Shopify Plus. Use Customer Events → Custom Pixels instead.

- In Shopify Admin, go to **Settings → Customer Events**.
- Click Add Custom Pixel.
- Place the PULSE DSP conversion pixel code inside Customer Events.
- Save and publish.
- Run a test checkout and verify via Pixel Helper + Customer Events logs.





Subscribing to Custom Events

You can also subscribe to custom-defined events over custom pixels.

Example: tracking an email signup event:

```
analytics.subscribe("special_email_signup", event => {  
    window.dataLayer.push({  
        event: "emailSignUp",  
        emailCampaignId:  
event.customData.email_campaign_id  
    });  
});
```



Using Google Tag Manager as a Custom Pixel

Shopify allows users to register as a Google Tag Manager (GTM) as a Custom Pixel.

- Tutorial: Shopify – [GTM Custom Pixel Guide](#)
- This method lets you manage all tags centrally in GTM while staying compatible with Shopify Plus Checkout.

Troubleshooting:

- Always test in a non-production environment or on a duplicate theme before publishing changes live.
- Use Shopify's Theme Preview to confirm your pixel code and preview any changes.
- Check Shopify's Customer Events activity log to see if the pixel fired on the checkout_completed event.
- Run a test order to confirm the pixel fires on real checkouts.
- Wait for 24 hours



Reference Documentation

The below can be referred to for further clarification on the use of Shopify Custom Pixels and GTM as a Custom Pixel.

- Shopify Custom Pixels: [Shopify – Custom Pixels Documentation](#)
- GTM as a Custom Pixel: [Shopify – GTM Tutorial](#)



Liquid Version

Shopify pixel implementation · Customer Events → Custom Pixels (GTM)

