

Cookbook - Google analytics and Tag manager

- Overview SaaS
- Integration
 - Google Analytics
 - Google Tag Manager
 - Google Analytics Commerce Events
- Resources
 - Google structured data tool

Overview SaaS

Understanding traffic of your commerce channel is extremely important in unifying your branding strategy, marketing strategy and personalised content strategy.

In this article we will go through step by step instructions to integrate Google analytics and Google tag manager in order to enable these insights in SaaS default themes.

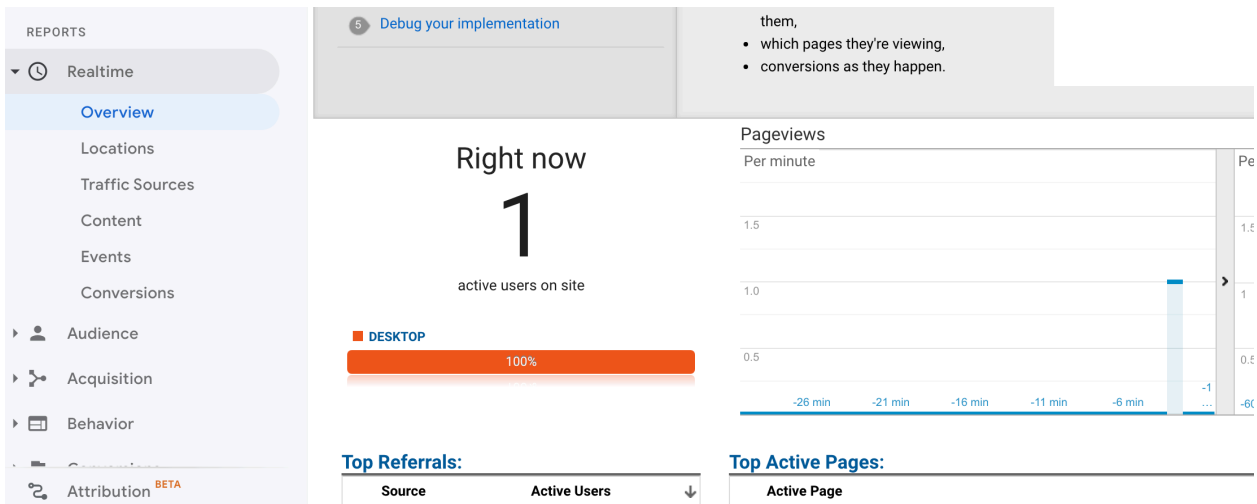
Prerequisites:

- You have a google account (<https://www.google.co.uk/> and click sign in to start the account creation process)
- You have enabled google analytics (<https://support.google.com/analytics/answer/1008015?hl=en>)
- You have enabled google tag manager (<https://support.google.com/tagmanager/answer/6103696?hl=en>)
- You have installed "Tag Assistant" Chrome extension (<https://chrome.google.com/webstore/detail/tag-assistant-by-google/kejbdjndbnbjgmefkgddjlbokphdefk?hl=en>)

Integration

Google Analytics

1. Start with creating analytics account
 1. Enter account name
 2. Specify that you wish to track website data
 3. Add website name, URL, industry and time zone
2. Under Admin > Property > Property settings > Tracking code, copy "Global Site Tag (gtag.js)" code
3. Login to Admin app and navigate to CMS section
4. Search for "header_include" and add the code to **all** CMS body element for all languages. If the include does not exist create one using SHOPCODE_header_include URI pattern
5. Save the content and clear cache
6. Go back to Analytics panel and click Property > All website data > Realtime > Overview
7. View home page on your website
8. Refresh the realtime overview - it should say that there are visitors on the website



