Shopify ↔ GA4 auto tracking

Shopify has automatic tracking for the following Google Analytics 4 events

1. **page_view** – A customer visited a page on your online store.
2. **search** – A customer searched for a product on your online store.
3. **view_item** – A customer viewed a product on your online store.
4. **add_to_cart** – A customer added a product to cart.
5. **begin_checkout** – A customer started the checkout process.
6. **add_payment_info** – A customer successfully entered payment information.
7. **purchase** – A customer completed their checkout.

Legend

- **we don't use** – search is hidden from site
- **checkout journey** – the conversion is part of the checkout process

These events we already have tracking for.

Micro conversions on website

All the below are browsing-based, except "Sign up" and unless marked with a **LEAD** *event. Lead event is any micro conversion where we get contact details as opposed to just a browsing event, such as opened configurator page.*

- Page views (multi-count)
  - **Seen a bike landing page** (RM1 link, RM1S link)
  - **Seen configurator**
  - **Seen Test Ride Booking page**
  - **Seen finance page**
  - **Seen New Riders / CBT (or another regional license) page**
- Page views (once-in-session count)
  - **Seen a bike landing page** (RM1 link, RM1S link)
  - **Seen configurator**
  - **Seen Test Ride Booking page**
  - **Seen finance page**
  - **Seen New Riders / CBT (or another regional license) page**
- Bike configurator
  1. **Opened configurator**
  2. **Progressed to page *n***
  3. **Saved configuration LEAD**
  4. **Reached Summary page**
  5. **Applied for finance UK ONLY**
  6. **Added to basket**

- Other products (incl. RM1S pre-order product, soon deprecated)
  1. **Seen product**
  2. **Added to basket**

- Test rides
  1. **Selected model**
  2. **Selected location**
  3. **Selected date and time**
  4. **Added contact details**
  5. **Booked a test ride LEAD**

- Finance Calculator

1. **Seen / Opened finance calculator**
2. **Used finance calculator**
3. **Clicked CTA DE ONLY**
4. **Applied for Finance DE ONLY**

- **Sign up**
    - Newsletter **LEAD**
    - New Rider Programme **LEAD**

- Used contact form **LEAD**
- Interacted with HubSpot chatbot **LEAD**

Offline conversions

- Email sent to Customer Service
- Inbound phone call with Customer Service / Sales
- Outbound phone call with Customer Service / Sales
- Finance application accepted
- ~~Pre-order edited with final configuration~~
- ~~Final configuration paid~~

Lead status

Lead status is defined based on the contact information we've got for them.

	Engaged with ads / website	Email address	Engaged with emails	Phone number	Address
Lead	✅				
Subscriber	possibly	✅			
MQL	✅	✅	✅		
SQL	✅	✅	possibly	✅	
Customer	✅	✅	possibly	✅	✅

Finance Calculator and finance applications – market differences

The Finance Calculator is an independent **theme section** that can be placed on different pages. The current placement includes:

- Embedded on the Finance page
- A drawer which can be opened in the Configurator

Category	Action	UK	US	DE	FR
Finance Calculator	**Can be used for illustrative purposes**	✅	✅	✅	✅
	CTA			✅	
	User can apply for finance			✅	
Finance application process	**Application not on maeving.com**	✅	✅	✅	
	Apply from Configurator Summary page	✅			
	Apply from Ekho Checkout		✅		
	Apply from Finance Calculator			✅	
	Apply from Shopify Checkout				✅

What happens with finance applications in offline? How is this recorded in HubSpot? ￼

- See section Offline conversions

Questions:

1. What about specific tracking on the homepage (e.g. how can we gauge the % of clicks from total homepage session and where these occur – i.e. when someone clicks the RM1 or the RM1S from the feature cards section or the new rider section or the about us section)…

Tracking dev briefs

Testing events to be pushed to GA4 in **developer / debug mode**.

Overview brief

We're looking for someone experienced with Google Analytics 4 and Shopify's Web Pixels API to 1) audit our current GA4 setup and 2) add custom event tracking for a few generic, site-wide events (e.g. button clicks) and a handful of section-specific events (e.g. feature card flips).

Current setup

We use the Shopify Google app to track 7 standard events. We also have the GA4 "Enhanced measurement events" feature switched on. For feeding Meta Ads, we use the Shopify Meta app. We don't use GTM and aren't planning to; instead, we want to build our tracking using Shopify's Web Pixels API and Custom pixels.

Deliverables

1. **GA4 audit**
   - Audit GA4, note good and bad practice, suggestions for improvement.
   - Document Custom events. Using GA4 definitions, that's any event not part of either of these 3 groups: 7 standard events fed by Shopify via the Google app, GA4 Automatically collected events and GA4 Enhanced measurement events (should we switch off the latter?).
2. **Shopify theme file edits**
   - Create a liquid snippet file to hold all tracking events. File should accept variables and load events conditionally (e.g. generic events should load on all pages but card flip events should only load with the Feature Cards section etc).
   - Publish events using Shopify's Web Pixels API (Shopify.analytics.publish()).
     - Standard Events (excluding all standard events Shopify already passes to GA4 – we don't want double-counting)
     - DOM Events
     - Custom Events (as specified in a detailed brief to follow)
   - Document every event in the snippet with code comments describing how the event is triggered.
3. **Create a Custom pixel for additional GA4 events**
   - Subscribe to all events in custom pixel and push them to GA4 with all their event parameters.