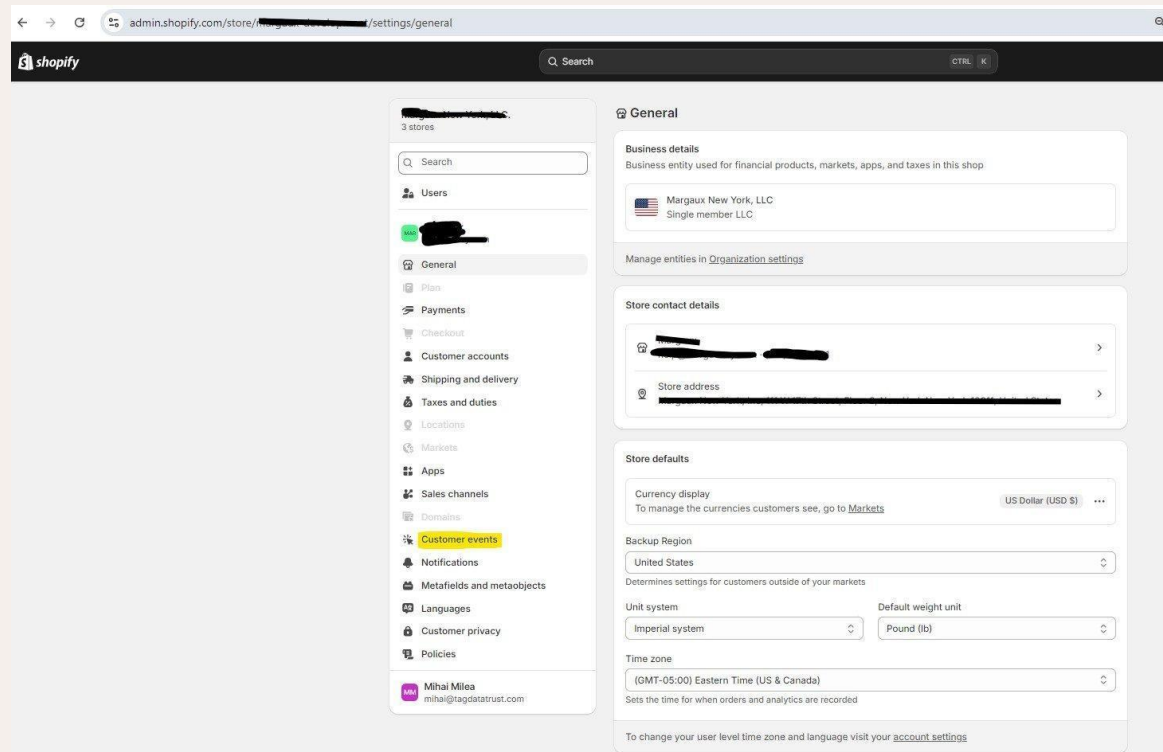
Before You Begin

- **Prerequisite:** Admin access to your Shopify store.
- **Prerequisite:** Your GTM Container ID (format: GTM-XXXXXXX), found in your Google Tag Manager account.
- **Note:** Depending on the Shopify plan type, some legacy checkout customization methods (including checkout.liquid customization and Additional Scripts behavior) may be restricted or deprecated by Shopify.
- **Recommended:** Implement GTM and conversion tracking through Settings → Customer Events → Custom Pixels, rather than relying on legacy Additional Scripts implementations.



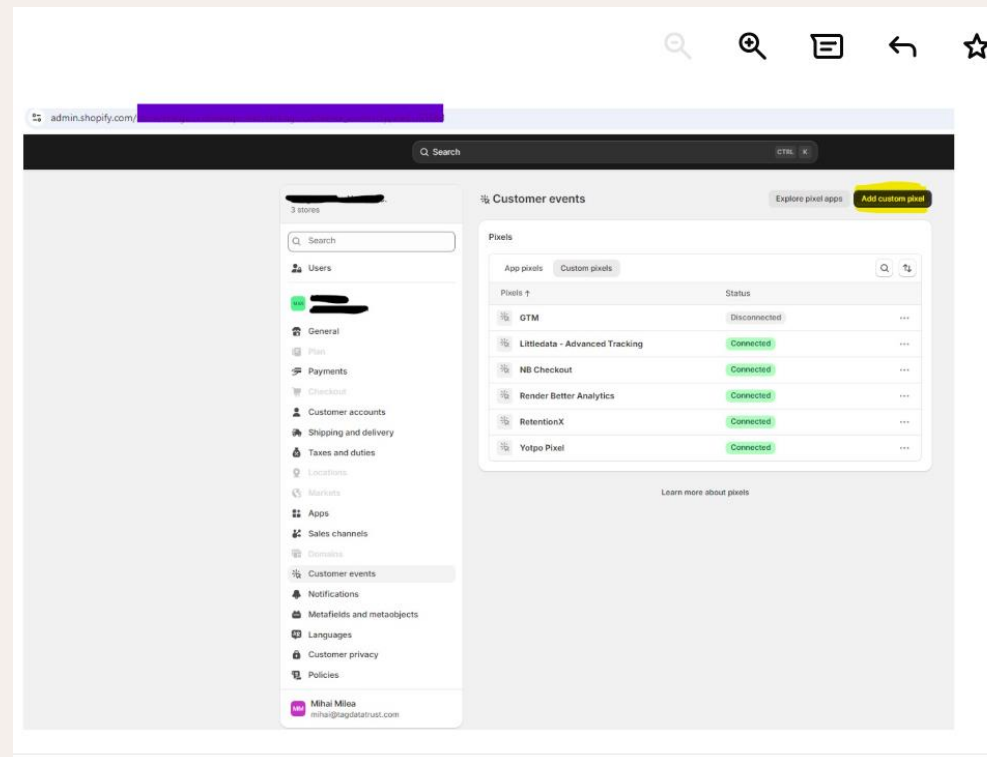
Step 1: Navigate to Customer Events

- From your Shopify Admin, go to Settings in the left-hand sidebar. Scroll down and click Customer Events.