Google Tag Manager

1. Start with creating tag manager account
 1. Specify account name, country, container URL and set target platform as Web
2. Copy two snippets provide for the "head" and "body" tags
3. Create a new Tag, for example Google Analytics: Universal and set it to track all pages, use your GA tracking ID as a variable
4. Submit and publish container
5. Login to Admin app and navigate to CMS section
6. Search for "header_include" and add the "head" code to **all** CMS body element for all languages. If the include does not exists create one using SHOPCODE_header_include URI pattern
7. Save the content
8. Search for "body_include" and add the "body" code to **all** CMS body element for all languages. If the include does not exists create one using SHOPCODE_body_include URI pattern
9. Save the content and clear cache
10. View home page on your website
11. Open "Tag Assistant" extension, click "enable" and click "start recording"

Google Tag Assistant

Result of Tag Analysis

3 In total

1. Global site tag (gtag.js)

UA-171280641-1

2. Google Analytics

UA-171280641-1

3. Google Tag Manager

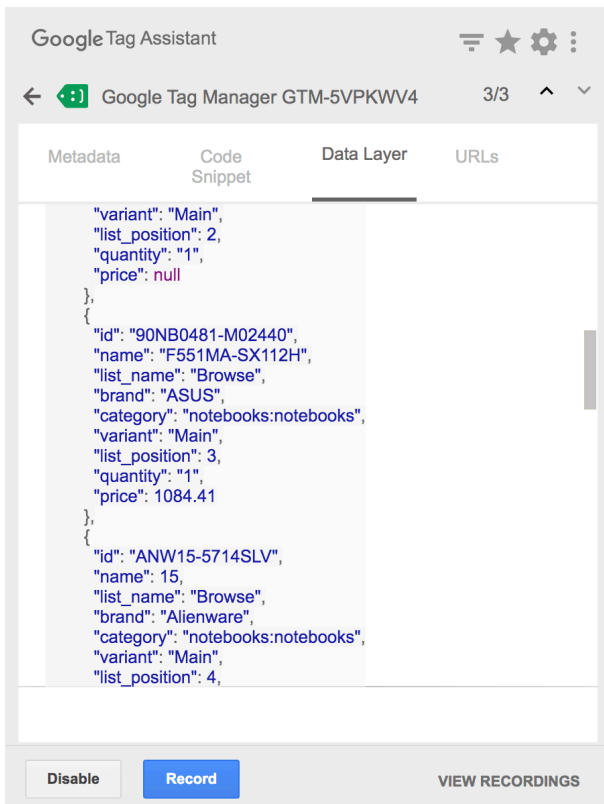
GTM-M34PGMJ

STOP RECORDING

VIEW RECORDINGS

Google Analytics Commerce Events

1. Login to Google Analytics account
2. Open property > Property view > Ecommerce settings and click enable Ecommerce and save changes
3. Login to Admin app and navigate to CMS section
4. Search for "footer_include" and add the below code (click expand button to view) to **all** CMS body element for all languages. If the include does not exist create one using SHOPCODE_footer_include URI pattern
5. Save the content and clear cache
6. Visit your site
7. Open "Tag Assistant" extension, click "enable" and click "start recording"
8. Go to the page and try to go through the checkout process ⚠️ Note that it may take up to 24h for the event to appear in google analytics
 1. Check tags in the extension, click "Check contained tags" to load data layer, click data layer tab



For versions up to 4.0.0 footer_include must contain full GA/GTag script. Below is an example integration script for default theme