Configurator tracking brief

Event names	Event parameters			Notes
	step_number	step_name (screen title)	Additional params	
cfg_step_view	✅	✅		
cfg_option_select	✅	✅	option_name (product title)	
cfg_option_pre-selected	✅	✅	option_name (product title)	triggered when the step is reached
cfg_save_email	✅	✅		
cfg_save_phone	✅	✅		
cfg_summary_option_change	✅	✅	option_name, new_value	
cfg_button_click	✅	✅	button_text	for all button clicks: arrows, Next, Save, Skip, Back, Add to Bag, Finance etc.
cfg_gallery_swipe	✅	✅	image_number_start, image_number_end, direction [left | right]	
cfg_page_swipe	✅	✅	direction [forward | backward]	
cfg_popup_opened	✅	✅	link_text	
cfg_finance_redirect	✅	✅	price	triggered when "Apply for HP finance" is clicked and the user is redirected to the finance partner's website. Note: different to clicking the Finance button on the Summary page – that should be recorded under cfg_button_click. Note: the price param can be fetched from the

				[data-price] attribute of DOM element #finance-application

Test Ride Booking tracking brief

Event names	Event parameters			Notes
	step_number	step_name (screen title)	Additional params	
trb_step_view	✅	✅		
trb_option_select	✅	✅	* option_name (product title / location name)	
trb_user_location_request	✅	✅		Clicking the “Near me” button
trb_user_location_allowed	✅	✅		User accepting location permission
trb_date_selected	✅	✅		Click on a date option
trb_time_selected	✅	✅		Click on a time option
trb_contact_details_added	✅	✅		Fires once per page visit when all 5 contact detail input fields have data that passes validation and none of them are in focus (i.e. the user has clicked away outside of input fields or clicked the Submit button)

Finance Calculator tracking brief

Events and event-specific parameters

1. **Finance calculator view** (applies to drawer and section instances)
   event name: finance_calculator_view
   event parameters:
   a. instance_type='drawer' || 'section'
2. **Finance calculator open** (applies to drawer instances only)
   event name: click
   event parameters:
   a. action='open drawer'
3. **Slider use**
   event name: slider
   event parameters:
   a. element='deposit' || 'term'
   b. action='increase' || 'decrease'
   c. start_position
   d. end_position
4. **Summary open**
   event name: click
   event parameters:
   a. action='open summary dropdown'
5. **Summary close**
   event name: click
   event parameters:
   a. action='close summary dropdown'
6. **Click on CTA** – record the values at the time of clicking the CTA
   event name: click
   event parameters:
   a. CTA_text
   b. URL – URL path for internal links / full URL for external links
   c. base_price
   d. deposit
   e. term
   f. repayments

7. **End state of finance calculator** – record the values at the time of closing the drawer or leaving the page (including when the CTA is clicked)
   event name: finance_calculator_use
   event parameters:
   a. CTA_text – only if CTA is present
   b. action='close drawer' || 'left page' || 'CTA clicked'
   c. base_price
   d. deposit
   e. term
   f. repayments

Generic event parameters

1. **All events** should also have the following parameter:
   a. product='RM1' || 'RM1S' – value from dropdown (though in some instances we don't show the dropdown, I believe we can still pass the product in the context of which the instance is deployed)
2. **Events 2-6** are standard events which apply to many other components so these should also have the following parameter:
   a. component='finance calculator'

Flip Cards tracking brief

Events and event-specific parameters

1. **Feature card seen** – triggered when element is inside the viewport
   event name: feature_card_view

2. **Feature card open**
   event name: feature_card_open

3. **Feature card close**
   event name: feature_card_close
   additional event parameters:
   a. open_duration_in_seconds = integer – number of seconds the flip card was open

Generic event parameters

All events should have the following parameters:

1. title = string
2. image = bool – does the card have an image or not
3. flip_side = bool – does the card have a flip side; is it open-able / flip-able
4. card_position = integer – the position number in the list of cards
5. total_cards = integer – the total number of cards in the list
6. cards_visible_on_page_load = integer – the number of cards visible on page load (in most cases that would be, for example, 2 on mobile, 3 on tablet, 5 on desktop etc.)
7. card_visible_on_page_load = bool – was the card which the event applies to visible on page load

Other website tracking

Using GA4 definitions, there are a few groups of events that are already automatically tracked by Shopify and/or/with GA4:

1. 7 standard events fed by Shopify via the Google app (more info in section Shopify ↔ GA4 auto tracking in this document)
2. GA4 Automatically collected events
3. GA4 Enhanced measurement events

Apart from those, we need to track **a few new custom events**.

New custom events

All events should have **event parameters** – such as page, button/dropdown text, href target, alt text etc – which we can use in GA4 to define the conversion events.

1. begin_checkout_external
   On our US store we don't use the Shopify checkout – we use an external checkout. We can't manually push standard Shopify events such as begin_checkout so we need a new custom event to track outbound clicks to external checkouts.

2. button_click
   Leveraging event filters, we can set up all sorts of conversions off the back of generic click events. Tracking all button clicks should require one single block of javascript code present on all pages that uses the correct CSS selectors to target buttons in different sections (i.e. the button in the banner section would use a different set of CSS selectors to that in the feature cards, for example). **What we don't want** is to have individual bits of code to track a set number of buttons we define now (for example, "the Configure button in the RM1 hero section on the homepage"); we want to track **all button clicks** as button_click, for example.

3. **Dropdowns**
   a. dropdown_open
   b. dropdown_close

   These should have parameter dropdown_title

4. **HubSpot form submits**
   Not sure if this can be set up server-side on HubSpot – please advise. If not

possible, we'd have to set up universal client-side tracking that loads on all pages and works for all forms, old and new.

~~5. **Card clicks**~~

~~We also want to **track clicks on product cards, article cards**. These are not necessarily buttons so I'm happy to add another event to track – card_click.~~

Page views of key pages | MAEVING TO ACTION

1. Test Rides
2. New Riders
   + set up cookie
3. Finance
4. …