Step 2: Open the Custom Pixels Tab

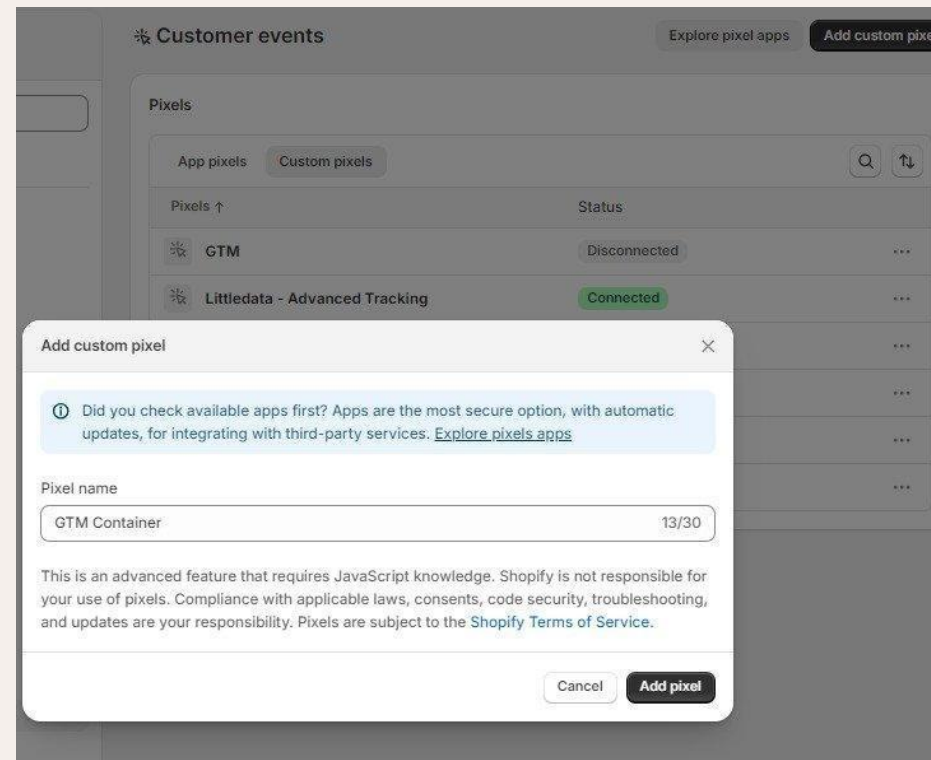
- On the Customer Events page, click the Custom pixels tab. Then click the Add custom pixel button in the top-right corner.