GA commerce integration (up to 4.0.0)

```
<script>
  /*
  https://developers.google.com/analytics/devguides/collection/gtagjs/enhanced-ecommerce */

  /* Home page includes browsing, search and product viewing */
  if (ctx.page.indexOf('HomePage') != -1) {
    try {
```

```

var _href = window.location.pathname;
if (_href.indexOf('?') != -1) {
    _href = _href.substring(0, _href.indexOf('?')); // cut away
query string
}

var _parts = _href.substr(ctx.root.length).split('/');
_loc = ctx.root;

var _view = 'category';
var _cat = 'x';
var _uri = '';
var _list = 'Browse';
var _dim = '';
for (var i = 0; i < _parts.length; i++) {
    var _part = _parts[i];
    if (_part == 'product' && (i+1 < _parts.length)) {
        _view = 'product';
        _uri = _parts[i+1];
        i++;
        break;
    }
    if (_part == 'sku' && (i+1 < _parts.length)) {
        _view = 'sku';
        _uri = _parts[i+1];
        i++;
        break;
    }
    if (_part == 'category' && (i+1 < _parts.length)) {
        _uri = _parts[i+1];
        _cat = _uri;
        _dim = _uri;
        i++;
    }
    if (_part == 'query' && (i+1 < _parts.length)) {
        _list = 'Search';
        _dim += 'query:' + decodeURIComponent(_parts[i+1]);
        i++;
    }
    if (_part == 'tag' && (i+1 < _parts.length)) {
        _list = 'Search';
        _dim += 'tag:' + decodeURIComponent(_parts[i+1]);
        i++;
    }
    if (_part == 'brand' && (i+1 < _parts.length)) {
        _list = 'Search';
        _dim += 'brand:' + decodeURIComponent(_parts[i+1]);
        i++;
    }
}

/* Record page views on page render */
if (_view == 'product' || _view == 'sku') {

```

```

        var _elp = jQuery('.js-ga-ecommerce-product');
        var _code = _elp.data('sku');
        var _name = _elp.data('name');
        var _brand = _elp.data('brand');
        var _fc = _elp.data('fc');
        var _elpp = _elp.find('.js-buy-panel
.js-ga-ecommerce-price');
        var _price = _elpp.length > 0 ? _elpp.data('price') :
'0.00';

        gtag('event', 'view_item', {          // Provide product
details
            'items': [
                {
                    'id': _code,
                    'name': _name,
                    'list_name': _list,
                    'brand': _brand,
                    'category': _cat + (_dim != '' ? ':' + _dim :
''),
                    'variant': _fc,
                    'list_position': 1,
                    'quantity': 1,
                    'price': _price
                }
            ],
            'event_label': 'pdp: View'
        });

        /* Bind buy buttons */
        _elp.find('.js-buy').click(function() {
            var _el = jQuery(this);
            var _sku = _el.data('sku');
            var _qty = _el.data('qty');
            var _fc = _el.data('fc');
            var _elpp =
_elpp.closest('.js-buy-panel').find('.js-ga-ecommerce-price');
            var _price = _elpp.length > 0 ? _elpp.data('price') :
'0.00';

            gtag('event', 'add_to_cart', {
                "items": [
                    {
                        'id': _sku,
                        'name': _name,
                        'list_name': _list,
                        'brand': _brand,
                        'category': _cat + (_dim != '' ? ':' +
_dim : ''),
                        'variant': _fc,
                        'list_position': 1,

```

```

        'quantity': _qty ? _qty : 1,
        'price': _price
    }
    ],
    'event_label': 'pdp: Add x'
});

});

} else {

    var _itemsImpr = [];

    jQuery('.js-ga-ecommerce-product-pod').each(function(_cnt,
_value) {

        var _item = jQuery(_value);
        var _sku = _item.data('sku');
        var _name = _item.data('name');
        var _price =
_item.find('.js-ga-ecommerce-price').data('price');
        var _qty = '1';
        var _fc = _item.data('fc');
        var _brand = _item.data('brand');

        _itemsImpr.push({
            'id': _sku,
            'name': _name,
            'list_name': _list,
            'brand': _brand,
            'category': _cat + (_dim != '' ? ':' + _dim : ''),
            'variant': _fc,
            'list_position': _cnt + 1,
            'quantity': _qty,
            'price': _price
        });

        _item.find('.js-buy').data('pos', _cnt +
1).click(function() {
            var _el = jQuery(this);
            var _sku = _el.data('sku');
            var _qty = _el.data('qty');
            var _fc = _el.data('fc');
            var _pos = _el.data('pos');

            gtag('event', 'add_to_cart', {
                'items': [
                    {
                        'id': _sku,
                        'name': _name,
                        'list_name': _list,
                        'brand': _brand,