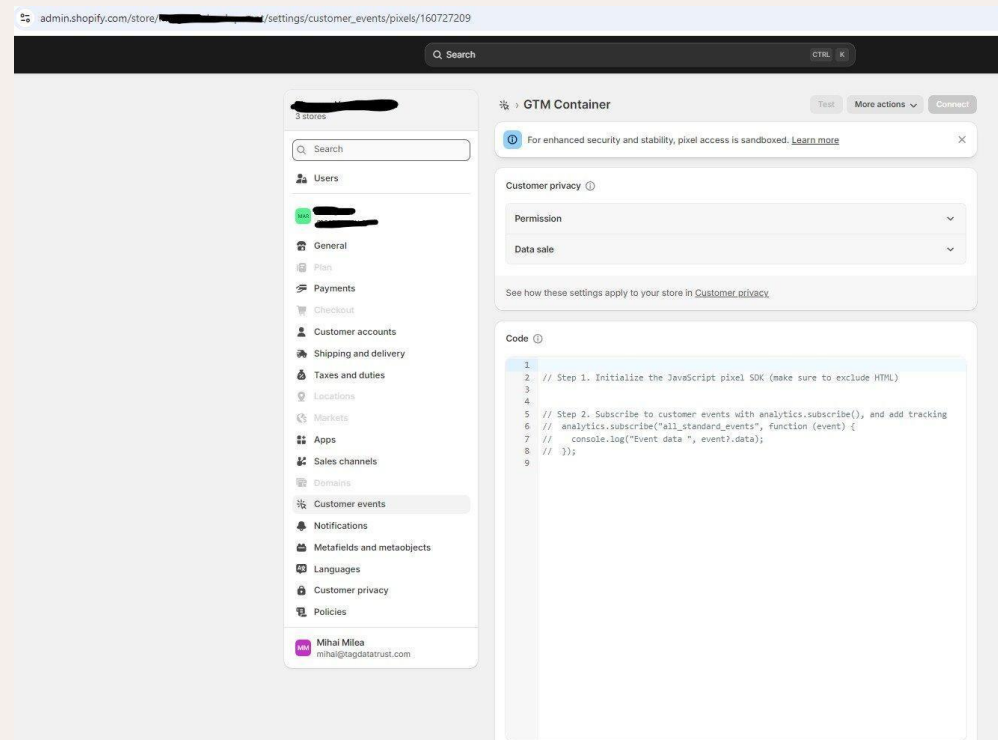
Step 3: Name Your Custom Pixel

- A dialog will appear asking for a Pixel name. Enter a descriptive name such as “GTM Container”. Then click Add pixel.



Step 4: Configure the Pixel Code Editor

- Shopify opens the pixel editor for your GTM Container pixel, showing a Code section with placeholder comments. Clear all the placeholder code, then paste the complete GTM pixel code provided next.



Step 5: Paste the GTM Pixel Code

Copy the full code block below and paste it into the Code editor, replacing all existing content.

```
const GTM_container_url = 'https://www.googletagmanager.com';
const GTM_container_id = 'GTM-XXXXXXX'; // ← REPLACE THIS
if (window.location.href.includes('/checkouts/cn/')) {
  // tag manager
  (function (w, d, s, l, i) {
    w[l] = w[l] || [];
    w[l].push({ 'gtm.start': new Date().getTime(), event: 'gtm.js' });
    var f = d.getElementsByTagName(s)[0],
        j = d.createElement(s),
        dl = l != 'dataLayer' ? '&l=' + l : '';
    j.async = true;
    j.src = GTM_container_url + '/gtm.js?id=' + i + dl;
    f.parentNode.insertBefore(j, f);
  })(window, document, 'script', 'dataLayer', GTM_container_id);
  analytics.subscribe('checkout_completed', (event) => {
    const checkout = event.data.checkout;
    // Optional: map product IDs from line items
    // const productIds = event.data.checkout.lineItems.map(item => item.product.id);
    window.dataLayer.push({
      event: 'transactionCompleted',
      transactionId: checkout.order?.id || checkout.id,
      revenue: checkout.totalPrice?.amount,
      currency: checkout.currencyCode,
      // productIds: productIds
    });
  });
  // end tag manager
}
```

Step 6: Replace the Required Variables

Before saving, you must replace the placeholder value for the GTM container ID with your own. You may also need to change the default variables. The table below lists every variable you may need to update:

Variable	Example value	Should change?
GTM_container_id	GTM-TN47S6P9	Must use your own GTM
transactionId	checkout.order?.id	May need to change
revenue	checkout.totalPrice?.amount	May need to change
currency	checkout.currencyCode	May need to change



Validation Requirements

Successful pixel firing alone does not confirm a valid conversion setup. After implementation, validate that the following values are correctly populated in the dataLayer after checkout completion:

- **transactionId**
- **revenue**
- **currency**
- **productIds**: optional, only if you map product IDs

If these values are missing or incorrect, conversions may fire without passing usable attribution or revenue data into reporting platforms.