```

```

        'category': _cat + (_dim != '' ? ':' +
_dim : ''),
        'variant': _fc,
        'list_position': _pos,
        'quantity': _qty ? _qty : 1,
        'price': _price
    }
    ],
    'event_label': 'list: Add x'
});

});

});

// Product impressions
if (_itemsImpr.length > 0) {
    gtag('event', 'view_item_list', {
        "items": _itemsImpr,
        'event_label': 'list: View'
    });
}

}

} catch(e) {
    log('Error initialising GA for HomePage', e);
}

}

/* Cart page, checkout step 0 */
if (ctx.page.indexOf('ShoppingCartPage') != -1) {
    try {

        var _cart = jQuery('.js-ga-ecommerce-cart-total
.js-ga-ecommerce-price').data('price');
        if (!_cart) {
            _cart = '0.00';
        }

        var _items = [];

        jQuery('.js-ga-ecommerce-cart-item').each(function(_cnt,
_value) {

            var _item = jQuery(_value);
            var _sku = _item.data('sku');
            var _name = _item.data('name');
            var _price = _item.data('price');
            var _qty = _item.data('qty');
            var _fc = _item.data('fc');

```

```

var _brand = _item.data('brand');

_items.push({
    'id': _sku,
    'name': _name,
    'list_name': 'Cart',
    'brand': _brand,
    //'category': 'zzzz',
    'variant': _fc,
    'list_position': _cnt + 1,
    'quantity': _qty,
    'price': _price
});

_item.find('.js-qty-update').data('pos', _cnt +
1).click(function() {

    var _newQty = _item.find('.js-qty-value').val();
    var _newQtyN = Number(_newQty);
    var _qtyN = Number(_qty);
    var _pos = jQuery(this).data('pos');

    if (_newQtyN > _qtyN) {
        gtag('event', 'add_to_cart', {
            'items': [
                {
                    'id': _sku,
                    'name': _name,
                    'list_name': 'Cart',
                    'brand': _brand,
                    //'category': 'zzzz',
                    'variant': _fc,
                    'list_position': _pos,
                    'quantity': _newQtyN - _qtyN,
                    'price': _price
                }
            ],
            'event_label': 'cart: Add X'
        });
    } else if (_newQtyN < _qtyN) {
        gtag('event', 'remove_from_cart', {
            'items': [
                {
                    'id': _sku,
                    'name': _name,
                    'list_name': 'Cart',
                    'brand': _brand,
                    //'category': 'zzzz',
                    'variant': _fc,
                    'list_position': _pos,
                    'quantity': _qtyN - _newQtyN,
                    'price': _price
                }
            ]
        });
    }
});

```



```

        ],
        'event_label': 'cart: Remove X'
    });
    }

    });

    _item.find('.js-cart-remove').data('pos', _cnt +
1).click(function() {
    try {
        var _pos = jQuery(this).data('pos');

        gtag('event', 'remove_from_cart', {
            'items': [
                {
                    'id': _sku,
                    'name': _name,
                    'list_name': 'Cart',
                    'brand': _brand,
                    // 'category': 'zzzzz',
                    'variant': _fc,
                    'list_position': _pos,
                    'quantity': _qty,
                    'price': _price
                }
            ],
            'event_label': 'cart: Remove all'
        });
    } catch(e) {
        log('Error executing GA event', e);
    }
});

});

var _coupons =
jQuery('.js-ga-ecommerce-applied-coupon').data('coupon');

    gtag('event', 'begin_checkout', { // track checkout steps
        'items': _items,
        'coupon': _coupons,
        'event_label': 'cart'
    });

} catch(e) {
    log('Error initialising GA for ShoppingCartPage', e);
}
}

/* Checkout pages */
if (ctx.page.indexOf('CheckoutPage') != -1) {
    try {