Step 7: Add product_id Values (Optional)

Un-comment the lines referencing product_id values, so your code looks like this:

```
// Un-comment the two product-ID lines:

const productIds =
  event.data.checkout.lineItems.map(item => item.product.id);

window.dataLayer.push({
  event: 'transactionCompleted',
  transactionId: checkout.order?.id || checkout.id,
  revenue: checkout.totalPrice?.amount,
  currency: checkout.currencyCode,
  productIds: productIds
});
```

Note

You may need to replace the default variables with values specific to the client's Shopify theme, i.e. how product IDs are read from lineItems. The highlighted values may need to change.

Testing Best Practices

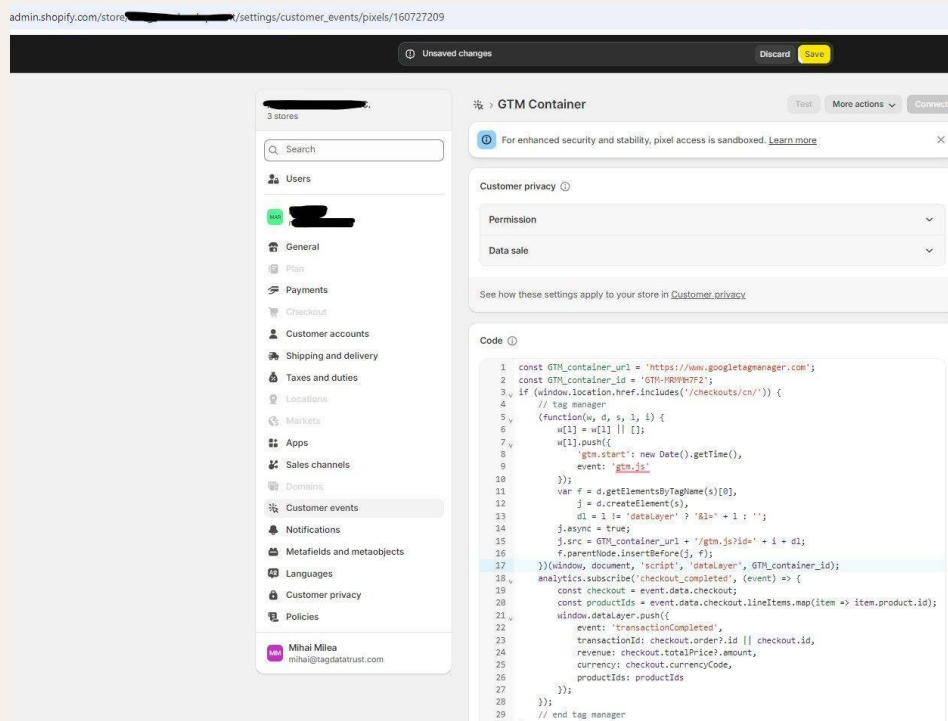
When validating a new implementation:

- Use a low-cost product or accessory for testing.
- Create a dedicated test discount code if possible.
- Notify the account or implementation team before placing test orders.
- Confirm **transactionId, revenue, and currency** values are properly passed into the dataLayer.
- Cancel / refund test orders after validation is complete.