```

```

function getStepParameterByName( name ){
    var regexS = "[\\?&]" + name + "=(^&#)*",
        regex = new RegExp( regexS ),
        results = regex.exec( window.location.search );
    if( results == null ){
        return 'address';
    } else{
        return decodeURIComponent(results[1].replace(/\+/g, "
));
    }
}

var _stepnum = 0;
var _stepopt = '';
var _step = getStepParameterByName('step');
var _loginf = jQuery('form.js-login-form');
if (_loginf.length > 0) {
    _stepnum = 0;
    _step = 'login';

    // Login button
    var _loginf = jQuery('.js-login-submit');
    if (_loginf.length > 0) {
        _loginf.click(function() {
            try {
                gtag('event', 'login', {
                    'method' : 'customer'
                });
                gtag('event', 'checkout_progress', { // track
                    'checkout_step': _stepnum,
                    'event_label': _step + ': customer'
                });
            } catch(e) {
                log('Error executing GA event', e);
            }
        })
    }
    // Guest registration button
    var _guestf = jQuery('.js-guest-submit');
    if (_guestf.length > 0) {
        _guestf.click(function () {
            try {
                gtag('event', 'login', {
                    'method' : 'guest'
                });
                gtag('event', 'checkout_progress', { // track
                    'checkout_step': _stepnum,
                    'event_label': _step + ': guest'
                });
            } catch(e) {
                log('Error executing GA event', e);
            }
        })
    }
}

```

```

        }
    });
}

// Registration button
var _regf = jQuery('.js-register-submit');
if (_regf.length > 0) {
    _regf.click(function () {
        try {
            gtag('event', 'login', { method :
'registration' });
        } catch(e) {
            log('Error executing GA event', e);
        }
        form.submit();
    });
}

} else if (_step == 'address') {
    _stepnum = 1;
} else if (_step == 'ship') {
    _stepnum = 2;
    _stepopt =
jQuery('.js-checkout-shipping').find('input[name=\'shippingmethod\']:check
ed').parent().find('span:first').html();
} else if (_step == 'payment') {
    _stepnum = 3;
    _stepopt =
jQuery('.js-checkout-pg').find('input[name=\'pg\']:checked').parent().find
('span:first').html();

    var _elp = jQuery('#paymentDiv');
    if (_elp.length > 0) {

        _elp.closest('form').submit(function() {
            try {

                var _el = jQuery('.js-ga-ecommerce-order');
                var _tx = _el.data('ordernum');
                var _rev = _el.data('orderrev');
                var _shp = _el.data('ordershp');
                var _tax = _el.data('ordertax');
                var _curr = _el.data('currency');

                var _items = [];
                var _itemsCnt = 1;

                jQuery('.js-ga-ecommerce-cart-item').each(function() {

                    var _item = jQuery(this);
                    var _sku = _item.data('sku');

```

```

        var _name = _item.data('name');
        var _price = _item.data('price');
        var _qty = _item.data('qty');
        var _fc = _item.data('fc');
        var _brand = _item.data('brand');

        _items.push({
            'id': _sku,
            'name': _name,
            'list_name': 'Cart',
            'brand': _brand,
            //'category': 'zzzz',
            'variant': _fc,
            'list_position': _itemsCnt++,
            'quantity': _qty,
            'price': _price
        });
    });

    gtag('event', 'purchase', {
        'transaction_id': _tx,
        'affiliation': null,
        'value': _rev,
        'currency': _curr,
        'tax': _tax,
        'shipping': _shp,
        'items': _items,
        'event_label': 'confirm'
    });

    } catch(e) {
        log('Error executing GA event', e);
    }
});

}

}

if ((_stepnum == 2 || _stepnum == 3) && _stepopt) {

    steps
        gtag('event', 'checkout_progress', { // track checkout

            'checkout_step': _stepnum,
            'event_label': _step + ': ' + _stepopt
        });

        steps options for shipping and payment
        gtag('event', 'set_checkout_option', { // track checkout

            'checkout_step': _stepnum,
            'checkout_option': _step,
            'value': _stepopt,
            'event_label': _step + ': ' + _stepopt
        });
    }
}

```

```

        });
    } else {

        steps
        gtag('event', 'checkout_progress', { // track checkout
            'checkout_step': _stepnum,
            'event_label': _step
        });

    }

    } catch(e) {
        log('Error initialising GA for CheckoutPage', e);
    }

}

/* Login/Registration pages */
if (ctx.page.indexOf('LoginPage') != -1) {
    try {
        // Login button
        var _loginf = jQuery('.js-login-submit');
        if (_loginf.length > 0) {
            _loginf.click(function() {
                try {
                    gtag('event', 'login', { method : 'customer' });
                } catch(e) {
                    log('Error executing GA event', e);
                }
            })
        }

        // Registration button
        var _regf = jQuery('.js-register-submit');
        if (_regf.length > 0) {
            _regf.click(function () {
                try {
                    gtag('event', 'login', { method : 'registration'
                });

                } catch(e) {
                    log('Error executing GA event', e);
                }
                form.submit();
            });
        }

    } catch(e) {
        log('Error initialising GA for LoginPage', e);
    }
}

```

```
</script>
```

In version 4.1.0+ variant of the above script was included as gtag.js resource file and hence in order to enable it only the following code snippet is required:

GA commerce integration (4.1.0+)

```
<script>  
ctx.objects.gtag.initialize();  
</script>
```

Provided script supports the following events:

Page	Event	Event label	Event structure	Comments
Login page	login		{ method : 'customer' }	tracks clicks of '.js-login-submit'
	login		{ method : 'registration' }	tracks clicks of '.js-register-submit'
Category and Search pages	view_item_list	list: View	'items': [{ 'id': _code, 'name': _name, 'list_name': _list, 'brand': _brand, 'category': _cat or _dim, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }]	tracks product impressions

	add_to_cart	list: Add x	<pre> 'items': [{ 'id': _code, 'name': _name, 'list_name': _list, 'brand': _brand, 'category': _cat or _dim, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	add to cart events triggered by 'js-buy' buttons
Product details page	view_item	pdp: View	<pre> 'items': [{ 'id': _code, 'name': _name, 'list_name': _list, 'brand': _brand, 'category': _cat or _dim, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	tracks product views

	add_to_cart	pdp: Add x	<pre> 'items': [{ 'id': _code, 'name': _name, 'list_name': _list, 'brand': _brand, 'category': _cat or _dim, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	add to cart events triggered by 'js-buy' buttons
Cart page	add_to_cart	cart: Add X	<pre> 'items': [{ 'id': _code, 'name': 'Cart', 'list_name': _list, 'brand': _brand, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	add to cart events triggered by quantity updates

	remove_from_cart	cart: Remove X / cart : Remove all	<pre> 'items': [{ 'id': _code, 'name': 'Cart', 'list_name': _list, 'brand': _brand, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	remove from cart event triggered by quantity updates
	begin_checkout	cart	<pre> 'items': [{ 'id': _code, 'name': 'Cart', 'list_name': _list, 'brand': _brand, 'variant': _fc, 'list_position': 1, 'quantity': 1, 'price': _price }] </pre>	tracks beginning of the checkout process
Checkout page	login + checkout_progress	login: customer	<pre> { method : 'customer' } </pre>	tracks clicks of '.js-login-submit'
	login + checkout_progress	login: registration	<pre> { method : 'registration' } </pre>	tracks clicks of '.js-register-submit'
	login + checkout_progress	login: guest	<pre> { method : 'guest' } </pre>	tracks clicks of '.js-guest-submit'
	checkout_progress + set_checkout_option	ship: "shipping method name"		tracks shipping method selections
	checkout_progress + set_checkout_option	payment: "payment method name"		tracks payment method selections

	purchase	confirm	<pre> { 'transaction_id': _tx, 'affiliation': null, 'value': _rev, 'currency': _curr, 'tax': _tax, 'shipping': _shp, 'items': _items, 'event_label': 'confirm' } </pre>	tracks clicking payment button
--	----------	---------	---	-----------------------------------

Resources

Google structured data tool

<https://search.google.com/structured-data/testing-tool/>

This is a useful tool to help analyse schema.org meta data in your pages.