Step 8: Save and Connect the Pixel

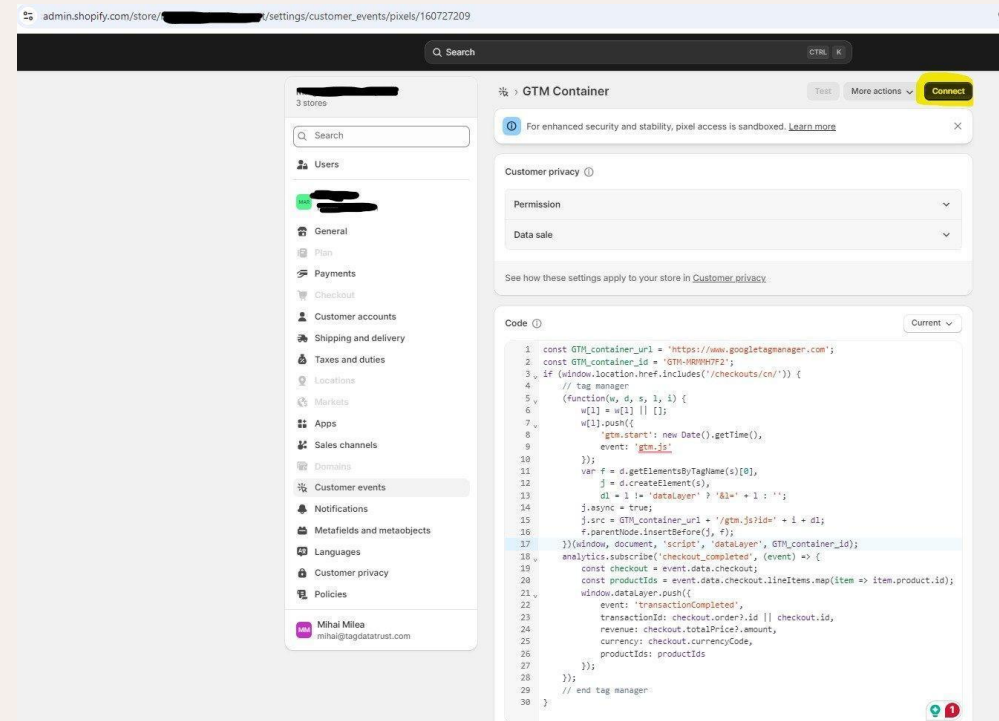
- Once you have replaced the variables and reviewed the code, click Save in the top banner; the unsaved-changes banner disappears. Then click Connect in the top-right corner to activate the pixel.

Save



The screenshot shows the Shopify admin interface for a store. The top banner displays "Unsaved changes" with a "Save" button highlighted in yellow. The left sidebar shows the navigation menu with "Customer events" selected. The main content area shows the "GTM Container" settings, including a warning about sandboxed pixel access, a "Customer privacy" section, and a "Code" section with a JavaScript snippet for the GTM container.

Connect



The screenshot shows the Shopify admin interface for a store. The top banner displays "Unsaved changes" with a "Connect" button highlighted in yellow. The left sidebar shows the navigation menu with "Customer events" selected. The main content area shows the "GTM Container" settings, including a warning about sandboxed pixel access, a "Customer privacy" section, and a "Code" section with a JavaScript snippet for the GTM container.

QA: Common Issues & Troubleshooting

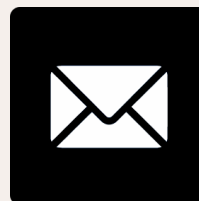
- **Pixel fires but no transactionId is captured.** Usually a variable-mapping issue, not a deployment failure. Validate: checkout.order?.id, checkout.id, the event.data.checkout structure, GTM dataLayer values, and event timing after checkout completion.
- **GTm not appearing during checkout.** Some Shopify plans restrict or alter checkout customization. Recommended path: Settings → Customer Events → Custom Pixels rather than legacy Additional Scripts methods.
- **Revenue or currency values missing.** Confirm checkout.totalPrice?.amount and checkout.currencyCode match the client's Shopify implementation and are available during checkout_completed events.

QA: Common Issues & Troubleshooting (cont.)

- **Duplicate conversion events.** Review: duplicate GTM containers, multiple Purchase event triggers, overlapping pixel implementations, and direct pixel installs alongside GTM-managed installs.
- **Event appears in GTM but not in platform reporting.** Validate: trigger conditions, event naming conventions, platform pixel IDs, deduplication settings, attribution windows, and browser consent settings.

Finally...

If you have issues or need assistance, let us know.
Once completed, please let us know so we can test and confirm that we are seeing traffic. Email us at **greenbergadops@gmail.com**



Once confirmed, get ready for your campaign to kick off! Please make sure you've granted GA4 access to **greenbergadops@gmail.com**

